9-20-2024 2:30 PM

Variational Bayesian inference for functional data clustering and survival data analysis

Chengqian Xian, *The University of Western Ontario*

Supervisor: Camila P.E. de Souza, *The University of Western Ontario*
Co-Supervisor: Wenqing He, *The University of Western Ontario*
Co-Supervisor: Felipe F. Rodrigues, *King's University College at Western University*
A thesis submitted in partial fulfillment of the requirements for the Doctor of Philosophy degree in Statistics and Actuarial Sciences

Abstract

Variational Bayesian inference is a method to approximate the posterior distribution under a Bayesian model analytically. As an alternative to Markov Chain Monte Carlo (MCMC) methods, variational inference (VI) produces an analytical solution to an approximation of the posterior but have a lower computational cost compared to MCMC methods. The main challenge of applying VI comes from deriving the equations used to update the approximated posterior parameters iteratively, especially when dealing with complex data. In this thesis, we apply the VI to the context of functional data clustering and survival data analysis. The main objective is to develop novel VI algorithms and investigate their performance under these complex statistical models.

In functional data analysis, clustering aims to identify underlying groups of curves without prior group membership information. The first project in this thesis presents a novel variational Bayes (VB) algorithm for simultaneous clustering and smoothing of functional data using a B-spline regression mixture model with random intercepts. The deviance information criterion is employed to select the optimal number of clusters.

The second project shifts focus to survival data analysis, proposing a novel mean-field VB algorithm to infer parameters of the log-logistic accelerated failure time (AFT) model. To address intractable calculations, we propose and incorporate a piecewise approximation technique into the VB algorithm, achieving Bayesian conjugacy.

The third project is motivated by invasive mechanical ventilation data from intensive care units (ICUs) in Ontario, Canada, which form multiple clusters. We assume that patients within the same ICU cluster are correlated. Extending the second project's methodology, a shared frailty log-logistic AFT model is introduced to account for intra-cluster correlation through a cluster-specific random intercept. A novel and fast VB algorithm for model parameter inference is presented.

Extensive simulation studies assess the performance of the proposed VB algorithms, com-

paring them with other methods, including MCMC algorithms. Applications to real data, such as ICU ventilation data from Ontario, illustrate the methodologies' practical use. The proposed VB algorithms demonstrate excellent performance in clustering functional data and analyzing survival data, while significantly reducing computational cost compared to MCMC methods.

Keywords: Bayesian inference, mean-field variational Bayes, functional data analysis, mixture models, survival analysis, accelerated failure time models, log-logistic regression, random effects, ventilation duration analysis

Summary for Lay Audience

Variational inference (VI) is a statistical technique used to estimate model parameters within a Bayesian framework. It provides similar accuracy to the traditional Markov Chain Monte Carlo (MCMC) method but does so more efficiently. My research explores applying VI in two key areas: clustering time-varying data (like daily temperature changes) and analyzing time-to-event data (such as hospital stay durations).

The first study introduces a new approach to group time-varying data into meaningful clusters, helping us identify underlying patterns without prior knowledge of these groups. For instance, this method could reveal geographical areas with similar weather patterns based on daily temperature data. We also use an information criterion to determine the optimal number of clusters.

The second study focuses on time-to-event (or survival time) data. We propose a new method to analyze the impact of risk factors on the time-to-event and predict survival times. This is particularly valuable in medical research, where understanding the lifespan of patients with specific conditions is crucial.

The third study is motivated by data on ventilation duration in intensive care units (ICUs) in Ontario, Canada. ICU patients often share similar environments, and our method accounts for these similarities to provide more accurate predictions of survival time (i.e., ventilation duration).

Each project includes extensive simulation studies that demonstrate the effectiveness of our proposed methods. We also apply these methodologies to various real-world datasets. Overall, my research aims to make advanced data analysis tools more accessible and efficient, ultimately supporting better decision-making in fields like healthcare.

Co-Authorship Statement

This thesis includes research that was conducted in collaboration with several co-authors. Materials from three jointly authored papers are contained in this thesis.

- Chapter 2 is based on a manuscript titled Clustering Functional Data via Variational Inference, co-authored with Dr. Camila P.E. de Souza (Supervisor), John Jewell and Dr. Ronaldo Dias. This manuscript has been published in *Advances in Data Analysis and Classification*.

- Chapter 3 is based on a manuscript titled Variational Bayesian Analysis of Survival Data Using a Log-logistic Accelerated Failure Time Model, co-authored with Dr. Camila P.E. de Souza (Supervisor), Dr. Wenqing He (Co-Supervisor), Dr. Felipe F. Rodrigues (Co-supervisor), Dr. Renfang Tian. This manuscript has been published in *Statistics and Computing*.

- Chapter 4 is based on a paper titled Fast Variational Bayesian Inference for Correlated Survival Data: an Application to Invasive Mechanical Ventilation Duration Analysis, co-authored with Dr. Camila P.E. de Souza (Supervisor), Dr. Wenqing He (Co-Supervisor), Dr. Felipe F. Rodrigues (Co-supervisor), Dr. Renfang Tian. This manuscript has been submitted for peer review.

I certify that I am the principal contributor for all these manuscripts. I extend my sincere gratitude to all my co-authors for their invaluable contributions and collaboration.

Acknowledgements

First and foremost, I extend my deepest appreciation and gratitude to my supervisors, Dr. Camila de Souza, Dr. Wenqing He, and Dr. Felipe Rodrigues. Their unwavering support, mentorship, and guidance have been indispensable throughout my PhD journey. Their expertise, insightful feedback, and constant encouragement have been instrumental in helping me achieve this milestone. I am immensely grateful for the mental and financial support they have provided, as well as for the invaluable introductions to many esteemed colleagues at various conferences. Thank you for fostering my scientific independence and nurturing my personal growth and development.

I would also like to express my gratitude to Dr. Simon Bonner, Dr. Shu Li, Dr. Osvaldo Espin-Garcia, and Dr. Alexandra Schmidt for serving as my thesis examiners and for their thorough review of my thesis.

Additionally, I want to thank Dr. Ronaldo Dias, Dr. Renfang Tian, and John Jewell for their co-authorship and significant contributions in reviewing and revising our manuscripts.

I am deeply thankful to Dr. Grace Y. Yi and all the members of the GW-DSRG (Grace-Wenqing Data Science Research Group). Dr. Yi's insightful comments and continuous encouragement after each of my presentations in the research group have been invaluable. The discussions during our weekly group meetings have enriched my knowledge in many research areas. I have also had the pleasure of befriending many talented and diligent peers: Ana Carolina Da Cruz, Dr. Dan Liu, Dr. Jingyu Cui, Dr. Yasin Khadem Charvadeh, Pingbo Hu, Ruimin Gao, Yu Shi, Dr. Dechen Gao, Yuan Bian, Gansen Deng, Yijia Weng, Hui Guo, Jingwei Lu, Dr. Yifan Sun, Duo Xu, Shiyu He, Dr. Yang Miao, Dr. Yiming Huang, Xinyi Zeng, Jiaxua Lu, Jet Yuhao Zhou, Jiachen Pan, Yunzhuo Zhang, Wenhao Chen, Xinshen Yang, and many others.

I would like to express my sincere thanks to my dear friends: Yuhong Peng, Yao Li, Sam Zhehao Xu, and Steven Zihan Cai. I cherished the study sessions, entertainments, and

outdoor activities we enjoyed together over the past four years. Thank you for your unwavering support and for being my best friends.

My family, undoubtedly, plays an essential role in my life. I am deeply grateful to my dad, my mom, and my brothers and sisters. Thank you all for your love and support. It was your encouragement that inspired me to study abroad and gave me the confidence to overcome difficulties.

Table of Contents

List of Tables

List of Figures

List of Appendices

Chapter 1
Introduction

As an important technique in statistics, Bayesian inference is widely developed and applied in a broad range of fields. The main objective of Bayesian inference is to derive the posterior distribution of parameters under a statistical model. Specifically, Bayesian inference computes the posterior density based on Bayes' theorem, which incorporates the prior distribution and the likelihood derived from the observed data. However, for complex statistical Bayesian models with many parameters, obtaining a closed form of the posterior distribution is usually challenging and even impossible. In such situations, the exact posterior distribution is intractable. Therefore, the fundamental problem changes to the approximation of the exact posterior. For decades, Markov Chain Monte Carlo (MCMC) algorithms such as the Gibbs sampler (Geman and Geman, 1984; Gelfand and Smith, 1990) and the Metropolis–Hastings algorithm (Hastings, Hastings) have been the dominant approximation technique to achieve this goal. As demonstrated in literature (Blei et al., 2017; Gunapati et al., 2022; Cabral et al., 2024), MCMC-based sampling algorithms are computationally intensive, particularly when handling large datasets.

1.1 Variational inference as an alternative to MCMC

To address the high computational cost of MCMC algorithms, researchers have begun exploring alternative methods that aim to achieve comparable posterior approximations with reduced computational demands. Variational Bayesian inference, developed from machine learning (Jordan et al., 1999), is utilized as an alternative to MCMC methods to approximate the posterior distribution under the Bayesian framework. The critical difference between the two methods is that variational Bayesian inference approximates the posterior via optimization while MCMC methods provide a numerical approximation by sampling

(Wainwright et al., 2008). On the one hand, as a Bayesian method, variational inference can incorporate prior information to accurately approximate the posterior distribution. On the other hand, many recent applications of variational Bayesian inference show that it can provide comparable results to sampling techniques but with a lower computational cost (Blei et al., 2017). However, the main challenge of applying variational Bayesian inference comes from deriving the set of equations used to update the posterior approximation parameters iteratively within an algorithm for the optimization problem.

The idea of variational inference (VI) can be traced back to Peterson and Anderson (1987), where a mean-field theory algorithm was proposed for a neural networks model. By making connection between variational Bayesian inference and the well-known expectation maximization (EM) algorithm (Dempster et al., 1977), faster development and broader application of variational Bayesian inference emerge in the field of machine learning (Bishop et al., 1997; Barber and Wiegerinck, 1998; Barber and de van Laar, 1999). Following Wainwright et al. (2008), Blei et al. (2017) provide a comprehensive review of VI for statisticians, including but not limited to the statistical motivations behind VI, particularly VI under the exponential family models, the coordinate ascent mean-field VI and comparison to MCMC. Blei et al. (2017) also present a complete example of applying VI to the Bayesian mixture of Gaussians to help understand the methodology of variational Bayesian inference. More recently, Lee (2022) provides a theoretical review and comparison of the Gibbs sampler and coordinate ascent variational inference (CAVI). The CAVI is a commonly utilized variational Bayesian inference method in which the solution to the optimization problem is obtained via the coordinate ascent algorithm. Such an interesting connection between the two essential techniques provides a complementary illustration for Bayesian approximation inference with practical implications.

1.2 Mean-field variational Bayes and the CAVI algorithm

Based on Blei et al. (2017), we present the idea of the most commonly used type of variational Bayesian inference, the mean-field variational Bayes (VB) and the CAVI algorithm to obtain the approximated posterior distribution. Let $\boldsymbol{\theta} \in \Theta$ be the parameter vector in the Bayesian model and the observed data be $\mathbf{y}$. The idea of VB is to find a variational density, denoted by $q^*(\boldsymbol{\theta})$ coming from a family of possible densities Q, to approximate $p(\boldsymbol{\theta}|\mathbf{y})$, which can be solved in terms of an optimization problem formed by the Kullback–Leibler (KL) divergence (Kullback and Leibler, 1951) as a minimization criterion. The KL divergence measures the closeness between the possible densities q in the family Q and the exact posterior density p. The optimization problem can be expressed as

$$q^*(\boldsymbol{\theta}) = \underset{q \in Q}{\operatorname{argmin}} \, \mathrm{KL}(q(\boldsymbol{\theta}) \| p(\boldsymbol{\theta}|\mathbf{y})). \tag{1.1}$$

Jordan et al. (1999) and Blei et al. (2017) show that minimizing the KL divergence is equivalent to maximizing the so-called evidence lower bound (ELBO) defined as

$$\mathrm{ELBO}(q) = \mathbb{E}_q \log p(\boldsymbol{\theta}, \mathbf{y}) - \mathbb{E}_q \log q(\boldsymbol{\theta}),$$

where $\log p(\boldsymbol{\theta}, \mathbf{y})$ is called the complete-data log-likelihood.

Maximizing the ELBO corresponding to a sophisticated variational family can be quite challenging. However, when we consider the mean-field variational family denoted by Q, where the sets of parameters and latent variables are assumed to be mutually independent, and each of them is governed by a distinct factor in the variational density, the optimization problem then changes to

$$q^*(\boldsymbol{\theta}) = \underset{q \in Q}{\operatorname{argmax}} \, \mathrm{ELBO}(q(\boldsymbol{\theta})) = \underset{q \in Q}{\operatorname{argmax}} \, \mathrm{ELBO}\Big(\prod_{k=1}^{K} q_k(\theta_k) \Big), \tag{1.2}$$

where we assume there are K sets of parameters and latent variables, $\boldsymbol{\theta} = \{\theta_1, ..., \theta_K\}$.

In what follows, we introduce the coordinate ascent algorithm under the mean-field VB (Bishop, 2006; Blei, 2011), namely the CAVI algorithm, which makes the variational Bayesian inference a popular alternative to MCMC methods. As in Equation (1.2), we aim to maximize the objective function, the ELBO, to find an optimal $q(\boldsymbol{\theta})$ to approximate the exact posterior. First, we decompose the ELBO

$$\text{ELBO} := \mathcal{L} = \log p(\mathbf{y}) + \sum_{k=1}^{K} \Big\{ \mathbb{E}_q[\log p(\theta_k | \theta_{1:(k-1)}, \mathbf{y})] - \mathbb{E}_{q_k}[\log q(\theta_k)] \Big\}, \tag{1.3}$$

using the following facts of chain rule and mean-field assumption, respectively,

$$p(\boldsymbol{\theta}, \boldsymbol{y}) = p(\mathbf{y}) \prod_{k=1}^{K} p(\theta_k | \theta_{1:(k-1)}, \mathbf{y}) \quad \text{and} \quad \mathbb{E}_q[\log p(\boldsymbol{\theta})] = \sum_{k=1}^{K} \mathbb{E}_{q_k}[\log q(\theta_k)].$$

In the next step, we write the ELBO as a function of $q(\theta_k)$ so that we can find the update equation for each variational component $q(\theta_k)$ to maximize the ELBO. To achieve this, we first consider the variable θ_k as the last variable in the list $\{\theta_1, ..., \theta_K\}$, and rewrite the objective function in Equation (1.3) as follows:

$$\mathcal{L} = \mathbb{E}_q[\log p(\theta_k | \boldsymbol{\theta}_{-k}, \mathbf{y})] - \mathbb{E}_{q_k}[\log q(\theta_k)] + \text{constant}, \tag{1.4}$$

where the term constant is a fixed value with respect to θ_k. Therefore, we have the the objective function as a function of $q(\theta_k)$, denoted by $\mathcal{L}_k$:

$$\mathcal{L}_k = \int q(\theta_k) \mathbb{E}_{-k} \log p(\theta_k | \boldsymbol{\theta}_{-k}, \mathbf{y}) d\theta_k - \int q(\theta_k) \log(q(\theta_k)) d\theta_k. \tag{1.5}$$

We now take the derivative of $\mathcal{L}_k$ with respect to $q(\theta_k)$ and obtain the coordinate ascent

update equation for $q(\theta_k)$, denoted by $q^*(\theta_k)$,

$$q^*(\theta_k) \propto \mathbb{E}_{-k}[\log p(\theta_k|\boldsymbol{\theta}_{-k}, \mathbf{y})] \propto \mathbb{E}_{-k}[\log p(\boldsymbol{\theta}, \mathbf{y})], \tag{1.6}$$

where $\mathbb{E}_{-k}$ means that the expectation is taken with respect to the variational density of all other parameters and latent variables except the one of interest.

There may be challenges when applying the mean-field variational inference using the CAVI algorithm. First, when the observed data is complicated, the closed form of the update equations in the CAVI algorithm may not be available. The complexity of the observed data can come from a complex distribution (e.g., outside the exponential family) or have a specific data scheme (e.g., censoring). In these situations, the complete-data log-likelihood may not have a closed form, or the expectation over the complete-data log-likelihood is challenging to compute. As a result, some update equations are intractable, and the model is non-conjugate. Implementing tractable variational Bayes for non-conjugate models can be found in Khan et al. (2012); Seeger and Bouchard (2012); Wang and Blei (2013); Wand (2014); Khan and Lin (2017); Galy-Fajou et al. (2020).

1.3 Thesis contributions

Consisting of three papers, this thesis aims to explore the performance of variational Bayesian inference under the context of functional data clustering and survival data analysis by developing novel and fast variational inference algorithms.

Current research in functional data clustering focuses mainly on two-stage models, or the inference in model-based clustering is conducted via the EM or MCMC algorithms. Our objective in the first project is to introduce variational Bayesian inference into functional data analysis and provide a new methodology for smoothing and clustering functional data simultaneously. We propose a B-spline regression mixture model and develop a two-fold

scheme to select the optimal number of clusters using the deviance information criterion. The proposed variational inference algorithm is evaluated and compared with other methods (k-means, functional k-means and two other model-based methods) via simulation studies with various scenarios. We compare the posterior estimation results from our proposed algorithm with the ones from MCMC. We apply our proposed methodology to two publicly available datasets: the Growth data (Tuddenham and Snyder, 1954) and the Canadian weather data (Ramsay and Silverman, 2005). We demonstrate that the proposed VB algorithm achieves satisfactory clustering performance in both simulation and real data analyses.

As an alternative to the Cox proportional hazard model, the accelerated failure time (AFT) model also plays an essential role in survival regression analysis. To the best of our knowledge, there was no research or related work on variational Bayesian inference under the AFT model setting. There are several challenges in applying VB in this context. First, the distribution of the survival data is usually right-skewed and may not belong to the exponential family, making it difficult and intractable to compute the expectation of complete-data log-likelihood over the variational density. Second, survival data have special data schemes, such as censoring, under the context of data incompleteness. Third, the survival data may come from multiple research centers, so that the subjects within the same cluster are no longer independent.

Therefore, our objective for the second and third projects is to fill the gap of variational Bayesian inference in the AFT survival regression analysis while addressing some of these challenges. More specifically, in the second project, we propose a novel VB algorithm to infer parameters of the log-logistic AFT model. To address intractable calculations, a piecewise approximation technique is integrated into the VB algorithm to achieve Bayesian conjugacy. The proposed VB algorithm is evaluated and compared with frequentist and MCMC techniques using simulated data under various scenarios. A publicly available

dataset, *rhDNASE* data (Fuchs et al., 1994; Therneau and Hamilton, 1997), is employed for illustration. The third project is motivated by invasive mechanical ventilation data from different intensive care units (ICUs) in Ontario, Canada, forming multiple clusters. The survival times from patients within the same ICU cluster are correlated. To address this association, we introduce a shared frailty log-logistic accelerated failure time model to account for the intra-cluster correlation through a cluster-specific random intercept. We present a novel, fast VB algorithm for parameter inference and evaluate its performance using simulation studies varying the number of clusters and their sizes. We further compare the performance of our proposed VB algorithm with the h-likelihood (Do Ha et al., 2002) method and the MCMC algorithm.

1.4 Thesis organization

The remainder of this thesis is organized as follows. In Chapter 2[1], we present the manuscript resulting from our first project applying VI to functional data clustering. In Chapter 3[2], we present the manuscript resulted from our second project, where we proposed a novel algorithm to infer parameters of the log-logistic AFT model. Chapter 4[3] presents the paper corresponding to the analysis of Ontario ICU data via a shared frailty log-logistic AFT model. A summary of research findings and future work is provided in Chapter 5.

[1]Chapter 2 has been published in *Advances in Data Analysis and Classification* (Xian et al., 2024)
[2]Chapter 3 has been published in *Statistics and Computing* (Xian et al., 2024)
[3]Chapter 4 has been submitted for peer review

References

Barber, D. and P. de van Laar (1999). Variational cumulant expansions for intractable distributions. *Journal of Artificial Intelligence Research 10*, 435–455. → page 2

Barber, D. and W. Wiegerinck (1998). Tractable variational structures for approximating graphical models. In M. Kearns, S. Solla, and D. Cohn (Eds.), *Advances in Neural Information Processing Systems*, Volume 11, pp. 183–189. MIT Press. → page 2

Bishop, C. (2006). *Pattern Recognition and Machine Learning*. Springer. → page 4

Bishop, C., N. Lawrence, T. Jaakkola, and M. Jordan (1997). Approximating posterior distributions in belief networks using mixtures. In M. Jordan, M. Kearns, and S. Solla (Eds.), *Advances in Neural Information Processing Systems*, Volume 10. MIT Press. → page 2

Blei, D. M. (2011). Variational inference. `https://www.cs.princeton.edu/courses/archive/fall11/cos597C/lectures/variational-inference-i.pdf`. Lecture notes. → page 4

Blei, D. M., A. Kucukelbir, and J. D. McAuliffe (2017). Variational inference: A review for statisticians. *Journal of the American Statistical Association 112*(518), 859–877. → page 1, 2, 3

Cabral, R., D. Bolin, and H. Rue (2024). Fitting latent non-Gaussian models using variational Bayes and Laplace approximations. *Journal of the American Statistical Association*, 1–13. → page 1

Dempster, A. P., N. M. Laird, and D. B. Rubin (1977). Maximum likelihood from in-

complete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological) 39*(1), 1–22. → page 2

Do Ha, I., Y. Lee, and J.-K. Song (2002). Hierarchical-likelihood approach for mixed linear models with censored data. *Lifetime Data Analysis 8*, 163–176. → page 7

Fuchs, H. J., D. S. Borowitz, D. H. Christiansen, E. M. Morris, M. L. Nash, B. W. Ramsey, B. J. Rosenstein, A. L. Smith, and M. E. Wohl (1994). Effect of aerosolized recombinant human DNase on exacerbations of respiratory symptoms and on pulmonary function in patients with cystic fibrosis. *New England Journal of Medicine 331*(10), 637–642. → page 7

Galy-Fajou, T., F. Wenzel, and M. Opper (2020). Automated augmented conjugate inference for non-conjugate gaussian process models. In *International Conference on Artificial Intelligence and Statistics*, pp. 3025–3035. PMLR. → page 5

Gelfand, A. E. and A. F. M. Smith (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association 85*(410), 398–409. → page 1

Geman, S. and D. Geman (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-6*(6), 721–741. → page 1

Gunapati, G., A. Jain, P. Srijith, and S. Desai (2022). Variational inference as an alternative to MCMC for parameter estimation and model selection. *Publications of the Astronomical Society of Australia 39*, e001. → page 1

Hastings, W. K. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika 57*(1), 97–109. → page 1

Jordan, M. I., Z. Ghahramani, T. Jaakkola, and L. Saul (1999). Introduction to variational methods for graphical models. *Machine Learning 37*, 183–233. → page 1, 3

Khan, E., S. Mohamed, and K. P. Murphy (2012). Fast Bayesian inference for non-conjugate Gaussian process regression. *Advances in Neural Information Processing Systems 25*. → page 5

Khan, M. and W. Lin (2017). Conjugate-computation variational inference: Converting variational inference in non-conjugate models to inferences in conjugate models. In *Artificial Intelligence and Statistics*, pp. 878–887. PMLR. → page 5

Kullback, S. and R. A. Leibler (1951). On information and sufficiency. *The Annals of Mathematical Statistics 22*(1), 79 – 86. → page 3

Lee, S. Y. (2022). Gibbs sampler and coordinate ascent variational inference: A set-theoretical review. *Communications in Statistics - Theory and Methods 51*(6), 1549–1568. → page 2

Peterson, C. and J. R. Anderson (1987). A mean field theory learning algorithm for neural networks. *Complex Systems 1*(1), 995–1019. → page 2

Ramsay, J. O. and B. W. Silverman (2005). *Functional Data Analysis* (2 ed.). Springer. → page 6

Seeger, M. and G. Bouchard (2012). Fast variational Bayesian inference for non-conjugate matrix factorization models. In *Artificial Intelligence and Statistics*, pp. 1012–1018. PMLR. → page 5

Therneau, T. M. and S. A. Hamilton (1997). rhDNase as an example of recurrent event analysis. *Statistics in Medicine 16*(18), 2029–2047. → page 7

Tuddenham, R. D. and M. M. Snyder (1954). Physical growth of California boys and

girls from birth to eighteen years. *Publications in Child Development. University of California, Berkeley 12*, 183–364. → page 6

Wainwright, M. J., M. I. Jordan, et al. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning 1*(1–2), 1–305. → page 2

Wand, M. P. (2014). Fully simplified multivariate normal updates in non-conjugate variational message passing. *Journal of Machine Learning Research*. → page 5

Wang, C. and D. M. Blei (2013). Variational inference in nonconjugate models. *Journal of Machine Learning Research*. → page 5

Xian, C., C. P. de Souza, W. He, F. F. Rodrigues, and R. Tian (2024). Variational Bayesian analysis of survival data using a log-logistic accelerated failure time model. *Statistics and Computing 34*(2), 67. → page 7

Xian, C., C. P. E. de Souza, J. Jewell, and R. Dias (2024). Clustering functional data via variational inference. *Advances in Data Analysis and Classification*, 1–50. → page 7

Chapter 2
Clustering functional data via variational inference

2.1 Introduction

Functional data analysis (FDA) [1], term first coined by Ramsay and Dalzell (1991), deals with the analysis of data that are defined on some continuum such as time. Theoretically, data are in the form of functions, but in practice they are observed as a series of discrete points representing an underlying curve. Ramsay and Silverman (2005) establish a foundation for FDA on topics including smoothing functional data, functional principal components analysis and functional linear models. Ramsay et al. (2009) provide a guide for analyzing functional data in R and Matlab using publicly available datasets. Wang et al. (2016) present a comprehensive review of FDA, in which clustering and classification methods for functional data are also discussed. Functional data analysis has been applied to various research areas such as energy consumption (Lenzi et al., 2017; De Souza et al., 2017; Franco et al., 2023), rainfall data visualization (Hael et al., 2020), income distribution (Hu et al., 2020), spectroscopy (Dias et al., 2015; Yang et al., 2021; Frizzarin et al., 2021), and Covid-19 pandemic (Boschi et al., 2021; Sousa et al., 2023; Collazos et al., 2023), to mention a few.

Cluster analysis of functional data aims to determine underlying groups in a set of observed curves when there is no information on the group label of each curve. As described in Jacques and Preda (2014), there are three main types of methods used for functional data clustering: dimension reduction-based (or filtering) methods, distance-based methods, and

[1] A version of this chapter has been published in *Advances in Data Analysis and Classification* (Xian et al., 2024).

model-based methods. Functional data generally belongs to the infinite-dimensional space, making those clustering methods for finite-dimensional data ineffective. Therefore, dimension reduction-based methods have been proposed to solve this problem. Before clustering, a dimension reduction step (also called *filtering* in James and Sugar, 2003) is carried out by the techniques including spline basis function expansion (Tarpey and Kinateder, 2003) and functional principal component analysis (Jones and Rice, 1992). Clustering is then performed using the basis expansion coefficients or the principal component scores, resulting in a two-stage clustering procedure. Distance-based methods are the most well-known and popular approaches for clustering functional data since no parametric assumptions are necessary for these algorithms. Nonparametric clustering techniques, including k-means clustering (Hartigan and Wong, 1979) and hierarchical clustering (Ward, 1963), are usually applied using specific distances or dissimilarities between curves (Delaigle et al., 2019; Martino et al., 2019; Zambom et al., 2019; Li and Ma, 2020). It is important to note that distance-based methods are sometimes equivalent to dimension reduction-based methods if, for example, distances are computed using the basis expansion coefficients. Another widely-used approach is model-based clustering, where functional data are assumed to arise from a mixture of underlying probability distributions. For example, in Bayesian hierarchical clustering, a common methodology is to assume that the set of coefficients in the basis expansion representing functional data follow a mixture of Gaussian distributions (Wang et al., 2016).

Chamroukhi and Nguyen (2019) recently provided a comprehensive review for model-based clustering of functional data. A common model-based approach is to represent functional data as a linear combination of basis functions (e.g., B-splines) and consider a finite regression mixture model (Grün, 2019) with the matrix of basis function evaluations as the design matrix and a set of basis expansion coefficients for each mixture component. The estimation and inference of the mixture parameters as well as the regression (or basis expansion) coefficients are usually conducted via the Expectation-Maximization (EM) al-

gorithm (Samé et al., 2011; Jacques and Preda, 2013; Giacofci et al., 2013; Chamroukhi, 2016a; Grün, 2019) or Markov Chain Monte Carlo (MCMC) sampling techniques (Ray and Mallick, 2006; Fruhwirth-Schnatter et al., 2019). An alternative approach to EM and MCMC is the use of variational inference techniques.

Bayesian variational inference has found versatile applications within the field of FDA. Variational Bayes for fast approximate inference was applied in functional regression analysis by Goldsmith et al. (2011). Beyond functional regression, another pivotal facet of FDA lies in functional data registration, with a growing interest in the joint clustering and registration of functional data (Zhang and Telesca, 2014). A novel adapted variational Bayes algorithm for smoothing and registration of functional data simultaneously via Gaussian processes was proposed by Earls and Hooker (2017). Nguyen and Gelfand (2011) considered a random allocation process, namely the Dirichlet labelling process, to cluster functional data and inferred model parameters by Gibbs sampling and variational Bayes. In a recent development, Rigon (2023) extended the work of Blei and Jordan (2006) and proposed an enriched Dirichlet mixture model for functional clustering via a variational Bayes algorithm. Rigon (2023) considered a Bayesian functional mixture model without random effects and introduced a functional Dirichlet multinomial process to allow the estimation of the number of clusters.

In this chapter, we develop a novel variational Bayes algorithm for clustering functional data via a regression mixture model. In contrast to Rigon (2023), we consider a regression mixture model with random intercepts and take on a two-fold scheme for choosing the best number of clusters using the deviance information criterion (Spiegelhalter et al., 2002). We model the raw data, simultaneously obtaining clustering assignments and cluster-specific smooth mean curves. We compare the posterior estimation results from our proposed VB with the ones from MCMC. Our proposed method is implemented in R, and codes are available at `https://github.com/chengqianxian/funclustVI`.

The remainder of this chapter is organized as follows. Section 2.2 presents our two model settings and proposed algorithms. In Section 2.3, we conduct simulation studies to assess the performance of our methods under various scenarios. In Section 2.4, we apply our proposed methodology to real datasets. A conclusion of our study and a discussion on the proposed method are provided in Section 2.5.

2.2 Methodology

2.2.1 Overview of variational inference

Variational inference (VI) is a method from machine learning that approximates the posterior density in a Bayesian model through optimization (Jordan et al., 1999; Wainwright et al., 2008). Blei et al. (2017) provide an interesting review of VI from a statistical perspective, including some guidance on when to use MCMC or VI. For example, one may apply VI to large datasets and scenarios where the interest is to develop probabilistic models. In contrast, one may apply MCMC to small datasets for more precise samples but with a higher computational cost. In Bayesian inference, our goal is to find the posterior density, denoted by $p(\cdot|y)$, where y corresponds to the observed data. One can apply Bayes' theorem to find the posterior, but this might not be easy if there are many parameters and non-conjugate prior distributions. Therefore, one can aim to find an approximation to the posterior. To be specific, one wants to find q^* coming from a family of possible densities Q to approximate $p(\cdot|y)$, which can be solved in terms of an optimization problem with criterion f as follows:

$$q^* = \underset{q \in Q}{\operatorname{argmin}} f(q(\cdot), p(\cdot|y)).$$

The criterion f measures the closeness between the possible densities q in the family Q and the exact posterior density p. When we consider the Kullback-Leibler (KL) divergence

(Kullback and Leibler, 1951) as criterion f, i.e.,

$$q^* = \underset{q \in Q}{\operatorname{argmin}} \, \mathrm{KL}(q(\cdot) \| p(\cdot|y)), \tag{2.1}$$

this optimization-based technique to approximate the posterior density is called Variational Bayes (VB). Jordan et al. (1999) and Blei et al. (2017) show that minimizing the KL divergence is equivalent to maximizing the so-called evidence lower bound (ELBO). Let θ be a set of latent model variables, the KL divergence is defined as

$$\mathrm{KL}(q(\cdot) \| p(\cdot|y)) := \int q(\theta) \log \frac{q(\theta)}{p(\theta|y)} d\theta,$$

and it can be shown that

$$\int q(\theta) \log \frac{q(\theta)}{p(\theta|y)} d\theta = \log p(y) - \int q(\theta) \log \frac{p(\theta, y)}{q(\theta)} d\theta,$$

where the last term is the ELBO. Since $\log p(y)$ is a constant with respect to $q(\theta)$, this changes the problem in (2.1) to

$$q^* = \underset{q \in Q}{\operatorname{argmax}} \, \mathrm{ELBO}(q). \tag{2.2}$$

We, therefore, derive a VB algorithm for clustering functional data. We consider the mean-field variational family in which the latent variables are mutually independent, and a distinct factor governs each of them in the variational density. Finally, we apply the coordinate ascent variational inference algorithm (Bishop, 2006) to solve the optimization problem in (2.2).

2.2.2 Assumptions and model settings

Let $\mathbf{Y}_i$, $\{i = 1, \ldots, N\}$, denote the observed data from N curves, and for each curve i there are n_i evaluation points, $t_{i1}, ..., t_{in_i}$, so that $\mathbf{Y}_i = (Y_i(t_{i1}), \ldots, Y_i(t_{in_i}))^T$. Let Z_i be a hidden variable taking values in $\{1, \ldots, K\}$ that determines which cluster $\mathbf{Y}_i$ belongs to. We assume $Z_1, \ldots, Z_N$ are independent and identically distributed with $P(Z_i = k) = \pi_k$, $k = 1, ..., K$, and $\sum_{k=1}^{K} \pi_k = 1$. For the ith curve from cluster k, there is a smooth function f_k evaluated at $\mathbf{t}_i = (t_{i1}, ..., t_{in_i})^T$ so that $f_k(\mathbf{t}_i) = (f_k(t_{i1}), \ldots, f_k(t_{in_i}))^T$. Given that $Z_i = k$, we consider two different models for $\mathbf{Y}_i$ based on the correlation structure of the errors. In Model 1, described in Section 2.2.2.1, we assume independent errors, and in Model 2, described in Section 2.2.2.2, we add a random intercept to induce a correlation between observations within each curve.

2.2.2.1 Model 1

Let us assume that

$$\mathbf{Y}_i \,|\, (Z_i = k) = f_k(\mathbf{t}_i) + \sigma_k \boldsymbol{\epsilon_i}, \tag{2.3}$$

with conditionally independent errors $\boldsymbol{\epsilon}_1, ..., \boldsymbol{\epsilon}_N$, where $\boldsymbol{\epsilon}_i = (\epsilon_{i1}, ..., \epsilon_{in_i})$ and $\boldsymbol{\epsilon_i} \sim MVN(\mathbf{0}, \mathrm{I}_{n_i})$, $i = 1, ..., N$, where I_{n_i} is an identity matrix of size n_i and MVN represents the multivariate normal distribution. The functions $f_1, \ldots, f_K$ can be written as a linear combination of M known B-spline basis functions, that is, $f_k(t_{ij}) = \sum_{m=1}^{M} B_m(t_{ij})\phi_{km}$, $j = 1, ..., n_i$, such that $f_k(\mathbf{t}_i) = \mathbf{B}_{i(n_i \times M)} \boldsymbol{\phi}_{k(M \times 1)}$, $i = 1, ..., N, k = 1, ..., K$, $\mathbf{B}_i$ is an $n_i \times M$ matrix for the ith curve whose each entry (j, m) is the mth basis function evaluated at t_{ij}, $B_m(t_{ij})$, and $\boldsymbol{\phi}_k$ is the basis coefficient vector for cluster k. Therefore,

$$\mathbf{Y}_i \,|\, (Z_i = k) \sim MVN(\mathbf{B}_i \boldsymbol{\phi}_k, \sigma_k^2 \mathrm{I}_{n_i}), \; i = 1, ..., N, \; k = 1, ..., K.$$

The proposed model is within the framework of a mixture of linear models, also known as the finite regression mixture model (Chamroukhi and Nguyen, 2019). The finite regression mixture model offers a statistical framework for characterizing complex data from various unknown classes of conditional probability distributions (Peel and MacLahlan, 2000; Melnykov and Maitra, 2010; Chamroukhi, 2016a; Grün, 2019; Fruhwirth-Schnatter et al., 2019; McLachlan et al., 2019; Rigon, 2023). In our model, we specifically consider Gaussian regression mixtures to deal with functional data that originate from a finite number of groups and are represented through a linear combination of B-spline basis functions plus some Gaussian random noise (Chamroukhi, 2016b). Our model aligns with the classical finite Gaussian regression mixture model of order K, which can be expressed as follows:

$$f(\mathbf{Y}_i|\mathbf{B}_i;\boldsymbol{\phi}_1,...,\boldsymbol{\phi}_K,\sigma_1^2,...,\sigma_K^2) = \sum_{k=1}^{K} \pi_k \; g(\mathbf{Y}_i;\mathbf{B}_i\boldsymbol{\phi}_k,\sigma_k^2\mathbf{I}_{n_i}),$$

where g is the density function of a $MVN(\mathbf{B}_i\boldsymbol{\phi}_k,\sigma_k^2\mathbf{I}_{n_i})$.

In our proposed models, we employ B-spline basis functions to represent and smooth functional data. However, it is worth noting that alternative basis systems, such as the Fourier bases, wavelets, and polynomial bases can also be considered for this purpose (Ramsay and Silverman, 2005). As discussed in Chamroukhi and Nguyen (2019), the B-spline basis system offers greater flexibility, allowing researchers to tailor their choice of B-spline order and the number of knots to suit their specific needs. For smoothing functional data, cubic B-splines, corresponding to an order of four, are sufficient and can provide satisfactory performance (Chamroukhi and Nguyen, 2019). As in previous studies of functional data, we use cubic B-splines with equally spaced knots and assume that the number of basis functions M is predefined and known (Dias et al., 2009, 2015; Lenzi et al., 2017; Franco et al., 2023).

Let $\mathbf{Z} = (Z_1,\ldots,Z_N)^T$, $\boldsymbol{\phi} = \{\boldsymbol{\phi}_1,\ldots,\boldsymbol{\phi}_K\}$, $\boldsymbol{\pi} = (\pi_1,\ldots,\pi_K)^T$ and $\boldsymbol{\tau} = (\tau_1,\ldots,\tau_K)^T$, where $\tau_k = 1/\sigma_k^2$ is the precision parameter. We take on a Bayesian approach to infer $\boldsymbol{Z}$, $\boldsymbol{\phi}$, $\boldsymbol{\pi}$ and

$\boldsymbol{\tau}$, and assume the following marginal prior distributions for parameters in Model 1:

- $\boldsymbol{\pi} \sim$ Dirichlet($\mathbf{d}^0$) where $\mathbf{d}^0$ is the parameter vector for a Dirichlet distribution;
- $Z_i|\boldsymbol{\pi} \sim$ Categorical($\boldsymbol{\pi}$);
- $\boldsymbol{\phi}_k \sim MVN(\mathbf{m}_k^0, s^0\mathbf{I})$ with precision $v^0 = 1/s^0$ and $\mathbf{I}$ an $M \times M$ identity matrix;
- $\tau_k = 1/\sigma_k^2 \sim$ Gamma(a^0, r^0), $k = 1, ..., K$.

We develop a novel VB algorithm which, for given data, approximates the posterior distribution by finding the variational distribution (VD), $q(\mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau})$, with smallest KL divergence to the posterior distribution $p(\mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau}|\mathbf{Y})$. Minimizing the KL divergence is equivalent to maximizing the ELBO given by

$$\text{ELBO}(q) = \mathbb{E}[\log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau})] - \mathbb{E}[\log q(\mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau})]. \tag{2.4}$$

where $\log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau})$ is the complete data log-likelihood.

2.2.2.2 Model 2

We extend Model 1 by adding a curve-specific random intercept a_i which induces correlation among observations within each curve. The model now becomes:

$$Y_{ij} \,|\, (Z_i = k) = a_i + f_k(t_{ij}) + \sigma_k \epsilon_{ij}, \tag{2.5}$$

where $\epsilon_{ij} \sim N(0, 1)$ and $a_i \sim N(0, \sigma_a^2)$ with a_i and ϵ_{ij} independent for all i and j. We can write Model 2 in a vector form as

$$\mathbf{Y}_i \,|\, (Z_i = k) = a_i \mathbf{1}_{n_i} + f_k(\mathbf{t}_i) + \sigma_k \boldsymbol{\epsilon_i},\ i = 1, 2, ..., N,$$

in which $\mathbf{1}_{n_i}$ is a column vector of length n_i with all elements equal to 1, and further assume that $\boldsymbol{\epsilon_i} \sim MVN(\mathbf{0}, \mathrm{I}_{n_i})$ and $a_i \sim N(0, \sigma_a^2)$. This model can be rewritten as a two-step

model:

$$\mathbf{Y}_i \,|\, (Z_i = k, a_i) \sim MVN(\mathbf{B}_i\boldsymbol{\phi}_k + a_i\mathbf{1}_{n_i}, \sigma_k^2\mathrm{I}_{n_i}),$$

and $a_i \sim N(0, \sigma_a^2), i = 1, 2, ..., N$. Let $\mathbf{a} = (a_1, \ldots, a_N)^T$ and $\tau_a = 1/\sigma_a^2$. We assume the following marginal prior distributions for parameters in Model 2:

- $\boldsymbol{\pi} \sim$ Dirichlet($\mathbf{d}^0$);
- $Z_i|\boldsymbol{\pi} \sim$ Categorical($\boldsymbol{\pi}$);
- $\boldsymbol{\phi}_k \sim MVN(\mathbf{m}_k^0, s^0\mathbf{I})$ with precision $v^0 = 1/s^0$;
- $\tau_k = 1/\sigma_k^2 \sim$ Gamma(b^0, r^0), $k = 1, ..., K$;
- $\tau_a = 1/\sigma_a^2 \sim$ Gamma(α^0, β^0);
- $a_i|\tau_a \sim N(0, \sigma_a^2)$ with $\tau_a = 1/\sigma_a^2$.

As in Model 1, we develop a VB algorithm to infer $\boldsymbol{Z}$, $\boldsymbol{\phi}$, $\boldsymbol{\pi}$, $\boldsymbol{\tau}$, $\mathbf{a}$ and τ_a. The ELBO under Model 2 is given by

$$\mathrm{ELBO}(q) = \mathbb{E}_{q^*}[\log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a}, \tau_a)] - \mathbb{E}_{q^*}[\log q(\mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a}, \tau_a)]. \tag{2.6}$$

2.2.3 Steps of the VB algorithm

This section describes the main steps of the VB algorithm under Model 2 for inferring $\boldsymbol{Z}$, $\boldsymbol{\phi}$, $\boldsymbol{\pi}$, $\boldsymbol{\tau}$, $\mathbf{a}$ and τ_a. The proposed VB is summarized in Algorithm 1. The VB algorithm's main steps and the ELBO calculation for Model 1 can be found in Appendix A.

First, we assume that the variational distribution belongs to the mean-field variational family, where $\boldsymbol{Z}$, $\boldsymbol{\phi}$, $\boldsymbol{\pi}$ $\boldsymbol{\tau}$, $\mathbf{a}$ and τ_a are mutually independent and each governed by a distinct

factor in the variational density, that is:

$$\begin{aligned} q(\mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a}, \tau_a) &= \prod_{i=1}^{N} q(Z_i) \times \prod_{k=1}^{K} q(\boldsymbol{\phi}_k) \times \prod_{k=1}^{K} q(\tau_k) \\ &\quad \times q(\boldsymbol{\pi}) \times \prod_{i=1}^{N} q(a_i) \times q(\tau_a). \end{aligned} \tag{2.7}$$

We then derive a coordinate ascent algorithm to obtain the VD (Jordan et al., 1999; Blei et al., 2017). That is, we derive an update equation for each term in the factorization (2.7) by calculating the expectation of $\log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a}, \tau_a)$ (the joint distribution of the observed data $\mathbf{Y}$, hidden variables $\mathbf{Z}$ and parameters $\boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a}, \tau_a$, which is also called complete-data log-likelihood) over the VD of all random variables except the one of interest, where

$$\begin{aligned} \log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a}, \tau_a) &= \log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a}) \; + \; \log p(\mathbf{Z}|\boldsymbol{\pi}) + \\ &\quad \log p(\boldsymbol{\phi}) + \log p(\boldsymbol{\tau}) \; + \; \log p(\boldsymbol{\pi}) \; + \\ &\quad \log p(\mathbf{a}|\tau_a) \; + \; \log p(\tau_a). \end{aligned} \tag{2.8}$$

So, for example, the optimal update equation for $q(\boldsymbol{\pi})$, $q^*(\boldsymbol{\pi})$, is given by calculating

$$\log q^*(\boldsymbol{\pi}) = \mathbb{E}_{-\boldsymbol{\pi}}\big(\log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a}, \tau_a)\big) \; + \; \text{constant},$$

where $-\boldsymbol{\pi}$ indicates that the expectation is taken with respect to the VD of all other latent variables but $\boldsymbol{\pi}$, i.e., $\mathbf{Z}, \boldsymbol{\phi}$, $\boldsymbol{\tau}$, $\mathbf{a}$ and τ_a. In what follows we derive the update equation for each component in our model. For convenience, we use $\overset{+}{\approx}$ to denote equality up to a constant additive factor.

2.2.3.1 VB update equations

i) Update equation for $q(\boldsymbol{\pi})$

Since only the second term, $\log p(\mathbf{Z}|\boldsymbol{\pi})$, and the fifth term, $\log p(\boldsymbol{\pi})$, in (2.8) depend on $\boldsymbol{\pi}$, the update equation $q^*(\boldsymbol{\pi})$ can be derived as follows.

$$\begin{aligned}
\log q^*(\boldsymbol{\pi}) &\overset{+}{\approx} \mathbb{E}_{-\boldsymbol{\pi}}\big(\log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a}, \tau_a)\big) \\
&\overset{+}{\approx} \mathbb{E}_{-\boldsymbol{\pi}}\big(\log p(\mathbf{Z}|\boldsymbol{\pi})\big) \;+\; \mathbb{E}_{-\boldsymbol{\pi}}\big(\log p(\boldsymbol{\pi})\big) \\
&= \mathbb{E}_{-\boldsymbol{\pi}}\Big[\sum_{i=1}^{N}\sum_{k=1}^{K} \mathrm{I}(Z_i = k)\log \pi_k\Big] + \log p(\boldsymbol{\pi}) \\
&\overset{+}{\approx} \sum_{k=1}^{K}\log \pi_k\Big[\sum_{i=1}^{N}\mathbb{E}_{q^*(Z_i)}(\mathrm{I}(Z_i = k))\Big] + \sum_{k=1}^{K}[d_k^0 - 1]\log \pi_k \\
&= \sum_{k=1}^{K}\log \pi_k\Big[\Big(\sum_{i=1}^{N}\mathbb{E}_{q^*(Z_i)}(\mathrm{I}(Z_i = k)) + d_k^0\Big) - 1\Big].
\end{aligned}$$

Therefore, $q^*(\boldsymbol{\pi})$ is a Dirichlet distribution with parameters $\mathbf{d}^* = (d_1^*, \ldots, d_K^*)$, where

$$d_k^* = d_k^0 + \sum_{i=1}^{N}\mathbb{E}_{q^*(Z_i)}(\mathrm{I}(Z_i = k)). \tag{2.9}$$

ii) Update equation for $q(Z_i)$

$$\begin{aligned}
\log q^*(Z_i) &\overset{+}{\approx} \mathbb{E}_{-Z_i}\big(\log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a}, \tau_a)\big) \\
&\overset{+}{\approx} \mathbb{E}_{-Z_i}\big(\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a})\big) \;+\; \mathbb{E}_{-Z_i}\big(\log p(\mathbf{Z}|\boldsymbol{\pi})\big).
\end{aligned} \tag{2.10}$$

Note that we can write $\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a})$ and $\log p(\mathbf{Z}|\boldsymbol{\pi})$ into two parts, one that depends on Z_i and one that does not, that is:

$$\begin{aligned}
\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a}) &= \sum_{k=1}^{K}\mathrm{I}(Z_i = k)\log p(\mathbf{Y}_i|Z_i = k, \boldsymbol{\phi}_k, \tau_k, a_i) \\
&\quad + \sum_{l:l\neq i}\sum_{k=1}^{K}\mathrm{I}(Z_l = k)\log p(\mathbf{Y}_l|Z_l = k, \boldsymbol{\phi}_k, \tau_k, a_l),
\end{aligned}$$

$$\log p(\mathbf{Z}|\boldsymbol{\pi}) \quad = \quad \sum_{k=1}^{K} \mathrm{I}(Z_i = k) \log \pi_k + \sum_{l:l\neq i} \sum_{k=1}^{K} \mathrm{I}(Z_l = k) \log \pi_k.$$

Now when taking the expectation in (2.10), the parts that do not depend on Z_i in $\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a})$ and $\log p(\mathbf{Z}|\boldsymbol{\pi})$ will be added as a constant in the expectation. So, we obtain

$$\begin{aligned}\log q^*(Z_i) \quad \overset{+}{\approx} \quad & \sum_{k=1}^{K} \mathrm{I}(Z_i = k)\Big\{\frac{n_i}{2}\mathbb{E}_{q^*(\tau_k)}(\log \tau_k) \\ & -\frac{1}{2}\mathbb{E}_{q^*(\tau_k)}(\tau_k)\mathbb{E}_{q^*(\boldsymbol{\phi}_k)\cdot q^*(a_i)}[(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})^T(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})] \\ & + \mathbb{E}_{q^*(\boldsymbol{\pi})}(\log \pi_k)\Big\}.\end{aligned}$$

Therefore, $q^*(Z_i)$ is a categorical distribution with parameters

$$p^*_{ik} = \frac{e^{\alpha_{ik}}}{\sum_{k=1}^{K} e^{\alpha_{ik}}}, \tag{2.11}$$

where

$$\begin{aligned}\alpha_{ik} \quad = \quad & \frac{n_i}{2}\mathbb{E}_{q^*(\tau_k)}(\log \tau_k) \\ & -\frac{1}{2}\mathbb{E}_{q^*(\tau_k)}(\tau_k)\mathbb{E}_{q^*(\boldsymbol{\phi}_k)q^*(a_i)}[(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})^T(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})] \\ & + \mathbb{E}_{q^*(\boldsymbol{\pi})}(\log \pi_k).\end{aligned}$$

iii) Update equation for $q(\boldsymbol{\phi}_k)$

Only the first term, $\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a})$, and the third term, $\log p(\boldsymbol{\phi})$, in (2.8) depend on $\boldsymbol{\phi}_k$. In addition, similarly to the previous case for $q^*(Z_i)$, we can write $\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a})$ and $\log p(\boldsymbol{\phi})$ in two parts, one that depends on $\boldsymbol{\phi}_k$ and the other that does not. Therefore, we obtain

$$
\begin{aligned}
\log q^*(\boldsymbol{\phi}_k) &\overset{+}{\approx} \mathbb{E}_{-\boldsymbol{\phi}_k}(\log p(\mathbf{Y}|\mathbf{Z},\boldsymbol{\phi},\boldsymbol{\tau},\mathbf{a})) + \mathbb{E}_{-\boldsymbol{\phi}_k}\log p(\boldsymbol{\phi}) \\
&\overset{+}{\approx} \mathbb{E}_{q^*(\tau_k)}(\log\tau_k)\sum_{i=1}^{N}\frac{n_i}{2}\mathbb{E}_{q^*(Z_i)}[\mathrm{I}(Z_i=k)] \\
&\quad -\frac{1}{2}\mathbb{E}_{q^*(\tau_k)}(\tau_k)\sum_{i=1}^{N}\Big\{\mathbb{E}_{q^*(Z_i)}[\mathrm{I}(Z_i=k)] \\
&\quad \times\mathbb{E}_{q^*(a_i)}[(\mathbf{Y}_i-\mathbf{B}_i\boldsymbol{\phi}_k-a_i\mathbf{1}_{n_i})^T(\mathbf{Y}_i-\mathbf{B}_i\boldsymbol{\phi}_k-a_i\mathbf{1}_{n_i})]\Big\} \qquad (2.12)\\
&\quad +\frac{M}{2}\log v^0 \;-\; \frac{1}{2}v^0(\boldsymbol{\phi}_k-\mathbf{m}_k^0)^T(\boldsymbol{\phi}_k-\mathbf{m}_k^0). \qquad (2.13)
\end{aligned}
$$

All expectations are defined in Section 2.2.3.2, but note that, for example, $\mathbb{E}_{q^*(Z_i)}[\mathrm{I}(Z_i = k)] = p^*_{ik}$ and

$$
\begin{aligned}
&\mathbb{E}_{q^*(a_i)}[(\mathbf{Y}_i-\mathbf{B}_i\boldsymbol{\phi}_k-a_i\mathbf{1}_{n_i})^T(\mathbf{Y}_i-\mathbf{B}_i\boldsymbol{\phi}_k-a_i\mathbf{1}_{n_i})] \\
\overset{+}{\approx}\;& (\mathbf{Y}_i \;-\; \mathbf{B}_i\boldsymbol{\phi}_k-\mu^*_{a_i}\mathbf{1}_{n_i})^T(\mathbf{Y}_i-\mathbf{B}_i\boldsymbol{\phi}_k-\mu^*_{a_i}\mathbf{1}_{n_i}),
\end{aligned}
$$

where $\mu^*_{a_i}$ is the posterior mean of $q^*(a_i)$ which is derived later. We focus on the quadratic forms that appear in (2.12) and (2.13). Let $\mathbf{Y}^*_i = \mathbf{Y}_i - \mu^*_{a_i}\mathbf{1}_{n_i}$, we can write:

$$
\begin{aligned}
\log q^*(\boldsymbol{\phi}_k) &\overset{+}{\approx} -\frac{1}{2}\mathbb{E}_{q^*(\tau_k)}(\tau_k)\sum_{i=1}^{N}p^*_{ik}(\mathbf{Y}^*_i \;-\; \mathbf{B}_i\boldsymbol{\phi}_k)^T(\mathbf{Y}^*_i-\mathbf{B}_i\boldsymbol{\phi}_k) - \frac{1}{2}v^0(\boldsymbol{\phi}_k-\mathbf{m}_k^0)^T(\boldsymbol{\phi}_k-\mathbf{m}_k^0) \\
&= -\frac{1}{2}\mathbb{E}_{q^*(\tau_k)}(\tau_k)\sum_{i=1}^{N}p^*_{ik}[\mathbf{Y}^{*T}_i\mathbf{Y}^*_i - 2\mathbf{Y}^{*T}_i\mathbf{B}_i\boldsymbol{\phi}_k + \boldsymbol{\phi}_k^T\mathbf{B}_i^T\mathbf{B}_i\boldsymbol{\phi}_k] \\
&\quad -\frac{1}{2}v^0[\boldsymbol{\phi}_k^T\boldsymbol{\phi}_k - 2(\mathbf{m}_k^0)^T\boldsymbol{\phi}_k + (\mathbf{m}_k^0)^T\mathbf{m}_k^0] \\
&\overset{+}{\approx} -\frac{1}{2}\boldsymbol{\phi}_k^T\Big[v^0\mathbf{I} + \mathbb{E}_{q^*(\tau_k)}(\tau_k)\sum_{i=1}^{N}p^*_{ik}\mathbf{B}_i^T\mathbf{B}_i\Big]\boldsymbol{\phi}_k \\
&\quad +\Big[v^0(\mathbf{m}_k^0)^T + \mathbb{E}_{q^*(\tau_k)}(\tau_k)\sum_{i=1}^{N}p^*_{ik}\mathbf{Y}^{*T}_i\mathbf{B}_i\Big]\boldsymbol{\phi}_k. \qquad (2.14)
\end{aligned}
$$

Now let

$$\boldsymbol{\Sigma}_k^* = \Big[v^0\mathbf{I} + \mathbb{E}_{q^*(\tau_k)}(\tau_k)\sum_{i=1}^{N} p_{ik}^*\mathbf{B}_i^T\mathbf{B}_i\Big]^{-1}. \tag{2.15}$$

We can then rewrite (2.14) as

$$-\frac{1}{2}\boldsymbol{\phi}_k^T\boldsymbol{\Sigma}_k^{*-1}\boldsymbol{\phi}_k - \frac{1}{2}(-2)\Big[v^0(\mathbf{m}_k^0)^T + \mathbb{E}_{q^*(\tau_k)}(\tau_k)\sum_{i=1}^{N} p_{ik}^*\mathbf{Y}_i^{*T}\mathbf{B}_i\Big]\boldsymbol{\Sigma}_k^*\boldsymbol{\Sigma}_k^{*-1}\boldsymbol{\phi}_k.$$

Therefore, $q^*(\boldsymbol{\phi}_k)$ is $MVN(\mathbf{m}_k^*, \boldsymbol{\Sigma}_k^*)$ with $\boldsymbol{\Sigma}_k^*$ as in (2.15) and mean vector

$$\mathbf{m}_k^* = [v^0(\mathbf{m}_k^0)^T + \mathbb{E}_{q^*(\tau_k)}(\tau_k)\sum_{i=1}^{N} p_{ik}^*\mathbf{Y}_i^{*T}\mathbf{B}_i]\boldsymbol{\Sigma}_k^*. \tag{2.16}$$

iv) Update equation for $q(\tau_k)$

Similarly to the calculations in *iii)* we can write

$$\begin{aligned}\log q^*(\tau_k) \;\overset{+}{\approx}\; & \log\tau_k\sum_{i=1}^{N}\frac{n_i}{2}p_{ik}^* \\ & -\frac{1}{2}\tau_k\sum_{i=1}^{N} p_{ik}^*\mathbb{E}_{q^*(\boldsymbol{\phi}_k)\cdot q^*(a_i)}[(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})^T(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})] \\ & + (b^0 - 1)\log\tau_k - r^0\tau_k.\end{aligned}$$

Therefore, $q^*(\tau_k)$ is a Gamma distribution with parameters

$$A_k^* = b^0 + \sum_{i=1}^{N}\frac{n_i}{2}p_{ik}^*, \tag{2.17}$$

and

$$R_k^* = r^0 + \frac{1}{2}\sum_{i=1}^{N}\Big\{p_{ik}^* \times \mathbb{E}_{q^*(\boldsymbol{\phi}_k)\cdot q^*(a_i)}[(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})^T(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})]\Big\}. \tag{2.18}$$

v) Update equation for $q(a_i)$

$$
\begin{aligned}
\log q^*(a_i) &\overset{+}{\approx} \mathbb{E}_{-a_i}(\log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a}, \tau_a)) \\
&\overset{+}{\approx} \mathbb{E}_{-a_i}(\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a})) \; + \; \mathbb{E}_{-a_i}(\log p(\mathbf{a}|\tau_a)) \\
&\overset{+}{\approx} \mathbb{E}_{-a_i}[\sum_{k=1}^{K} \mathrm{I}(Z_i = k) \log p(\mathbf{Y}_i|Z_i = k, \boldsymbol{\phi}_k, \tau_k, a_i)] + \mathbb{E}_{-a_i}[\sum_{k=1}^{K} \mathrm{I}(Z_i = k) \log p(a_i|\tau_a)] \\
&\overset{+}{\approx} \sum_{k=1}^{K} p^*_{ik}\{\frac{n_i}{2}\mathbb{E}_{q^*(\tau_k)} \log \tau_k \\
&\quad -\frac{1}{2}\mathbb{E}_{q^*(\tau_k)}\tau_k \mathbb{E}_{q^*(\boldsymbol{\phi}_k)}[(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})^T(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})] \\
&\quad -\frac{1}{2}a_i^2\mathbb{E}_{q^*(\tau_a)}\tau_a\} \\
&\overset{+}{\approx} \sum_{k=1}^{K} p^*_{ik}\{-\frac{1}{2}\mathbb{E}_{q^*(\tau_k)}\tau_k[(\mathbf{Y}_i - \mathbf{B}_i\mathbf{m}^*_k - a_i\mathbf{1}_{n_i})^T(\mathbf{Y}_i - \mathbf{B}_i\mathbf{m}^*_k - a_i\mathbf{1}_{n_i})] - \frac{1}{2}a_i^2\mathbb{E}_{q^*(\tau_a)}\tau_a\}.
\end{aligned}
$$

Let $\mathbf{Y}^*_{ik} = \mathbf{Y}_i - \mathbf{B}_i\mathbf{m}^*_k$, then

$$
\begin{aligned}
\log q^*(a_i) &\overset{+}{\approx} \sum_{k=1}^{K} p^*_{ik}\{-\frac{1}{2}\mathbb{E}_{q^*(\tau_k)}\tau_k[(\mathbf{Y}^*_{ik} - a_i\mathbf{1}_{n_i})^T(\mathbf{Y}^*_{ik} - a_i\mathbf{1}_{n_i})] - \frac{1}{2}a_i^2\mathbb{E}_{q^*(\tau_a)}\tau_a\} \\
&\overset{+}{\approx} -\frac{n_i}{2}a_i^2\sum_{k=1}^{K} p^*_{ik}\mathbb{E}_{q^*(\tau_k)}\tau_k + a_i\sum_{k=1}^{K} p^*_{ik}\mathbb{E}_{q^*(\tau_k)}\tau_k\mathbf{1}^T_{n_i}\mathbf{Y}^*_{ik} - \frac{1}{2}a_i^2\mathbb{E}_{q^*(\tau_a)}\tau_a \\
&= -\frac{1}{2}a_i^2[n_i\sum_{k=1}^{K} p^*_{ik}\mathbb{E}_{q^*(\tau_k)}\tau_k + \mathbb{E}_{q^*(\tau_a)}\tau_a] + a_i\sum_{k=1}^{K} p^*_{ik}\mathbb{E}_{q^*(\tau_k)}\tau_k\mathbf{1}^T_{n_i}\mathbf{Y}^*_{ik}.
\end{aligned}
$$

Let

$$
\sigma^{2*}_{a_i} = (n_i\sum_{k=1}^{K} p^*_{ik}\mathbb{E}_{q^*(\tau_k)}\tau_k + \mathbb{E}_{q^*(\tau_a)}\tau_a)^{-1}, \tag{2.19}
$$

and

$$
\mu^*_{a_i} = \sigma^{2*}_{a_i}\sum_{k=1}^{K} p^*_{ik}\mathbb{E}_{q^*(\tau_k)}\tau_k\mathbf{1}^T_{n_i}\mathbf{Y}^*_{ik}. \tag{2.20}
$$

Then $q^*(a_i)$ is $N(\mu^*_{a_i}, \sigma^{*2}_{a_i})$.

vi) Update equation for $q(\tau_a)$

$$\begin{aligned}
\log q^*(\tau_a) &\overset{+}{\approx} \mathbb{E}_{-\tau_a}(\log p(\mathbf{a}|\tau_a) + \log p(\tau_a)) \\
&\overset{+}{\approx} \mathbb{E}_{-\tau_a}\Big(\sum_{i=1}^{N} \log p(a_i|\tau_a)\Big) + (\alpha^0 - 1)\log\tau_a - \beta^0\tau_a \\
&\overset{+}{\approx} \frac{N}{2}\log\tau_a - \frac{1}{2}\tau_a \sum_{i=1}^{N} \mathbb{E}_{q^*(a_i)} a_i^2 + (\alpha^0 - 1)\log\tau_a - \beta^0\tau_a \\
&= (\alpha^0 + \frac{N}{2} - 1)\log\tau_a - \Big(\beta^0 + \frac{1}{2}\sum_{i=1}^{N} \mathbb{E}_{q^*(a_i)} a_i^2\Big)\tau_a.
\end{aligned}$$

Let

$$\alpha^* = \alpha^0 + \frac{N}{2},$$

and

$$\beta^* = \beta^0 + \frac{1}{2}\sum_{i=1}^{N} \mathbb{E}_{q^*(a_i)} a_i^2. \tag{2.21}$$

$q^*(\tau_a)$ is Gamma(α^*, β^*).

2.2.3.2 Expectations

In this section, we calculate the expectations in the update equations for each component in the VD.

Let $\mathbf{\Psi}$ be the digamma function defined as

$$\mathbf{\Psi}(x) = \frac{d}{dx}\log\Gamma(x), \tag{2.22}$$

which can be easily calculated via numerical approximation. The values of the expectations taken with respect to the approximated distributions are given as follows.

$$\mathbb{E}_{q^*(Z_i)}[\mathrm{I}(Z_i = k)] = p_{ik}^*, \tag{2.23}$$

$$\mathbb{E}_{q^*(\tau_k)}(\tau_k) = \frac{A_k^*}{R_k^*}, \tag{2.24}$$

$$\mathbb{E}_{q^*(\tau_k)}(\log \tau_k) = \boldsymbol{\Psi}(A_k^*) - \log R_k^*, \tag{2.25}$$

$$\mathbb{E}_{q^*(\boldsymbol{\pi})}(\log \pi_k) = \boldsymbol{\Psi}(d_k^*) - \boldsymbol{\Psi}\Big(\sum_{k=1}^{K} d_k^*\Big), \tag{2.26}$$

$$\mathbb{E}_{q^*(\tau_a)}(\tau_a) = \frac{\alpha^*}{\beta^*}, \tag{2.27}$$

$$\mathbb{E}_{q^*(\tau_a)}(\log \tau_a) = \boldsymbol{\Psi}(\alpha^*) - \log \beta^*, \tag{2.28}$$

$$\mathbb{E}_{q^*(a_i)} a_i^2 = \sigma_{a_i}^{*2} + \mu_{a_i}^{*2}. \tag{2.29}$$

In addition, using the fact that $\mathbb{E}(\mathbf{X}^T\mathbf{X}) = \text{trace}[\text{Var}(\mathbf{X})] + \mathbb{E}(\mathbf{X})^T\mathbb{E}(\mathbf{X})$, we obtain

$$\begin{aligned} & \mathbb{E}_{q^*(\boldsymbol{\phi}_k)}[(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})^T(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})] \\ = \;& \text{trace}(\mathbf{B}_i\boldsymbol{\Sigma}_k^*\mathbf{B}_i^T) + (\mathbf{Y}_i - \mathbf{B}_i\mathbf{m}_k^* - a_i\mathbf{1}_{n_i})^T(\mathbf{Y}_i - \mathbf{B}_i\mathbf{m}_k^* - a_i\mathbf{1}_{n_i}), \end{aligned} \tag{2.30}$$

and

$$\begin{aligned} & \mathbb{E}_{q^*(\boldsymbol{\phi}_k)\cdot q^*(a_i)}[(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})^T(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})] \\ = \;& \mathbb{E}_{q^*(a_i)}\Big[\mathbb{E}_{q^*(\boldsymbol{\phi}_k)}[(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})^T(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})]\Big] \\ = \;& \mathbb{E}_{q^*(a_i)}\Big[\text{trace}(\mathbf{B}_i\boldsymbol{\Sigma}_k^*\mathbf{B}_i^T) + (\mathbf{Y}_i - \mathbf{B}_i\mathbf{m}_k^* - a_i\mathbf{1}_{n_i})^T(\mathbf{Y}_i - \mathbf{B}_i\mathbf{m}_k^* - a_i\mathbf{1}_{n_i})\Big] \\ = \;& \text{trace}(\mathbf{B}_i\boldsymbol{\Sigma}_k^*\mathbf{B}_i^T) + n_i\sigma_{a_i}^{*2} + (\mathbf{Y}_i - \mathbf{B}_i\mathbf{m}_k^* - \mu_{a_i}^*\mathbf{1}_{n_i})^T(\mathbf{Y}_i - \mathbf{B}_i\mathbf{m}_k^* - \mu_{a_i}^*\mathbf{1}_{n_i}). \end{aligned} \tag{2.31}$$

2.2.4 ELBO calculation

In this section, we show how to calculate the ELBO under Model 2, which is the convergence criterion of our proposed VB algorithm and is updated at the end of each iteration until convergence. Equation (2.6) gives the ELBO:

$$\text{ELBO}(q) = \mathbb{E}_{q^*}[\log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a}, \tau_a)] - \mathbb{E}_{q^*}[\log q(\mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a}, \tau_a)],$$

where

$$\begin{aligned}\mathbb{E}_{q^*}[\log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a}, \tau_a)] &= \mathbb{E}_{q^*}[\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a})] + \mathbb{E}_{q^*}[\log p(\mathbf{Z}|\boldsymbol{\pi})] \\ &\quad + \mathbb{E}_{q^*}[\log p(\boldsymbol{\phi})] + \mathbb{E}_{q^*}[\log p(\boldsymbol{\tau})] \\ &\quad + \mathbb{E}_{q^*}[\log p(\boldsymbol{\phi})] + \mathbb{E}_{q^*}[\log p(\mathbf{a}|\tau_a)] \\ &\quad + \mathbb{E}_{q^*}[\log p(\tau_a)],\end{aligned}$$

and

$$\begin{aligned}\mathbb{E}_{q^*}[\log q(\mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a}, \tau_a)] &= \mathbb{E}_{q^*}[\log q(\mathbf{Z})] + \mathbb{E}_{q^*}[\log q(\boldsymbol{\phi})] + \mathbb{E}_{q^*}[\log q(\boldsymbol{\pi})] \\ &\quad + \mathbb{E}_{q^*}[\log q(\boldsymbol{\tau})] + \mathbb{E}_{q^*}[\log q(\mathbf{a})] + \mathbb{E}_{q^*}[\log q(\tau_a)].\end{aligned}$$

Therefore, we can write the ELBO as the summation of 7 terms:

$$\begin{aligned}\text{ELBO}(q) &= \mathbb{E}_{q^*}[\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{a})] + diff_{\mathbf{Z}} + diff_{\boldsymbol{\phi}} \\ &\quad + diff_{\boldsymbol{\tau}} + diff_{\boldsymbol{\pi}} + diff_{\mathbf{a}} + diff_{\tau_a},\end{aligned} \tag{2.32}$$

where,

$$diff_{\mathbf{Z}} = \mathbb{E}_{q^*}[\log p(\mathbf{Z}|\boldsymbol{\pi})] - \mathbb{E}_{q^*}[\log q(\mathbf{Z})].$$

Specifically,

$$diff_{\mathbf{Z}} = \sum_{i=1}^{N}\sum_{k=1}^{K} p_{ik}^{*}\mathbb{E}_{q^{*}(\boldsymbol{\pi})}(\log \pi_k) - \sum_{i=1}^{N}\sum_{k=1}^{K} p_{ik}^{*}\log p_{ik}^{*}. \tag{2.33}$$

The other terms in (2.32) are calculated as follows:

$$diff_{\boldsymbol{\phi}} = -\frac{1}{2}\sum_{k=1}^{K} v_k^0\{\text{trace}(\boldsymbol{\Sigma}_k^{*}) + (\mathbf{m}_k^{*} - \mathbf{m}_k^0)^T(\mathbf{m}_k^{*} - \mathbf{m}_k^0)\} + \frac{1}{2}\sum_{k=1}^{K}\log|\boldsymbol{\Sigma}_k^{*}|,$$

$$\begin{aligned} diff_{\boldsymbol{\tau}} &= \sum_{k=1}^{K}\{(b^0-1)\mathbb{E}_{q^{*}(\tau_k)}(\log\tau_k) - r^0\mathbb{E}_{q^{*}(\tau_k)}(\tau_k)\} \\ &\quad - \sum_{k=1}^{K}\{A_k^{*}\log R_k^{*} - \log\Gamma(A_k^{*}) \\ &\quad + (A_k^{*}-1)\mathbb{E}_{q^{*}(\tau_k)}(\log\tau_k) - R_k^{*}\mathbb{E}_{q^{*}(\tau_k)}(\tau_k)\}, \end{aligned} \tag{2.34}$$

$$diff_{\boldsymbol{\pi}} \equiv \sum_{k=1}^{K}(d_k^0 - d_k^{*})\mathbb{E}_{q^{*}(\boldsymbol{\pi})}(\log\pi_k),$$

$$diff_{\mathbf{a}} = -\frac{1}{2}\mathbb{E}_{q^{*}(\tau_a)}\tau_a\sum_{i=1}^{N}\mathbb{E}_{q^{*}(a_i)}a_i^2 + \sum_{i=1}^{N}\log\sigma_{a_i}^{*},$$

$$\begin{aligned} diff_{\tau_a} &= (\alpha^0-1)\mathbb{E}_{q^{*}(\tau_a)}(\log\tau_a) - \beta^0\mathbb{E}_{q^{*}(\tau_a)}\tau_a \\ &\quad - \alpha^{*}\log\beta^{*} - (\alpha^{*}-1)\mathbb{E}_{q^{*}(\tau_a)}(\log\tau_a) + \beta^{*}\mathbb{E}_{q^{*}(\tau_a)}\tau_a \\ &= (\alpha^0-\alpha^{*})\mathbb{E}_{q^{*}(\tau_a)}(\log\tau_a) - (\beta^0-\beta^{*})\mathbb{E}_{q^{*}(\tau_a)}\tau_a - \alpha^{*}\log\beta^{*}, \end{aligned}$$

and

$$\begin{aligned} \mathbb{E}_{q^{*}}[\log p(\mathbf{Y}|\mathbf{Z},\boldsymbol{\phi},\boldsymbol{\tau},\mathbf{a})] &= \sum_{i=1}^{N}\sum_{k=1}^{K} p_{ik}^{*}\Big\{\frac{n_i}{2}\mathbb{E}_{q^{*}(\tau_k)}(\log\tau_k) \\ &\quad -\frac{1}{2}\frac{A_k^{*}}{R_k^{*}}\mathbb{E}_{q^{*}(\boldsymbol{\phi}_k)\cdot q^{*}(a_i)}[(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})^T(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k - a_i\mathbf{1}_{n_i})]\Big\}. \end{aligned}$$

Therefore, at iteration c, we calculate $\text{ELBO}^{(c)}$ using all parameters obtained at the end of iteration c. Convergence of the algorithm is achieved if $\text{ELBO}^{(c)} - \text{ELBO}^{(c-1)}$ is smaller than a given threshold. It is important to note that we use the fact that $\lim_{p^*_{ik}\to 0} p^*_{ik} \log p^*_{ik} = 0$ to avoid numerical issues when calculating (2.33). Numerical issues also exist in calculating the term $\{A^*_k \log R^*_k - \log\Gamma(A^*_k) + (A^*_k - 1)\mathbb{E}_{q^*(\tau_k)}(\log\tau_k) - R^*_k\mathbb{E}_{q^*(\tau_k)}(\tau_k)\}$ in (2.34), so we will approximate it by the following digamma and log-gamma approximations. Note that we use (2.24) and (2.25) for $\mathbb{E}_{q^*(\tau_k)}(\tau_k)$ and $\mathbb{E}_{q^*(\tau_k)}(\log\tau_k)$, respectively.

(1) digamma approximation based on asymptotic expansion:

$$\boldsymbol{\Psi}(A^*_k) \approx \log A^*_k - 1/(2A^*_k).$$

(2) log-gamma Stirling's series approximation:

$$\log\Gamma(A^*_k) \approx A^*_k \log(A^*_k) - A^*_k - \frac{1}{2}\log(A^*_k).$$

Therefore, plugging in these two approximations, we obtain

$$\begin{aligned}
& A^*_k \log R^*_k - \log\Gamma(A^*_k) + (A^*_k - 1)\mathbb{E}_{q^*(\tau_k)}(\log\tau_k) - R^*_k\mathbb{E}_{q^*(\tau_k)}(\tau_k) \\
= \;& A^*_k \log R^*_k - \log\Gamma(A^*_k) + (A^*_k - 1)(\boldsymbol{\Psi}(A^*_k) - \log R^*_k) - R^*_k\frac{A^*_k}{R^*_k} \\
\approx \;& \frac{1}{2}\log A^*_k + \frac{1}{2A^*_k} - \frac{1}{2} \\
\overset{+}{\approx} \;& \frac{1}{2}\log A^*_k + \frac{1}{2A^*_k} = \frac{1}{2}(\log A^*_k + \frac{1}{A^*_k}).
\end{aligned}$$

Algorithm 1: Clustering functional data via variational inference with random intercepts

Data: N original curves with n_i evaluation points for the ith curve and the $\mathbf{B}_i$ matrix containing the evaluation values of the basis functions, $i = 1, ..., N$; number of clusters K; values of hyperparameters: $\mathbf{d}^0$, $\mathbf{m}_k^0, k = 1, ..., K$, s^0, b^0, r^0, α^0, β^0; convergence threshold and maximum number of iterations

Result: VB estimated mean curves for each cluster and the cluster index for each original curve

Initialization: initialize R_k^*, μ_a^* and β^* with arbitrary values (e.g., $R_k^* = r^0$, $\mu_a^* = 0$, $\beta^* = \beta^0$) and p_{ik}^* from k-means, and set $c = 0$;

while $c <$ *maximum number of iterations and difference of ELBO > convergence threshold* **do**

  $\alpha^* = \alpha^0 + \frac{N}{2}$;

  repeat

    $c = c + 1$;

    update $A_k^{*(c)}$ using $p_{1k}^{*(c-1)}, \ldots, p_{Nk}^{*(c-1)}$ with equation (2.17);

    update $\mathbf{\Sigma}_k^{*(c)}$ using $A_k^{*(c)}$, $R_k^{*(c-1)}$ and $p_{1k}^{*(c-1)}, \ldots, p_{Nk}^{*(c-1)}$ with equations (2.15) and (2.24);

    update $\mathbf{m}_k^{*(c)}$ using $\mathbf{\Sigma}_k^{*(c)}$, $A_k^{*(c)}$, $R_k^{*(c-1)}$, $\mu_a^{*(c-1)}$ and $p_{1k}^{*(c-1)}, \ldots, p_{Nk}^{*(c-1)}$ with equations (2.16) and (2.24);

    update $\sigma_{a_i}^{*2(c)}$ using $A_k^{*(c)}$, $R_k^{*(c-1)}$, α^*, $\beta^{*(c-1)}$ and $p_{ik}^{*(c-1)}, \ldots, p_{iK}^{*(c-1)}$ with equations (2.19), (2.24) and (2.27) ;

    update $\mu_{a_i}^{*(c)}$ using $\sigma_{a_i}^{*2(c)}$, $A_k^{*(c)}$, $R_k^{*(c-1)}$ and $p_{ik}^{*(c-1)}, \ldots, p_{iK}^{*(c-1)}$ with equations (2.20) and (2.24);

    update $R_k^{*(c)}$ using $\mathbf{m}_k^{*(c)}$, $\mathbf{\Sigma}_k^{*(c)}$, $\sigma_{a_i}^{*2(c)}$, $\mu_{a_i}^{*(c)}$ and $p_{1k}^{*(c-1)}, \ldots, p_{Nk}^{*(c-1)}$ with equations (2.18) and (2.31);

    update $\beta^{*(c)}$ using $\sigma_{a_i}^{*2(c)}$ and $\mu_{a_i}^{*(c)}$ with equations (2.21) and (2.29);

    update $\boldsymbol{d}^{*(c)}$ using $p_{1k}^{*(c-1)}, \ldots, p_{Nk}^{*(c-1)}$ with equations (2.9) and (2.23);

    update $p_{1k}^{*(c)}, \ldots, p_{Nk}^{*(c)}$ using $A_k^{*(c)}$, $R_k^{*(c)}$, $\boldsymbol{d}^{*(c)}$, $\sigma_{a_i}^{*2(c)}$, $\mu_{a_i}^{*(c)}$, $\mathbf{m}_k^{*(c)}$ and $\mathbf{\Sigma}_k^{*(c)}$ with equations (2.11), (2.24), (2.25), (2.26) and (2.31);

    calculate the current ELBO, $\text{ELBO}^{(c)}$ using equation (2.32) ;

    calculate difference of ELBO = $\text{ELBO}^{(c)} - \text{ELBO}^{(c-1)}$;

  until *maximum iteration is achieved or the ELBO converges*;

end

2.3 Simulation studies

In Section 2.3.1, we present the metrics used to evaluate the performance our proposed methodology. Sections 2.3.2 and 2.3.3 present the simulation scenarios and results for

Model 1 and Model 2, respectively.

2.3.1 Performance metrics

We evaluate the clustering performance of our proposed algorithm by two metrics: mismatches (Zambom et al., 2019) and V-measure (Rosenberg and Hirschberg, 2007). Mismatch rate is the proportion of subjects misclassified by the clustering procedure. In our case, each subject corresponds to a curve in our functional dataset. V-measure, a score between zero and one, evaluates the subject-to-cluster assignments and indicates the homogeneity and completeness of a clustering procedure result. Homogeneity is satisfied if the clustering procedure assigns only those subjects that are members of a single group to a single cluster. Completeness is symmetrical to homogeneity, and it is satisfied if all those subjects that are members of a single group are assigned to a single cluster. The V-measure is one when all subjects are assigned to their correct groups by the clustering procedure. One may also consider alternative metrics to evaluate clustering performance, such as the Rand index (Rand, 1971) and the mutual information (Cover, 1999). The Rand index measures the similarity between two data partitions by counting the number of pairs of observations that are either correctly grouped together (i.e., true positives) or correctly separated (i.e., true negatives) in both partitions. Mutual information, on the other hand, quantifies the information shared between two data partitions. Along with the V-measure, these metrics are commonly used for clustering and partition evaluation, but they each have different mathematical formulations and emphasize different aspects of clustering performance.

For comparison purposes, we also investigate the performance, in terms of mismatch and V-measure, of the classical clustering algorithms including k-means for raw data (discrete observed points), and k-means for functional data (referred to as functional k-means, Febrero-Bande and de la Fuente (2012)), and two other model-based algorithms: funFEM (Bouveyron et al., 2015) and SaS-Funclust (Centofanti et al., 2023). The funFEM method was

proposed for the inference of the discriminative functional mixture model to cluster functional data via the EM algorithm. The SaS-Funclust method, short for sparse and smooth functional clustering, was developed to facilitate sparse clustering for functional data via a functional Gaussian mixture model and penalized maximum likelihood estimation.

To further evaluate the performance of the proposed VB algorithm in terms of the estimated mean curves, we calculate the empirical mean integrated squared error (EMISE) for each cluster in each simulation scenario. For simplicity, we generate curves with equal number of observed values, that is n, in our simulation study. The EMISE is obtained as follows:

$$\text{EMISE}_k = \frac{T}{n} \sum_{j=1}^{n} \text{EMSE}_k(t_j), \tag{2.35}$$

where T is the curve evaluation interval length, n is total number of observed evaluation points, and the empirical mean squared error (EMSE) at point t_j for cluster k, $\text{EMSE}_k(t_j)$, is given by

$$\text{EMSE}_k(t_j) = \frac{1}{S} \sum_{s=1}^{S} [f_k(t_j) - \hat{f}_k^s(t_j)]^2,$$

in which s corresponds to the sth simulated dataset among S datasets in total, $f_k(t_j)$ is the value of the true mean function in cluster k evaluated at point t_j and $\hat{f}_k^s(t_j)$ is its corresponding estimated value for the sth simulated dataset. The estimated value $\hat{f}_k^s(t_j)$ is calculated using the B-spline basis expansion with coefficients corresponding the to posterior mean (2.16) obtained at the convergence of the VB algorithm.

2.3.2 Simulation study on Model 1

In Sections 2.3.2.1 and 2.3.2.2, we first conduct simulation studies for Model 1 which comprises six different scenarios, five of which have three clusters ($K = 3$) while the last scenario has four clusters ($K = 4$). For each simulation scenario, we generate 50

datasets and apply the proposed VB algorithm to each dataset, considering the number of basis functions to be six except for Scenario 5, which uses 12 basis functions. The ELBO convergence threshold is 0.01, with a maximum of 100 iterations. We use the clustering results of k-means to initialize p^*_{ik} in our VB algorithm.

We further conduct simulation studies on Model 1 to investigate the performance of the VB algorithm, including a prior sensitivity analysis in Section 2.3.2.3, choice of the number of clusters in Section 2.3.2.4 and misspecification of the type of basis functions in Section 2.3.2.5. We compare the posterior estimation results from VB to the ones from MCMC in Section 2.3.2.6.

2.3.2.1 Simulation scenarios

Scenarios 1 and 2 are adopted from Zambom et al. (2019). Each dataset is generated from 3 possible clusters ($k = 1, 2, 3$) with $N = 50$ curves per cluster. For each curve, we assume there are $n = 100$ observed values across a grid of equally spaced points in the interval $[0, \pi/3]$.

Scenario 1, $K = 3$***:***

$$Y_{ik}(t_j) = a_i + b_k + c_k \sin(1.3t_j) + t_j^3 + \delta_{ij}; i = 1, ..., 50; j = 1, ..., 100; k = 1, 2, 3,$$

where $Y_{ik}(t_j)$ denotes the value at point t_j of the ith curve from cluster k, $a_i \sim U(-1/4, 1/4)$, $\delta_{ij} \sim N(0, 0.4^2)$, $b_1 = 0.3$, $b_2 = 1$, $b_3 = 0.2$, $c_1 = 1/1.3$, $c_2 = 1/1.2$, and $c_3 = 1/4$.

Scenario 2, $K = 3$***:***

$$Y_{ik}(t_j) = a_i + b_k \exp(c_k t_j) - t_j^3 + \delta_{ij}; i = 1, ..., 50; j = 1, ..., 100; k = 1, 2, 3,$$

where $Y_{ik}(t_j)$ denotes the value at point t_j of the ith curve from cluster k, $a_i \sim U(-1/4, 1/4)$,

$\delta_{ij} \sim N(0, 0.3^2)$, $b_1 = 1/1.8$, $b_2 = 1/1.7$, $b_3 = 1/1.5$, $c_1 = 1.1$, $c_2 = 1.4$, and $c_3 = 1.5$.

In Scenarios 3 and 4, each dataset is also generated considering three clusters ($k = 1, 2, 3$) with 50 curves each. The mean curve of the functional data in each cluster is generated from a pre-specified linear combination of B-spline basis functions. The number of basis functions is the same across clusters but the coefficients of the linear combination are different, one set per cluster (see Table 2.1). We apply the function *create.bspline.basis* in the R package *fda* to generate six B-spline basis functions of order 4, $B_l(\cdot)$, $l = 1, ..., 6$, evaluated on equally spaced points, t_j, $j = 1, ..., 100$, in the interval $[0, 1]$.

Scenarios 3 and 4, $K = 3$***:***

$$Y_{ik}(t_j) = \sum_{l=1}^{6} B_l(t_j)\phi_{kl} + \delta_{ij}; i = 1, ..., 50; j = 1, ..., 100; k = 1, 2, 3,$$

where $Y_{ik}(t_j)$ denotes the value at point t_j of the ith curve from cluster k and $\delta_{ij} \sim N(0, 0.4^2)$. Table 2.1 presents the vector of coefficients for each cluster k, $\boldsymbol{\phi}_k = (\phi_{k1}, \ldots, \phi_{k6})^T$, used in Scenarios 3 and 4. Figure 2.1 illustrates the true mean curves for the three clusters and their corresponding basis functions for Scenarios 3 and 4.

Table 2.1: Coefficient vectors of six B-spline basis functions for each cluster in Scenarios 3 and 4

	Scenario 3						Scenario 4					
$\boldsymbol{\phi}_k$	ϕ_{k1}	ϕ_{k2}	ϕ_{k3}	ϕ_{k4}	ϕ_{k5}	ϕ_{k6}	ϕ_{k1}	ϕ_{k2}	ϕ_{k3}	ϕ_{k4}	ϕ_{k5}	ϕ_{k6}
$k = 1$	1.5	1	1.8	2	1	1.5	1.5	1	1.6	1.8	1	1.5
$k = 2$	2.8	1.4	1.8	0.5	1.5	2.5	1.8	0.6	0.4	2.6	2.8	1.6
$k = 3$	0.4	0.6	2.4	2.6	0.1	0.4	1.2	1.8	2.2	0.8	0.6	1.8

Scenario 5 ($K = 3$) is based on one of the simulation scenarios used in Dias et al. (2009) in which the curves mimic the energy consumption of different types of consumers in Brazil. There are 50 curves per cluster and for each curve we generate 96 points based on equally spaced time points, t_j, $j = 1, ..., 96$ in the interval $[0, 24]$ (corresponding to one observation

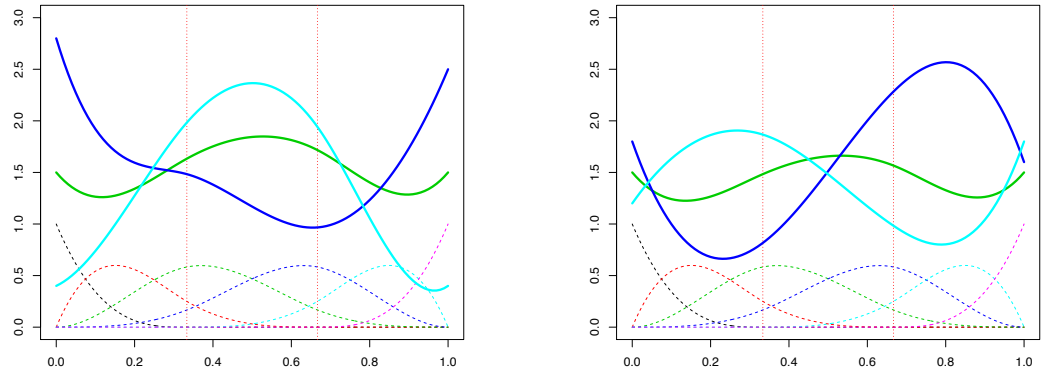


Figure 2.1: Cluster true mean curves (solid curves) and their corresponding six B-splines basis functions (dashed curves) for simulation scenarios 3 (left) and 4 (right).

every 15 minutes over a 24-hour period).

Scenario 5, $K = 3$***:***

$$\begin{aligned} Y_{i1}(t_j) &= 0.1(0.4 + \exp(-(t_j - 6)^2/3) \\ &\quad + 0.2\exp(-(t_j - 12)^2/25) \\ &\quad + 0.5\exp(-(t_j - 19)^2/4)) + \delta_{ij}, \end{aligned}$$

$$\begin{aligned} Y_{i2}(t_j) &= 0.1(0.2 + \exp(-(t_j - 5)^2/4) \\ &\quad + 0.25\exp(-(t_j - 18)^2/5)) + \delta_{ij}, \end{aligned}$$

$$\begin{aligned} Y_{i3}(t_j) &= 0.1(0.2 + \exp(-(t_j - 3)^2/4) \\ &\quad + 0.25\exp(-(t_j - 16)^2/5)) + \delta_{ij}, \end{aligned}$$

where $Y_{ik}(t_j)$ denotes the value at time t_j of the ith curve from cluster k, $i = 1, ..., 50$, $j = 1, ..., 96$, $k = 1, 2, 3$, and $\delta_{ij} \sim N(0, 0.012^2)$.

Scenario 6 also corresponds to one of the simulation scenarios considered by Zambom et al. (2019), where there are $K = 4$ clusters with 50 curves each. Each curve has 100 observed values based on equally spaced points, t_j, $j = 1, ..., 100$, in the interval $[0, \pi/3]$.

Scenario 6, $K = 4$***:***

$$Y_{ik}(t_j) = a_i + b_k - \sin(c_k \pi t_j) + t_j^3 + \delta_{ij}; i = 1, ..., 50; j = 1, ..., 100; k = 1, 2, 3, 4,$$

where $Y_{ik}(t_j)$ denotes the value at point t_j of the ith curve from cluster k, $a_i \sim U(-1/3, 1/3)$, $\delta_{ij} \sim N(0, 0.4^2)$, $b_1 = 0.2$, $b_2 = 0.5$, $b_3 = 0.7$, $b_4 = 1.3$, $c_1 = 1.1$, $c_2 = 1.4$, $c_3 = 1.6$ and $c_4 = 1.8$.

2.3.2.2 Simulation results for Model 1

Figure 2.2 shows the raw curves (color-coded by cluster) from one of the 50 generated datasets for each simulation scenario. In addition, the true mean curves ($f_k(\mathbf{t}), k = 1, \ldots, K$) and the estimated smoothed mean curves ($\hat{f}_k(\mathbf{t}) = \mathbf{B}\mathbf{m}_k^*, k = 1, \ldots, K$) are shown in black and red, respectively. We can observe that the true and estimated mean curves almost coincide within each cluster in all scenarios.

Table 2.2 displays the mean and standard deviation of mismatch rates (M) and V-measure values (V) across 50 simulated datasets for each scenario. For the sake of completeness, we have included the results from Scenario 7 in Section 2.3.2.4 and Scenario 8 in Section 2.3.2.5 in Table 2.2 as they pertain to the study of Model 1. The proposed VB algorithm performs the best in all scenarios except for Scenario 5 where we simulate the curves that mimic daily energy consumption. Across Scenarios 1 to 6, VB demonstrates impressive results with a mean mismatch rate of 5.13% and a mean V-measure of 88.06%. Notably, the mean mismatch rate achieved by VB is 55.71%, 83.6%, 85.86%, and 73.41% lower than that of classical k-means, functional k-means, funFEM, and SaS-Funclust, respectively. Meanwhile, VB's mean V-measure surpasses the compared methods by 5.36%, 38.75%,

Table 2.2: Simulation results for Model 1. Mismatch rate and V-measure values for each simulation scenario.

	VB		*k*-means		functional *k*-means		funFEM		SaS-Funclust	
Scenario	M[1] (sd[2])	V[3] (sd)	M (sd)	V (sd)	M (sd)	V (sd)	M (sd)	V (sd)	M (sd)	V (sd)
1	0.0409 (0.0153)	0.8654 (0.0350)	0.0488 (0.0181)	0.8594 (0.0388)	0.2844 (0.1063)	0.6031 (0.1020)	0.5569 (0.0333)	0.0569 (0.0319)	0.0552 (0.0246)	0.8489 (0.0424)
2	0.1416 (0.0334)	0.6300 (0.0655)	0.1739 (0.0517)	0.6188 (0.0650)	0.3252 (0.0591)	0.4882 (0.0438)	0.4081 (0.1683)	0.2924 (0.2570)	0.2119 (0.0563)	0.5786 (0.0576)
3	0.0000 (0.0000)	1.0000 (0.0000)	0.1715 (0.2312)	0.8738 (0.1700)	0.3096 (0.0799)	0.5580 (0.0812)	0.1901 (0.2197)	0.7263 (0.3332)	0.3333 (0.0000)	0.7337 (0.0000)
4	0.0000 (0.0000)	1.0000 (0.0000)	0.0559 (0.1531)	0.9581 (0.1145)	0.3421 (0.1119)	0.6480 (0.0428)	0.1796 (0.2670)	0.7386 (0.4151)	0.0233 (0.0825)	0.9788 (0.0726)
5	0.0200 (0.0800)	0.9840 (0.0639)	0.1053 (0.2005)	0.9227 (0.1469)	0.0261 (0.0852)	0.9638 (0.1117)	0.1500 (0.2213)	0.8882 (0.1646)	0.0133 (0.0660)	0.9893 (0.0527)
6	0.1054 (0.0197)	0.8043 (0.0262)	0.1398 (0.0655)	0.7819 (0.0546)	0.5900 (0.1354)	0.5469 (0.1030)	0.6932 (0.0692)	0.1398 (0.0271)	0.5208 (0.1624)	0.7424 (0.0305)
7[4]	0.3001 (0.0944)	0.7528 (0.0592)	0.3002 (0.0946)	0.7497 (0.0608)	0.7761 (0.1120)	0.5525 (0.0694)	0.8183 (0.0279)	0.0504 (0.0146)	0.7167 (0.1300)	0.6179 (0.0249)
8[5]	0.0667 (0.1347)	0.9467 (0.1076)	0.0960 (0.1940)	0.9281 (0.1455)	0.2321 (0.1448)	0.6323 (0.1729)	0.5401 (0.1833)	0.1166 (0.2958)	0.6667 (0.0000)	0.0000 (0.0000)

[1]M: mean mismatch rate from 50 runs.

[2]sd: standard deviation.

[3]V: mean V-measure from 50 runs.

[4]Scenario 7 is in Section 2.3.2.4

[5]Scenario 8 is in Section 2.3.2.5

85.9%, and 8.46%, respectively. In Scenarios 3 and 4, where data is simulated through a linear combination of six predefined basis functions, VB exhibits perfect classification, with $M = 0$ and $V = 1$, which aligns with expectations since the raw data in these scenarios share the same structure as the proposed model. Comparatively, classical k-means generally outperforms functional k-means, funFEM, and SaS-Funclust in Scenarios 1, 2, 3, and 6, as similarly found in Zambom et al. (2019). The SaS-Funclust method excels in Scenario 5, with a slightly (0.0067) lower mismatch rate and a marginally (0.0053) higher V-measure than VB. Functional k-means also demonstrates competitive performance in Scenario 5, comparable to VB and SaS-Funclust.

In terms of computational efficiency, the run times for the proposed VB algorithm of Model 1 across the 50 simulated datasets from Scenarios 1 to 6 are as follows: 1.97 minutes, 5.41 minutes, 1.41 minutes, 1.61 minutes, 3.60 minutes, and 5.32 minutes. For comparison, SaS-Funclust required significantly longer computation times: 60.16 minutes, 68.94 minutes, 65.04 minutes, 68.19 minutes, 72.26 minutes, and 129.47 minutes for the respective scenarios. On average, the proposed VB algorithm demonstrates exceptional speed, being approximately 20 times faster than SaS-Funclust. The algorithm was implemented in R version 3.6.3 on a computer using the Mac OS X operating system with a 1.6 GHz processor and 8 GBytes of random access memory, same for the simulation study for Model 2 in Section 2.3.3.

Table 2.3 presents the EMISE for each cluster in each Scenario. We can observe small EMISE values, which are consistent with the results shown in Figure 2.2, where there is a small difference between the red curves (i.e., the estimated mean functions) and the black curves (i.e., the true mean functions). A plot of EMSE values versus observed points for each cluster in Scenario 1 is presented in Figure 2.3 while plots of EMSE values for Scenarios 2, 3, 4, 5 and 6 are provided in Figure B.1 in Appendix B.

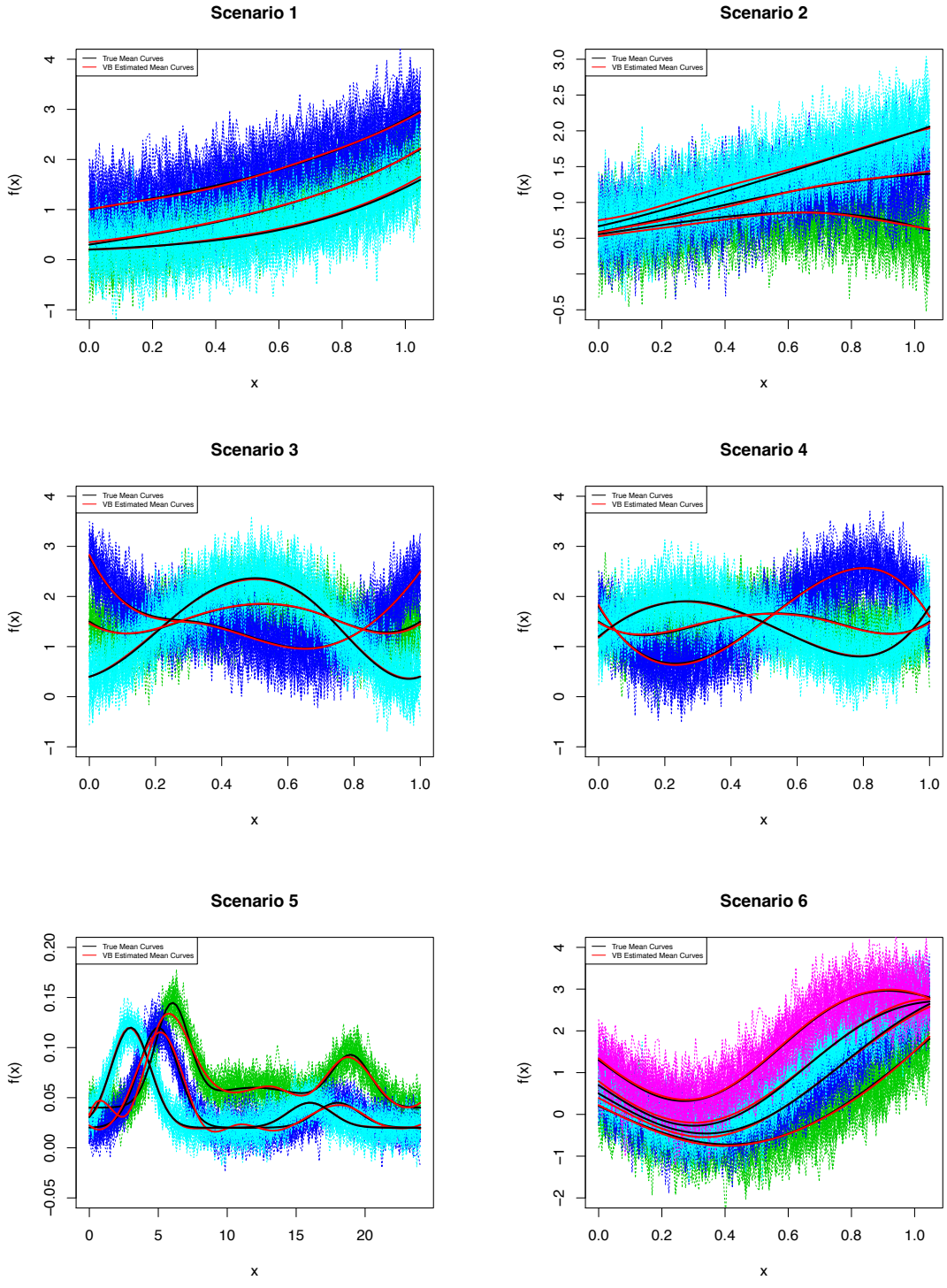


Figure 2.2: Simulation results for Model 1. Example of simulated data under each proposed scenario. Raw curves (different colors correspond to different clusters), cluster-specific true mean curves (in black) and corresponding estimated mean curves (in red).

Table 2.3: Simulation results for Model 1. The empirical mean integrated squared error (EMISE) for the estimated mean curve in each cluster in each scenario.

Scenario	Cluster	EMISE	Scenario	Cluster	EMISE
1	1	0.00096	2	1	0.00164
	2	0.00077		2	0.00246
	3	0.00080		3	0.00169
3	1	0.00031	4	1	0.00023
	2	0.00045		2	0.00034
	3	0.00042		3	0.00033
5	1	0.00001	6	1	0.00076
	2	0.00114		2	0.00419
	3	0.00022		3	0.00472
				4	0.00130

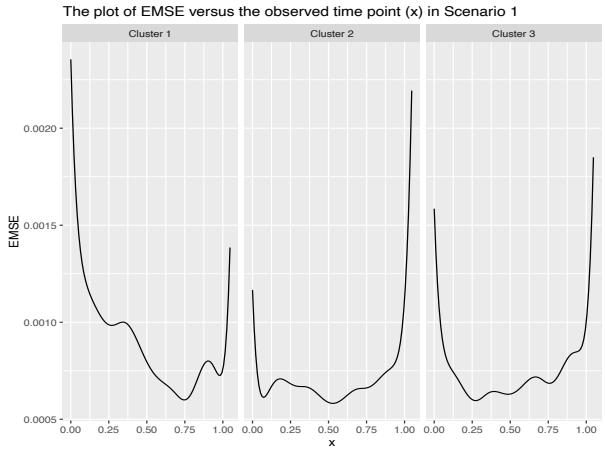


Figure 2.3: Simulation results for Model 1. Empirical mean squared error (EMSE) versus each evaluation point x for each cluster in Scenario 1.

2.3.2.3 Prior sensitivity analysis

In Bayesian analysis, it is important to assess the effects of different prior settings in the posterior estimation. In this section, we carry out a sensitivity analysis on how different prior settings may affect the results of our proposed VB algorithm. Our sensitivity analysis focuses on the prior distribution of the coefficients $\boldsymbol{\phi}_k$ of the B-spline basis expansion of each cluster-specific mean curve. We assume $\boldsymbol{\phi}_k$ follows a multivariate normal prior distribution with a mean vector $\mathbf{m}_k^0$ and $s^0\mathbf{I}$ as the covariance matrix. We simulated data according to Scenario 3 in Section 2.3.2.1 and four different prior settings as follows:

- Setting 1: use the true coefficients as the prior mean vector and consider a small variance ($s^0 = 0.01$).
- Setting 2: use the true coefficients as the prior mean vector but consider a larger variance than in Setting 1 ($s^0 = 1$).
- Setting 3: use a prior mean vector that is different than the true vector of coefficients with a small variance ($s^0 = 0.01$).
- Setting 4: set the prior mean vector of coefficients to a vector of zeros with a small variance ($s^0 = 0.01$).

Setting 1 has the strongest prior information among these four prior settings, while setting 4 is the most non-informative prior case. In setting 3, the prior mean vector of coefficients is generated from sampling from a multivariate normal distribution with a mean vector corresponding to the true coefficients and covariance matrix $\sigma^2\mathbf{I}$, with $\sigma^2 = 0.5$. For each prior setting, we simulate 50 datasets as in Scenario 3, obtaining the average mismatch rate and V-measure, which are displayed in Table 2.4. First, we can observe that all the curves are correctly clustered under Setting 1, which has the strongest prior information. Then, as we relax the prior assumptions in two possible directions (i.e., more considerable variance or less informative mean vector), the mismatch rate increases, and the V-measure decreases. However, the clustering performance does not decrease much, only 4.67% higher in mismatches and 3.73% lower in V-measure.

Table 2.4: Simulation results for Model 1. Mean mismatch rate and V-measure value from prior sensitivity analysis in Scenario 3

Setting	1	2	3	4
M[1]	0.0000	0.0067	0.0067	0.0467
V[2]	1.0000	0.9947	0.9947	0.9627

[a] M: mean mismatch rate from 50 runs.
[b] V: mean V-measure from 50 runs.

2.3.2.4 Choosing the number of clusters

Choosing an appropriate number of clusters, denoted as K, holds paramount importance within clustering procedures. This decision aligns with determining the number of mixture components in a regression mixture model. One of the most widely applied methodologies to deal with uncertainty in the cluster numbers is the two-fold scheme that one first fits the mixture model with different predefined numbers of mixtures and then use some information criteria to select the best one (Chen et al., 2012; Nieto-Barajas and Contreras-Cristán, 2014; Wang and Lin, 2022). Alternatively, one can explore concurrent approaches for optimal cluster number selection, including techniques such as overfitted Bayesian mixtures, tailored to address scenarios with large unknown K (Rousseau and Mengersen, 2011), selection through penalized maximum likelihood (Chamroukhi, 2016b), and the application of infinite mixture models such as Dirichlet process mixture models (Escobar and West, 1995; Ray and Mallick, 2006; Petrone et al., 2009; Rodríguez et al., 2009; Angelini et al., 2012; Heinzl and Tutz, 2013; Rigon, 2023).

In our study, we employ the afterward model selection (i.e., two-fold) scheme to determine the most suitable number of clusters. Assuming some prior knowledge of K, we establish a clustering model for a range of integers based on this prior information, employing the VB algorithm for each K. For model comparison, we utilize the deviance information criterion (DIC) (Spiegelhalter et al., 2002), which can be applied to select the optimal number of clusters within a comparable Bayesian clustering framework (Gao et al., 2011; Anderson et al., 2014; Komárek, 2009). DIC is built to balance the model fitness and complexity under a Bayesian framework, and a lower DIC indicates a better model. Nonetheless, the DIC is not an integral component of the core methodology and can be substituted with alternative model selection criteria such as the WAIC (Watanabe and Opper, 2010) and LPML (Geisser and Eddy, 1979) when someone's concern is predictive goodness-of-fit. In

our Model 1 setting, the DIC can be obtained as follows:

$$DIC = -4\mathbb{E}_{q^*}[\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau})] + 2\overline{D},$$

where $\mathbb{E}_{q^*}[\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau})]$ can be computed after the convergence of our proposed VB algorithm based on the ELBO. The term $\overline{D}$ corresponds to the log-likelihood $\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau})$ evaluated at the expected value of each parameter posterior. For example, when we calculate the term $\log \tau_k$ in $\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau})$, we replace it by $\log(\mathbb{E}_{q^*(\tau_k)}(\tau_k))$.

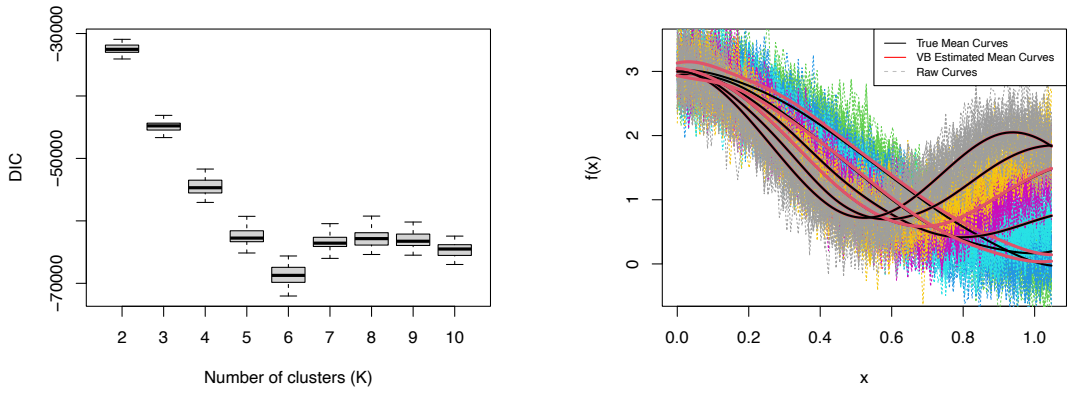


Figure 2.4: Simulation results for Model 1, Scenario 7, $K = 6$. Left: boxplots of DIC values under different $K \in \{1, 2, ..., 10\}$. The best number of clusters is six which has the smallest DIC. Right: the clustering results for $K = 6$ for one of the simulated data sets. Raw curves (different colors correspond to different clusters), cluster-specific true mean curves (in black) and corresponding VB estimated mean curves (in red).

We consider a more complex scenario, namely Scenario 7, where $K = 6$ in this simulation study which was also analyzed in Zambom et al. (2019). The data are generated as follows:

Scenario 7, $K = 6$***:***

$$Y_{ik}(t_j) = a_i + \cos(b_k \pi t_j) - t_j^2 + \delta_{ij}; i = 1, ..., 50; j = 1, ..., 100; k = 1, 2, ..., 6,$$

where $Y_{ik}(t_j)$ denotes the value at point t_j of the ith curve from cluster k, $a_i \sim U(-1/4, 1/4)$,

$\delta_{ij} \sim N(0, 0.3^2)$, $b_1 = 1$, $b_2 = 1.2$, $b_3 = 1.4$, $b_4 = 1.6$, $b_5 = 1.8$ and $b_6 = 2$.

We assume a prior information of the number of clusters that K is around 6. Accordingly, we evaluate a range of potential K values, specifically $\{2, 3, ..., 10\}$. For each K, we apply the VB algorithm to cluster the observed functional data and calculate the resulting DIC. Within this scope, for each $K \in \{2, 3, ..., 10\}$, we repeat the simulation analysis for 50 times utilizing different random seeds to generate data. The left plot in Figure 2.4 displays a boxplot representation of the DIC values for each K. It is evident that our DIC-based approach adeptly identifies the correct K (in this case, $K = 6$), yielding the lowest DIC. The accompanying right plot in Figure 2.4 showcases the clustering results for one of the simulated data sets under Scenario 7, demonstrating a highly satisfactory estimation of the true mean curves.

The quantitative evaluation of VB clustering performance in Scenario 7, along with a comparison to the other methods, is presented in Table 2.2. The VB algorithm performs the best among the others with a mean mismatch rate of 0.3001 and a mean V-measure of 0.7528. The mean mismatch rate of VB is 0.03%, 61.33%, 63.33%, and 58.13% lower than that of the classical k-means, functional k-means, funFEM and SaS-Funclust methods, while the mean V-measure is 0.41%, 36.25%, 1393.65%, and 21.83% higher, respectively. It is important to note that Scenario 7, characterized by a more complex structure with multiple groups of curves and overlapping patterns, poses a greater challenge for all methods, leading to overall reduced performance compared to other scenarios. FunFEM, in particular, encounters significant difficulties, with a V-measure approaching 0 due to the misclassification of more than 80% of curves.

2.3.2.5 Misspecification of the type of basis functions

This section illustrates the performance of the VB algorithm in case of misspecification of the type of basis functions via a simulation study, namely Scenario 8. We generate seven Fourier basis functions with equally spaced points on the interval $[0, 1]$, which are

shown in Figure 5(b), and simulate the data for three clusters ($k = 1, 2, 3$) with 50 curves ($i = 1, 2, ..., 50$) and 100 values (t_j, $j = 1, 2, ..., 100$) on each curve in each cluster using a linear combination of these Fourier basis functions as follows:

Scenario 8, $K = 3$***:***

$$Y_{ik}(t_j) = \sum_{l=1}^{7} G_l(t_j)\phi_{kl} + \delta_{ij}; i = 1, ..., 50; j = 1, ..., 100; k = 1, 2, 3,$$

where $Y_{ik}(t_j)$ denotes the value at point t_j for the ith curve from cluster k, $G_l(t_j)$ is the lth Fourier basis function evaluated at point t_j, ϕ_{kl} is the corresponding basis function coefficient, and $\delta_{ij} \sim N(0, 4)$. In this simulation study, the vectors of basis function coefficients for each cluster are:

$\boldsymbol{\phi}_1 = (\phi_{11}, \phi_{12}, ..., \phi_{17})^T = (0.75, 0.50, 0.90, 1.25, 0.90, 0.50, 0.40)^T$,

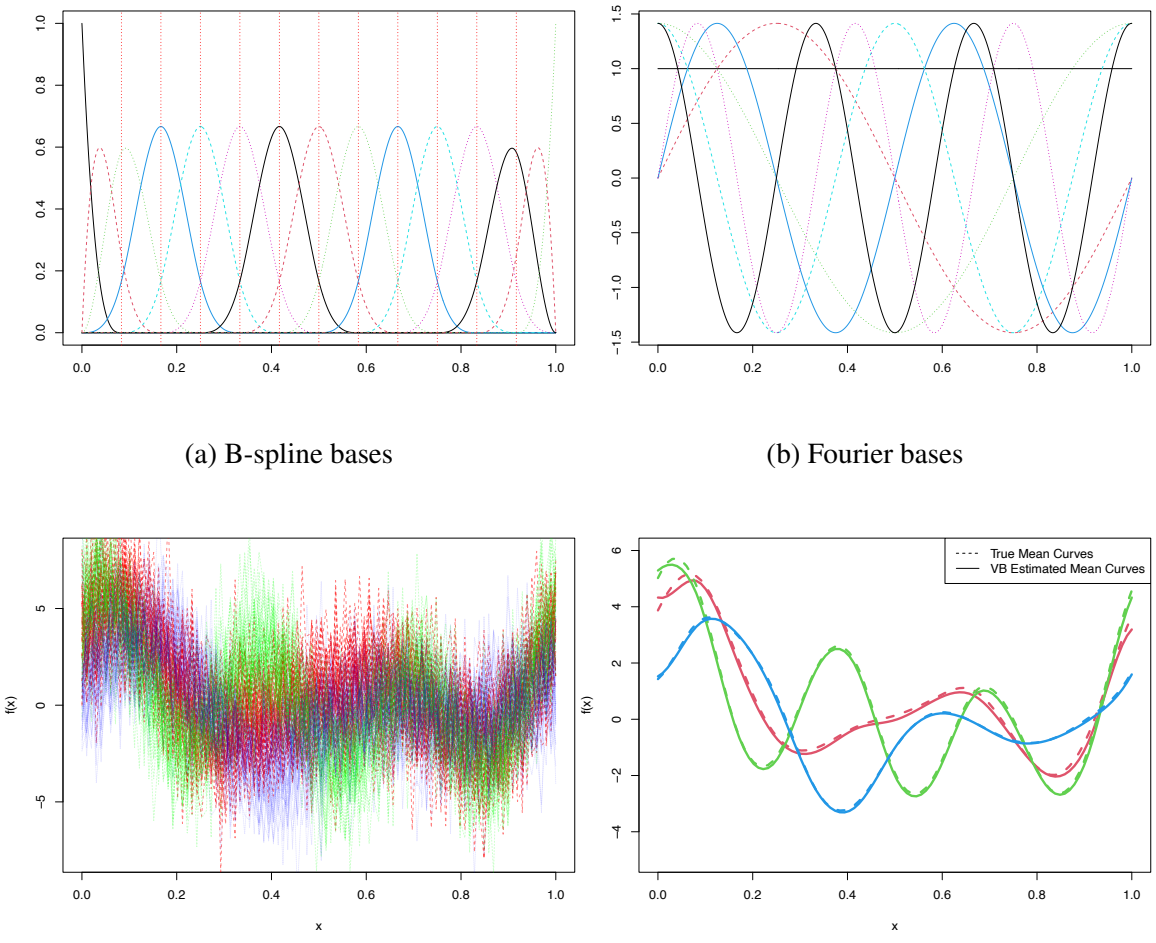
$\boldsymbol{\phi}_2 = (\phi_{21}, \phi_{22}, ..., \phi_{27})^T = (0.40, 0.70, 0.90, 0.25, 0.75, 1.25, 1.50)^T$, and

$\boldsymbol{\phi}_3 = (\phi_{31}, \phi_{32}, ..., \phi_{37})^T = (0.10, 0.30, 1.20, 1.30, 0.05, -0.20, -0.30)^T$.

Figure 2.5(c) presents the raw curves with each cluster distinguished by a unique color. Notably, when compared to the B-spline bases, the Fourier bases exhibit a more intricate curve structure, suggesting the potential need for an increased number of B-spline basis functions to adequately represent these functional curves, as observed in Sousa et al. (2023). Consequently, we have generated 15 B-spline bases from the interval $[0, 1]$, as illustrated in Figure 2.5(a), to cluster the curves derived from a linear combination of the Fourier bases. The resulting VB estimated mean curves (solid lines) are juxtaposed with the true mean curves (dashed lines) in Figure 2.5(d) from one of the simulated data sets.

While a minor discrepancy is observable between the true and estimated mean curves at the left boundary for the red and green groups, it is evident that the VB algorithm achieves highly accurate estimations of the true mean curves across all clusters. As shown in Table 2.2, the computed mean mismatch rate (sd) and mean V-measure (sd) from clustering 50 different simulated datasets are 0.067 (0.135) and 0.947 (0.108), respectively. In compari-

son to classical k-means, functional k-means, and funFEM, the mean mismatch rate from VB is 30.52%, 71.26%, and 87.65% lower, while the mean V-measure is 2%, 49.72%, and 711.92% higher. Unfortunately, SaS-Funclust struggles to cluster the curves, resulting in a V-measure of zero. This simulation illustrates the robustness of the VB algorithm in clustering functional data, even when confronted with the misspecification of basis function types.



(a) B-spline bases

(b) Fourier bases

(c) Raw curves

(d) True and estimated curves

Figure 2.5: Simulation results for Model 1, Scenario 8, $K = 3$. (a) B-spline basis functions for model fit. (b) Fourier basis functions for data generation. (c) Raw curves from three clusters (distinct colors for each cluster). (d) Cluster-specific true mean curves (dashed) and corresponding VB estimated mean curves (solid).

2.3.2.6 Comparison with MCMC posterior estimation

In our simulation study on Model 1, VB is shown to yield accurate mean curve estimates and satisfactory outcomes in clustering functional data. Although mean-field VB, as an alternative to MCMC, boasts a lower computational cost, it may potentially underestimate the posterior variance (Wang and Titterington, 2005). To investigate this concern in the context of clustering functional data through a B-spline regression mixture model, we employ the MCMC-based Gibbs sampling algorithm for simulated data under Scenario 1. The resulting posterior distribution from Gibbs is based on 9000 MCMC samples following a 1000-sample burn-in and with a thinning of 1 from one chain. The convergence of the MCMC algorithm was well assessed and checked by the trace plot. Figure 2.6 illustrates the marginal posterior density of each basis coefficient ϕ_{km}, $k = 1, 2, 3$, $m = 1, \ldots, 6$, and the precision parameter τ_k, $k = 1, 2, 3$, for each cluster, organized by columns. In each plot, the dashed red line represents the corresponding posterior density from VB, while the solid blue line is derived from MCMC. We observe a robust consistency in the estimated posterior distributions between MCMC and VB. A similar consistency between VB and MCMC in posterior estimation under a regression setting was found by Faes et al. (2011); Luts and Wand (2015); Xian et al. (2024).

To elucidate the uncertainty from the estimated mean curves, we utilize Scenarios 1 and 3 as illustrative examples. We construct 95% credible bands, both from MCMC and VB, for the true mean curves based on the posterior distribution of the B-spline coefficients. Figure 2.7 presents the results, with the first row corresponding to Scenario 1 and the second row to Scenario 3. In each plot, the solid colored lines depict the estimated mean curves from VB or MCMC, while the black solid lines represent the true mean curves. The 95% credible bands are shown as dashed lines, with different colors for different clusters. In Scenario 1, VB provides comparable point and interval estimation results with MCMC. In contrast, in Scenario 3, VB provides more accurate estimated mean curves, particularly at the left tails.

Importantly, we observed no substantial differences in the resulting credible bands between VB and MCMC. In terms of computational cost for one simulation, VB took 5.5 seconds to produce the results, while the Gibbs sampler took 2.9 minutes for Scenario 1. In Scenario 3, VB took 5.8 seconds, while MCMC took 2.6 minutes. Overall, VB was more than 20 times faster than MCMC.

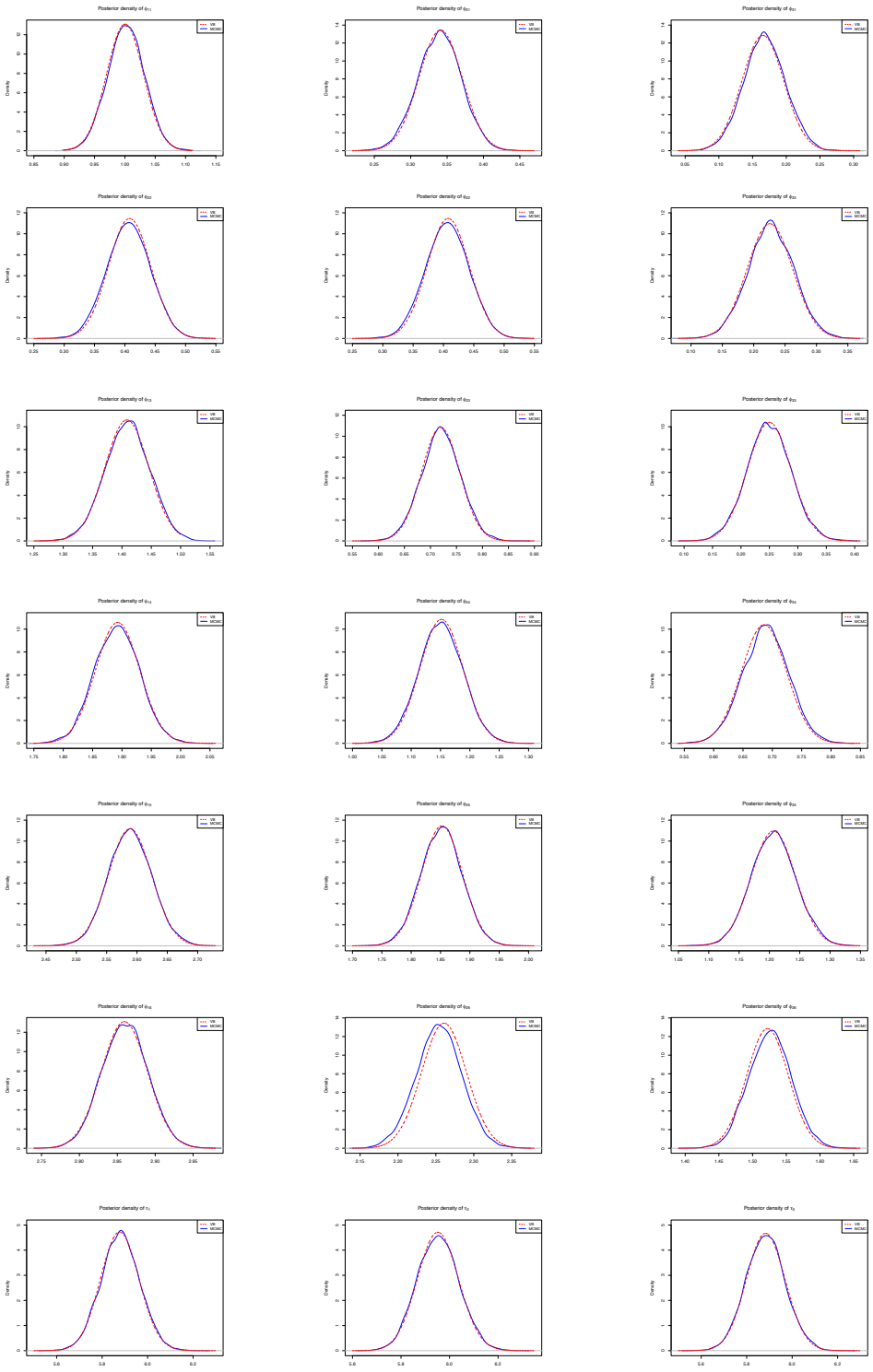


Figure 2.6: Simulation results for Model 1, Scenario 1, $K = 3$. Posterior distributions of the B-spline basis coefficients and the precision parameter for each cluster (one column for each cluster). In each plot, the dashed red line is from the VB algorithm and the solid blue line from MCMC.

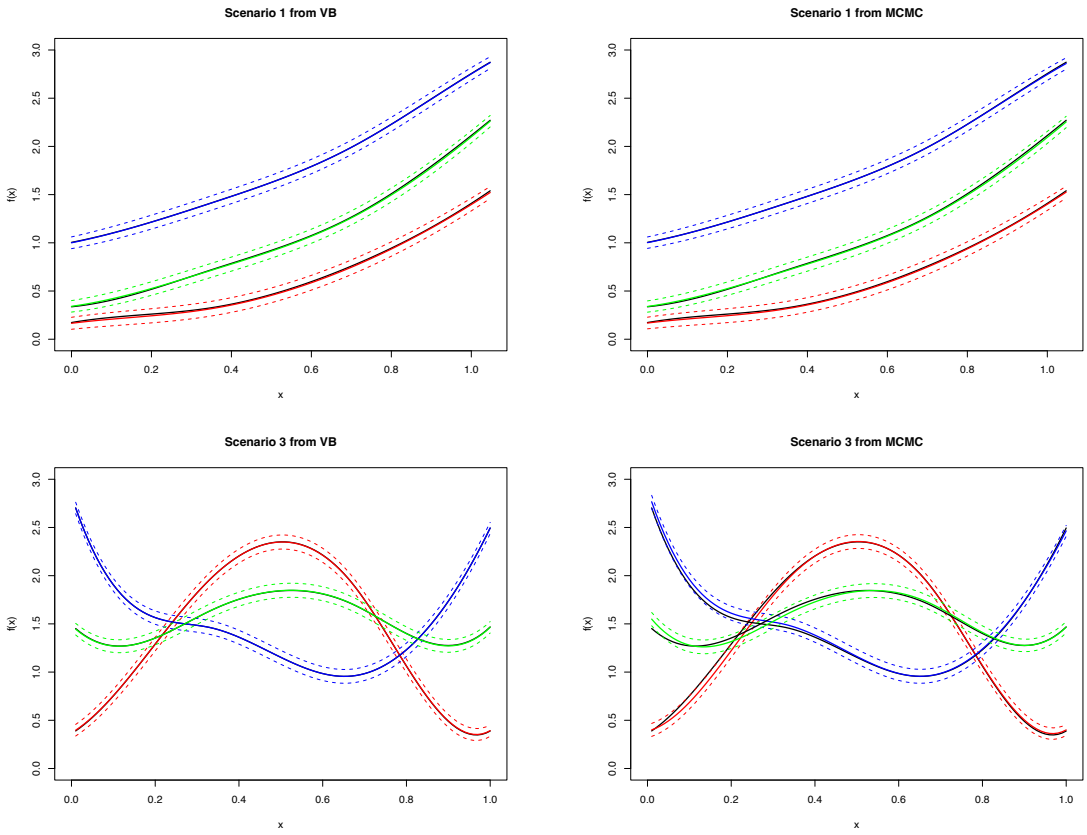


Figure 2.7: Simulation results for Model 1, Scenarios 1 and 3. The 95% credible bands for the true mean curves from VB (the left column) and MCMC (the right column). The solid colored lines represent the estimated mean curves, with the true mean curves depicted by black solid lines. The 95% credible bands are illustrated by the corresponding dashed lines.

2.3.3 Simulation study on Model 2

2.3.3.1 Simulation scenarios

We also investigate the performance of our proposed VB algorithm under Model 2 using simulated data. We consider the simulation schemes of Scenario 1 and Scenario 3 in Section 2.3.2.1, but add a random intercept to each curve, to construct four different scenarios namely Scenario 9, Scenario 10, Scenario 11, and Scenario 12.

Scenario 9, $K = 3$***:***

Scenario 9 is constructed based on Scenario 1. The data are simulated as follows.

$$Y_{ik}(t_j) = a_{ik} + b_k + c_k \sin(1.3t_j) + t_j^3 + \delta_{ij}; i = 1, ..., 50; j = 1, ..., 100; k = 1, 2, 3,$$

where $Y_{ik}(t_j)$ denotes the value at point t_j of the ith curve from cluster k, $a_{ik} \sim N(0, 0.4^2)$, $\delta_{ij} \sim N(0, 0.2^2)$, $b_1 = -0.25$, $b_2 = 1.25$, $b_3 = 2.50$, $c_1 = 1/1.3$, $c_2 = 1/1.2$, and $c_3 = 1/4$.

Scenario 10, $K = 3$***:***

Scenario 10 is developed based on Scenario 3. In this scenario, we consider a very small variance for the random intercept which almost resembles the case without a random intercept. Data are generated as follows.

$$Y_{ik}(t_j) = a_{ik} + \sum_{l=1}^{6} B_l(t_j)\phi_{kl} + \delta_{ij}; i = 1, ..., 50; j = 1, ..., 100; k = 1, 2, 3,$$

where $Y_{ik}(t_j)$ denotes the value at point t_j of the ith curve from cluster k, $a_{ik} \sim N(0, 0.05^2)$, $\delta_{ij} \sim N(0, 0.4^2)$. The B-spline coefficients, ϕ_{kl}, remain the same and are presented in Table 2.1, which are also used in Scenarios 9 and 10.

Scenario 11, $K = 3$***:***

Scenario 11 is similar to Scenario 10, but with larger variance for the random intercept but smaller variance for the random error. Data are generated as follows.

$$Y_{ik}(t_j) = a_{ik} + \sum_{l=1}^{6} B_l(t_j)\phi_{kl} + \delta_{ij}; i = 1, ..., 50; j = 1, ..., 100; k = 1, 2, 3,$$

where $Y_{ik}(t_j)$ denotes the value at point t_j of the ith curve from cluster k, $a_{ik} \sim N(0, 0.3^2)$, $\delta_{ij} \sim N(0, 0.15^2)$.

Scenario 12, $K = 3$***:***

Scenario 12 is similar to Scenario 10, but with larger variance for the random intercept. In this scenario, we use larger variance for the random error compared with that in Scenario 11, indicating a more complex case. Data are generated as follows.

$$Y_{ik}(t_j) = a_{ik} + \sum_{l=1}^{6} B_l(t_j)\phi_{kl} + \delta_{ij}; i = 1, ..., 50; j = 1, ..., 100; k = 1, 2, 3,$$

where $Y_{ik}(t_j)$ denotes the value at point t_j of the ith curve from cluster k, $a_{ik} \sim N(0, 0.6^2)$, $\delta_{ij} \sim N(0, 0.4^2)$.

2.3.3.2 Simulation results for Model 2

Figure 2.8 shows the curves from one of the 50 simulated datasets for Scenarios 9 and 11. Due to the similarity among Scenarios 10, 11 and 12, the curves for Scenarios 10 and 12 are presented in Figure B.2 of Appendix B. In Figure 2.8, we can observe a slight difference between each cluster's true mean curve and the estimated mean curve. Furthermore, more variation occurs after adding the random intercept. Especially in Scenario 12, with large variances, there is a more substantial overlap among curves from different clusters, resulting in a more complex scenario for clustering than the corresponding Scenario 3 in Section 2.3.2.

Table 2.5 presents the numerical results, including the mean mismatch rate and the mean V-measure with their corresponding standard deviations from the 50 different simulated datasets under each scenario considered. In Scenario 9, where the true mean curves exhibit relative parallelism, we do not observe a significant difference in the mean mismatch rate (approximately 10%) and the mean V-measure (approximately 0.7) among our VB model, the classical k-means, and SaS-Funclust. In contrast, in Scenario 9, the functional k-means and funFEM methods exhibit a larger mean mismatch rate and an 18.78% lower mean V-measure than VB. In Scenario 10, where the true mean curves intersect, our proposed model achieves a significantly lower mean mismatch rate of 0.0299, in contrast to

Table 2.5: Simulation results for Model 2. Mismatch rate and V-measure values for each simulation scenario.

	VB		*k*-means		functional *k*-means		funFEM		SaS-Funclust	
Scenario	M[1] (sd[2])	V[3] (sd)	M (sd)	V (sd)	M (sd)	V (sd)	M (sd)	V (sd)	M (sd)	V (sd)
9	0.1045 (0.0265)	0.7077 (0.0565)	0.1069 (0.0259)	0.7033 (0.0505)	0.2795 (0.0865)	0.5767 (0.0810)	0.2308 (0.0869)	0.5738 (0.0891)	0.1012 (0.0259)	0.7137 (0.0513)
10	0.0299 (0.1040)	0.9767 (0.0804)	0.1404 (0.2169)	0.8937 (0.1641)	0.2799 (0.0938)	0.5768 (0.1085)	0.1845 (0.2136)	0.7284 (0.3288)	0.3333 (0.0000)	0.7337 (0.0000)
11	0.1453 (0.1485)	0.7923 (0.1865)	0.1571 (0.1400)	0.7580 (0.1793)	0.3427 (0.0591)	0.4802 (0.0831)	0.2029 (0.1411)	0.6917 (0.1738)	0.1972 (0.0308)	0.6644 (0.0513)
12	0.2493 (0.1416)	0.6078 (0.2285)	0.3824 (0.0367)	0.3774 (0.0581)	0.5131 (0.086)	0.1751 (0.150)	0.3844 (0.0425)	0.3104 (0.0499)	0.5961 (0.0417)	0.0280 (0.0498)

[1]M: mean mismatch rate from 50 runs.

[2]sd: standard deviation.

[3]V: mean V-measure from 50 runs.

the other methods: 0.1404 for classical k-means, 0.2799 for functional k-means, 0.1845 for funFEM, and 0.3333 for SaS-Funclust. Moreover, the mean V-measure obtained from VB is 0.9767, which is 9.28%, 69.33%, 34.09%, and 33.12% higher than the results from the aforementioned methods, respectively.

When the random intercept variance becomes larger in Scenario 11, even with a smaller random error variance, clustering curves via our proposed model becomes more challenging. The mean mismatch rate increases to 0.1453 from 0.0299, while the mean V-measure drops to 0.7923 from 0.9767 in Scenario 10. Nonetheless, our model continues to outperform the other considered methods, with differences in mismatch rates of 0.0118 for classical k-means, 0.1974 for functional k-means, 0.0576 for funFEM, and 0.0519 for SaS-Funclust. In Scenario 12, where there is a further increase in variance in the random intercept, we observe that the clustering performance of all methods deteriorates, leading to higher mismatch rates and lower V-measure values. Nevertheless, the VB algorithm still stands out by achieving the lowest mean mismatch rate and the highest mean V-measure compared to the other methods. The larger standard deviation of mismatch rates and V-measure of VB compared to other methods happen because, among the 50 different runs, there are 11 runs where our method can 100% correctly assign each curve to the cluster it belongs to, resulting in a mismatch rate of zero and a V-measure of one. At the same time, using the classical k-means as an example, there is no run where the classical k-means provides such perfect clustering results. Besides, among the 50 different runs, there are 41 runs where our method provides lower mismatch rates and higher V-measures than the classical k-means.

Table 2.6 shows the EMISE for each cluster in Scenarios 9, 10, 11 and 12 based on Model 2. Small EMISE values once again indicate that the true mean curves and the corresponding curves have a small difference. We also find that compared with Table 2.3 based on Model 1, the EMISE values based on Model 2 are larger. This is in our expectation since adding

a random intercept to each curve will bring more variation to the curves, and as a result, more variation in the estimated mean curves, in Scenario 12 especially when we have a larger variance for generating random intercepts. Plots of EMSE values in Scenarios 7, 8, 9, and 10 based on Model 2 are provided in Figure B.3 in Appendix B.

For the computational cost, the run times of the proposed VB algorithm of Model 2 for 50 simulated datasets from Scenarios 9, 10, 11 and 12 are 40.96 min, 1.52 min, 10.46 min, and 11.52 min, respectively. For comparison, SaS-Funclust takes longer computation times: 45.06 min, 65.17 min, 64.35 min and 64.2 min for the respective scenarios.

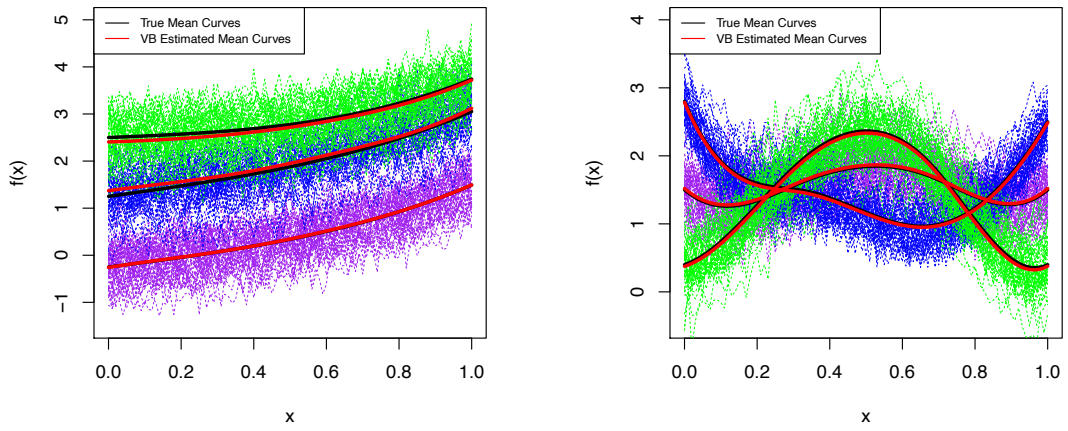


Figure 2.8: Simulation results for Model 2. Example of simulated data under Scenario 9 (left) and Scenario 11 (right). Raw curves (different colors correspond to different clusters), cluster-specific true mean curves (in black) and corresponding estimated mean curves (in red).

Table 2.6: Simulations results for Model 2. The empirical mean integrated squared error (EMISE) for the estimated mean curve in each cluster in each scenario.

Scenario	Cluster	EMISE	Scenario	Cluster	EMISE
9	1	0.07666	10	1	0.00498
	2	0.03109		2	0.00203
	3	0.06953		3	0.00316
11	1	0.05171	12	1	0.25312
	2	0.01938		2	0.13287
	3	0.02638		3	0.12465

2.4 Application to real data

In this section, we apply our proposed method in Section 2 to the growth and the Canadian weather datasets, which are both publicly available in the R package *fda*.

The Growth data (Tuddenham and Snyder, 1954) includes heights (in cm) of the 93 children over 31 unevenly spaced time points from the age of one to eighteen. Raw curves without any smoothing are shown in Figure 2.9, where the green curves correspond to boys and blue curves to girls. In this case, we apply our proposed method to the growth curves considering two clusters and compare the inferred cluster assignments (boys or girls) to the true ones.

The Canadian weather data (raw data are presented in Figure B.4 in Appendix B) contains the daily temperature at 35 different weather stations (cities) in Canada, averaged out from the year of 1960 to 1994. However, unlike the growth data, we do not know the true number of clusters in the weather data. Therefore, in order to find the best number of clusters, we apply the DIC for model comparison.

The number of B-spline basis functions is fixed and known within the VB algorithm. As discussed in Rossi et al. (2004), a low number of basis functions can be applied to get rid of the measurement noise. Another feature of the B-spline basis system is that increasing the number of B-spline bases does not always improve certain aspects of the fit to the data (Ramsay and Silverman, 2005). Based on Liu and Yang (2009), ten B-spline basis functions are relatively reasonable for clustering the Growth data with two clusters. The Canadian weather data presents a higher variation (larger noise) than the Growth data. Therefore, curves with a moderate smoothing, rather than with more roughness, may more accurately reflect the underlying functional structures, and the underlying clusters. So, we use six B-spline basis functions to represent the weather data within the VB algorithm. It is important to note that we do not have a strong prior knowledge of these real datasets but still need

to provide appropriate prior hyperparameters for the VB algorithm. As a solution, we randomly select one underlying curve in each dataset and fit a B-spline regression to obtain a vector of coefficients which is then modified across different clusters resulting in the prior mean vectors $\mathbf{m}_k^0$ for $k = 1, ..., K$. We set $s^0 = 0.1$, corresponding to a precision of 10, as the prior variance of these coefficients which provides a useful information as assumed in real world. For the Dirichlet prior distribution of $\boldsymbol{\pi}$, we use $\mathbf{d}^0 = (1/K, ..., 1/K)$, indicating that for each curve, the probability of assignment to each cluster is a *priori* equal across clusters. For the Gamma prior distribution of the precision, $\tau_k = 1/\sigma_k^2$, we prefer a large prior mean (e.g., 10) and a small prior variance (e.g., 0.1) which serve as informative prior knowledge, and therefore, we set $a^0 = 2000$ and $r^0 = 100$ for the growth data, and $a^0 = 1000$ and $r^0 = 800$ for the weather data. The ELBO convergence threshold is 0.001.

Since we know there are two clusters (boys and girls) in the growth dataset, $K = 2$ is preset for the clustering procedure. We apply the proposed VB algorithms under Models 1 and 2 to cluster the growth curves with 50 runs corresponding to 50 different initializations. The classical k-means method is also applied to the raw curves for performance comparison purposes. Figure 2.9 presents the estimated mean curves for each cluster corresponding to the the best VB run (the one with maximum ELBO after convergence) along with the empirical mean curves from both models (left graph for Model 1 while right for Model 2). The empirical mean curves are calculated by considering the true clusters and calculating their corresponding point-wise mean at each time point. Some difference between the estimated and the empirical curves can be observed for the girls due to a potential outlier. Regarding clustering performance, the mean mismatch rates for the VB algorithms under Model 1 and Model 2, and k-means are 33.33%, 20.47% and 34.41%, respectively. V-measure is more sensitive to misclassification than mismatch rate and, therefore, we obtain low mean V-measure values of 7.75% for VB under Model 1, 33.75% for VB under Model 2, and 6.37% for k-means. We can see the clustering performance significantly improved after adding a random intercept to each curve. Compared with Model 1, the mean mismatch

rate from Model 2 is lower by 12.86% , and the mean V-measure is higher by 26%.

For the Canadian weather dataset analysis, we considered temperature data from all stations except those located in Vancouver and Victoria because they present relatively flat temperature curves compared to other locations. We applied the proposed VB algorithm under Model 1 to the weather data. The left plot in Figure 2.10 shows the DIC values for different possible numbers of clusters ($K = 2, 3, 4, 5$). We can observe that the best number of clusters for separating the Canadian weather data is three, which corresponds to the smallest DIC. Finally, we present the clustering results with $K = 3$ on a map of Canada in the right plot in Figure 2.10. As can be seen, when $K = 3$, we have three resulting groups in three different colors. In general, most of the weather stations in purple are located in northern Canada. In contrast, stations in southern Canada are separated into two groups color-coded in blue and red on the map of Canada. Although some stations may be incorrectly clustered, we can still see a potential pattern that makes sense geographically.

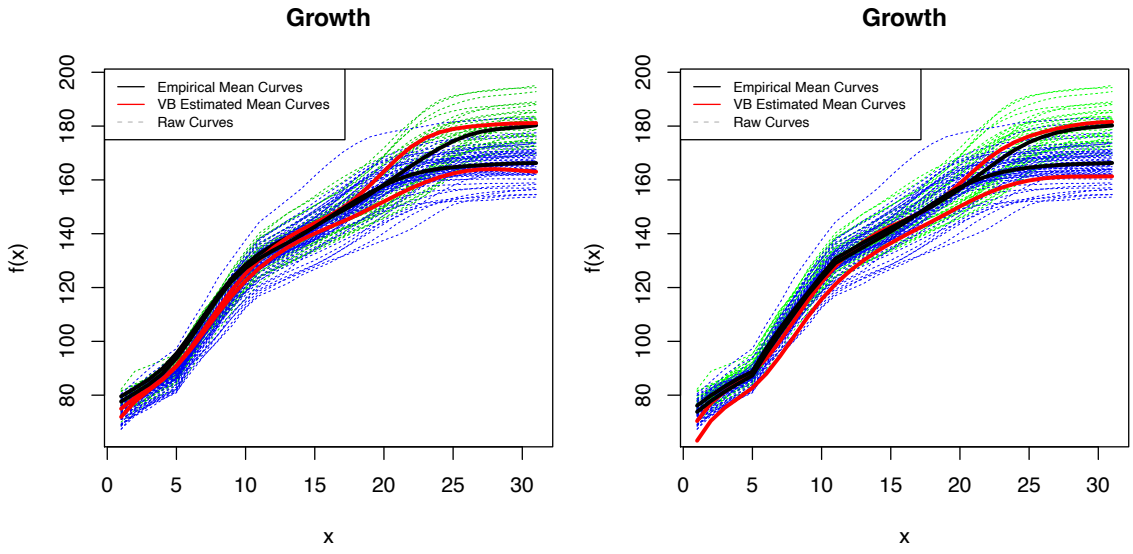


Figure 2.9: Raw curves (dashed curves) from the Growth dataset where green curves refer to the boys' heights while the blue ones are for the girls', with empirical mean curves (in solid black) and our VB estimated mean curves (in solid red). The left graph is resulted from Model 1 while the right is from Model 2.

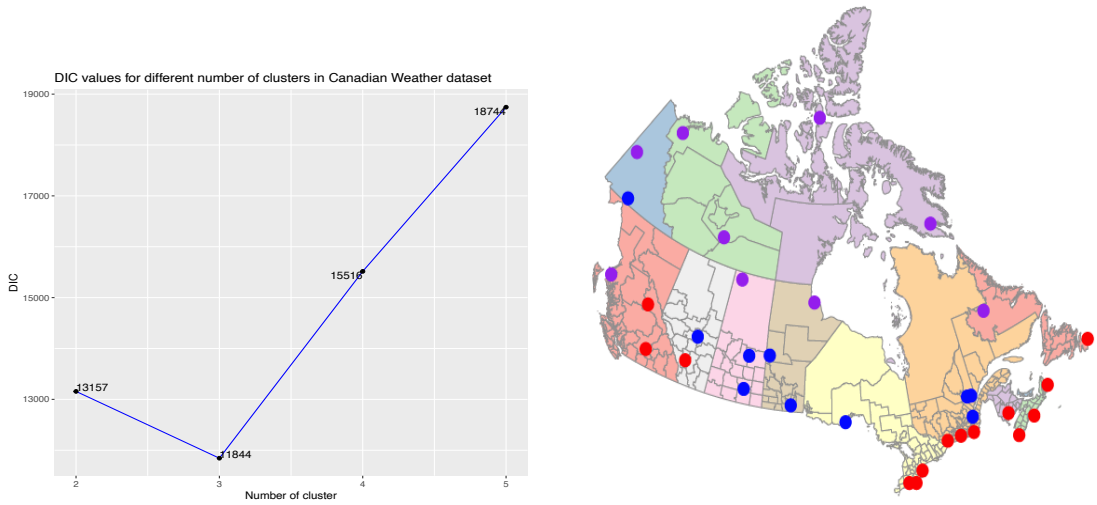


Figure 2.10: Left: DIC values for different clusters ($K = 2, 3, 4, 5$) in Canadian weather data. The best number of clusters is three which has the smallest DIC. Right: Clustering results under Model 1 (cities with same color are predicted in the same cluster) for Canadian weather data with preset three clusters ($K = 3$).

2.5 Conclusion and Discussion

This chapter develops a new model-based algorithm to cluster functional data via Bayesian variational inference. We derive a mean-field Variational Bayes (VB) algorithm. Next, the coordinate ascent variational inference is applied to update each term in the variational distribution factorization until convergence of the evidence lower bound. Finally, each observed curve is assigned to the cluster with the largest posterior probability.

We build our proposed VB algorithm under two different models. In Model 1, we assume the errors are independent, which may be a strong assumption. Motivated by the Growth data for the children's heights, which show a parallel structure indicating a shift among curves, we extended our approach to Model 2, which includes more complex variance-covariance structures by adding a random intercept for each curve.

The performance of our proposed VB algorithm in clustering functional data is supported by simulations and real data analyses. In simulation studies, VB accurately estimates mean curves, closely aligning with true curves, resulting in minimal empirical mean integrated

squared errors and demonstrating a good fit. In most scenarios, VB consistently outperforms other considered methods (classical k-means, functional k-means, funFEM, and SaS-Funclust) with the highest V-measure and the lowest mismatch rate. We provide insight into the selection of the number of clusters (mixture components) through a two-fold scheme based on DIC. Robustness is assessed via a sensitivity analysis across different prior settings and a study involving a misspecified type of basis functions. In our simulations, the proposed VB algorithm demonstrated computational efficiency, averaging 4 seconds to cluster each simulated dataset. In particular, for simulated data under Scenarios 1 and 3, VB is over 20 times faster than MCMC (Gibbs sampler). Moreover, VB demonstrates strong consistency with MCMC in estimating the marginal posterior distribution of B-spline basis coefficients and precision parameters. In addition to simulation studies, applying the VB algorithm to the Growth data reveals that Model 2 with a random intercept surpasses Model 1 in both mean curve estimation and clustering performance when the curves from the same cluster show a parallel structure.

The main advantage of our proposed VB algorithm is that we model the raw data and obtain clustering assignments and cluster-specific smooth mean curves simultaneously. In other words, compared to some previous methods where researchers first smooth the data and then cluster the data using only the information after smoothing (e.g., the coefficients of B-spline basis functions); our model, as a regression mixture model, directly uses the raw data as input, performing smoothing and clustering simultaneously. In addition, as we take a Bayesian inference approach, we can measure the uncertainty of our proposed clustering using the obtained cluster assignment posterior probabilities.

While our study has introduced the VB algorithm to cluster functional data using a B-spline regression mixture model, it is important to recognize its limitations. Although our Model 2, which includes a random intercept, provides a more flexible dependence structure, one could explore more intricate Gaussian processes for modeling the random errors. Addi-

tionally, it is worth noting that VB is not the sole method for clustering functional data with regression mixtures; alternatives like Gibbs sampler (as used for comparison here) or other MCMC-based algorithms can also be considered. In this work, we focus on the case where, for each curve, the number of basis functions is smaller than the number of evaluation points ($M < n$). So, future work may include investigation and further extension of the proposed VB under high-dimensional settings ($M >> n$), paying special attention to the issue of underestimation of the variability of the posterior estimates (Mukherjee and Sen, 2022; Devijver, 2017). For large datasets (large number of curves, N), the coordinate ascent variational inference algorithm, which considers all data points, may result in a high computational cost. Therefore, one may consider scalable algorithms such as the stochastic variational inference (Hoffman et al., 2013) for approximating the posterior distributions.

Furthermore, our approach relies on the assumption that the number of B-spline basis functions (M) is known prior to applying the VB algorithm. This assumption aligns with practical scenarios where researchers may subjectively determine M based on their expertise and/or visual inspection of the curves (Franco et al., 2023; Günther et al., 2021; Lenzi et al., 2017). However, to enhance the model's adaptability and automate the selection process, future investigations could explore the integration of a mechanism for selecting the number of B-spline bases directly within the VB algorithm itself. Relevant approaches and references for the selection of the number of basis functions include Sousa et al. (2023); Devijver et al. (2020); Gálvez et al. (2015); Yuan et al. (2013); Dias and Garcia (2007), and DeVore et al. (2003).

References

Anderson, C., D. Lee, and N. Dean (2014). Identifying clusters in Bayesian disease mapping. *Biostatistics 15*(3), 457–469. → page 44

Angelini, C., D. De Canditiis, and M. Pensky (2012). Clustering time-course microarray data using functional Bayesian infinite mixture model. *Journal of Applied Statistics 39*(1), 129–149. → page 44

Bishop, C. (2006). *Pattern Recognition and Machine Learning*. Springer. → page 16

Blei, D. M. and M. I. Jordan (2006). Variational inference for Dirichlet process mixtures. *Bayesian Analysis 1*(1), 121 – 143. → page 14

Blei, D. M., A. Kucukelbir, and J. D. McAuliffe (2017). Variational inference: A review for statisticians. *Journal of the American Statistical Association 112*(518), 859–877. → page 15, 16, 21

Boschi, T., J. Di Iorio, L. Testa, M. A. Cremona, and F. Chiaromonte (2021). Functional data analysis characterizes the shapes of the first COVID-19 epidemic wave in Italy. *Scientific Reports 11*(1). → page 12

Bouveyron, C., E. Côme, and J. Jacques (2015). The discriminative functional mixture model for a comparative analysis of bike sharing systems. *The Annals of Applied Statistics*, 1726–1760. → page 33

Centofanti, F., A. Lepore, and B. Palumbo (2023). Sparse and smooth functional data clustering. *Statistical Papers*, 1–31. → page 33

Chamroukhi, F. (2016a). Piecewise regression mixture for simultaneous functional data

clustering and optimal segmentation. *Journal of Classification 33*(3), 374–411. → page 14, 18

Chamroukhi, F. (2016b). Unsupervised learning of regression mixture models with unknown number of components. *Journal of Statistical Computation and Simulation 86*(12), 2308–2334. → page 18, 44

Chamroukhi, F. and H. D. Nguyen (2019). Model-based clustering and classification of functional data. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 9*(4), e1298. → page 13, 18

Chen, T., N. L. Zhang, T. Liu, K. M. Poon, and Y. Wang (2012). Model-based multidimensional clustering of categorical data. *Artificial Intelligence 176*(1), 2246–2269. → page 44

Collazos, J. A. A., R. Dias, and M. C. Medeiros (2023). Modeling the evolution of deaths from infectious diseases with functional data models: The case of COVID-19 in Brazil. *Statistics in Medicine*. → page 12

Cover, T. M. (1999). *Elements of Information Theory*. John Wiley & Sons. → page 33

De Souza, C. P. E., N. E. Heckman, and F. Xu (2017). Switching nonparametric regression models for multi-curve data. *Canadian Journal of Statistics 45*(4), 442–460. → page 12

Delaigle, A., P. Hall, and T. Pham (2019). Clustering functional data into groups by using projections. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 81*(2), 271–304. → page 13

Devijver, E. (2017). Model-based regression clustering for high-dimensional data: application to functional data. *Advances in Data Analysis and Classification 11*, 243–279. → page 63

Devijver, E., Y. Goude, and J.-M. Poggi (2020). Clustering electricity consumers using high-dimensional regression mixture models. *Applied Stochastic Models in Business and Industry 36*(1), 159–177. → page 63

DeVore, R., G. Petrova, and V. Temlyakov (2003). Best basis selection for approximation in Lp. *Foundations of Computational Mathematics 3*, 161–185. → page 63

Dias, R. and N. L. Garcia (2007). Consistent estimator for basis selection based on a proxy of the Kullback–Leibler distance. *Journal of Econometrics 141*(1), 167–178. → page 63

Dias, R., N. L. Garcia, G. Ludwig, and M. A. Saraiva (2015). Aggregated functional data model for near-infrared spectroscopy calibration and prediction. *Journal of Applied Statistics 42*(1), 127–143. → page 12, 18

Dias, R., N. L. Garcia, and A. Martarelli (2009). Non-parametric estimation for aggregated functional data for electric load monitoring. *Environmetrics 20*, 111 – 130. → page 18, 36

Earls, C. and G. Hooker (2017). Variational Bayes for functional data registration, smoothing, and prediction. *Bayesian Analysis 12*(2), 557 – 582. → page 14

Escobar, M. D. and M. West (1995). Bayesian density estimation and inference using mixtures. *Journal of the American Statistical Association 90*(430), 577–588. → page 44

Faes, C., J. T. Ormerod, and M. P. Wand (2011). Variational Bayesian inference for parametric and nonparametric regression with missing data. *Journal of the American Statistical Association 106*(495), 959–971. → page 49

Febrero-Bande, M. and M. O. de la Fuente (2012). Statistical computing in functional data analysis: The R Package fda.usc. *Journal of Statistical Software 51*(4), 1–28. → page 33

Franco, G., C. P. E. de Souza, and N. L. Garcia (2023). Aggregated functional data model applied on clustering and disaggregation of UK electrical load profiles. *Journal of the Royal Statistical Society Series C: Applied Statistics 72*(1), 48–75. → page 12, 18, 63

Frizzarin, M., A. Bevilacqua, B. Dhariyal, K. Domijan, F. Ferraccioli, E. Hayes, G. Ifrim, A. Konkolewska, T. L. Nguyen, U. Mbaka, G. Ranzato, A. Singh, M. Stefanucci, and A. Casa (2021). Mid infrared spectroscopy and milk quality traits: a data analysis competition at the International Workshop on Spectroscopy and Chemometrics 2021. → page 12

Fruhwirth-Schnatter, S., G. Celeux, and C. P. Robert (2019). *Handbook of Mixture Analysis*. CRC press. → page 14, 18

Gálvez, A., A. Iglesias, A. Avila, C. Otero, R. Arias, and C. Manchado (2015). Elitist clonal selection algorithm for optimal choice of free knots in B-spline data fitting. *Applied Soft Computing 26*, 90–106. → page 63

Gao, H., K. Bryc, and C. D. Bustamante (2011). On identifying the optimal number of population clusters via the deviance information criterion. *PloS one 6*(6), e21014. → page 44

Geisser, S. and W. F. Eddy (1979). A predictive approach to model selection. *Journal of the American Statistical Association 74*(365), 153–160. → page 44

Giacofci, M., S. Lambert-Lacroix, G. Marot, and F. Picard (2013). Wavelet-based clustering for mixed-effects functional models in high dimension. *Biometrics 69*(1), 31–40. → page 14

Goldsmith, J., M. P. Wand, and C. Crainiceanu (2011). Functional regression via variational Bayes. *Electronic Journal of Statistics 5*, 572. → page 14

Grün, B. (2019). Model-based clustering. In *Handbook of Mixture Analysis*, pp. 157–192. CRC Press, Taylor & Francis Group. → page 13, 14, 18

Günther, S., W. Pazner, and D. Qi (2021). Spline parameterization of neural network controls for deep learning. *arXiv preprint arXiv:2103.00301*. → page 63

Hael, M. A., Y. Yongsheng, and B. I. Saleh (2020). Visualization of rainfall data using functional data analysis. *SN Applied Sciences 2*(3), 461. → page 12

Hartigan, J. and M. Wong (1979). A k-means clustering algorithm. *J R Stat Soc Ser C 28*, 100–108. → page 13

Heinzl, F. and G. Tutz (2013). Clustering in linear mixed models with approximate Dirichlet process mixtures using EM algorithm. *Statistical Modelling 13*(1), 41–67. → page 44

Hoffman, M. D., D. M. Blei, C. Wang, and J. Paisley (2013). Stochastic variational inference. *Journal of Machine Learning Research*. → page 63

Hu, G., J. Geng, Y. Xue, and H. Sang (2020). Bayesian Spatial Homogeneity Pursuit of Functional Data: an Application to the U.S. Income Distribution. → page 12

Jacques, J. and C. Preda (2013). Funclust: A curves clustering method using functional random variables density approximation. *Neurocomputing 112*, 164–171. → page 14

Jacques, J. and C. Preda (2014). Functional data clustering: a survey. *Advances in Data Analysis and Classification 8*(3), 24. → page 12

James, G. and C. Sugar (2003). Clustering for sparsely sampled functional data. *Journal of the American Statistical Association 98*(462), 397–408. → page 13

Jones, M. C. and J. A. Rice (1992). Displaying the important features of large collections of similar curves. *The American Statistician 46*(2), 140. → page 13

Jordan, M. I., Z. Ghahramani, T. Jaakkola, and L. Saul (1999). Introduction to variational methods for graphical models. *Machine Learning 37*, 183–233. → page 15, 16, 21

Komárek, A. (2009). A new R package for Bayesian estimation of multivariate normal mixtures allowing for selection of the number of components and interval-censored data. *Computational Statistics & Data Analysis 53*(12), 3932–3947. → page 44

Kullback, S. and R. A. Leibler (1951). On information and sufficiency. *The Annals of Mathematical Statistics 22*(1), 79 – 86. → page 16

Lenzi, A., C. P. de Souza, R. Dias, N. L. Garcia, and N. E. Heckman (2017). Analysis of aggregated functional data from mixed populations with application to energy consumption. *Environmetrics 28*(2), e2414. → page 12, 18, 63

Li, T. and J. Ma (2020). Functional data clustering analysis via the learning of Gaussian processes with Wasserstein distance. In *Neural Information Processing*, pp. 393–403. Springer International Publishing. → page 13

Liu, X. and M. C. Yang (2009). Simultaneous curve registration and clustering for functional data. *Computational Statistics & Data Analysis 53*(4), 1361–1376. → page 58

Luts, J. and M. P. Wand (2015). Variational inference for count response semiparametric regression. *Bayesian Analysis 10*(4), 991 – 1023. → page 49

Martino, A., A. Ghiglietti, F. Ieva, and A. M. Paganoni (2019). A k-means procedure based on a mahalanobis type distance for clustering multivariate functional data. *Statistical Methods & Applications 28*(2), 301–322. → page 13

McLachlan, G. J., S. X. Lee, and S. I. Rathnayake (2019). Finite mixture models. *Annual Review of Statistics and Its Application 6*, 355–378. → page 18

Melnykov, V. and R. Maitra (2010). Finite mixture models and model-based clustering. *Statistics Surveys 4*(none), 80 – 116. → page 18

Mukherjee, S. and S. Sen (2022). Variational inference in high-dimensional linear regression. *The Journal of Machine Learning Research 23*(1), 13703–13758. → page 63

Nguyen, X. and A. E. Gelfand (2011). The Dirichlet labeling process for clustering functional data. *Statistica Sinica*, 1249–1289. → page 14

Nieto-Barajas, L. E. and A. Contreras-Cristán (2014). A Bayesian nonparametric approach for time series clustering. *Bayesian Analysis 9*(1), 147 – 170. → page 44

Peel, D. and G. MacLahlan (2000). *Finite Mixture Models*. John & Sons. → page 18

Petrone, S., M. Guindani, and A. E. Gelfand (2009). Hybrid Dirichlet mixture models for functional data. *Journal of the Royal Statistical Society Series B: Statistical Methodology 71*(4), 755–782. → page 44

Ramsay, J., G. Hooker, and S. Graves (2009). *Functional Data Analysis with R and MATLAB*. Springer New York. → page 12

Ramsay, J. O. and C. J. Dalzell (1991). Some tools for functional data analysis. *Journal of the Royal Statistical Society: Series B (Methodological) 53*(3), 539–561. → page 12

Ramsay, J. O. and B. W. Silverman (2005). *Functional Data Analysis* (2 ed.). Springer. → page 12, 18, 58

Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association 66*(336), 846–850. → page 33

Ray, S. and B. Mallick (2006). Functional clustering by Bayesian wavelet methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 68*(2), 305–332. → page 14, 44

Rigon, T. (2023). An enriched mixture model for functional clustering. *Applied Stochastic Models in Business and Industry 39*(2), 232–250. → page 14, 18, 44

Rodríguez, A., D. B. Dunson, and A. E. Gelfand (2009). Bayesian nonparametric functional data analysis through density estimation. *Biometrika 96*(1), 149–162. → page 44

Rosenberg, A. and J. Hirschberg (2007). V-Measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, Prague, Czech Republic, pp. 410–420. Association for Computational Linguistics. → page 33

Rossi, F., B. Conan-Guez, and A. El Golli (2004). Clustering functional data with the SOM algorithm. In *ESANN*, pp. 305–312. Citeseer. → page 58

Rousseau, J. and K. Mengersen (2011). Asymptotic behaviour of the posterior distribution in overfitted mixture models. *Journal of the Royal Statistical Society Series B: Statistical Methodology 73*(5), 689–710. → page 44

Samé, A., F. Chamroukhi, G. Govaert, and P. Aknin (2011). Model-based clustering and segmentation of time series with changes in regime. *Advances in Data Analysis and Classification 5*(4), 301–321. → page 14

Sousa, P. H. T. O., C. P. E. de Souza, and R. Dias (2023). Bayesian adaptive selection of basis functions for functional data representation. *Journal of Applied Statistics*, 1–35. → page 12, 47, 63

Spiegelhalter, D. J., N. G. Best, B. P. Carlin, and A. Van Der Linde (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 64*(4), 583–639. → page 14, 44

Tarpey, T. and K. Kinateder (2003). Clustering functional data. *Journal of Classification 20*(1), 93–114. → page 13

Tuddenham, R. D. and M. M. Snyder (1954). Physical growth of California boys and girls from birth to eighteen years. *Publications in Child Development. University of California, Berkeley 12*, 183–364. → page 58

Wainwright, M. J., M. I. Jordan, et al. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning 1*(1–2), 1–305. → page 15

Wang, B. and D. M. Titterington (2005). Inadequacy of interval estimates corresponding to variational Bayesian approximations. In *International Workshop on Artificial Intelligence and Statistics*, pp. 373–380. PMLR. → page 49

Wang, J.-L., J.-M. Chiou, and H.-G. Müller (2016). Functional data analysis. *Annual Review of Statistics and Its Application 3*, 257–295. → page 12, 13

Wang, W.-L. and T.-I. Lin (2022). Model-based clustering via mixtures of unrestricted skew normal factor analyzers with complete and incomplete data. *Statistical Methods & Applications*, 1–31. → page 44

Ward, J. (1963). Hierarchical grouping to optimize an objective function. *J Am Stat Assoc 58*, 236–244. → page 13

Watanabe, S. and M. Opper (2010). Asymptotic equivalence of Bayes cross validation and widely applicable information criterion in singular learning theory. *Journal of Machine Learning Research 11*(12). → page 44

Xian, C., C. P. de Souza, W. He, F. F. Rodrigues, and R. Tian (2024). Variational Bayesian analysis of survival data using a log-logistic accelerated failure time model. *Statistics and Computing 34*(2), 67. → page 49

Xian, C., C. P. E. de Souza, J. Jewell, and R. Dias (2024). Clustering functional data via variational inference. *Advances in Data Analysis and Classification*, 1–50. → page 12

Yang, Y., Y. Yang, and H. L. Shang (2021). Feature extraction for functional time series: Theory and application to NIR spectroscopy data. → page 12

Yuan, Y., N. Chen, and S. Zhou (2013). Adaptive B-spline knot selection using multi-resolution basis set. *Iie Transactions 45*(12), 1263–1277. → page 63

Zambom, A., J. Collazos, and R. Dias (2019). Functional data clustering via hypothesis testing k-means. *Computational Statistics 34*(2), 527–549. → page 13, 33, 35, 38, 40, 45

Zhang, Y. and D. Telesca (2014). Joint clustering and registration of functional data. *arXiv preprint arXiv:1403.7134*. → page 14

Chapter 3
Variational Bayesian analysis of survival data using a log-logistic accelerated failure time model

3.1 Introduction

As an alternative to Cox proportional hazards model (Cox, 1972) [1], the accelerated failure time (AFT) model has been widely utilized in survival analysis recently (Webber et al., 2022; Longo et al., 2022; Xu et al., 2022) due to its intuitive interpretation (Wei, 1992). Estimation of parameters and inference under an AFT model are usually likelihood-based under a frequentist framework (Kalbfleisch and Prentice, 2002; Lawless, 2003). Recent developments have made Bayesian estimation and inference for an AFT model an attractive alternative to likelihood-based methods (Ibrahim et al., 2001). Implementations of the AFT model under the framework of Bayesian survival analysis can be found in different scenarios; see, for example, Lambert et al. (2004); Komárek and Lesaffre (2008); Zhang and Lawson (2011) and Tang et al. (2022). As for the distributions considered in the parametric AFT model, common choices include log-logistic, Weibull, log-normal, and Gamma distributions. The log-logistic distribution, exhibiting a non-monotonic hazard function, is commonly used in survival analysis when the hazard function presents an inverse U-shape. Empirical analyses in various applications show that the log-logistic distribution is well-suited to model a variety of survival data (Patel et al., 2006; Weng et al., 2014; Thiruvengadam et al., 2021; Rivas-López et al., 2022).

Variational inference (VI), a method developed from machine learning, is used to approximate the posterior distribution of a Bayesian model via optimization (Jordan et al., 1999;

[1] A version of this chapter has been published in *Statistics and Computing* (Xian et al., 2024).

Bishop, 2006). Blei et al. (2017) presented a comprehensive review of VI from a statistical perspective. As an alternative to Markov Chain Monte Carlo (MCMC) algorithms in Bayesian analysis, the main advantage of VI is its much lower computational cost (Blei et al., 2017). In addition, as a Bayesian approach, VI can make use of prior information obtained from similar studies, which are commonly available in survival analysis. Another advantage of VI is that it enables us to conduct inference for small sample sizes since it does not rely on asymptotics (Ibrahim et al., 2001), although asymptotic properties for VI methods may still be obtained in some scenarios. For example, Wang and Blei (2019) provided a study on the frequentist consistency of VI when the Kullback–Leibler (KL) minimizer (Kullback and Leibler, 1951) of a normal distribution is considered.

Variational Bayes (VB) is a variational inference method when the KL divergence is used as a criterion to measure the closeness between an approximated posterior density and the exact posterior density in the optimization. VB has been utilized in regression analysis for different statistical problems, such as parametric and nonparametric regression with missing data (Faes et al., 2011), nonparametric regression with measurement error (Pham et al., 2013), semiparametric regression for count response (Luts and Wand, 2015), high-dimensional linear regression with sparse priors (Ray and Szabó, 2022), clustering of functional data via a regression mixture model (Xian et al., 2024) and regression analysis of right censored survival data from the exponentiated-Weibull distribution (Abubakar et al., 2023).

In this chapter, we consider the AFT survival model with survival times following a log-logistic distribution and being right censored. We take on a Bayesian approach and develop a VB algorithm to infer the model parameters. In our approach, we employ the coordinate ascent algorithm and introduce a piecewise approximation technique when computing expectations involving the complete-data log-likelihood to overcome the intractability of deriving update equations for the variational density. The purpose of the piecewise approx-

imation is to attain a closed-form solution for variational inference while avoiding reliance on methods such as black-box variational inference (Ranganath et al., 2014) or stochastic variational inference (Hoffman et al., 2013; Murphy, 2023). To the best of our knowledge, we are the first to build and investigate a VB approach for the log-logistic AFT survival regression analysis. Our proposed method is implemented in R and codes are available at `https://github.com/chengqianxian/vbaft`.

The remainder of this chapter is organized as follows. Section 3.2 presents a background of the log-logistic AFT model and the VB inference. We present our methodology including the proposed VB algorithm in Section 3.3. In Section 3.4, we conduct simulation studies to evaluate the performance of our method under various scenarios and compare the analysis results with both frequentist analysis and the MCMC analysis. In Section 3.5, we apply our proposed method to a real dataset. A discussion on the proposed method is provided in Section 3.6.

3.2 Background

3.2.1 Log-logistic accelerated failure time model

Let T_i be the survival time and C_i be the censoring time of the i^{th} subject in the sample, $i = 1, ..., n$. Let $t_i = \min(T_i, C_i)$ and $\delta_i = \mathbb{1}(T_i \leq C_i)$ be the observed time and the indicator for right censoring of the i^{th} subject, respectively. Then the log-logistic AFT model can be expressed as follows:

$$\log(T_i) = \mathbf{X}_i^T \boldsymbol{\beta} + bz_i, \tag{3.1}$$

where $\mathbf{X}_i$ is a column vector with length $p, p \geq 2$, containing $p - 1$ fixed effects (covariates) and a constant one to incorporate the intercept (i.e., $\mathbf{X}_i = (1, x_{i1}, ..., x_{i(p-1)})^T$), $\boldsymbol{\beta}$ is the corresponding vector of coefficients for the fixed effects, z_i is a random variable following a standard logistic distribution, and b is a scale parameter. The survival time T_i and cen-

soring time C_i are assumed independent given the covariates $\mathbf{X}_i$. For the standard logistic distribution, the survival function and density are

$$S_0(z) = \frac{1}{1+e^z}, \quad f_0(z) = \frac{e^z}{(1+e^z)^2}, \ -\infty < z < \infty.$$

Then the log-likelihood for $\boldsymbol{\beta}$ and b is

$$l(\boldsymbol{\beta}, b) = -r \log b + \sum_{i=1}^{n} [\delta_i \log f_0(z_i) + (1-\delta_i) \log S_0(z_i)], \tag{3.2}$$

where $r = \sum_{i=1}^{n} \delta_i$ is the number of observed survival times, and $z_i = (y_i - \mathbf{X}_i^T \boldsymbol{\beta})/b, y_i = \log(t_i)$.

3.2.2 Elements of variational Bayes inference

In a generic Bayesian model, the posterior density of the parameters is of interest to conduct statistical inference. Consider a Bayesian model with parameter vector $\boldsymbol{\theta} \in \Theta$ and observed data $\mathbf{D}$. Using the Bayes' theorem, we can obtain the posterior density function by

$$p(\boldsymbol{\theta}|\mathbf{D}) = \frac{p(\boldsymbol{\theta}, \mathbf{D})}{p(\mathbf{D})}. \tag{3.3}$$

However, calculating the posterior density in (3.3) might not be feasible if there are many parameters and no conjugate prior distributions exist. Therefore, one may alternatively find an approximation to the posterior. While for many years MCMC has stood as the conventional method for attaining this objective, the subsequent paragraphs introduce the elements of variational Bayes inference.

The idea of variational Bayes is to find a variational density $q^*(\boldsymbol{\theta})$ from a family of possible densities Q to approximate $p(\boldsymbol{\theta}|\mathbf{D})$, which can be solved in terms of an optimization problem using the Kullback-Leibler (KL) divergence as a minimization criterion. The KL divergence measures the closeness between the possible densities q in the family Q and the

exact posterior density p. The KL divergence is defined as

$$\mathrm{KL}(q||p) = \mathbb{E}_{q(\boldsymbol{\theta})}[\log q(\boldsymbol{\theta})] - \mathbb{E}_{q(\boldsymbol{\theta})}[\log p(\boldsymbol{\theta}|\mathbf{D})].$$

It can be shown that

$$\begin{aligned}\mathbb{E}_{q(\boldsymbol{\theta})}[\log q(\boldsymbol{\theta})] - \mathbb{E}_{q(\boldsymbol{\theta})}[\log p(\boldsymbol{\theta}|\mathbf{D})] &= \int_{\Theta} q(\boldsymbol{\theta}) \log \frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta}|\mathbf{D})} d\boldsymbol{\theta} \\ &= \log p(\mathbf{D}) - \int_{\Theta} q(\boldsymbol{\theta}) \log \frac{p(\boldsymbol{\theta}, \mathbf{D})}{q(\boldsymbol{\theta})} d\boldsymbol{\theta},\end{aligned}$$

where the last term is the so-called evidence lower bound (ELBO). Since $\log p(\mathbf{D})$ is a constant with respect to q,

$$q^* = \underset{q \in Q}{\operatorname{argmin}}\, \mathrm{KL}(q||p) = \underset{q \in Q}{\operatorname{argmax}}\, \mathrm{ELBO}(q) = \underset{q \in Q}{\operatorname{argmax}}\, (\mathbb{E}_q \log p(\boldsymbol{\theta}, \mathbf{D}) - \mathbb{E}_q \log q(\boldsymbol{\theta})). \quad (3.4)$$

That is, minimizing the KL divergence is equivalent to maximizing the ELBO (Jordan et al., 1999; Blei et al., 2017).

The complexity of the variational family, Q, determines the complexity of such an optimization problem. It is a great challenge to solve a complex optimization problem corresponding to a complicated variational family. However, when we restrict Q to be the mean-field variational family, Q_{MF}, where the parameters and the latent variables are all assumed to be mutually independent and each of them is governed by a distinct factor in the variational density, $q(\boldsymbol{\theta}) = \prod_{k=1}^{K} q_k(\theta_k)$ for $q(\boldsymbol{\theta}) \in Q_{MF}$, the optimization problem in (3.4) is then changed to

$$q^*(\boldsymbol{\theta}) = \underset{q \in Q_{MF}}{\operatorname{argmax}}\, \mathrm{ELBO}(q(\boldsymbol{\theta})) = \underset{q \in Q_{MF}}{\operatorname{argmax}}\, \mathrm{ELBO}\Big(\prod_{k=1}^{K} q_k(\theta_k) \Big), \quad (3.5)$$

where we assume there are K sets of parameters, $\boldsymbol{\theta} = \{\theta_1, ..., \theta_K\}$.

The coordinate ascent algorithm under the mean-field variational inference (Bishop, 2006),

namely coordinate ascent variational inference (CAVI), can be utilized to solve the optimization problem in (3.5). The CAVI algorithm iteratively updates each mean-field variational density factor while keeping the other factors fixed, which makes the variational Bayesian inference a popular alternative to MCMC methods. As shown in Bishop (2006) and Blei et al. (2017), the update equation for the k^{th} factor ($k = 1, ..., K$) in the variational density can be obtained by calculating

$$\log q_k^*(\theta_k) = \mathbb{E}_{-\theta_k}[\log p(\boldsymbol{\theta}, \mathbf{D})] + \text{constant}, \tag{3.6}$$

where the constant refers to a term which does not depend on θ_k, and $\log p(\boldsymbol{\theta}, \mathbf{D})$ is the log of the joint density of the observed data $\mathbf{D}$, the parameters and the latent variables, which is also called the complete-data log-likelihood. The expectation is taken with respect to the variational density of all other parameters and latent variables except the one of interest. The update equation indicates that the expectation on the right-hand side does not involve the k^{th} factor, and therefore can be considered as a coordinate update. With the aid of the CAVI algorithm, the optimization problem (3.5) can be solved by climbing the ELBO to a local optimum (Blei et al., 2017).

3.3 Methodology

For the log-logistic AFT model specified in (3.1), we estimate the model parameters, $\boldsymbol{\beta}$ and b, using a Bayesian framework by further assuming the following prior distributions for $\boldsymbol{\beta}$ and b:

$$\boldsymbol{\beta} \sim N_p(\boldsymbol{\mu}_0, \sigma_0^2 I_{p\times p}) \text{ with precision } v_0 = 1/\sigma_0^2, \quad b \sim \text{Inverse-Gamma}\,(\alpha_0, \omega_0),$$

where μ_0, v_0, α_0 and ω_0 are known hyperparameters (Gelman et al., 2004; Faes et al., 2011).

Our goal is to derive a VB algorithm to approximate $p(\boldsymbol{\beta}, b \,|\, \mathbf{D})$, the posterior joint distribution of $\boldsymbol{\beta}$ and b given the data $\mathbf{D} := \{(t_i, \delta_i, \mathbf{X}_i), i = 1, ..., n\}$, with $q^* \in Q_{MF}$. That is, we assume that $q(\boldsymbol{\beta}, b) = q(\boldsymbol{\beta})\, q(b)$. The complete-data log-likelihood is then

$$\log p(\mathbf{D}, \boldsymbol{\beta}, b) = \log p(\mathbf{D} \,|\, \boldsymbol{\beta}, b) + \log p(\boldsymbol{\beta}) + \log p(b),$$

where

$$\log p(\mathbf{D} \,|\, \boldsymbol{\beta}, b) = -r \log b + \sum_{i=1}^{n} \left[\delta_i \frac{y_i - \mathbf{X}_i^T \boldsymbol{\beta}}{b} - (1 + \delta_i) \log \left\{ 1 + \exp\left(\frac{y_i - \mathbf{X}_i^T \boldsymbol{\beta}}{b} \right) \right\} \right]. \quad (3.7)$$

By maximizing the ELBO, we have the following solutions (Bishop, 2006):

$$\log q^*(\boldsymbol{\beta}) \overset{+}{\approx} \mathbb{E}_{q(b)}[\log p(\mathbf{D} \,|\, \boldsymbol{\beta}, b) + \log p(\boldsymbol{\beta})],$$

and

$$\log q^*(b) \overset{+}{\approx} \mathbb{E}_{q(\boldsymbol{\beta})}[\log p(\mathbf{D} \,|\, \boldsymbol{\beta}, b) + \log p(b)],$$

where we use $\overset{+}{\approx}$ to denote equality up to a constant additive factor for convenience. However, due to the complexity of the logistic distribution and the right censoring scheme, the expectation over the complete-data log-likelihood is challenging to compute. To achieve conjugacy and tractable expectation calculation of $\log p(\mathbf{D} \,|\, \boldsymbol{\beta}, b)$ in (3.7), we propose piecewise approximations of the function, $f(x) = \log(1 + \exp(x)), x \in (-\infty, \infty)$, embedded in deriving the update equations of $q(\boldsymbol{\beta})$ and $q(b)$. More specifically, taking the derivation for the update equation of $q^*(\boldsymbol{\beta})$ as an example, we need to calculate the expectation, $\mathbb{E}_{q(b)}[\log p(\mathbf{D} \,|\, \boldsymbol{\beta}, b) + \log p(\boldsymbol{\beta})]$. To obtain the conjugate posterior distribution for $\boldsymbol{\beta}$ which is a multivariate normal distribution allowing a quadratic form of $\boldsymbol{\beta}$, we propose a quadratic

piecewise approximation:

$$\log\left(1+\exp\left(\frac{y_i-\mathbf{X}_i^T\boldsymbol{\beta}}{b}\right)\right)\approx\rho_i\frac{y_i-\mathbf{X}_i^T\boldsymbol{\beta}}{b}+\zeta_i\left(\frac{y_i-\mathbf{X}_i^T\boldsymbol{\beta}}{b}\right)^2,$$

where ρ_i and ζ_i are the corresponding piecewise approximation coefficients depending on the value of $y_i-\mathbf{X}_i^T\boldsymbol{\beta}/b$. We present more details for the derivation of update equations in Appendix C) and the illustration of the proposed piecewise approximations in Appendix D).

3.3.1 Update equations and the VB algorithm

The optimal variational densities of $\boldsymbol{\beta}$ and b, $q^*(\boldsymbol{\beta})$ and $q^*(b)$, which are the corresponding approximated posterior distributions, are given as follows:

$q^*(\boldsymbol{\beta})$ is a $N_p(\boldsymbol{\mu},\Sigma)$ density function, and

$q^*(b)$ is an Inverse-Gamma(α,ω) density function,

where the parameters $\boldsymbol{\mu},\Sigma,\alpha$ and ω are obtained or updated according to Algorithm 2 (see derivation details in Appendix C) and ρ_i,ζ_i and φ_i are the piecewise approximation coefficients with formulas provided in Appendix D).

3.3.2 ELBO calculation

Our goal is to find $q^*(\cdot)$ by maximizing the ELBO. The ELBO in our model can be derived as follows:

$$\text{ELBO}(q)=\mathbb{E}_q[\log p(\mathbf{D},\boldsymbol{\beta},b)]-\mathbb{E}_q[\log q(\boldsymbol{\beta},b)],$$

where $\log p(\mathbf{D},\boldsymbol{\beta},b)=\log p(\mathbf{D}\,|\,\boldsymbol{\beta},b)+\log p(\boldsymbol{\beta})+\log p(b)$ and $\log q(\boldsymbol{\beta},b)=\log q(\boldsymbol{\beta})+\log q(b)$.

Algorithm 2: Variational Bayes Inference of Survival Data using a Log-logistic AFT Model

Data: a sample of independent log observed time y_i, their corresponding covariate vectors $\mathbf{X}_i$ and the right censoring indicator $\delta_i, i = 1, 2, \cdots, n$, where n is the sample size; values of hyperparameters: $\boldsymbol{\mu}_0, \sigma_0^2$, α_0 and ω_0; convergence threshold γ and maximum number of iterations M

Result: posterior distributions of $\boldsymbol{\beta}$ and b and their parameters: $\Sigma, \boldsymbol{\mu}, \alpha, \omega$

Initialization: initialize $\omega = \omega_0$ and $\boldsymbol{\mu} = \boldsymbol{\mu}_0$, set $m = 0$ and $ELBO = 0$;

Calculation: obtain α by $\alpha = \alpha_0 + r$ with $r = \sum_{i=1}^{n} \delta_i$;

while *iteration* $m < M$ *and difference of ELBO* $> \gamma$ **do**

$m = m + 1$;

$$\Sigma^{(m)} \leftarrow \left[v_0 \mathbf{I} + 2\mathbb{E}_{q(b)}\Big(\frac{1}{b^2}\Big) \sum_{i=1}^{n} (1+\delta_i)\zeta_i \mathbf{X}_i \mathbf{X}_i^T \right]^{-1};$$

$$\boldsymbol{\mu}^{(m)} \leftarrow \left[\left\{ v_0 \boldsymbol{\mu}_0^T + \sum_{i=1}^{n} \left(\mathbb{E}_{q(b)}\Big(\frac{1}{b}\Big)\Big(-\delta_i + (1+\delta_i)\rho_i \Big)\mathbf{X}_i^T + 2\mathbb{E}_{q(b)}\Big(\frac{1}{b^2}\Big)(1+\delta_i) y_i \zeta_i \mathbf{X}_i^T \right) \right\} \Sigma^{(m)} \right]^T ;$$

$$\omega^{(m)} \leftarrow \omega_0 - \sum_{i=1}^{n} \Big(\delta_i - (1+\delta_i)\varphi_i \Big)\Big(y_i - \mathbf{X}_i^T \boldsymbol{\mu}^{(m)} \Big);$$

calculate the current ELBO, $\text{ELBO}^{(m)}$;

calculate the difference of $\text{ELBO} = \text{ELBO}^{(m)} - \text{ELBO}^{(m-1)}$;

end

Let $diff_{\boldsymbol{\beta}} = \mathbb{E}_q[\log p(\boldsymbol{\beta})] - \mathbb{E}_q[\log q(\boldsymbol{\beta})]$ and $diff_b = \mathbb{E}_q[\log p(b)] - \mathbb{E}_q[\log q(b)]$, then

$$\text{ELBO}(q) = \mathbb{E}_q[\log p(\mathbf{D} \,|\, \boldsymbol{\beta}, b)] + diff_{\boldsymbol{\beta}} + diff_b. \tag{3.8}$$

With some algebraic manipulations (see details in Appendix A), we have

$$\mathbb{E}_q[\log p(\mathbf{D} \,|\, \boldsymbol{\beta}, b)] \overset{+}{\approx} -r\mathbb{E}_{q(b)}\Big(\log b \Big) + \mathbb{E}_{q(b)}\Big(\frac{1}{b}\Big) \sum_{i=1}^{n} (\delta_i - (1+\delta_i)\varphi_i)\,(y_i - \mathbf{X}_i^T \boldsymbol{\mu}),$$

$$diff_{\boldsymbol{\beta}} \overset{+}{\approx} -\frac{1}{2} v_0 [\text{trace}(\Sigma) + (\boldsymbol{\mu} - \boldsymbol{\mu}_0)^T (\boldsymbol{\mu} - \boldsymbol{\mu}_0)] + \frac{1}{2} \log(|\Sigma|),$$

$$diff_b \overset{+}{\approx} (\alpha - \alpha_0)\mathbb{E}_{q(b)}(\log b) + (\omega - \omega_0)\mathbb{E}_{q(b)}(\frac{1}{b}) - \alpha \log \omega.$$

3.3.3 Expectations

In what follows, we calculate the expectations in the update equations in Algorithm 2 in Section 3.3.1 and the ELBO calculations. All the expectations are taken with respect to the approximated variational distributions. Since $q(b)$ is an Inverse-Gamma(α, ω), we have

$$\mathbb{E}_{q(b)}\Big(\frac{1}{b}\Big) = \frac{\alpha}{\omega},$$

$$\mathbb{E}_{q(b)}\Big(\frac{1}{b^2}\Big) = \mathbb{E}_{q(b)}\Big[\Big(\frac{1}{b}\Big)^2\Big] = \mathrm{Var}_{q(b)}\Big[(\frac{1}{b})\Big] + \Big[\mathbb{E}_{q(b)}(\frac{1}{b})\Big]^2 = \frac{\alpha}{\omega^2} + \frac{\alpha^2}{\omega^2} = \frac{\alpha + \alpha^2}{\omega^2},$$

$$\mathbb{E}_{q(b)}(\log b) = \log(\omega) - \Psi(\alpha),$$

where Ψ is the digamma function defined as $\Psi(x) = \frac{d}{dx}\log\Gamma(x)$.

3.4 Simulation studies

We conduct simulation studies under various scenarios with different sample sizes and censoring percentages to assess the performance of the proposed VB algorithm (i.e., Algorithm 2 in Section 3.3.1).

3.4.1 Simulation scenarios and performance metrics

We generate the log of survival time for the i^{th} subject, $\log(T_i)$, $i = 1, ..., n$, as follows:

$$\log(T_i) = 0.5 + 0.2x_{i1} + 0.8x_{i2} + 0.8z_i,$$

where x_{i1}, x_{i2}, and z_i are mutually independently generated with $x_{i1} \sim N(1, 0.2^2)$, $x_{i2} \sim$ Bernoulli(0.5) and $z_i \sim$ logistic$(0, 1)$. The censoring time for the i^{th} subject, C_i, is generated from a uniform distribution, uniform$(0, u)$, where u is a positive value controlling the percentage of censoring. Then $t_i = \min(T_i, C_i)$ and $\delta_i = \mathbb{1}(T_i \leq C_i)$. Take $u = 48$ to achieve a 15% censoring rate and $u = 17$ to achieve a 30% censoring rate in our simulations.

In the first study, we consider sample sizes of $n = 300$ and $n = 600$, and varying censoring percentages of 0%, 15%, and 30%. These combinations yield a total of six distinct scenarios. We consider a prior setting with $\boldsymbol{\mu}_0 = (0, 0, 0)^T$, $v_0 = 0.1$, $\alpha_0 = 11$ and $\omega_0 = 10$, which indicates no strong prior information on the parameters. The ELBO convergence threshold is set as 0.01 which is the default recommendation (Yao et al., 2018), and the maximum number of iterations is 100. The performance of our VB algorithm are compared against that from the likelihood-based survival regression, *survreg* in the R package *survival* (Therneau and Grambsch, 2000; Therneau, 2023) and from the MCMC-based algorithm, the Hamiltonian Monte Carlo (HMC) sampling in the R package *rstan* (Stan Development Team, 2023).

The second study is designed to assess the performance of the proposed VB algorithms when the sample size is small. When the sample size is small, the likelihood-based estimation methods may fail to achieve satisfactory results. We change the sample size to $n = 30$ from $n = 300$ or 600 in the previous study to evaluate the proposed method for the performance with a small sample size. We also consider a different prior setting with $\boldsymbol{\mu}_0 = (0.3, 0.1, 1.0)^T$, $v_0 = 0.15$, $\alpha_0 = 11$ and $\omega_0 = 8$, which indicates partial information about the hyperparameters is known, although they do not precisely match the true parameter values.

We conduct $N = 500$ runs (replicates) for each scenario. In each of the 500 replicates, we apply our proposed method to derive an approximate posterior distribution for each parameter. The mean of the posterior distribution serves as our parameter estimate. The

empirical bias and sample standard deviation (SD) as well as the empirical mean squared error (MSE) for each estimate are obtained, where

$$\text{MSE} = \frac{\sum_{i=1}^{N}(\theta_0 - \hat{\theta}_i)^2}{N},$$

and $\hat{\theta}_i$ is the estimate of parameter θ in the i^{th} replicate, and θ_0 is the true value.

In Bayesian statistics, we also assess estimation accuracy by comparing the advertised coverage of approximate credible intervals to their true proposed coverage. We compute 95% credible intervals for each parameter in 500 replicates. We prefer equal-tailed intervals (ETI) for fixed effects ($\boldsymbol{\beta}$) and highest density intervals (HDI) for the scale parameter (b) due to the Inverse-Gamma distribution's asymmetry, as suggested by Kruschke (2015). We also calculate the average interval length from these replicates to gauge estimation precision. For comparison, we contrast the empirical credible interval coverage obtained through VB and MCMC with the empirical confidence interval coverage derived from likelihood estimations using the *survreg* method.

High computational cost is a common issue in MCMC-based Bayesian inference algorithms. We compare the performance of our VB algorithm with the MCMC-based HMC algorithm with respect to total run time of 500 replicates. The HMC algorithm in *rstan* (Ashraf-Ul-Alam and Ali Khan, 2021) is employed to produce four chains with 2000 iterations for each chain. MCMC summaries are based on 4000 MCMC samples after a 1000 sample burn-in for each of the four chains and with the default thinning of 1. Both the VB and HMC algorithms are implemented within R version 4.2.2 on a computer running the Mac OS X operating system with 1.6 GHz CPU and 8 GB RAM.

3.4.2 Simulation results

The numerical results from the first study are presented in Table 3.1. The empirical bias, SD and MSE pertaining to parameters β_2 and b exhibit notable similarity among all three

methods in all the scenarios. The proposed VB algorithm has smaller empirical standard deviation but similar bias, and, therefore, smaller MSE for parameters β_0 and β_1 than those of *survreg* under all considered scenarios. The empirical MSEs from the VB method are approximately 5.8% smaller for β_0 and 6.1% smaller for β_1 than that of *survreg*. This advantage is sustained even when compared to MCMC with a sample size of 300, exhibiting empirical MSE reductions of approximate 9.6% for β_0 and 10.5% for β_1. When the sample size is 600, the proposed VB algorithm provides similar MSEs for parameters β_0 and β_1 with MCMC in each scenario with different censoring percentages. The 95% coverage rates yielded by all three methods exhibit remarkable consistency and closely align with the expected credible or confidence level of 0.95, ranging from 0.93 to 0.96.

Table 3.2 presents the run time required in minutes for 500 replicates for the proposed VB method and MCMC under each scenario. We see that the VB algorithm is approximately 300 times faster than MCMC.

As expected, the sample size and censoring percentage affect the MSEs. The MSE experiences an increase with higher censoring percentages and a decrease as the sample size increases. Through empirical observation, the proposed VB algorithm exhibits analogous asymptotic properties when compared to both MCMC and the likelihood-based method. To visually capture the distribution of parameter estimates across the three methods, we present side-by-side boxplots in Figure 3.1 for each parameter, considering sample sizes of $n = 300$ and $n = 600$.

Table 3.1: Results for the first simulation study. A comparison of numerical estimation results including the empirical Bias, sample SD, MSE, coverage rate (CR) and average interval length (AL), from our VB method, the *survreg* and MCMC method under different sample sizes (n) and censoring percentages (p).

			VB algorithm					*survreg*					MCMC				
n	p		Bias	SD	MSE	CR[1]	AL	Bias	SD	MSE	CR[2]	AL	Bias	SD	MSE	CR[1]	AL
300	0%	β_0	0.017	0.410	0.168	95	1.59	0.023	0.423	0.179	94	1.63	0.018	0.426	0.182	95	1.65
		β_1	-0.013	0.393	0.154	95	1.53	-0.020	0.405	0.164	95	1.57	-0.017	0.412	0.170	95	1.59
		β_2	-0.002	0.161	0.026	94	0.62	0.001	0.161	0.026	94	0.63	0.006	0.161	0.026	95	0.63
		b	0.001	0.038	0.001	96	0.16	-0.004	0.037	0.001	95	0.15	0.004	0.037	0.001	96	0.15
	15%	β_0	0.011	0.412	0.170	95	1.62	0.018	0.426	0.181	94	1.65	0.002	0.434	0.188	96	1.68
		β_1	-0.008	0.398	0.158	95	1.56	-0.014	0.411	0.169	95	1.59	0.003	0.419	0.175	95	1.61
		β_2	-0.003	0.163	0.027	94	0.63	0.001	0.164	0.027	94	0.63	-0.006	0.165	0.027	95	0.64
		b	0.002	0.041	0.002	96	0.18	-0.004	0.041	0.002	95	0.16	0.003	0.040	0.002	96	0.16
	30%	β_0	0.012	0.421	0.177	95	1.65	0.021	0.440	0.194	94	1.71	0.001	0.448	0.200	96	1.74
		β_1	-0.013	0.404	0.163	95	1.60	-0.017	0.423	0.179	94	1.65	0.006	0.431	0.186	95	1.68
		β_2	-0.015	0.165	0.027	94	0.65	-0.003	0.168	0.028	93	0.66	-0.001	0.171	0.029	95	0.67
		b	-0.003	0.045	0.002	96	0.19	-0.006	0.045	0.002	95	0.18	0.006	0.045	0.002	95	0.18
600	0%	β_0	-0.015	0.308	0.095	93	1.13	-0.012	0.312	0.097	94	1.15	-0.010	0.306	0.094	94	1.16
		β_1	0.013	0.299	0.089	94	1.09	0.010	0.303	0.092	94	1.11	0.009	0.296	0.088	94	1.11
		β_2	-0.001	0.113	0.013	94	0.44	-0.001	0.113	0.013	95	0.44	0.002	0.113	0.013	95	0.44
		b	-0.002	0.028	0.001	95	0.12	-0.003	0.027	0.001	94	0.11	0.002	0.027	0.001	95	0.11
	15%	β_0	-0.015	0.316	0.100	94	1.15	-0.011	0.321	0.103	93	1.17	0.017	0.307	0.095	96	1.18
		β_1	0.011	0.306	0.094	94	1.11	0.008	0.311	0.097	93	1.13	-0.009	0.301	0.092	96	1.14
		β_2	-0.001	0.114	0.013	95	0.45	0.001	0.114	0.013	95	0.45	-0.004	0.113	0.013	95	0.45
		b	-0.002	0.031	0.001	95	0.13	-0.004	0.029	0.001	94	0.12	0.003	0.029	0.001	96	0.12
	30%	β_0	-0.018	0.315	0.100	94	1.18	-0.014	0.325	0.106	94	1.21	0.018	0.327	0.098	96	1.22
		β_1	0.014	0.306	0.094	94	1.14	0.013	0.316	0.100	93	1.17	-0.009	0.304	0.093	95	1.18
		β_2	-0.010	0.117	0.014	94	0.46	0.002	0.119	0.014	95	0.47	-0.002	0.116	0.013	95	0.47
		b	-0.004	0.033	0.001	96	0.14	-0.004	0.032	0.001	95	0.13	0.005	0.031	0.001	96	0.13

[1]Empirical coverage rate corresponding to a 95% credible interval for VB and MCMC

[2]Empirical coverage rate corresponding to a 95% confidence interval for *survreg*

Table 3.2: Results for the first simulation study. Times in minutes for 500 replicates from the VB and MCMC algorithms, respectively, under scenarios with different sample sizes (n) and censoring percentages (p). The corresponding ratio of MCMC's time to VB's is also calculated and presented.

n	300			600		
p	0%	15%	30%	0%	15%	30%
VB	1.72	1.96	2.07	2.81	3.09	3.18
MCMC	544.53	549.64	581.22	1064.22	1071.30	1109.06
Ratio	317	280	281	379	347	349

When the sample size is small, as we considered in the second study, the MCMC provides similar estimation results as the VB method but is substantially more time-intensive in contrast to VB. We focus on the comparison between the likelihood-based *survreg* method and the VB algorithm, shown in Table 3.3. We observe in Table 3.3 that when the sample size is 30, VB consistently yields smaller MSEs across both weak and strong prior settings when contrasted with *survreg*. Specifically, within the weak prior setting, VB achieves reductions in MSEs of approximately 46.6% for β_0, 46.5% for β_1, 8.2% for β_2, and 42.2% for b, relative to the corresponding estimates obtained via *survreg*. In the strong prior setting, the reductions in MSEs are more substantial, amounting to approximately 63.4% for β_0, 63.5% for β_1, 15.1% for β_2, and 39.1% for b. We see that both VB and *survreg* exhibit similar empirical bias for each parameter. However, estimates derived from the *survreg* method are characterized by greater sample SDs, consequently leading to larger MSEs. Compared with the results in the weak prior setting, the VB method with useful prior information exhibits superior performance in estimating the regression coefficients (i.e., β's) with smaller MSEs.

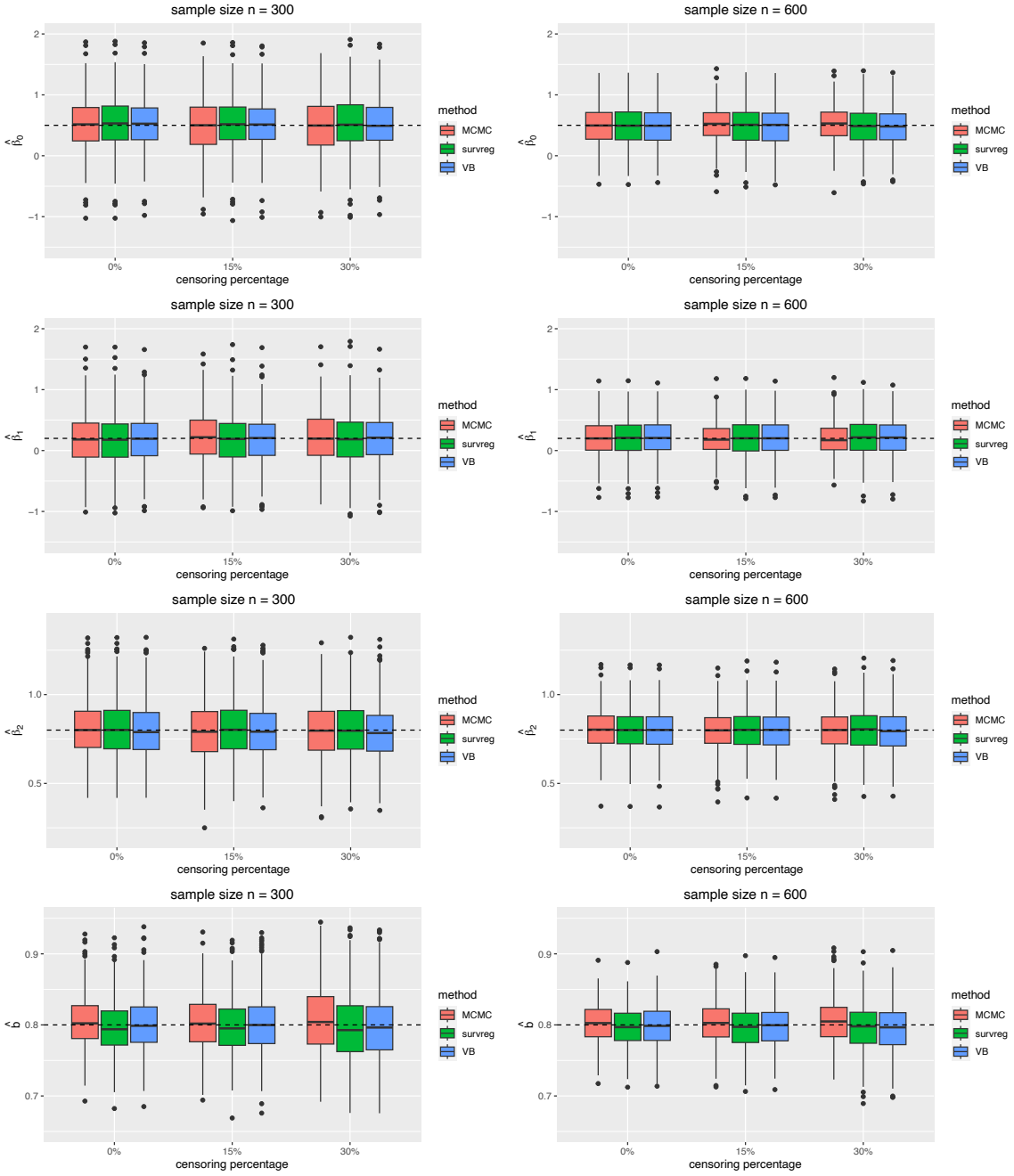


Figure 3.1: Results for the first simulation study. A comparison of results from our VB method, the *survreg* and MCMC via boxplots. The horizontal dashed line on each plot represents the true value of the corresponding parameter.

Table 3.3: Results for the second simulation study. A comparison of numerical estimation results including the empirical Bias, sample SD, MSE, coverage rate (CR) and average interval length (AL), from our VB method with two prior settings, the *survreg* under a small sample size ($n = 30$) with different censoring percentages (p).

			VB algorithm + weak prior[1]					VB algorithm + strong prior[2]					*survreg*				
n	p		Bias	SD	MSE	CR[3]	AL	Bias	SD	MSE	CR[3]	AL	Bias	SD	MSE	CR[4]	AL
30	0%	β_0	-0.055	1.102	1.214	95	4.45	-0.043	0.917	0.841	95	3.87	-0.034	1.473	2.167	91	5.18
		β_1	0.047	1.054	1.111	95	4.28	0.023	0.877	0.768	96	3.73	0.020	1.410	1.984	92	4.97
		β_2	-0.081	0.500	0.256	92	1.93	-0.055	0.486	0.239	91	1.82	-0.067	0.521	0.275	92	1.97
		b	0.010	0.106	0.011	96	0.48	-0.032	0.106	0.012	94	0.45	-0.036	0.130	0.018	91	0.47
	15%	β_0	-0.066	1.101	1.214	95	4.51	-0.054	0.912	0.833	95	3.90	-0.047	1.501	2.250	92	5.26
		β_1	0.057	1.057	1.118	95	4.34	0.029	0.875	0.766	95	3.75	0.033	1.438	2.064	92	5.06
		β_2	-0.081	0.503	0.259	93	1.96	-0.056	0.489	0.242	91	1.83	-0.063	0.526	0.280	92	2.00
		b	0.012	0.109	0.012	96	0.51	-0.035	0.109	0.013	94	0.48	-0.041	0.138	0.021	91	0.50
	30%	β_0	-0.067	1.107	1.228	95	4.62	-0.057	0.909	0.828	96	3.96	-0.051	1.558	2.426	92	5.49
		β_1	0.057	1.056	1.115	95	4.46	0.026	0.866	0.749	96	3.82	0.041	1.484	2.201	92	5.29
		β_2	-0.083	0.513	0.270	94	2.04	-0.061	0.492	0.245	93	1.89	-0.057	0.545	0.300	92	2.09
		b	0.018	0.116	0.014	96	0.55	-0.031	0.116	0.014	95	0.51	-0.038	0.155	0.025	90	0.56

[1]Weak prior setting: $\boldsymbol{\mu}_0 = (0, 0, 0)^T$, $v_0 = 0.1$, $\alpha_0 = 11$ and $\omega_0 = 10$

[2]Strong prior setting: $\boldsymbol{\mu}_0 = (0.3, 0.1, 1.0)^T$, $v_0 = 0.15$, $\alpha_0 = 11$ and $\omega_0 = 8$

[3]Empirical coverage rate corresponding to a 95% credible interval for VB and MCMC

[4]Empirical coverage rate corresponding to a 95% confidence interval for *survreg*

3.5 Application to real data

In this section, we apply our proposed VB algorithm in Section 3.3.1 to a real data set, *rhDNASE*, which is publicly available in the R package *survival*. The data, first introduced in Fuchs et al. (1994) and further analyzed in Therneau and Hamilton (1997), were used to investigate the effect of recombinant human deoxyribonuclease I (rhDNase) on pulmonary function among patients with cystic fibrosis. The rhDNase can digest extracellular DNA released by leukocytes that accumulate in the airways in response to chronic bacterial infection. Therefore, administering rhDNase would reduce the incidence of exacerbation and improve lung function. Among 645 subjects, 324 were randomly assigned to the Placebo group, and the rest were assigned to the treatment group (i.e., the rhDNase group). The event time, T, was defined as the time until the first pulmonary exacerbation, and the follow-up period was 169 days. The forced expiratory volume (FEV) at enrollment was considered a risk factor (i.e., covariate) measuring lung capacity. In Lawless (2003), a log-logistic AFT model was applied to this data set, and estimates were obtained by maximizing the likelihood. Model diagnostic in Lawless (2003) shows that the parametric assumption that the event time follows a log-logistic distribution was satisfied. Therefore, we want to fit the AFT regression model:

$$\log(T) := Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + bz,$$

where $x_1 = \mathbb{1}(\text{treatment} = \text{rhDNase})$ with $\mathbb{1}$ being an indicator function, x_2 is the FEV, and z follows a standard logistic distribution with a scale parameter b. We follow the Bayesian paradigm to make inference about the unknowns in the AFT model, and the approximation to the posterior distribution is performed following the proposed VB algorithm.

Unlike simulation studies, we do not have informative priors in this real data set. However, we can choose priors using historical data and similar analyses on this type of data. In

a similar study by Shah and Hodson (1996) on the effect of rhDNase on improving lung function, researchers found that daily treatment of rhDNase could reduce the risk of developing an exacerbation by 28%. That is, a daily administration of rhDNase can prolong the occurrence of an exacerbation by 28%. Therefore, we can choose $\log(1.28) \approx 0.25$ as the prior mean of β_1. Similarly, based on Block et al. (2006), the odds ratio of developing an exacerbation with one unit increase of FEV is 0.96, which indicates the corresponding time to an exacerbation occurrence increase by 4%. Therefore, we can choose $\log(1.04) \approx 0.04$ as the prior mean of β_2. For the mean of the intercept (i.e., β_0) prior distribution, we can choose the log of half of the follow-up period length, $\log(169/2) \approx 4.4$. For the precision hyperparameter v_0, we use a low precision, with $v_0 = 1$, to obtain a flat prior. For the prior of the scale parameter, we use $\alpha_0 = 501$ and $\omega_0 = 500$ to have a mean scale of one. To summarize, we consider the following prior distributions for the model parameters:

- $\boldsymbol{\beta} \sim N_p(\boldsymbol{\mu}_0, \sigma_0^2 I_{p\times p})$ with $\boldsymbol{\mu}_0 = (4.4, 0.25, 0.04)^T$ and $v_0 = 1/\sigma_0^2 = 1$
- $b \sim$ Inverse-Gamma (α_0, ω_0) with $\alpha_0 = 501$ and $\omega_0 = 500$.

We compared the estimation results obtained using our proposed VB algorithm to those from the MCMC-based HMC algorithm and the likelihood-based survival regression, *survreg*, as shown in Table 3.4. The convergence of the MCMC algorithm was well assessed and checked by the trace plot and autocorrelation plot (Ashraf-Ul-Alam and Ali Khan, 2021). Remarkably, all three methods exhibited a strong agreement in both point and interval estimations of each parameter. Figure 3.2 depicts the approximated posterior densities of each parameter obtained from MCMC and VB, further confirming a strong agreement in the estimation of regression coefficients and the scale parameter. Notably, the computational efficiency of the proposed VB algorithm was outstanding, completing in only 0.88 seconds, whereas the MCMC method took 2.56 minutes, making it over 170 times slower than VB.

Based on the results from our VB method, the estimated coefficient of the treatment, rhD-

Nase, is 0.416 with a 95% credible interval of $[0.139, 0.692]$, indicating that rhDNase can significantly prolong the time to the first pulmonary exacerbation. Furthermore, the acceleration factor is $\exp(0.416) \approx 1.516$ with a 95% credible interval of [1.1491.998] for a patient treated with rhDNase. The time to the first pulmonary exacerbation of a patient treated with rhDNase is therefore delayed by a factor of about 1.5 compared to a patient from the placebo group with the same FEV under a log-logistic AFT model. Besides, FEV is a significant risk factor on the event time, with an estimated coefficient of 0.021 (95% credible interval $[0.016, 0.027]$). The acceleration factor of FEV is $\exp(0.021) \approx 1.021$, meaning that one unit increase in FEV would delay the event time by 2.1% with a 95% credible interval of $[1.6\%, 2.7\%]$. Our results from the VB algorithm highly agree with the results obtained by *survreg* and the MCMC algorithm.

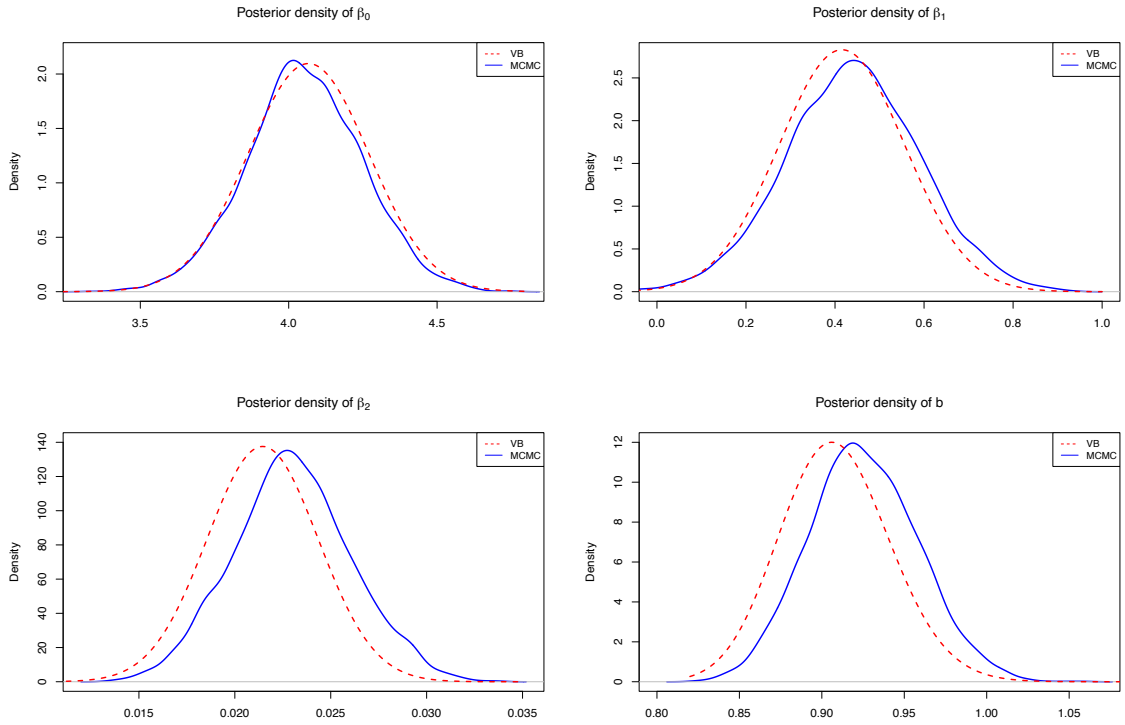


Figure 3.2: Results from analysis on rhDNASE data. Approximated posterior density for each parameter (dashed red line for VB and solid blue line for MCMC).

Table 3.4: Results from analysis on rhDNASE data. Posterior means (Mean) with posterior standard deviations (SD), and 95% credible intervals (95% Cred. Int.) from our proposed VB algorithm and MCMC, respectively. Point estimates (Est.) with standard errors (SE), and 95% confidence interval (95% Conf. Int.) from *survreg* in the R package *survival*.

	VB algorithm			*survreg*			MCMC		
	Mean	SD	95% Cred. Int.[1]	Est.	SE	95% Conf. Int.[2]	Mean	SD	95% Cred. Int.[3]
β_0	4.113	0.190	[3.740, 4.486]	4.086	0.175	[3.743, 4.429]	4.046	0.198	[3.650, 4.424]
β_1	0.416	0.141	[0.139, 0.692]	0.402	0.130	[0.146, 0.657]	0.440	0.147	[0.165, 0.737]
β_2	0.021	0.003	[0.016, 0.027]	0.021	0.003	[0.015, 0.026]	0.023	0.003	[0.017, 0.030]
b	0.908	0.033	[0.844, 0.974]	0.796	0.045[4]	[0.712, 0.891]	0.926	0.033	[0.866, 0.995]

[1] 95% Cred. Int.: highest density interval (HDI) was applied. Note that for a symmetric distribution, HDI is the same as the equal-tailed interval.

[2] 95% Conf. Int.: for regression coefficient estimates, the likelihood-based confidence interval was used, while for the scale estimate, the Wald-based interval was used.

[3] 95% Cred. Int.: for MCMC, we obtained the credible interval based on the percentiles of the sample from the posterior distribution.

[4] Standard error (SE) for the scale estimate is not available for *survreg* in the R package, but the SE for log scale which is 0.0570, is provided. We calculated the SE for the scale estimate via Delta method.

3.6 Discussion

This chapter introduces a novel inference approach for the log-logistic AFT model, offering an alternative to the MCMC-based Bayesian algorithm for modeling survival data. The study utilizes mean-field variational Bayes (VB) and applies coordinate ascent variational inference to formulate update equations within the VB framework. To achieve conjugacy under the Bayesian paradigm, the linear and quadratic piecewise approximations are embedded in the update equations for parameters. Simulation studies and the application to a real data set show that our proposed VB algorithm provides satisfactory results.

Our proposed VB approach presents several notable advantages. Similar to other Bayesian methods, our proposed VB technique accommodates the integration of prior information obtained from historical data or related studies, which is more particular in clinical research. Our VB algorithm is particularly prominent in the small sample scenario where the typical likelihood-based methods may not work well. The proposed VB algorithm performs well under a large sample size and a weak prior setting, while also having a significantly lower computational cost compared to MCMC.

In principle, VB can be applied to the AFT regression model with other different censoring schemes, including left censored and interval censored data. However, such adaptations for a log-logistic AFT model with different censoring schemes necessitate adjustments to the likelihood function and thus to the update equation for each variational density. We anticipate that more extensive modification or a different approach altogether may be required if we consider alternative parametric distributions for survival data, for example, such as the log-normal distribution, which lacks a closed-form survival function.

It is important to note that, while closed-form update equations are obtained through the CAVI algorithm with a piecewise approximation technique, alternative variational inference procedures such as the stochastic variational inference (Hoffman et al., 2013) and the

black-box variational inference (Ranganath et al., 2014; Murphy, 2023) may be considered. These types of variational inference are gradient-based which can be implemented stochastically, for example, using the automatic variational inference algorithm (Kucukelbir et al., 2015). However, potential challenges associated with large variance in such gradient-based algorithms need to be addressed. For instance, in the black-box variational inference, Ranganath et al. (2014) applied a variance-control method to reduce the variance of the score function estimator of the ELBO gradient. A more comprehensive discussion can be found in Murphy (2023). Numerical calculations of the gradients when implementing the gradient-based variational inference are available via some autograd based Python libraries such as Pyro (Bingham et al., 2019) and Tensorflow (Dillon et al., 2017).

To the best of our knowledge, our work stands as a pioneering effort in the application of Bayesian variational inference to model survival data via a log-logistic AFT regression. In addition, we introduce a piecewise approximation technique within the VB algorithm to obtain closed-form update equations. The piecewise approximations have been widely used in other optimization problems to overcome the issue of intractable calculations (Stein, 1995; Powell et al., 2004; Rewieński and White, 2006; Zhou et al., 2020; Asghari and Fathollahi-Fard, 2022). The piecewise polynomial approximation in the update equations is shown to work well based on our simulation studies. This approximation provides a new insight to apply Bayesian variational inference under complex models to achieve conjugacy.

References

Abubakar, J., M. A. A. Abdullah, and O. R. Olaniran (2023). Variational Bayesian inference for exponentiated Weibull right censored survival data. *Statistics, Optimization & Information Computing 11*(4), 1027–1040. → page 75

Asghari, M. and A. Fathollahi-Fard (2022). Transformation and linearization techniques in optimization: A State-of-the-Art survey. *Mathematics 10*, 283. → page 96

Ashraf-Ul-Alam, M. and A. Ali Khan (2021). Comparison of accelerated failure time models: A Bayesian study on head and neck cancer data. *Journal of Statistics Applications & Probability 10*(3), 715–738. → page 85, 92

Bingham, E., J. P. Chen, M. Jankowiak, F. Obermeyer, N. Pradhan, T. Karaletsos, R. Singh, P. Szerlip, P. Horsfall, and N. D. Goodman (2019). Pyro: Deep universal probabilistic programming. *The Journal of Machine Learning Research 20*(1), 973–978. → page 96

Bishop, C. (2006). *Pattern Recognition and Machine Learning*. Springer. → page 75, 78, 79, 80

Blei, D. M., A. Kucukelbir, and J. D. McAuliffe (2017). Variational inference: A review for statisticians. *Journal of the American Statistical Association 112*(518), 859–877. → page 75, 78, 79

Block, J. K., K. L. Vandemheen, E. Tullis, D. Fergusson, S. Doucette, D. Haase, Y. Berthiaume, N. Brown, P. Wilcox, P. Bye, S. Bell, M. Noseworthy, L. Pedder, A. Freitag, N. Paterson, and S. D. Aaron (2006). Predictors of pulmonary exacerbations in patients with cystic fibrosis infected with multi-resistant bacteria. *Thorax 61*, 969–974. → page 92

Cox, D. R. (1972). Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological) 34*(2), 187–202. → page 74

Dillon, J. V., I. Langmore, D. Tran, E. Brevdo, S. Vasudevan, D. Moore, B. Patton, A. Alemi, M. Hoffman, and R. A. Saurous (2017). Tensorflow distributions. *arXiv preprint arXiv:1711.10604*. → page 96

Faes, C., J. T. Ormerod, and M. P. Wand (2011). Variational Bayesian inference for parametric and nonparametric regression with missing data. *Journal of the American Statistical Association 106*(495), 959–971. → page 75, 79

Fuchs, H. J., D. S. Borowitz, D. H. Christiansen, E. M. Morris, M. L. Nash, B. W. Ramsey, B. J. Rosenstein, A. L. Smith, and M. E. Wohl (1994). Effect of aerosolized recombinant human DNase on exacerbations of respiratory symptoms and on pulmonary function in patients with cystic fibrosis. *New England Journal of Medicine 331*(10), 637–642. → page 91

Gelman, A., J. B. Carlin, H. S. Stern, and D. B. Rubin (2004). *Bayesian Data Analysis*. Chapman and Hall/CRC. → page 79

Hoffman, M. D., D. M. Blei, C. Wang, and J. Paisley (2013). Stochastic variational inference. *Journal of Machine Learning Research*. → page 76, 95

Ibrahim, J. G., M.-H. Chen, and D. Sinha (2001). *Bayesian Survival Analysis*. New York: Springer. → page 74, 75

Jordan, M. I., Z. Ghahramani, T. Jaakkola, and L. Saul (1999). Introduction to variational methods for graphical models. *Machine Learning 37*, 183–233. → page 74, 78

Kalbfleisch, J. D. and R. L. Prentice (2002). *The Statistical Analysis of Failure Time Data*. Hoboken, N.J: J. Wiley. → page 74

Komárek, A. and E. Lesaffre (2008). Bayesian accelerated failure time model with multivariate doubly interval-censored data and flexible distributional assumptions. *Journal of the American Statistical Association 103*(482), 523–533. → page 74

Kruschke, J. K. (2015). Chapter 12 - Bayesian Approaches to Testing a Point ("Null") Hypothesis. In J. K. Kruschke (Ed.), *Doing Bayesian Data Analysis (Second Edition)* (Second Edition ed.)., pp. 335–358. Boston: Academic Press. → page 85

Kucukelbir, A., R. Ranganath, A. Gelman, and D. Blei (2015). Automatic variational inference in Stan. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett (Eds.), *Advances in Neural Information Processing Systems*, Volume 28, pp. 1–9. Curran Associates, Inc. → page 96

Kullback, S. and R. A. Leibler (1951). On information and sufficiency. *The Annals of Mathematical Statistics 22*(1), 79 – 86. → page 75

Lambert, P., D. Collett, A. Kimber, and R. Johnson (2004). Parametric accelerated failure time models with random effects and an application to kidney transplant survival. *Statistics in Medicine 23*(20), 3177–3192. → page 74

Lawless, J. F. (2003). *Statistical Models and Methods for Lifetime Data*. Hoboken, N.J: Wiley-Interscience. → page 74, 91

Longo, A., M. M. Bambo, and M. G. Gebremariam (2022). Statistical analysis on time to blindness of glaucoma patients at Jimma University Specialized Hospital: Application of accelerated failure time model. *Journal of Ophthalmology 2022*, 914–921. → page 74

Luts, J. and M. P. Wand (2015). Variational inference for count response semiparametric regression. *Bayesian Analysis 10*(4), 991 – 1023. → page 75

Murphy, K. P. (2023). *Probabilistic Machine Learning: Advanced Topics*. MIT Press. → page 76, 96

Patel, K., R. Kay, and L. Rowell (2006). Comparing proportional hazards and accelerated failure time models: An application in influenza. *Pharmaceutical Statistics 5*(3), 213–224. → page 74

Pham, T. H., J. T. Ormerod, and M. Wand (2013). Mean field variational Bayesian inference for nonparametric regression with measurement error. *Computational Statistics & Data Analysis 68*, 375–387. → page 75

Powell, W., A. Ruszczyński, and H. Topaloglu (2004). Learning algorithms for separable approximations of discrete stochastic optimization problems. *Mathematics of Operations Research 29*(4), 814–836. → page 96

Ranganath, R., S. Gerrish, and D. Blei (2014). Black box variational inference. In *Artificial Intelligence and Statistics*, pp. 814–822. PMLR. → page 76, 96

Ray, K. and B. Szabó (2022). Variational Bayes for high-dimensional linear regression with sparse priors. *Journal of the American Statistical Association 117*(539), 1270–1281. → page 75

Rewieński, M. and J. White (2006). Model order reduction for nonlinear dynamical systems based on trajectory piecewise-linear approximations. *Linear Algebra and Its Applications 415*(2-3), 426–454. → page 96

Rivas-López, M., R. Martín-Martín, and I. García-Camacha Gutiérrez (2022). Recent advances in robust design for accelerated failure time models with type I censoring. *Mathematics 10*(3), 379. → page 74

Shah, P. and M. Hodson (1996). New treatment strategies in cystic fibrosis: rhDNase. *Monaldi Archives for Chest Disease 51*(2), 125—129. → page 92

Stan Development Team (2023). RStan: the R interface to Stan. R package version 2.21.8. → page 84

Stein, D. W. (1995). Detection of random signals in gaussian mixture noise. *IEEE Transactions on Information Theory 41*(6), 1788–1801. → page 96

Tang, Y., X. Song, and G. Y. Yi (2022). Bayesian analysis under accelerated failure time models with error-prone time-to-event outcomes. *Lifetime Data Analysis 28*(1), 139–168. → page 74

Therneau, T. M. (2023). *A Package for Survival Analysis in R*. R package version 3.5-5. → page 84

Therneau, T. M. and P. M. Grambsch (2000). *Modeling Survival Data: Extending the Cox Model*. New York: Springer. → page 84

Therneau, T. M. and S. A. Hamilton (1997). rhDNase as an example of recurrent event analysis. *Statistics in Medicine 16*(18), 2029–2047. → page 91

Thiruvengadam, G., R. Ramanujam, and L. Marappa (2021). Modeling the recovery time of patients with coronavirus disease 2019 using an accelerated failure time model. *Journal of International Medical Research 49*(8), 1–7. → page 74

Wang, Y. and D. M. Blei (2019). Frequentist consistency of variational bayes. *Journal of the American Statistical Association 114*(527), 1147–1161. → page 75

Webber, C., M. Brundage, T. P. Hanna, C. M. Booth, E. Kennedy, W. Kong, Y. Peng, M. Whitehead, and P. A. Groome (2022). Explaining regional variations in colon cancer survival in Ontario, Canada: A population-based retrospective cohort study. *BMJ Open 12*(9), 1–11. → page 74

Wei, L. J. (1992). The accelerated failure time model: A useful alternative to the Cox

regression model in survival analysis. *Statistics in Medicine 11*(14-15), 1871–1879. → page 74

Weng, J., Y. Zheng, X. Yan, and Q. Meng (2014). Development of a subway operation incident delay model using accelerated failure time approaches. *Accident Analysis & Prevention 73*, 12–19. → page 74

Xian, C., C. P. de Souza, W. He, F. F. Rodrigues, and R. Tian (2024). Variational Bayesian analysis of survival data using a log-logistic accelerated failure time model. *Statistics and Computing 34*(2), 67. → page 74

Xian, C., C. P. E. de Souza, J. Jewell, and R. Dias (2024). Clustering functional data via variational inference. *Advances in Data Analysis and Classification*, 1–50. → page 75

Xu, D., S. Zhao, and J. Sun (2022). Regression analysis of dependent current status data with the accelerated failure time model. *Communications in Statistics - Simulation and Computation 51*(10), 6188–6196. → page 74

Yao, Y., A. Vehtari, D. Simpson, and A. Gelman (2018). Yes, but did it work?: Evaluating variational inference. In *Proceedings of the 35th International Conference on Machine Learning*, Volume 80 of *Proceedings of Machine Learning Research*, pp. 5581–5590. PMLR. → page 84

Zhang, J. and A. B. Lawson (2011). Bayesian parametric accelerated failure time spatial model and its application to prostate cancer. *Journal of Applied Statistics 38*(3), 591–603. → page 74

Zhou, S., X. Zhuo, Z. Chen, and Y. Tao (2020). A new separable piecewise linear learning algorithm for the stochastic empty container repositioning problem. *Mathematical Problems in Engineering 2020*, 1–16. → page 96

Chapter 4
Fast variational Bayesian inference for correlated survival data: an application to invasive mechanical ventilation duration analysis

4.1 Introduction

Correlated survival data are commonplace in various research contexts and have been extensively studied in the literature (Luo et al., 2013; Honerkamp-Smith and Xu, 2016; Liu et al., 2017) [1]. One such context of correlated survival data is observed in clustered survival data, which are derived from multiple entities such as families or hospitals. Within each cluster, survival data exhibit correlation because of the shared environmental factors, and random effects are often introduced to capture the shared characteristics within the cluster. Conditional on the random effects (frailty), the survival times within a cluster can be assumed independent, which leads to a shared frailty model to account for cluster-level uncertainty (Hougaard, 1995; Hanagal, 2011; Gorfine and Zucker, 2023).

Our research is motivated by data from intensive care units (ICUs) across multiple clinical centers in Ontario, Canada. The ICU data were provided by the Critical Care Information System (CCIS) Ontario database. In Canadian ICUs, invasive mechanical ventilation is prevalent, with approximately one third of patients requiring ventilation during their ICU stay (Canadian Institute for Health Information, 2020). The duration of ventilation has significant implications for clinical outcomes and is associated with an increased risk of complications (Canadian Institute for Health Information, 2020). Analysis on the ventilation duration for ICU patients utilizing the associated risk factors, such as patient categories

[1] A version of this chapter has been submitted (Xian et al., 2024b).

(e.g., medical or surgical), admission diagnoses, and patient severity, can help determine the number of beds with ventilators and therefore, support capacity planning and effective clinical resource management. The severity of patients in ICU is often assessed using the Multiple Organ Dysfunction Score (MODS) (Marshall et al., 1995) that evaluates organ function. Kobara et al. (2023) conducted a study on ventilation duration for ICU patients from Ontario using a survival analysis framework. In their study, patient ventilation duration is considered as a time-to-event (survival) outcome and parametric accelerated failure time (AFT) models are applied to predict the ventilation time. If a patient is transferred to another facility without subsequent follow-up information, the ventilation time is considered right censored. Kobara et al. (2023) found that the log-logistic AFT model well describes the association between risk factors and patients' ventilation duration in Ontario ICUs. However, Kobara et al. (2023) did not consider a possible correlation of ventilation duration times among patients within the same ICU. Previous studies have indicated that patient outcomes within the same ICU site may be correlated, and ignoring this hierarchical structure can result in flawed prediction models (Burgess Jr et al., 2000; Glance et al., 2003).

As aforementioned, shared frailty can be used to accommodate cluster-level association among patients within the cluster. Gorfine and Zucker (2023) recently provided a comprehensive review of shared frailty methods for complex survival data. To incorporate risk factors in survival data, the Cox proportional hazards (PH) model with a multiplicative shared frailty on the hazard rates has been widely developed (Gorfine and Zucker, 2023). As an alternative to the Cox PH model, AFT models offer an intuitive interpretation of covariate effects on survival time. Lambert et al. (2004) developed shared frailty AFT models with different parametric distributions assumed for the survival time and conducted maximum likelihood estimation by integrating out the unobserved frailties. They empirically found that the choice of distribution for the shared frailty is not critical, recommending the normal distribution. Do Ha et al. (2002) considered a shared frailty log-normal AFT

model and used the hierarchical-likelihood (h-likelihood) approach to estimate the model parameters. The h-likelihood method obtains fixed effects estimates by maximizing the h-likelihood, while utilizing the restricted maximum likelihood estimate for the estimation of the variance of the shared frailty (Do Ha et al., 2017). An R package, `frailtyHL` (Do Ha et al., 2012) has been developed to implement the h-likelihood estimation in the shared frailty log-normal AFT model. Building on this, Park and Do Ha (2019) introduced a penalized variable selection technique for the shared frailty log-normal AFT model. Additionally, Zhou et al. (2017) developed a Markov chain Monte Carlo (MCMC)-based algorithm, called *survregBayes*, to estimate the shared frailty AFT model, and an R package, `spBayesSurv` (Zhou et al., 2020), is available for its implementation. To our knowledge, there is no work on variational Bayesian inference for shared frailty AFT models.

As an alternative to MCMC methods, which are the gold standard for obtaining posterior distributions under a Bayesian framework, variational inference (VI) has gained popularity due to its favourable results and lower computational cost than MCMC. Recently, several types of VI algorithms have been developed, including mean-field VI (Bishop, 2006), stochastic VI (Hoffman et al., 2013), and black-box VI (Ranganath et al., 2014). A special case of mean-field VI, called mean-field variational Bayes (VB), arises when the Kullback–Leibler (KL) divergence is utilized to quantify the dissimilarity between exact and approximated posterior distributions. In addition, the approximated posterior distribution, referred to as the variational posterior, is assumed to belong to a mean-field variational family. Under the mean-field VB framework, the solutions to minimizing the KL divergence can be obtained by utilizing the coordinate ascent algorithm (Bishop, 2006; Jordan et al., 1999). Mean-field VB has been widely applied to regression models such as the generalized additive model (Neville et al., 2011), nonparametric regression with measurement error (Pham et al., 2013), count response semiparametric regression (Luts and Wand, 2015), high-dimensional linear regression (Ray and Szabó, 2022), multilevel regression modelling (Lee and Wand, 2016a), B-spline regression mixture model for functional data clustering

(Xian et al., 2024), basis selection for functional data representation (da Cruz et al., 2024), among others. Applications of other types of VI can be found in a comprehensive review of VI from a statistical perspective by Blei et al. (2017).

In this study, we propose a shared frailty log-logistic AFT model to account for the correlation among patient ventilation durations within the ICU sites. We develop a novel and fast mean-field VB algorithm to infer the model parameters. By applying the piece-wise approximation techniques proposed by Xian et al. (2024a) to avoid intractable calculations, we obtain closed-form posterior distributions. We conduct extensive simulation studies with various numbers of clusters and cluster sizes to evaluate the performance of the proposed method, and compare the performance of our VB algorithm with the h-likelihood method (Do Ha et al., 2017) and the MCMC-based algorithm *survregBayes* (Zhou et al., 2020). Finally, we apply our methodology to investigate ventilation duration for ICU patients using the same dataset as Kobara et al. (2023). This study was approved by the Research Ethics Review Committee at King's University College at Western University.

The remainder of this chapter is organized as follows. Section 4.2 presents the log-logistic AFT model with a shared frailty under the Bayesian framework. We introduce our proposed VB algorithm in Section 4.3. In Section 4.4, simulation studies are conducted to evaluate the performance of our VB algorithm. Section 4.5 illustrates the application of the proposed method to the ICU ventilation duration data. A discussion is provided in Section 4.6.

4.2 Bayesian log-logistic AFT model with a shared frailty

Let T_{ij} and C_{ij} be the survival and censoring times, respectively, of the j^{th} subject from the i^{th} group (i.e., cluster) in a sample, $i = 1, ..., K$ and $j = 1, ..., n_i$. Let $t_{ij} = \min(T_{ij}, C_{ij})$ and $\delta_{ij} = \mathbb{1}(T_{ij} \leq C_{ij})$ be the subject's observed time and the indicator for right censoring, respectively. We consider a log-logistic AFT model with shared frailty (a random intercept)

specified as follows:

$$\log(T_{ij}) = \gamma_i + \mathbf{X}_{ij}^T \boldsymbol{\beta} + b\epsilon_{ij}, \tag{4.1}$$

where $\mathbf{X}_{ij}$ is a column vector with length p, $p \geq 2$, containing $p-1$ fixed effects (covariates) and a constant one to incorporate the constant intercept (i.e., $\mathbf{X}_{ij} = (1, x_{ij1}, ..., x_{ij(p-1)})^T$), $\boldsymbol{\beta}$ is the corresponding vector of coefficients for the fixed effects, where γ_i is a random intercept for the i^{th} cluster and ϵ_{ij} is a random variable following a standard logistic distribution, and b is a scale parameter. We further assume that $\gamma_i \overset{\text{iid}}{\sim} N(0, \sigma_\gamma^2)$ representing the discrepancy between clusters with iid denoting identically and independent distributed. The survival time T_{ij} and censoring time C_{ij} are assumed independent given the covariates $\mathbf{X}_{ij}$. Our model follows a structure similar to one presented by Robinson (1991) and Nolan et al. (2020) for a Gaussian linear mixed effect model.

In our proposed Model (4.1), we incorporate the unknown and unobserved shared risk through cluster-specific random intercepts and estimate the model parameters, $\boldsymbol{\beta}$, b, and σ_γ^2 using a Bayesian framework by further assuming the following independent marginal prior distributions:

- $\boldsymbol{\beta} \sim N_p(\boldsymbol{\mu}_0, \sigma_0^2 \mathbf{I}_{p\times p})$ with precision $v_0 = 1/\sigma_0^2$ and $\mathbf{I}_{p\times p}$ being a $p \times p$ identity matrix
- $b \sim$ Inverse-Gamma (α_0, ω_0)
- $\gamma_i | \sigma_\gamma^2 \overset{\text{iid}}{\sim} N(0, \sigma_\gamma^2), i = 1, ..., K$
- $\sigma_\gamma^2 \sim$ Inverse-Gamma (λ_0, η_0)

where $\mu_0, v_0, \alpha_0, \omega_0, \lambda_0$ and η_0 are known hyperparameters (parameters of the prior distributions).

4.3 Variational Bayes algorithm

In what follows, we outline our methodology for deriving a mean-field VB algorithm for Model (4.1). We summarize the resulted VB algorithm in Algorithm 3.

Given the observed data $\mathbf{D} := \{(t_{ij}, \delta_{ij}, \mathbf{X}_{ij}), i = 1, ..., K, j = 1, ..., n_i\}$, we denote the complete data log-likelihood by $\log p(\mathbf{D}, \boldsymbol{\beta}, \boldsymbol{\gamma}, b, \sigma_\gamma^2)$, where $\boldsymbol{\gamma} = (\gamma_1, ..., \gamma_K)$. Our objective is to derive a VB algorithm to approximate the exact posterior joint distribution of $\boldsymbol{\beta}$, b, $\boldsymbol{\gamma}$ and σ_γ^2 given the data $\mathbf{D}$ by maximizing the evidence lower bound (ELBO) defined as

$$\text{ELBO}(q) = \mathbb{E}_q[\log p(\mathbf{D}, \boldsymbol{\beta}, \boldsymbol{\gamma}, b, \sigma_\gamma^2)] - \mathbb{E}_q[\log q(\boldsymbol{\beta}, b, \boldsymbol{\gamma}, \sigma_\gamma^2)], \tag{4.2}$$

where $q(\boldsymbol{\beta}, b, \boldsymbol{\gamma}, \sigma_\gamma^2)$ is the approximated posterior joint distribution, which is also called the variational density, and the expectation is taken with respect to the variational density (Blei et al., 2017). We consider the mean-field variational family which assumes that $q(\boldsymbol{\beta}, b, \boldsymbol{\gamma}, \sigma_\gamma^2) = q(\boldsymbol{\beta})\, q(b)\, q(\sigma_\gamma^2) \prod_{i=1}^K q(\gamma_i)$, and apply the coordinate ascent variational inference (CAVI) algorithm (Bishop, 2006) to obtain each variational component (e.g., $q(\boldsymbol{\beta})$) in $q(\boldsymbol{\beta}, b, \boldsymbol{\gamma}, \sigma_\gamma^2)$). Under our Model (4.1), the complete data log-likelihood $\log p(\mathbf{D}, \boldsymbol{\beta}, \boldsymbol{\gamma}, b, \sigma_\gamma^2)$ can be obtained by

$$\begin{aligned} \log p(\mathbf{D}, \boldsymbol{\beta}, \boldsymbol{\gamma}, b, \sigma_\gamma^2) &= \log p(\mathbf{D} \,|\, \boldsymbol{\beta}, \boldsymbol{\gamma}, b) + \log p(\boldsymbol{\beta}) + \log p(b) \\ &\quad + \sum_{i=1}^K \log p(\boldsymbol{\gamma}_i \,|\sigma_\gamma^2) + \log p(\sigma_\gamma^2), \end{aligned} \tag{4.3}$$

where

$$\begin{aligned} \log p(\mathbf{D} \,|\, \boldsymbol{\beta}, \boldsymbol{\gamma}, b) &= -\delta \log b + \sum_{i=1}^K \sum_{j=1}^{n_i} \Bigg[\delta_{ij} \frac{y_{ij} - \mathbf{X}_{ij}^T \boldsymbol{\beta} - \gamma_i}{b} - (1 + \delta_{ij}) \\ &\quad \log \Big\{ 1 + \exp\big(\frac{y_{ij} - \mathbf{X}_{ij}^T \boldsymbol{\beta} - \gamma_i}{b}\big) \Big\} \Bigg], \end{aligned} \tag{4.4}$$

with $\delta = \sum_{i=1}^{K} \sum_{j=1}^{n_i} \delta_{ij}$ being the number of observed uncensored times and $y_{ij} = \log(t_{ij}), i = 1, ..., K, j = 1, ..., n_i$, being the log observed survival time.

By maximizing the ELBO, the following solutions are provided by the CAVI algorithm:

$$\log q^*(\boldsymbol{\beta}) \overset{+}{\approx} \mathbb{E}_{-\boldsymbol{\beta}}[\log p(\mathbf{D} \,|\, \boldsymbol{\beta}, \boldsymbol{\gamma}, b) + \log p(\boldsymbol{\beta})],$$

$$\log q^*(\gamma_i) \overset{+}{\approx} \mathbb{E}_{-\gamma_i}\Big[\log p(\mathbf{D} \,|\, \boldsymbol{\beta}, \boldsymbol{\gamma}, b) + \sum_{i=1}^{K} \log p(\gamma_i \,|\sigma_\gamma^2)\Big],$$

$$\log q^*(b) \overset{+}{\approx} \mathbb{E}_{-b}[\log p(\mathbf{D} \,|\, \boldsymbol{\beta}, \boldsymbol{\gamma}, b) + \log p(b)],$$

$$\log q^*(\sigma_\gamma^2) \overset{+}{\approx} \mathbb{E}_{-\sigma_\gamma^2}\Big[\sum_{i=1}^{K} \log p(\gamma_i \,|\sigma_\gamma^2) + \log p(\sigma_\gamma^2)\Big],$$

where we use $\overset{+}{\approx}$ to denote equality up to a constant additive factor for convenience, and $-\boldsymbol{\beta}$ indicates the expectation is taken with respect to the variational density of other latent variables but $\boldsymbol{\beta}$, same for other solutions. To achieve conjugacy and tractable expectation calculation of $\log p(\mathbf{D} \,|\, \boldsymbol{\beta}, \boldsymbol{\gamma}, b)$ as specified in (4.4), we apply the proposed method by Xian et al. (2024a), piecewise approximations of the function, $f(x) = \log(1 + \exp(x)), x \in (-\infty, \infty)$, embedded in deriving the update equations for each parameter. As in Nolan et al. (2020) and Lee and Wand (2016b), we are also interested in the posterior distribution of the random effects, $q^*(\gamma_i), i = 1, ..., K$, in our proposed VB framework.

4.3.1 Update equation for each variational density

The update equations to obtain the optimal variational densities of $\boldsymbol{\beta}$, γ_i, b and σ_γ^2 denoted by $q^*(\boldsymbol{\beta})$, $q^*(\gamma_i)$, $q^*(b)$ and $q^*(\sigma_\gamma^2)$, respectively, which are the corresponding approximated

posterior distributions, are presented as follows within the chapter while their derivations are given in the Appendix E.1. The calculation of expectations in the update equations are given in Section 4.3.2. In the update equations, φ_{ij}, ζ_{ij}, and ρ_{ij} represent the piece-wise approximation coefficients proposed by Xian et al. (2024a) for the j^{th} subject in the i^{th} cluster. Detailed information regarding these coefficients is also provided in the Appendix E.1.

(1) Update equation for $q^*(\boldsymbol{\beta})$

$q^*(\boldsymbol{\beta})$ is an $N_p(\boldsymbol{\mu}^*, \Sigma^*)$ where

$$\Sigma^* = \left[v_0 \mathbf{I}_{p\times p} + 2\mathbb{E}_{q(b)}(\frac{1}{b^2}) \sum_{i=1}^{K} \sum_{j=1}^{n_i} (1+\delta_{ij})\zeta_{ij}\mathbf{X}_{ij}\mathbf{X}_{ij}^T \right]^{-1},$$

and

$$\begin{aligned} \boldsymbol{\mu}^* \quad = \quad & \left[\left\{ v_0 \boldsymbol{\mu}_0^T + \sum_{i=1}^{K} \sum_{j=1}^{n_i} \Big(\big(\mathbb{E}_{q(b)}(\frac{1}{b}) \big(-\delta_{ij} + (1+\delta_{ij})\rho_{ij} \big) \right. \right. \\ & \left. \left. + 2\mathbb{E}_{q(b)}(\frac{1}{b^2})(1+\delta_{ij})\zeta_{ij}\big(y_{ij} - \mathbb{E}_{q(\gamma_i)}(\gamma_i)\big) \big) \mathbf{X}_{ij}^T \Big) \right\} \Sigma^* \right]^T. \end{aligned}$$

(2) Update equation for $q^*(\gamma_i)$

$q^*(\gamma_i)$ is an $N_l(\tau_i^*, \sigma_i^{2*})$ where

$$\tau_i^* = \sigma_i^{2*} \sum_{j=1}^{n_i} \Big[\mathbb{E}_{q(b)}(\frac{1}{b})(-\delta_{ij} + (1+\delta_{ij})\rho_{ij}) + 2\mathbb{E}_{q(b)}(\frac{1}{b^2})(1+\delta_{ij})\zeta_{ij}(y_{ij} - \mathbf{X}_{ij}^T \mathbb{E}_{q(\boldsymbol{\beta})}\boldsymbol{\beta}) \Big],$$

and

$$\sigma_i^{2*} = [\mathbb{E}_{q(\sigma_\gamma^2)}(\frac{1}{\sigma_\gamma^2}) + 2\mathbb{E}_{q(b)}(\frac{1}{b^2}) \sum_{j=1}^{n_i} (1+\delta_{ij})\zeta_{ij}]^{-1}.$$

(3) Update equation for $q^*(b)$

$q^*(b)$ is an Inverse-Gamma (α^*, ω^*) where $\alpha^* = \alpha_0 + \delta$ and

$$\omega^* = \omega_0 - \sum_{i=1}^{K} \sum_{j=1}^{n_i} \Big(\delta_{ij} - (1 + \delta_{ij})\varphi_{ij}\Big)\Big(y_{ij} - \mathbf{X}_{ij}^T \mathbb{E}_{q(\boldsymbol{\beta})}(\boldsymbol{\beta}) - \mathbb{E}_{q(\gamma_i)}\gamma_i\Big).$$

(4) Update equation for $q^*(\sigma_\gamma^2)$

$q^*(\sigma_\gamma^2)$ is an Inverse-Gamma (λ^*, η^*) where $\lambda^* = \lambda_0 + K/2$ and

$$\eta^* = \eta_0 + \frac{1}{2} \sum_{i=1}^{K} \mathbb{E}_{q(\gamma_i)} \gamma_i^2.$$

4.3.2 ELBO calculation

The ELBO under Model (4.1) is defined in (4.2) with the complete-data log-likelihood calculated by (4.3) and note that $q(\boldsymbol{\beta}, b, \boldsymbol{\gamma}, \sigma_\gamma^2) = q(\boldsymbol{\beta})\, q(b)\, q(\sigma_\gamma^2) \prod_{i=1}^{K} q(\gamma_i)$. Let

$$diff_{\boldsymbol{\beta}} = \mathbb{E}_q[\log p(\boldsymbol{\beta})] - \mathbb{E}_q[\log q(\boldsymbol{\beta})],$$

$$diff_{\boldsymbol{\gamma}} = \mathbb{E}_q\Big[\sum_{i=1}^{K} \log p(\gamma_i \,|\sigma_\gamma^2)\Big] - \mathbb{E}_q\Big[\sum_{i=1}^{K} \log q(\gamma_i)\Big],$$

$$diff_b = \mathbb{E}_q[\log p(b)] - \mathbb{E}_q[\log q(b)],$$

$$diff_{\sigma_\gamma^2} = \mathbb{E}_q[\log p(\sigma_\gamma^2)] - \mathbb{E}_q[\log q(\sigma_\gamma^2)].$$

We can calculate the ELBO as follows with proof details provided in the Appendix E.2.

$$\text{ELBO}(q) = \mathbb{E}_q[\log p(\mathbf{D} \,|\boldsymbol{\beta}, \boldsymbol{\gamma}, b)] + diff_{\boldsymbol{\beta}} + diff_{\boldsymbol{\gamma}} + diff_b + diff_{\sigma_\gamma^2}, \tag{4.5}$$

where

$$\mathbb{E}_q[\log p(\mathbf{D}\,|\,\boldsymbol{\beta},\boldsymbol{\gamma},b)] \overset{+}{\approx} -\delta\,\mathbb{E}_{q(b)}(\log b) + \mathbb{E}_{q(b)}(\frac{1}{b})\sum_{i=1}^{K}\sum_{j=1}^{n_i}\Big(\delta_{ij}-(1+\delta_{ij})\varphi_{ij}\Big)\Big(y_{ij}-\mathbf{X}_{ij}^T\mathbb{E}_{q(\boldsymbol{\beta})}(\boldsymbol{\beta})-\mathbb{E}_{q(\gamma_i)}\gamma_i\Big),$$

$$diff_{\boldsymbol{\beta}} \overset{+}{\approx} -\frac{1}{2}v_0[\text{trace}(\Sigma^*)+(\boldsymbol{\mu}^*-\boldsymbol{\mu}_0)^T(\boldsymbol{\mu}^*-\boldsymbol{\mu}_0)]+\frac{1}{2}\log(|\Sigma^*|),$$

$$diff_{\boldsymbol{\gamma}} \overset{+}{\approx} -\frac{K}{2}\mathbb{E}_{q(\sigma_\gamma^2)}(\log\sigma_\gamma^2)-\frac{1}{2}\mathbb{E}_{q(\sigma_\gamma^2)}(\frac{1}{\sigma_\gamma^2})\sum_{i=1}^{K}\mathbb{E}_{q(\gamma_i)}(\gamma_i^2)-\frac{1}{2}\sum_{i=1}^{K}(\log\sigma_i^{2*}),$$

$$diff_b \overset{+}{\approx} (\alpha^*-\alpha_0)\mathbb{E}_{q(b)}(\log b)+(\omega^*-\omega_0)\mathbb{E}_{q(b)}(\frac{1}{b})-\alpha^*\log\omega^*, \text{ and}$$

$$diff_{\sigma_\gamma^2} \overset{+}{\approx} (\lambda^*-\lambda_0)\mathbb{E}_{q(\sigma_\gamma^2)}(\log\sigma_\gamma^2)+(\eta^*-\eta_0)\mathbb{E}_{q(\sigma_\gamma^2)}(\frac{1}{\sigma_\gamma^2})-\lambda^*\log\eta^*.$$

We now present the calculation of the expectations in the update equations and the ELBO calculation. All the expectations are taken with respect to the approximated variational distributions. Since $q^*(b)$ is an Inverse-Gamma(α^*,ω^*), we have $\mathbb{E}_{q(b)}(1/b)=\alpha^*/\omega^*$,

$$\mathbb{E}_{q(b)}\Big(\frac{1}{b^2}\Big)=\mathbb{E}_{q(b)}\Big[\Big(\frac{1}{b}\Big)^2\Big]=\text{Var}_{q(b)}\Big[\Big(\frac{1}{b}\Big)\Big]+\Big[\mathbb{E}_{q(b)}\Big(\frac{1}{b}\Big)\Big]^2=\frac{\alpha^*}{\omega^{*2}}+\frac{\alpha^{*2}}{\omega^{*2}}=\frac{\alpha^*+\alpha^{*2}}{\omega^{*2}},$$

and $\mathbb{E}_{q(b)}(\log b)=\log(\omega^*)-\psi(\alpha^*)$, where ψ is the digamma function defined as $\psi(x)=\frac{d}{dx}\log\Gamma(x)$. Similarly, $\mathbb{E}_{q(\sigma_\gamma^2)}(1/\sigma_\gamma^2)=\lambda^*/\eta^*$ and $\mathbb{E}_{q(\sigma_\gamma^2)}(\log\sigma_\gamma^2)=\log(\eta^*)-\psi(\lambda^*)$.

Algorithm 3: Variational Bayesian inference of correlated survival data using a shared frailty log-logistic AFT model

Data: a sample of independent log observed time y_{ij}, their corresponding covariate vectors $\mathbf{X}_{ij}$ and the right censoring indicator $\delta_{ij}, i = 1, ..., K, j = 1, ..., n_i$ for the j^{th} observation from the i^{th} group; values of hyperparameters: $\boldsymbol{\mu}_0, v_0, \alpha_0, \omega_0, \lambda_0$ and η_0; convergence threshold Δ and maximum number of iterations M

Result: posterior distributions of $\boldsymbol{\beta}, \gamma_i, i = 1, ..., K, b$ and σ^2_γ, and their parameters: $\Sigma, \boldsymbol{\mu}, \sigma^2_i, \tau_i, \alpha, \omega, \lambda, \eta$

Initialization: initialize $\omega = \omega_0$, $\boldsymbol{\mu} = \boldsymbol{\mu}_0$, $\tau_i = 0$ and $\eta = \eta_0$, set $m = 0$ and ELBO = 0 ;

Calculation: obtain α by $\alpha = \alpha_0 + \delta$ with $\delta = \sum_{i=1}^{K} \sum_{j=1}^{n_i} \delta_{ij}$ and λ by $\lambda = \lambda_0 + K/2$;

while *iteration* $m < M$ *and difference of ELBO* $> \Delta$ **do**

$m = m + 1$;

$$\Sigma^{(m)} \leftarrow \left[v_0 \mathbf{I}_{p\times p} + 2\mathbb{E}_{q(b)}(\tfrac{1}{b^2}) \sum_{i=1}^{K} \sum_{j=1}^{n_i} (1+\delta_{ij}) \zeta_{ij} \mathbf{X}_{ij} \mathbf{X}_{ij}^T \right]^{-1} ;$$

$$\boldsymbol{\mu}^{(m)} \leftarrow \left[\left\{ v_0 \boldsymbol{\mu}_0^T + \sum_{i=1}^{K} \sum_{j=1}^{n_i} \left(\left(\mathbb{E}_{q(b)}(\tfrac{1}{b}) \Big(-\delta_{ij} + (1+\delta_{ij}) \rho_{ij} \Big) + 2\mathbb{E}_{q(b)}(\tfrac{1}{b^2})(1 + \delta_{ij}) \zeta_{ij} \big(y_{ij} - \mathbb{E}_{q(\gamma_i)}(\gamma_i) \big) \right) \Big) \mathbf{X}_{ij}^T \right\} \Sigma^{(m)} \right]^T ;$$

$$\sigma_i^{2(m)} \leftarrow \left[\mathbb{E}_{q(\sigma^2_\gamma)}(\tfrac{1}{\sigma^2_\gamma}) + 2\mathbb{E}_{q(b)}(\tfrac{1}{b^2}) \sum_{j=1}^{n_i} (1+\delta_{ij}) \zeta_{ij} \right]^{-1}, i = 1, ..., K;$$

$$\tau_i^{(m)} \leftarrow \sigma_i^{2(m)} \sum_{j=1}^{n_i} \Big[\mathbb{E}_{q(b)}(\tfrac{1}{b}) \big(-\delta_{ij} + (1+\delta_{ij}) \rho_{ij} \big) + 2\mathbb{E}_{q(b)}(\tfrac{1}{b^2})(1+\delta_{ij}) \zeta_{ij} \big(y_{ij} - \mathbf{X}_{ij}^T \mathbb{E}_{q(\boldsymbol{\beta})} \boldsymbol{\beta} \big) \Big], i = 1, ..., K ;$$

$$\omega^{(m)} \leftarrow \omega_0 - \sum_{i=1}^{K} \sum_{j=1}^{n_i} \Big(\delta_{ij} - (1+\delta_{ij}) \varphi_{ij} \Big) \Big(y_{ij} - \mathbf{X}_{ij}^T \mathbb{E}_{q(\boldsymbol{\beta})}(\boldsymbol{\beta}) - \mathbb{E}_{q(\gamma_i)} \gamma_i \Big);$$

$$\eta^{(m)} \leftarrow \eta_0 + \tfrac{1}{2} \sum_{i=1}^{K} \mathbb{E}_{q(\gamma_i)} \gamma_i^2;$$

calculate the current ELBO, $\text{ELBO}^{(m)}$;

calculate the difference of ELBO = $\text{ELBO}^{(m)} - \text{ELBO}^{(m-1)}$;

end

4.4 Simulation study

4.4.1 Design of simulation

We conduct simulation studies to evaluate the performance of our proposed VB algorithms across different scenarios by varying the number of clusters and the number of observations within each cluster.

We generate the log of survival time for the j^{th} subject in the i^{th} cluster, $\log(T_{ij})$, $i = 1, ..., K$ and $j = 1, ..., n_i$ as follows:

$$\log T_{ij} = 0.5 + \beta_1 x_{ij1} + \beta_2 x_{ij2} + \gamma_i + b\epsilon_{ij},$$

where x_{ij1}, x_{ij2}, and ϵ_{ij} are mutually independently generated with $x_{ij1} \sim N(1, 0.2^2)$, $x_{ij2} \sim$ Bernoulli(0.5) and $\epsilon_{ij} \sim$ logistic$(0, 1)$. The values of β_1, β_2 and b are chosen as 0.2, 0.8 and 0.8, respectively. The random intercept for the i^{th} cluster, γ_i, is generated from $N(0, \sigma_\gamma^2)$ with $\sigma_\gamma^2 = 1$. The censoring time for the j^{th} subject in the i^{th} cluster, C_{ij}, is generated from a uniform distribution, uniform$(0, d)$, where d is a positive value controlling the percentage of censoring. Then $t_{ij} = \min(T_{ij}, C_{ij})$ and $\delta_{ij} = \mathbb{1}(T_{ij} \leq C_{ij})$. Take $d = 48$ to achieve a 15% censoring rate in our simulations. Investigation of the effect of censoring rates can be found in Xian et al. (2024a) where they show that as the censoring rate increases, an increase in mean squared error (MSE) of estimating parameters in an AFT model using a VB algorithm was observed.

We explore various scenarios by varying the number of clusters, K, and the number of observations within each cluster, $n_i = n$ for all $i = 1, ..., K$. Across the experiments, we consider K values from the set $\{15, 30, 50, 80\}$ and n values from $\{5, 15, 30, 50\}$, resulting in a total of 16 unique scenarios. Our objective is to assess the estimation performance of the VB algorithm concerning the variations in K and n. We consider a prior setting with $\boldsymbol{\mu}_0 = (0, 0, 0)^T$, $\nu_0 = 0.1$, $\alpha_0 = \lambda_0 = 3$, and $\omega_0 = \eta_0 = 2$, which indicates no strong prior information on the parameters. The ELBO convergence threshold is set as 0.01 which is the default recommendation (Yao et al., 2018), and the maximum number of iterations is 100.

We conduct $N = 500$ runs (replicates) in each considered scenario and apply our proposed VB algorithm to derive the approximated posterior distribution of each parameter to each run. The mean of each approximated posterior distribution is used as the parameter estimate

for the corresponding parameter. The empirical bias and sample standard deviation (SD) as well as the empirical MSE for each estimate are obtained, where

$$\text{MSE} = \frac{\sum_{i=1}^{N}(\theta_0 - \hat{\theta}_i)^2}{N},$$

and $\hat{\theta}_i$ is the estimate of parameter θ in the i^{th} replicate, and θ_0 is the true value. In addition, for each parameter of interest, we report the empirical 95% coverage rate (CR) calculated as

$$\text{CR} = \frac{\sum_{i=1}^{N} I_i}{N},$$

where I is the indicator variable which takes 1 if the true parameter value θ_0 falls into the 95% credible interval.

4.4.2 Simulation results

The empirical bias, sample SD, empirical MSE and empirical CR of estimating β_1, β_2, b, and σ_γ^2 from 500 replicates are summarized in Table 4.1. Considering β_1, β_2, and b, we observe that when the number of clusters K is fixed, increasing the cluster size n from 5 to 50 does not always significantly affect the empirical bias. However, there is a noticeable decrease in the sample SD, leading to a pronounced reduction in the empirical MSE. A similar trend is observed when the cluster size is fixed while increasing K. Figure 4.1 visually illustrates this asymptotic property through boxplots. Our primary focus lies in understanding the impact of K and n on estimating the variance of the random intercept, denoted as σ_γ^2. In most of the scenarios, when the number of clusters K is fixed, we observe that the bias of estimating σ_γ^2 decreases as the cluster size increases, while the sample SD remains relatively stable. However, when K is increased, both the empirical bias and the sample SD decrease noticeably, except for the case when $n = 30$ as K grows from 15 to 30. Particularly noteworthy is the scenario with $K = 80$ and $n = 50$, where we achieve a less biased and less variable estimation of the variance of the random intercept.

In addition, the empirical CRs corresponding to a 95% credible interval for each parameter across all scenarios are close to the nominal level of 95%, with a mean of 94.1% and a standard deviation of 0.014. Furthermore, based on a nominal level of 95% with 500 replicates, the coverage with two standard deviations is $95\% \pm 2\text{SD} = [93.1\%, 97.0\%]$ since $\text{SD} = \sqrt{0.95 \times 0.05/500} \approx 0.0097$. We observe that the majority of our empirical CRs are within the range of $[93.1\%, 97.0\%]$.

The left part of Figure 4.2 presents the computational run time of 500 replicates in minutes against the number of clusters (K) with different colors for different cluster sizes n. Fixing K, as n increases, the used time increases. The scatter plot in the right of Figure 4.2 shows the run time against the sample size ($K \times n$). As anticipated, the computational cost rises with increasing sample size.

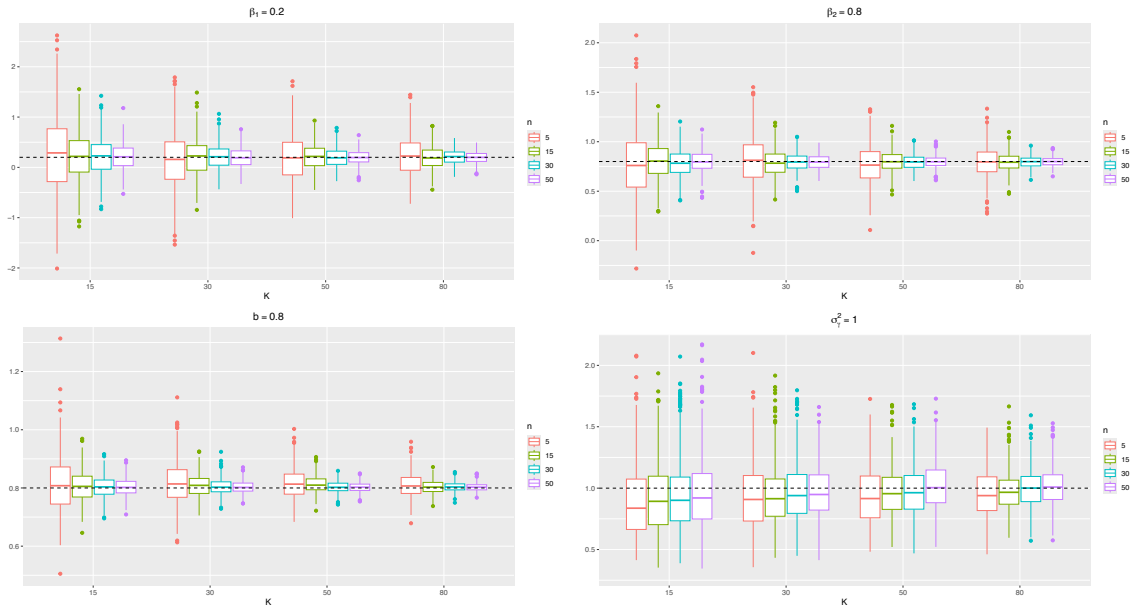


Figure 4.1: Boxplots of parameter estimates using posterior means from 500 replicates under various scenarios with different number of clusters K and cluster sizes n based on our proposed VB algorithm. The horizontal dashed line on each plot represents the true value of the corresponding parameter used when generating the data.

Furthermore, among those considered 16 scenarios, we select three scenarios ($K = 30$ with $n = 5$, $K = 50$ with $n = 15$, and $K = 80$ with $n = 30$) and conduct a comparative analysis of estimation results obtained from the proposed VB algorithm with those from two alter-

native methods which we introduced in Section 4.1: the h-likelihood method proposed by Do Ha et al. (2017) and the MCMC-based *survregbayes* developed by Zhou et al. (2017). Summaries from the MCMC-based *survregbayes* algorithm with adaptive Metropolis samplers are derived from one Markov chain, subsampled every 5 iterates to achieve a final chain size of 2,000 after a burn-in period of 5,000 iterates (Zhou et al., 2017). We further refer to our VB method and the h-likelihood method as *survregVBfrailty* and *survregHL*, respectively. Numerical estimation results are presented in Table 4.2, where we focus on the comparison of estimation for β_1, β_2 and σ_γ^2 since the MCMC-based *survregbayes* does not directly return the estimate for b. In the scenario with $K = 30$ and $n = 5$, the proposed *survregVBfrailty* exhibits a larger empirical bias compared to both the *survregHL* and *survregbayes* methods. However, as K and n increase, the discrepancy in empirical bias among the three methods becomes negligible. In terms of the sample SD, on average, across the three parameters β_1, β_2, and σ_γ^2, *survregVBfrailty* yields a 16.5% smaller sample SD when $K = 30$ and $n = 5$, a 7.6% smaller sample SD when $K = 50$ and $n = 15$, and a 3.7% smaller sample SD when $K = 80$ and $n = 30$, compared to *survregHL*. Compared with the *survregBayes* algorithm, we observe a decrease in sample SD of 7.8% in the $K = 30$, $n = 5$ scenario, mainly due to β_1, and only 2.0% in the $K = 50$, $n = 15$ scenario. However, in the scenario with $K = 80$, $n = 30$, no average difference in sample SD is observed between VB and MCMC. Therefore, the *survregVBfrailty* algorithm generally yields a smaller SD, resulting in a lower MSE compared to the other two methods in most cases. On average, across the three parameters β_1, β_2, and σ_γ^2, *survregVBfrailty* achieves a 27.2% lower MSE when $K = 30$, $n = 5$, a 12.0% lower MSE when $K = 50$, $n = 15$, and a 13.4% lower MSE when $K = 80$, $n = 30$, compared to *survregHL*. Compared with *survregBayes*, we observe a reduction in MSE of 12.4% in the $K = 30$, $n = 5$ scenario, mainly because of β_1, and only 3.0% in the $K = 50$, $n = 15$ scenario. However, in the scenario with $K = 80$, $n = 30$, no difference in MSE is observed between VB and MCMC. In addition, we compare the posterior densities for β_1, β_2, and σ_γ^2 obtained from both *survregVBfrailty*

and the MCMC-based *survregBayes* using a simulated dataset with $K = 50$ and $n = 15$. These densities present a strong consistency between the VB and MCMC algorithms as illustrated in Figure 4.3, which is also reflected in the results presented in Table 4.2.

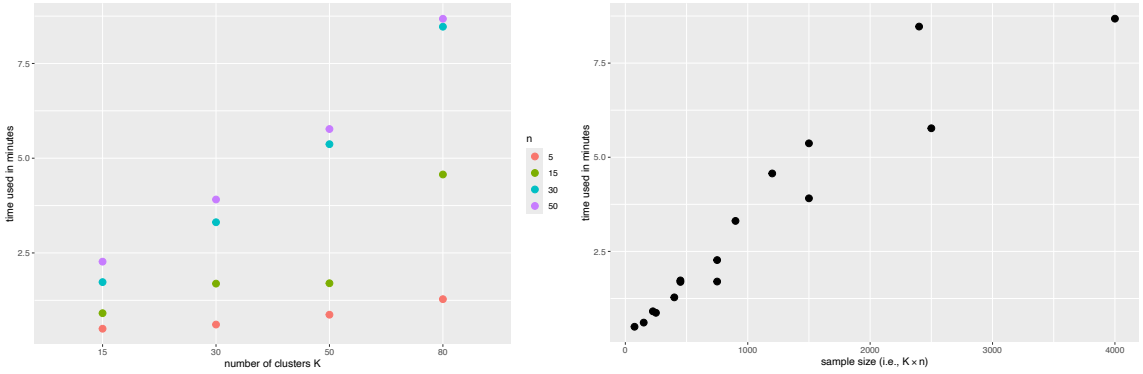


Figure 4.2: Left: the run time in minutes used for 500 replicates under various scenarios with different number of clusters K and cluster sizes n based on our proposed VB algorithm. Right: the run time in minutes used for 500 replicates in different sample sizes based on VB.

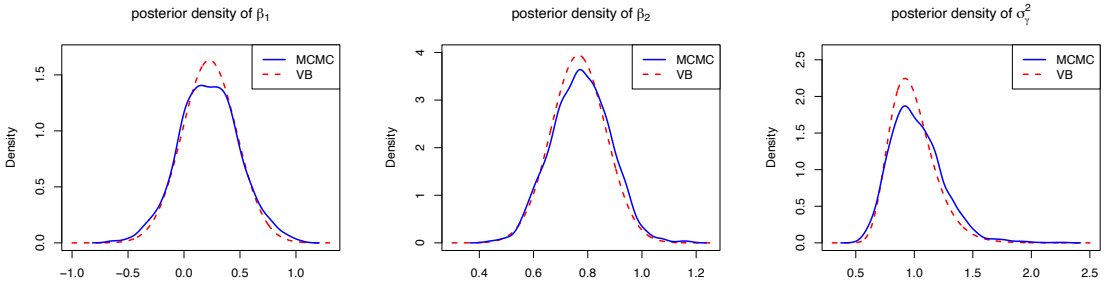


Figure 4.3: A comparison of posterior densities of β_1, β_2 and σ^2_γ obtained from our VB method *survregVBfrailty* and MCMC-based *survregbayes* from a simulated data set in the scenario with $K = 50$ and $n = 15$.

To assess the computational efficiency of the proposed VB algorithm, we compare the run times in minutes for 500 replicates in these three scenario across *survregVBfrailty*, *survregHL*, and *survregbayes* methods and present the results in Table 4.3. In Table 4.3, we observe that in the scenario with a small sample size ($K = 30$ and $n = 5$), there is minimal difference in the run times between the VB and h-likelihood methods. However, as both K and n increase, the h-likelihood method requires progressively longer computation times to

obtain estimates. Particularly in the scenario with $K = 80$ and $n = 30$, the computational time is over 10 times longer than that required by the VB algorithm. Comparatively, the VB algorithm demonstrates significantly higher computational efficiency when contrasted with the MCMC-based *survregbayes* algorithm. Notably, in the scenario with $K = 80$ and $n = 30$, the *survregVBfrailty* algorithm runs approximately 150 times faster than the *survregbayes* algorithm. All algorithms were implemented within R version 4.3.2 and simulations were conducted on a computer operating the Mac OS X platform, equipped with a 4.05 GHz CPU and 8 GB RAM.

Table 4.1: Numerical estimation results (point estimate using the mean of the corresponding posterior distribution) including the empirical Bias, sample SD, empirical MSE and coverage rate (CR) for parameters in each scenario with different number of clusters K and cluster sizes n from the proposed VB algorithm.

		β_1				β_2				b				σ^2_γ			
K	n	Bias	SD	MSE	CR	Bias	SD	MSE	CR	Bias	SD	MSE	CR	Bias	SD	MSE	CR
15	5	0.066	0.764	0.587	94.2	-0.026	0.352	0.124	90.0	0.005	0.095	0.009	94.5	-0.101	0.312	0.107	91.4
	15	0.018	0.489	0.239	93.2	0.007	0.192	0.037	94.0	0.006	0.052	0.003	94.8	-0.081	0.287	0.089	93.1
	30	0.006	0.354	0.125	93.2	-0.017	0.139	0.019	92.4	-0.006	0.036	0.001	96.0	0.003	0.290	0.087	95.5
	50	0.009	0.254	0.065	95.6	-0.001	0.104	0.011	94.8	0.003	0.029	0.001	94.7	-0.040	0.299	0.100	94.2
30	5	0.004	0.589	0.346	92.8	-0.015	0.257	0.066	92.8	0.015	0.071	0.005	94.4	-0.092	0.274	0.084	92.6
	15	0.002	0.360	0.129	93.0	-0.018	0.140	0.020	93.0	0.008	0.038	0.001	96.1	-0.057	0.242	0.062	91.1
	30	0.012	0.239	0.057	94.6	-0.004	0.088	0.008	95.7	0.005	0.026	0.001	95.6	-0.036	0.248	0.063	92.4
	50	-0.007	0.192	0.037	93.0	-0.005	0.074	0.005	95.0	0.003	0.020	< 0.001	95.8	-0.026	0.228	0.052	95.2
50	5	-0.011	0.470	0.220	94.3	-0.033	0.199	0.041	91.4	0.014	0.051	0.003	95.1	-0.062	0.235	0.059	93.2
	15	0.003	0.253	0.064	95.2	-0.001	0.104	0.011	94.8	0.018	0.029	0.001	94.2	-0.033	0.203	0.042	92.4
	30	-0.006	0.192	0.037	92.3	-0.006	0.074	0.006	94.4	0.003	0.020	< 0.001	95.6	-0.027	0.191	0.037	93.3
	50	-0.002	0.143	0.020	94.4	-0.004	0.057	0.003	95.6	0.002	0.016	< 0.001	95.7	0.016	0.201	0.040	93.4
80	5	0.013	0.384	0.147	93.2	-0.005	0.161	0.026	91.3	0.014	0.041	0.002	95.2	-0.048	0.197	0.041	94.4
	15	-0.008	0.223	0.050	93.4	-0.005	0.090	0.008	92.4	0.008	0.023	0.001	95.5	-0.023	0.160	0.026	92.2
	30	0.008	0.141	0.020	95.8	-0.005	0.058	0.003	95.1	-0.003	0.015	< 0.001	95.6	0.003	0.161	0.026	92.8
	50	-0.005	0.108	0.012	95.4	-0.002	0.043	0.002	95.5	0.003	0.013	< 0.001	95.5	0.014	0.157	0.025	94.4

Table 4.2: A comparison of numerical estimation results including the empirical Bias, sample SD and MSE, from our VB method *survregVBfrailty*, the h-likelihood method *survregHL* and MCMC-based *survregbayes* method in each scenario (K: number of clusters, n: number of observations in each cluster).

		survregVBfrailty			*survregHL*			*survregbayes*		
Scenario		Bias	SD	MSE	Bias	SD	MSE	Bias	SD	MSE
$K = 30, n = 5$	β_1	0.004	0.589	0.346	0.001	0.664	0.440	0.001	0.664	0.440
	β_2	-0.015	0.257	0.066	-0.002	0.270	0.073	-0.002	0.266	0.071
	σ^2_γ	-0.092	0.274	0.084	0.001	0.412	0.170	-0.042	0.300	0.092
$K = 50, n = 15$	β_1	0.003	0.253	0.064	0.001	0.268	0.072	-0.005	0.259	0.067
	β_2	-0.001	0.104	0.011	0.007	0.107	0.011	-0.004	0.104	0.011
	σ^2_γ	-0.033	0.203	0.042	0.011	0.237	0.056	-0.001	0.211	0.044
$K = 80, n = 30$	β_1	0.008	0.141	0.020	0.009	0.143	0.021	-0.009	0.142	0.020
	β_2	-0.005	0.058	0.003	0.001	0.060	0.004	0.001	0.058	0.003
	σ^2_γ	0.003	0.161	0.026	0.007	0.172	0.029	0.001	0.160	0.026

Table 4.3: Times in minutes for 500 replicates from our VB algorithm *survregVBfrailty*, the h-likelihood method *survregHL* and the MCMC-based *survregbayes* algorithm, respectively, under each scenario.

Scenario	*survregVBfrailty*	*survregHL*	*survregbayes*
$K = 30, n = 5$	0.61	0.67	47.65
$K = 50, n = 15$	1.70	9.59	232.31
$K = 80, n = 30$	8.47	124.53	1302.80

4.5 Application to ventilation duration analysis

In this section, we apply the shared frailty log-logistic AFT model with the proposed VB algorithm to the ICU data as we descried in the Introduction section and conduct a retrospective study on the ventilation duration time. We aim to investigate the ICU site-specific random effect on patient's ventilation duration. We extend the work by Kobara et al. (2023) by incorporating the uncertainty from the ICU sites via a group-specific random intercept under a Bayesian analysis framework.

The CCIS ICU data were collected between July 2015 and December 2016 and contained 49,467 patients receiving invasive mechanical ventilation upon arrival to ICU. About 3%

of these patients were discharged/transferred to a Complex Continuing Care Facility, other hospitals, the Level 3 Unit, and Outside the ICU while still on a ventilator, and therefore, their ventilation time were considered as right-censored data (Kobara et al., 2023). The data were from 66 ICU sites (centers) and each site has a unique site code, for example, 3970. In the CCIS dataset, the average number of ventilated patients per ICU site over the study period of about 1.5 years is 749.5. We consider the significant covariates investigated by Kobara et al. (2023) as fixed effects which are admission source (e.g., from operation rooms), admission diagnosis, patient type (medical or surgical), scheduled admission (yes or no), scheduled surgery (yes or no), referring physician specialty, other interventions (yes or no), central venous line (CVL, yes or no), arterial line (AL, yes or no), intra-cranial pressure monitor (IPM, yes or no), extracorporeal membrane oxygen (EMO, yes or no), intra-aortic balloon pump (IABP, yes or no), age group (18-39, 40-80 or above 80 years of age), pre-LOS (no more than 1 day, between 2 and 7 days, or no less than 7 days), and the MODS score (none with MODS ≤ 1, minimal with $1-4$ scores, mild with $4-8$ scores, moderate with $8-12$ scores or, severe with scores > 12).

To apply our proposed VB algorithm in real data analysis, we consider the same weak prior setting in our simulation study described in Section 4.4.1: $\boldsymbol{\mu}_0 = \mathbf{0}$, $v_0 = 0.1$, $\alpha_0 = \lambda_0 = 3$, and $\omega_0 = \eta_0 = 2$. Table 4.4 displays the estimated regression coefficients from the fitted shared frailty log-logistic AFT model. For comparative analysis, we also employed the h-likelihood method *survregHL* and the MCMC-based *survregbayes* to model the data incorporating shared frailty. In the absence of shared frailty, we used the likelihood-based *survreg* from the `survival` package in R, and the VB method *survregVB* proposed by Xian et al. (2024a). Furthermore, for the Bayesian methods (*survregVBfrailty*, *survregVB*, and *survregbayes*), we report the 95% credible intervals, whereas for the likelihood-based methods, we provide the 95% confidence intervals. In Table 4.4, we first observe that there is no significant difference in the estimated regression coefficients between *survregVBfrailty* and *survregbayes*. However, some of the estimated coefficients from *survregHL*, such as sched-

uled surgery, CVL, AL, IPM and EMO, are different from those based on *survregVBfrailty* and *survregbayes*, which is further discussed in Section 4.6. In comparison to the models with frailty, the *survregVB* and *survreg* methods, which do not account for frailty, exhibit some differences in estimating certain coefficients. However, they maintain a strong overall consistency in estimating the regression coefficients with the results from the frailty models. This consistency is expected, as the inclusion of frailty does not affect the fixed effects, implying that the regression lines from models with or without frailty should be parallel. To illustrate the difference in interval estimation between models with and without frailty, we calculated the mean overlap percentage of the two 95% credible intervals from VB with or without frailty. Using the 95% credible intervals from VB without frailty (*survregVB*) as references, we found a 77.3% overlap on average. This means that if we fit a model using VB without considering the ICU site as a shared frailty, about 77% of the credible interval will fall within the corresponding credible interval obtained from the model that includes the shared frailty.

In what follows, we summarize the fixed effects on ventilation duration based on estimates from our proposed VB algorithm, *survregVBfrailty*. Regarding the admission source, patients arriving from the emergency department (ED), operating room (OR), or ward have ventilation times that are $(1 - \exp(-0.134))100\% = 12.54\%$ shorter (credible interval (CI): [9.24%, 12.72%]), 27.89% shorter (CI: [25.62%, 30.02%]), and 13.06% shorter (CI: [9.70%, 16.31%]), respectively, compared to patients admitted from a downstream unit (baseline). Conversely, patients admitted from home or another hospital have ventilation durations that are 10.63% (CI: [-10.40%, 23.49%]) and 10.30% (CI: [7.04%, 13.54%]) longer, respectively. Since the CI for patients admitted from home includes zero, we conclude that admission from home is not a statistically significant factor. Additionally, patients from other sources, such as from outside the province, have ventilation durations that are 12.98% longer (CI: [5.13%, 21.29%]).

Compared to cardiovascular patients, those with gastrointestinal, neurological, and trauma diagnoses experience significantly longer ventilation times. Specifically, ventilation duration increases by 29.69% (CI: [25.36%, 34.18%]) for gastrointestinal patients, 33.11% (CI: [29.05%, 37.44%]) for neurological patients, and 77.71% (CI: [69.72%, 86.08%]) for trauma patients. Patient categories (surgical or medical) do not show significant differences in ventilation time. However, scheduled ICU admissions or surgeries are important factors. Patients with a scheduled ICU admission have a 24.95% longer ventilation time (CI: [21.89%, 27.82%]), while those with a scheduled surgery have a 13.32% shorter ventilation time (CI: [9.70%, 16.81%]) compared to patients without scheduled admissions or surgeries. The referral physician service is also a significant risk factor. Compared with medical referrals, surgical referrals result in a 5.82% shorter ventilation time (CI: [8.97%, 2.57%]), while respirology referrals result in a 16.77% longer ventilation time (CI: [11.29%, 22.51%]).

Specific treatment interventions upon ICU arrival significantly impact ventilation duration. Patients receiving a CVL, AL, IPM, EMO, or IABP have increased ventilation times by 20.20% (CI: [17.82%, 22.51%]), 21.90% (CI: [19.36%, 24.48%]), 70.40% (CI: [60.16%, 81.48%]), 148.93% (CI: [127.50%, 172.10%]), and 40.92% (CI: [32.45%, 49.78%]), respectively. Age is also a significant factor. Compared to patients aged 18 to 39, those aged 40 to 80 and those over 80 have longer ventilation periods, increasing by 14.22% (CI: [11.29%, 17.23%]) and 7.79% (CI: [4.50%, 11.18%]), respectively.

Both pre-ICU LOS and patients' severity scores, as measured by MODS, are significant risk factors for longer ventilation duration. Longer pre-ICU LOS correlates with longer ventilation times. Patients with a pre-ICU LOS of 2-7 days or $\geq$ 7 days experience increases in ventilation duration by 3.87% (CI: [1.71%, 6.08%]) and 12.98% (CI: [10.52%, 15.60%]), respectively, compared to those with $\leq$ 1 day pre-ICU LOS. Compared to patients with a MODS score $\leq$ 1, those with minimal (1-4), mild (4-8), moderate (8-12), or severe (>

12) MODS scores experience increases in ventilation duration by 10.63% (CI: [7.90%, 13.54%]), 23.49% (CI: [20.56%, 26.49%]), 30.34% (CI: [26.49%, 34.31%]), and 14.22% (CI: [7.47%, 21.53%]), respectively.

We now turn to the analysis of variance in terms of the variations from individual patients and from ICU sites. Using the mean of the posterior distribution for σ_γ^2, the estimated variance (σ_γ^2) for the shared ICU site effect from *survregVBfrailty* is 0.1, with a square root of 0.316, which indicates that the average spread of the ventilation time among ICU sites is exp(0.316), or 1.372. The estimated scale parameter b is 0.444, resulting in an estimated variance of the logarithms of individual patient ventilation times of $0.444^2 \times \pi^2/3$, or 0.649, where $\pi^2/3$ is the variance of a standard logistic distribution. To measure the strength of the correlation between patients within the same ICU site, the intra-class correlation coefficient between the logarithms of ventilation time can be estimated using the plug-in method as follows: $0.1/(0.1 + 0.649) \approx 0.134$.

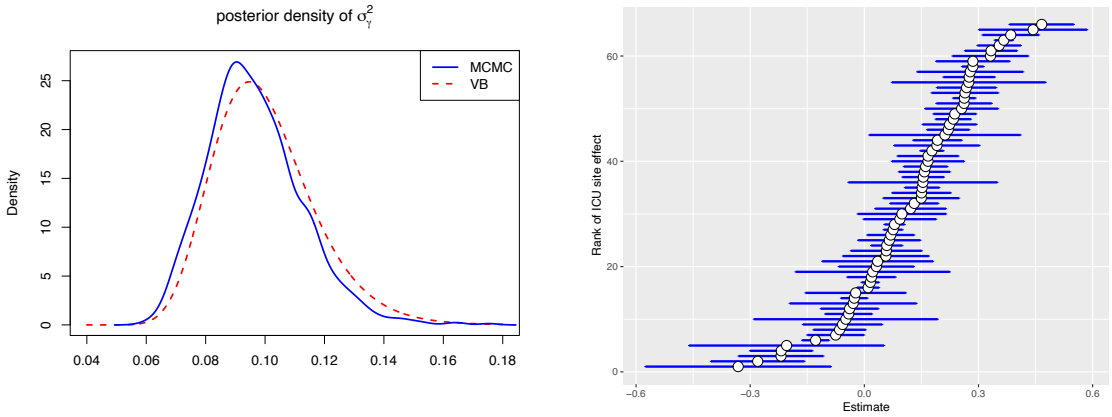


Figure 4.4: Real data analysis. Left: Posterior distribution of the variance of the random intercept from VB and MCMC. Right: Estimated ICU site specific random effects with their 95% credible interval from the proposed VB algorithm. The random effects have been ranked in an increasing order.

In the left of Figure 4.4, we observe a strong consistency in estimating the variance of the ICU-site specific random effect between *survregVBfrailty* and the MCMC-based *survregbayes*. We further visualize the estimated ICU-site random effects with the 95% credible

intervals based on their posterior distributions obtained from *survregVBfrailty*, as shown in the right of Figure 4.4, where the random effects are ranked from smallest to largest. Wider credible intervals correspond to ICUs with larger numbers of patients. As discussed in Lambert et al. (2004), if the intervals overlap, there are no significant center random effects. We observe that some of the intervals do not overlap, indicating that these ICU sites perform differently in terms of patient ventilation duration.

For reference on computational efficiency, the run times in minutes for each method are as follows: *survregVBfrailty* took 1.45 minutes, h-likelihood took 106.18 minutes, MCMC-based *survregbayes* took 267.04 minutes, *survreg* took 0.02 minutes, and *survregVB* took 0.13 minutes.

Table 4.4: Results for ICU ventilation duration analysis. Posterior means (Mean) with 95% credible intervals (95% Cred. Int.) from the VB algorithms and MCMC-based *survregbayes*, respectively. Point estimates (Est.) with 95% confidence interval (95% Conf. Int.) from the h-likelihood *survregHL* and *survreg* methods.

	With frailty						Without frailty			
	survregVBfrailty		*survregHL*		*survregbayes*		*survreg*		*survregVB*	
	Mean	95% Cred. Int.	Est.	95% Conf. Int.	Mean	95% Cred. Int.	Est.	95% Conf. Int.	Mean	95% Cred. Int.
Admission Source										
Downstream (ref)	-	-	-	-	-	-	-	-	-	-
ED	-0.134	[-0.171, -0.097]	-0.134	[-0.181, -0.086]	-0.134	[-0.172, -0.103]	-0.135	[-0.173, -0.098]	-0.134	[-0.171, -0.097]
Home	0.101	[-0.009, 0.211]	0.088	[-0.016, 0.191]	0.096	[-0.014, 0.211]	0.095	[-0.011, 0.201]	0.092	[-0.013, 0.197]
Hospital	0.098	[0.068, 0.127]	0.084	[0.033, 0.135]	0.097	[0.064, 0.128]	0.077	[0.037, 0.117]	0.079	[0.039, 0.119]
OR	-0.327	[-0.357, -0.296]	-0.319	[-0.372, -0.266]	-0.312	[-0.342, -0.281]	-0.333	[-0.374, -0.292]	-0.334	[-0.375, -0.293]
Other	0.122	[0.050, 0.193]	0.062	[-0.062, 0.186]	0.115	[0.039, 0.191]	0.146	[0.043, 0.248]	0.151	[0.049, 0.252]
Ward	-0.140	[-0.178, -0.102]	-0.142	[-0.191, -0.094]	-0.140	[-0.178, -0.101]	-0.135	[-0.174, -0.097]	-0.137	[-0.175, -0.099]
Diagnosis										
Cardiovascular (ref)	-	-	-	-	-	-				
Gastrointestinal	0.260	[0.226, 0.294]	0.265	[0.219, 0.310]	0.260	[0.226, 0.294]	0.265	[0.231, 0.300]	0.265	[0.230, 0.299]
Neurological	0.286	[0.255, 0.318]	0.286	[0.244, 0.329]	0.280	[0.250, 0.317]	0.304	[0.272, 0.336]	0.303	[0.270, 0.335]
Other	0.379	[0.356, 0.402]	0.366	[0.335, 0.397]	0.370	[0.349, 0.394]	0.386	[0.363, 0.410]	0.386	[0.363, 0.410]
Trauma	0.575	[0.529, 0.621]	0.527	[0.467, 0.586]	0.574	[0.530, 0.622]	0.585	[0.539, 0.631]	0.589	[0.543, 0.635]
Patient type										
Medical (ref)	-	-	-	-	-	-				
Surgical	-0.021	[-0.060, 0.017]	-0.019	[-0.031, 0.069]	-0.023	[-0.058, 0.013]	-0.022	[-0.061, 0.017]	-0.024	[-0.064, 0.015]
Scheduled Admission										
No (ref)	-	-	-	-	-	-				
Yes	-0.287	[-0.326, -0.247]	-0.281	[-0.336, -0.226]	-0.285	[-0.329, -0.246]	-0.289	[-0.329, -0.249]	-0.277	[-0.317, -0.237]
Scheduled Surgery										
No (ref)	-	-	-	-	-	-				
Yes	-0.143	[-0.184, -0.102]	-0.162	[-0.218, -0.106]	-0.146	[-0.188, -0.104]	-0.141	[-0.183, -0.099]	-0.144	[-0.186, -0.102]

Table 4.4 Continued: Results for ICU ventilation duration analysis. Posterior means (Mean) with 95% credible intervals (95% Cred. Int.) from the VB algorithms and MCMC-based *survregbayes*, respectively. Point estimates (Est.) with 95% confidence interval (95% Conf. Int.) from the h-likelihood *survregHL* and *survreg* methods.

	With frailty						Without frailty			
	survregVBfrailty		*survregHL*		*survregbayes*		*survreg*		*survregVB*	
	Mean	95% Cred. Int.	Est.	95% Conf. Int.	Mean	95% Cred. Int.	Est.	95% Conf. Int.	Mean	95% Cred. Int.
Referral										
Medical (ref)	-	-	-	-	-	-				
Other	-0.028	[-0.058, 0.001]	-0.012	[-0.052, 0.027]	-0.034	[-0.065, -0.004]	-0.051	[-0.080, -0.021]	-0.052	[-0.081, -0.022]
Respirology	0.155	[0.107, 0.203]	0.112	[0.047, 0.177]	0.156	[0.108, 0.205]	0.165	[0.117, 0.214]	0.168	[0.120, 0.216]
Surgical	-0.060	[-0.094, -0.026]	-0.052	[-0.098, -0.005]	-0.061	[-0.099, -0.024]	-0.048	[-0.083, -0.013]	-0.049	[-0.085, -0.014]
Other Interventions										
No (ref)	-	-	-	-	-	-				
Yes	0.088	[0.073, 0.104]	0.091	[0.069, 0.112]	0.089	[0.075, 0.105]	0.063	[0.047, 0.079]	0.062	[0.046, 0.078]
CVL										
No (ref)	-	-	-	-	-	-				
Yes	0.184	[0.164, 0.203]	0.163	[0.137, 0.188]	0.178	[0.157, 0.198]	0.198	[0.178, 0.218]	0.201	[0.181, 0.220]
AL										
No (ref)	-	-	-	-	-	-				
Yes	0.198	[0.177, 0.219]	0.177	[0.149, 0.205]	0.193	[0.172, 0.213]	0.199	[0.178, 0.220]	0.202	[0.181, 0.223]
IPM										
No (ref)	-	-	-	-	-	-				
Yes	0.533	[0.471, 0.596]	0.476	[0.401, 0.552]	0.535	[0.473, 0.599]	0.544	[0.481, 0.606]	0.542	[0.479, 0.604]
EMO										
No (ref)	-	-	-	-	-	-				
Yes	0.912	[0.822, 1.001]	0.798	[0.691, 0.906]	0.908	[0.817, 0.996]	0.993	[0.902, 1.083]	0.994	[0.904, 1.085]
IABP										
No (ref)	-	-	-	-	-	-				
Yes	0.343	[0.281, 0.404]	0.297	[0.216, 0.377]	0.340	[0.274, 0.398]	0.324	[0.261, 0.387]	0.317	[0.254, 0.379]

Table 4.4 Continued: Results for ICU ventilation duration analysis. Posterior means (Mean) with 95% credible intervals (95% Cred. Int.) from the VB algorithms and MCMC-based *survregbayes*, respectively. Point estimates (Est.) with 95% confidence interval (95% Conf. Int.) from the h-likelihood *survregHL* and *survreg* methods.

	With frailty						Without frailty			
	survregVBfrailty		*survregHL*		*survregbayes*		*survreg*		*survregVB*	
	Mean	95% Cred. Int.	Est.	95% Conf. Int.	Mean	95% Cred. Int.	Est.	95% Conf. Int.	Mean	95% Cred. Int.
Age										
18-39 (ref)	-	-	-	-	-	-				
40-80	0.133	[0.107, 0.159]	0.128	[0.093, 0.162]	0.130	[0.102, 0.158]	0.115	[0.089, 0.142]	0.116	[0.089, 0.143]
> 80	0.075	[0.044, 0.106]	0.063	[0.022, 0.105]	0.077	[0.040, 0.110]	0.057	[0.026, 0.089]	0.058	[0.027, 0.090]
Pre-LOS										
≤ 1 day (ref)	-	-	-	-	-	-				
2-7 days	0.038	[0.017, 0.059]	0.040	[0.010, 0.069]	0.038	[0.016, 0.060]	0.040	[0.019, 0.062]	0.040	[0.019, 0.062]
≥ 7 days	0.122	[0.100, 0.145]	0.140	[0.109, 0.171]	0.121	[0.100, 0.143]	0.130	[0.107, 0.153]	0.129	[0.106, 0.152]
MODS										
None (ref)	-	-	-	-	-	-				
Minimal	0.101	[0.076, 0.127]	0.112	[0.076, 0.148]	0.103	[0.078, 0.130]	0.083	[0.057, 0.109]	0.082	[0.056, 0.108]
Mild	0.211	[0.187, 0.235]	0.228	[0.191, 0.264]	0.210	[0.186, 0.233]	0.182	[0.158, 0.207]	0.180	[0.156, 0.205]
Moderate	0.265	[0.235, 0.295]	0.290	[0.247, 0.332]	0.262	[0.231, 0.293]	0.248	[0.217, 0.278]	0.245	[0.215, 0.276]
Severe	0.133	[0.072, 0.195]	0.198	[0.122, 0.274]	0.130	[0.070, 0.189]	0.137	[0.075, 0.198]	0.132	[0.071, 0.193]

4.6 Discussion

In this chapter, we have proposed a fast variational Bayesian algorithm, called *survregVBfrailty*, for statistical inference using a shared frailty log-logistic AFT model, which can be applied to analyze clustered survival data. We demonstrated that our proposed *survregVBfrailty* algorithm achieves satisfactory estimation performance through simulation studies under various scenarios with different numbers of clusters and cluster sizes. A strong consistency in the estimated posterior distributions was observed between VB and MCMC methods in both the simulation study and the application to ICU ventilation data. Moreover, the *survregVBfrailty* algorithm is significantly more computationally efficient, running over 150 times faster than the MCMC-based *survregbayes* algorithm.

The h-likelihood method proposed by Do Ha et al. (2002) for analyzing clustered survival data in a log-normal AFT model has been shown to be robust against violations of the normality assumption (e.g., extreme value distribution) for the logarithm of survival time. In our simulation study, we investigated the performance of the h-likelihood method when the survival data, given the covariates in a specific cluster, follow a log-logistic distribution. We compared its estimation results to those obtained using VB and MCMC algorithms. We found that the h-likelihood method, *survregHL*, results in higher mean squared errors (MSEs) compared to VB and MCMC algorithms. Specifically, *survregHL* produced a 49.1% higher MSE than VB and a 41.2% higher MSE than MCMC for estimating the variance of the random intercept in our simulation study. Additionally, in our application to ICU ventilation duration analysis, we observed significant differences in some regression coefficients between h-likelihood and VB or MCMC methods. Therefore, our proposed shared frailty log-logistic AFT model using variational Bayes can be viewed as a better approach to the shared frailty log-normal AFT model using h-likelihood.

Our application of the proposed method to the CCIS ICU data for ventilation duration anal-

ysis reveals a moderate correlation of 0.134 among patients within the same ICU site. By incorporating the ICU site-specific random effect as an unknown shared frailty, we further validate the significant risk factors including the patient severity score MODS identified in the study by Kobara et al. (2023) based on the 95% credible intervals obtained from the posterior distributions of each regression coefficient. We observe a similar trend regarding the effect of MODS on ventilation duration as reported by Kobara et al. (2023). Specifically, when MODS reaches a severe level, the increase in ventilation duration becomes less pronounced. This may be due to patient mortality, which warrants further investigation in future studies. We demonstrate that different ICU sites have varying effects on patient ventilation duration, as observed through differences in the estimated ICU site-specific random effects. Our research provides valuable insights and practical implications for clinical practice and resource management. Specifically, ICUs with smaller estimated random intercepts tend to have shorter overall ventilation durations compared to other ICUs. Understanding the patient characteristics and ventilation practices within these ICUs may help improve clinical performance. For ICUs with larger estimated random intercepts, equipping more ventilators could enhance the overall efficiency of the ICU ventilation procedures.

We present several open problems and directions for future work related to our proposed methodology. As discussed, we assume that the survival time from a specific cluster follows a log-logistic distribution. The robustness of the proposed shared frailty log-logistic AFT model to other distributions remains unknown and warrants further study and comparison with the shared frailty log-normal AFT model. Additionally, extending the current VB algorithm to accommodate other survival distributions, such as the Weibull distribution, could be an interesting avenue of research. In our current framework, we account for cluster-level uncertainty using a random intercept. This approach could be extended to a more general model that includes cluster-level covariates (e.g., the ICU type, general or specialized), resulting in a mixed-effects model. Such a general model can be used to

better assess differences in performance across ICU sites regarding the duration of invasive mechanical ventilation. Another potential area for development is the integration of variable selection techniques within the VB algorithm, see Park and Do Ha (2019) as a reference.

References

Bishop, C. (2006). *Pattern Recognition and Machine Learning*. Springer. → page 105, 108

Blei, D. M., A. Kucukelbir, and J. D. McAuliffe (2017). Variational inference: A review for statisticians. *Journal of the American Statistical Association 112*(518), 859–877. → page 106, 108

Burgess Jr, J. F., C. L. Christiansen, S. E. Michalak, and C. N. Morris (2000). Medical profiling: improving standards and risk adjustments using hierarchical models. *Journal of Health Economics 19*(3), 291–309. → page 104

Canadian Institute for Health Information (2020). Care in Canadian ICUs. Technical report, Canadian Institute for Health Information, Canada. → page 103

da Cruz, A. C., C. P. E. de Souza, and P. H. T. O. Sousa (2024). Fast Bayesian basis selection for functional data representation with correlated errors. *arXiv*. → page 106

Do Ha, I., J.-H. Jeong, and Y. Lee (2017). Statistical modelling of survival data with random effects. *Statistics for Biology and Health*. → page 105, 106, 117

Do Ha, I., Y. Lee, and J.-K. Song (2002). Hierarchical-likelihood approach for mixed linear models with censored data. *Lifetime Data Analysis 8*, 163–176. → page 104, 130

Do Ha, I., M. Noh, and Y. Lee (2012). frailtyHL: A package for fitting frailty models with h-likelihood. *R J. 4*(2), 28. → page 105

Glance, L. G., A. W. Dick, T. M. Osler, and D. Mukamel (2003). Using hierarchical modeling to measure ICU quality. *Intensive Care Medicine 29*(12), 2223–2229. → page 104

Gorfine, M. and D. M. Zucker (2023). Shared frailty methods for complex survival data: a review of recent advances. *Annual Review of Statistics and Its Application 10*, 51–73. → page 103, 104

Hanagal, D. D. (2011). *Modeling Survival Data Using Frailty Models*. Springer. → page 103

Hoffman, M. D., D. M. Blei, C. Wang, and J. Paisley (2013). Stochastic variational inference. *Journal of Machine Learning Research*. → page 105

Honerkamp-Smith, G. and R. Xu (2016). Three measures of explained variation for correlated survival data under the proportional hazards mixed-effects model. *Statistics in Medicine 35*(23), 4153–4165. → page 103

Hougaard, P. (1995). Frailty models for survival data. *Lifetime Data Analysis 1*, 255–273. → page 103

Jordan, M. I., Z. Ghahramani, T. Jaakkola, and L. Saul (1999). Introduction to variational methods for graphical models. *Machine Learning 37*, 183–233. → page 105

Kobara, Y. M., M. Wismer, F. F. Rodrigues, and C. P. E. de Souza (2023). Invasive mechanical ventilation duration prediction using survival analysis. *International Journal of Healthcare Management*, 1–11. → page 104, 106, 121, 122, 131

Lambert, P., D. Collett, A. Kimber, and R. Johnson (2004). Parametric accelerated failure time models with random effects and an application to kidney transplant survival. *Statistics in Medicine 23*(20), 3177–3192. → page 104, 126

Lee, C. Y. Y. and M. P. Wand (2016a). Streamlined mean field variational Bayes for longitudinal and multilevel data analysis. *Biometrical Journal 58*(4), 868–895. → page 105

Lee, C. Y. Y. and M. P. Wand (2016b). Variational methods for fitting complex Bayesian mixed effects models to health data. *Statistics in Medicine 35*(2), 165–188. → page 109

Liu, X.-R., Y. Pawitan, and M. S. Clements (2017). Generalized survival models for correlated time-to-event data. *Statistics in Medicine 36*(29), 4743–4762. → page 103

Luo, S., M. Yi, X. Huang, and K. K. Hunt (2013). A Bayesian model for misclassified binary outcomes and correlated survival data with applications to breast cancer. *Statistics in Medicine 32*(13), 2320–2334. → page 103

Luts, J. and M. P. Wand (2015). Variational inference for count response semiparametric regression. *Bayesian Analysis 10*(4), 991 – 1023. → page 105

Marshall, J. C., D. J. Cook, N. V. Christou, G. R. Bernard, C. L. Sprung, and W. J. Sibbald (1995). Multiple organ dysfunction score: a reliable descriptor of a complex clinical outcome. *Critical Care Medicine 23*(10), 1638–1652. → page 104

Neville, S. E., M. Palmer, and M. Wand (2011). Generalized extreme value additive model analysis via mean field variational Bayes. *Australian & New Zealand Journal of Statistics 53*(3), 305–330. → page 105

Nolan, T. H., M. Menictas, and M. P. Wand (2020). Streamlined variational inference with higher level random effects. *Journal of Machine Learning Research 21*(157), 1–62. → page 107, 109

Park, E. and I. Do Ha (2019). Penalized variable selection for accelerated failure time models with random effects. *Statistics in Medicine 38*(5), 878–892. → page 105, 132

Pham, T. H., J. T. Ormerod, and M. Wand (2013). Mean field variational Bayesian inference for nonparametric regression with measurement error. *Computational Statistics & Data Analysis 68*, 375–387. → page 105

Ranganath, R., S. Gerrish, and D. Blei (2014). Black box variational inference. In *Artificial Intelligence and Statistics*, pp. 814–822. PMLR. → page 105

Ray, K. and B. Szabó (2022). Variational Bayes for high-dimensional linear regression with sparse priors. *Journal of the American Statistical Association 117*(539), 1270–1281. → page 105

Robinson, G. K. (1991). That blup is a good thing: the estimation of random effects. *Statistical Science*, 15–32. → page 107

Xian, C., C. P. de Souza, W. He, F. F. Rodrigues, and R. Tian (2024a). Variational Bayesian analysis of survival data using a log-logistic accelerated failure time model. *Statistics and Computing 34*(2), 67. → page 106, 109, 110, 114, 122

Xian, C., C. P. E. de Souza, W. He, F. F. Rodrigues, and R. Tian (2024b). Fast variational Bayesian inference for correlated survival data: an application to invasive mechanical ventilation duration analysis. → page 103

Xian, C., C. P. E. de Souza, J. Jewell, and R. Dias (2024). Clustering functional data via variational inference. *Advances in Data Analysis and Classification*, 1–50. → page 106

Yao, Y., A. Vehtari, D. Simpson, and A. Gelman (2018). Yes, but did it work?: Evaluating variational inference. In *Proceedings of the 35th International Conference on Machine Learning*, Volume 80 of *Proceedings of Machine Learning Research*, pp. 5581–5590. PMLR. → page 114

Zhou, H., T. Hanson, and J. Zhang (2017). Generalized accelerated failure time spatial frailty model for arbitrarily censored data. *Lifetime Data Analysis 23*, 495–515. → page 105, 117

Zhou, S., X. Zhuo, Z. Chen, and Y. Tao (2020). A new separable piecewise linear learn-

ing algorithm for the stochastic empty container repositioning problem. *Mathematical Problems in Engineering 2020*, 1–16. → page 105, 106

Chapter 5
Summary and Future Work

In this thesis, we demonstrate the satisfactory performance of variational Bayesian inference in clustering functional data via a B-spline regression mixture model and modeling survival data via a log-logistic AFT model.

In Chapter 2, we develop a novel VB algorithm for clustering and smoothing functional data simultaneously via a B-spline regression mixture model with random intercepts. We employ the deviance information criterion to select the best number of clusters. The proposed VB algorithm is evaluated and compared with k-means, functional k-means, and two additional model-based methods through simulation studies under various scenarios. Our results demonstrate that the proposed VB algorithm achieves satisfactory clustering performance in both simulation and real data analyses. Furthermore, VB shows strong consistency with MCMC in estimating the marginal posterior distribution of B-spline basis coefficients and precision parameters, while offering a lower computational cost.

There are several promising directions for future work in Chapter 2. First, we currently employ a two-stage procedure to select the optimal number of clusters using the deviance information criterion. It would be valuable to develop a more comprehensive VB algorithm that enables the automatic selection of the optimal number of clusters via a Dirichlet process mixture model (Rigon, 2023). Additionally, as discussed in Section 2.5, extending our model to incorporate a more flexible dependence structure on the random errors is another potential avenue for exploration. For instance, we could assume that the errors in Model 1 (Section 2.2.2) follow a Gaussian process with mean zero and covariance function $\sigma^2\Psi(t, s)$, where t and s are two observed points on a functional curve. Specifically, the correlation function of an Ornstein–Uhlenbeck process, $\Psi(t, s) = \exp(-\omega|s - t|)$ with $\omega > 0$ (Williams and Rasmussen, 2006; Dias et al., 2013; da Cruz et al., 2024), could be consid-

ered. Another significant area for future work is the development of integrated software to implement the VB algorithm for functional data clustering, such as a user-friendly R package.

In Chapter 3, we develop an alternative approach to MCMC methods and infer the parameters of the log-logistic AFT model via a mean-field VB algorithm. A piecewise approximation technique is embedded in deriving the VB algorithm to achieve conjugacy. The proposed VB algorithm is evaluated and compared with frequentist and MCMC techniques using simulated data under various scenarios. We have demonstrated that our proposed VB algorithm consistently produces satisfactory estimation results and, in most scenarios, outperforms the likelihood-based method in terms of empirical MSE. When compared to MCMC, similar performance was achieved by our proposed VB, and, in certain scenarios, VB yielded the lowest MSE. Furthermore, the proposed VB algorithm offers a significantly reduced computational cost compared to MCMC, with an average speedup of 300 times. A publicly available dataset is employed to illustrate our proposed methodology.

In Chapter 4, motivated by invasive mechanical ventilation data from different ICUs in Ontario, Canada, we introduce a shared frailty log-logistic accelerated failure time model that accounts for intra-cluster correlation through a cluster-specific random intercept. We present a novel and fast VB algorithm for parameter inference and evaluate its performance using simulation studies that vary the number and sizes of clusters. Additionally, we compare the performance of our proposed VB algorithm with the h-likelihood method and a MCMC algorithm. The proposed algorithm delivers satisfactory results and demonstrates computational efficiency, being over 150 times faster than a MCMC algorithm. We apply our method to the ICU ventilation data from Ontario to investigate the ICU site random effect on ventilation duration. We demonstrate that ICU sites perform differently regarding patient ventilation duration, as shown by the observed differences in the estimated site-specific random effects. Our research offers insights and implications for clinical practice

and resource management. For example, ICUs with smaller estimated random intercepts perform better in ventilation operations, as patient ventilation durations in these ICUs are shorter than in others. Other ICUs can learn from these high-performing ICUs to improve their ventilation practices.

We identify several open problems and future research directions related to our proposed methodology in Chapters 3 and 4. As noted, we assume that survival time follows a log-logistic distribution. However, the robustness of the proposed (shared frailty) log-logistic AFT model with VB algorithm to alternative distributions remains unexplored and requires further examination. Extending the current VB algorithm to accommodate other survival distributions, such as the Weibull and log-gamma distributions, could be interesting. Another promising area for advancement is the integration of variable selection techniques within the VB algorithm, as studied in Park and Do Ha (2019). In our existing framework presented in Chapter 4, we address cluster-level uncertainty using a random intercept. This method could be extended to a more general model that includes cluster-level covariates, resulting in a mixed-effects model.

References

da Cruz, A. C., C. P. E. de Souza, and P. H. T. O. Sousa (2024). Fast Bayesian basis selection for functional data representation with correlated errors. *arXiv*. → page 138

Dias, R., N. L. Garcia, and A. M. Schmidt (2013). A hierarchical model for aggregated functional data. *Technometrics 55*(3), 321–334. → page 138

Park, E. and I. Do Ha (2019). Penalized variable selection for accelerated failure time models with random effects. *Statistics in Medicine 38*(5), 878–892. → page 140

Rigon, T. (2023). An enriched mixture model for functional clustering. *Applied Stochastic Models in Business and Industry 39*(2), 232–250. → page 138

Williams, C. K. and C. E. Rasmussen (2006). *Gaussian Processes for Machine Learning*, Volume 2. MIT press Cambridge, MA. → page 138

This thesis contains five appendices, each covering different supplementary materials. Appendix A presents the VB algorithm for Model 1, discussed in Chapter 2, detailing the main steps of the algorithm and the ELBO calculation. Appendix B includes additional plots from the simulation study and the raw curves of the Canadian weather data referenced in the real data analysis section of Chapter 2. Appendix C provides the derivation details of the VB algorithm for the log-logistic AFT model in Chapter 3. The details of the piecewise approximation proposed in Chapter 3 are presented in Appendix D. Appendix E gives the details of the VB algorithm for the shared frailty log-logistic AFT model proposed in Chapter 4.

A Chapter 2: VB algorithm for Model 1

A.1 Main steps

This section describes the main steps of the VB algorithm for inferring $\boldsymbol{Z}$, $\boldsymbol{\phi}$, $\boldsymbol{\pi}$ and $\boldsymbol{\tau}$ in Model 1 in Section 2.2.2.1, which is summarized in Algorithm 4.

1. VD factorization:

$$q(\mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau}) = \prod_{i=1}^{N} q(Z_i) \times \prod_{k=1}^{K} q(\boldsymbol{\phi}_k) \times \prod_{k=1}^{K} q(\tau_k) \times q(\boldsymbol{\pi}). \tag{A.1}$$

2. Complete data log-likelihood:

$$\begin{aligned} \log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau}) \;\; &= \;\; \log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\phi}, \boldsymbol{\tau}) \;+\; \log p(\mathbf{Z}|\boldsymbol{\pi}) \\ &\quad + \log p(\boldsymbol{\phi}) \;+\; \log p(\boldsymbol{\tau}) \;+\; \log p(\boldsymbol{\pi}). \end{aligned} \tag{A.2}$$

3. Update equations:

i) Update equation for $q(\boldsymbol{\pi})$

Since only the second term, $\log p(\mathbf{Z}|\boldsymbol{\pi})$, and the last term, $\log p(\boldsymbol{\pi})$, in (A.2) depend on $\boldsymbol{\pi}$, the update equation $q^*(\boldsymbol{\pi})$ can be derived as follows.

$$\begin{aligned} & \log q^*(\boldsymbol{\pi}) \\ \overset{+}{\approx} \;\; & \mathbb{E}_{-\boldsymbol{\pi}}(\log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau})) \overset{+}{\approx} \mathbb{E}_{-\boldsymbol{\pi}}(\log p(\mathbf{Z}|\boldsymbol{\pi})) \;+\; \mathbb{E}_{-\boldsymbol{\pi}}(\log p(\boldsymbol{\pi})) \\ = \;\; & \mathbb{E}_{-\boldsymbol{\pi}}\Big[\sum_{i=1}^{N}\sum_{k=1}^{K} \mathrm{I}(Z_i = k) \log \pi_k\Big] + \log p(\boldsymbol{\pi}) \\ \overset{+}{\approx} \;\; & \sum_{k=1}^{K} \log \pi_k \Big[\sum_{i=1}^{N} \mathbb{E}_{q^*(Z_i)}(\mathrm{I}(Z_i = k))\Big] + \sum_{k=1}^{K} [d_k^0 - 1] \log \pi_k \\ = \;\; & \sum_{k=1}^{K} \log \pi_k \Big[\Big(\sum_{i=1}^{N} \mathbb{E}_{q^*(Z_i)}(\mathrm{I}(Z_i = k)) + d_k^0\Big) - 1\Big]. \end{aligned}$$

Therefore, $q^*(\boldsymbol{\pi})$ is a Dirichlet distribution with parameters $\mathbf{d}^* = (d_1^*, \ldots, d_K^*)$, where

$$d_k^* = d_k^0 + \sum_{i=1}^{N} \mathbb{E}_{q^*(Z_i)}(\mathrm{I}(Z_i = k)). \tag{A.3}$$

ii) *Update equation for* $q(Z_i)$

$$\log q^*(Z_i) \overset{+}{\approx} \mathbb{E}_{-Z_i}(\log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau})). \tag{A.4}$$

When taking the expectation above we just need to consider the first term, $\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\phi}, \boldsymbol{\tau})$, and the second term, $\log p(\mathbf{Z}|\boldsymbol{\pi})$, in (A.2). Note that we can write $\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\phi}, \boldsymbol{\tau})$ and $\log p(\mathbf{Z}|\boldsymbol{\pi})$ into two parts, one that depends on Z_i and one that does not.

$$\begin{aligned} \log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\phi}, \boldsymbol{\tau}) &= \sum_{k=1}^{K} \mathrm{I}(Z_i = k) \log p(\mathbf{Y}_i|Z_i = k, \boldsymbol{\phi}_k, \tau_k) \\ &\quad + \sum_{l:l\neq i} \sum_{k=1}^{K} \mathrm{I}(Z_l = k) \log p(\mathbf{Y}_l|Z_l = k, \boldsymbol{\phi}_k, \tau_k), \end{aligned}$$

$$\log p(\mathbf{Z}|\boldsymbol{\pi}) = \sum_{k=1}^{K} \mathrm{I}(Z_i = k) \log \pi_k + \sum_{l:l\neq i} \sum_{k=1}^{K} \mathrm{I}(Z_l = k) \log \pi_k.$$

Now when taking the expectation in (A.4) the parts that do not depend on Z_i in $\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\phi}, \boldsymbol{\tau})$ and $\log p(\mathbf{Z}|\boldsymbol{\pi})$ in (A.2) will be added as a constant in the expectation. So, we obtain

$$\begin{aligned} \log q^*(Z_i) \overset{+}{\approx} &\sum_{k=1}^{K} I(Z_i = k)\Big\{\frac{n_i}{2}\mathbb{E}_{q^*(\tau_k)}(\log \tau_k) \\ &-\frac{1}{2}\mathbb{E}_{q^*(\tau_k)}(\tau_k)\mathbb{E}_{q^*(\boldsymbol{\phi}_k)}[(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k)^T(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k)] \\ &+ \mathbb{E}_{q^*(\boldsymbol{\pi})}(\log \pi_k)\Big\}. \end{aligned}$$

Therefore, $q^*(Z_i)$ is a categorical distribution with parameters

$$p^*_{ik} = \frac{e^{\alpha_{ik}}}{\sum_{k=1}^{K} e^{\alpha_{ik}}}, \tag{A.5}$$

where

$$\begin{aligned}\alpha_{ik} &= \frac{n_i}{2}\mathbb{E}_{q^*(\tau_k)}(\log \tau_k) - \frac{1}{2}\mathbb{E}_{q^*(\tau_k)}(\tau_k)\mathbb{E}_{q^*(\boldsymbol{\phi}_k)}[(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k)^T(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k)] \\ &\quad + \mathbb{E}_{q^*(\boldsymbol{\pi})}(\log \pi_k).\end{aligned}$$

iii) Update equation for $q(\boldsymbol{\phi}_k)$

Note that only the first term, $\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\phi}, \boldsymbol{\tau})$, and the third term, $\log p(\boldsymbol{\phi})$, in (A.2) depend on $\boldsymbol{\phi}_k$. Similarly to the previous case for $q^*(Z_i)$, we can write $\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\phi}, \boldsymbol{\tau})$ and $\log p(\boldsymbol{\phi})$ in two parts, one that depends on $\boldsymbol{\phi}_k$ and the other that does not. Therefore, we obtain

$$\begin{aligned}\log q^*(\boldsymbol{\phi}_k) &\overset{+}{\approx} \mathbb{E}_{-\boldsymbol{\phi}_k}(\log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau})) \\ &\overset{+}{\approx} \mathbb{E}_{q^*(\tau_k)}(\log \tau_k) \sum_{i=1}^{N} \frac{n_i}{2}\mathbb{E}_{q^*(Z_i)}[\mathrm{I}(Z_i = k)] \\ &\quad - \frac{1}{2}\mathbb{E}_{q^*(\tau_k)}(\tau_k) \sum_{i=1}^{N} \\ &\quad \left\{\mathbb{E}_{q^*(Z_i)}[\mathrm{I}(Z_i = k)](\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k)^T(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k)\right\} \qquad \text{(A.6)} \\ &\quad - \frac{M}{2}\log v^0 - \frac{1}{2}v^0(\boldsymbol{\phi}_k - \mathbf{m}^0_k)^T(\boldsymbol{\phi}_k - \mathbf{m}^0_k). \qquad \text{(A.7)}\end{aligned}$$

All expectations will be later defined, but note that, for example, $\mathbb{E}_{q^*(Z_i)}[\mathrm{I}(Z_i = k)] = p^*_{ik}$.

First, we will focus on the quadratic forms that appear in (A.6) and (A.7).

$$
\begin{aligned}
&-\frac{1}{2}\mathbb{E}_{q^*(\tau_k)}(\tau_k)\sum_{i=1}^{N} p_{ik}^*(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k)^T(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k)\\
&-\frac{1}{2}v^0(\boldsymbol{\phi}_k - \mathbf{m}_k^0)^T(\boldsymbol{\phi}_k - \mathbf{m}_k^0) =\\
&-\frac{1}{2}\mathbb{E}_{q^*(\tau_k)}(\tau_k)\sum_{i=1}^{N} p_{ik}^*[\mathbf{Y}_i^T\mathbf{Y}_i - 2\mathbf{Y}_i^T\mathbf{B}_i\boldsymbol{\phi}_k + \boldsymbol{\phi}_k^T\mathbf{B}_i^T\mathbf{B}_i\boldsymbol{\phi}_k]\\
&-\frac{1}{2}v^0[\boldsymbol{\phi}_k^T\boldsymbol{\phi}_k - 2(\mathbf{m}_k^0)^T\boldsymbol{\phi}_k + (\mathbf{m}_k^0)^T\mathbf{m}_k^0] \overset{+}{\approx}\\
&-\frac{1}{2}\boldsymbol{\phi}_k^T\Big[v^0\mathbf{I} + \mathbb{E}_{q^*(\tau_k)}(\tau_k)\sum_{i=1}^{N} p_{ik}^*\mathbf{B}_i^T\mathbf{B}_i\Big]\boldsymbol{\phi}_k\\
&-\frac{1}{2}(-2)\Big[v^0(\mathbf{m}_k^0)^T + \mathbb{E}_{q^*(\tau_k)}(\tau_k)\sum_{i=1}^{N} p_{ik}^*\mathbf{Y}_i^T\mathbf{B}_i\Big]\boldsymbol{\phi}_k. \qquad \text{(A.8)}
\end{aligned}
$$

Now let

$$
\boldsymbol{\Sigma}_k^* = \Big[v^0\mathbf{I} + \mathbb{E}_{q^*(\tau_k)}(\tau_k)\sum_{i=1}^{N} p_{ik}^*\mathbf{B}_i^T\mathbf{B}_i\Big]^{-1}. \qquad \text{(A.9)}
$$

We can then rewrite (A.8) as

$$
-\frac{1}{2}\boldsymbol{\phi}_k^T\boldsymbol{\Sigma}_k^{*-1}\boldsymbol{\phi}_k - \frac{1}{2}(-2)\Big[v^0(\mathbf{m}_k^0)^T + \mathbb{E}_{q^*(\tau_k)}(\tau_k)\sum_{i=1}^{N} p_{ik}^*\mathbf{Y}_i^T\mathbf{B}_i\Big]\boldsymbol{\Sigma}_k^*\boldsymbol{\Sigma}_k^{*-1}\boldsymbol{\phi}_k.
$$

Therefore, $q^*(\boldsymbol{\phi}_k)$ is $MVN(\mathbf{m}_k^*, \boldsymbol{\Sigma}_k^*)$ with $\boldsymbol{\Sigma}_k^*$ as in (A.9) and mean vector

$$
\mathbf{m}_k^* = [v^0(\mathbf{m}_k^0)^T + \mathbb{E}_{q^*(\tau_k)}(\tau_k)\sum_{i=1}^{N} p_{ik}^*\mathbf{Y}_i^T\mathbf{B}_i]\boldsymbol{\Sigma}_k^*. \qquad \text{(A.10)}
$$

iv) Update equation for $q(\tau_k)$

Similarly to the calculations in i) and ii) we can write

$$
\begin{aligned}
\log q^*(\tau_k) \quad \overset{+}{\approx} \quad & \log\tau_k\sum_{i=1}^{N}\frac{n_i}{2}p_{ik}^* - \frac{1}{2}\tau_k\sum_{i=1}^{N} p_{ik}^*\mathbb{E}_{q^*(\boldsymbol{\phi}_k)}\Big[(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k)^T(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k)\Big]\\
& + (a^0 - 1)\log\tau_k - r^0\tau_k.
\end{aligned}
$$

Therefore, $q^*(\tau_k)$ is a Gamma distribution with parameters

$$A_k^* = a^0 + \sum_{i=1}^{N} \frac{n_i}{2} p_{ik}^*, \tag{A.11}$$

and

$$R_k^* = \Big(r^0 + \frac{1}{2}\sum_{i=1}^{N} p_{ik}^* \mathbb{E}_{q^*(\boldsymbol{\phi}_k)}\Big[(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k)^T(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k)\Big]\Big). \tag{A.12}$$

4. Expectations:

Next, we calculate the expectations in the update equations for each component in the VD. Let $\boldsymbol{\Psi}$ be the digamma function defined as

$$\boldsymbol{\Psi}(x) = \frac{d}{dx}\log\Gamma(x), \tag{A.13}$$

which can be easily calculated via numerical approximation. The values of the expectations taken with respect to the approximated distributions are given as follows.

$$\mathbb{E}_{q^*(Z_i)}[\mathrm{I}(Z_i = k)] = p_{ik}^*, \tag{A.14}$$

$$\mathbb{E}_{q^*(\tau_k)}(\tau_k) = \frac{A_k^*}{R_k^*}, \tag{A.15}$$

$$\mathbb{E}_{q^*(\tau_k)}(\log\tau_k) = \boldsymbol{\Psi}(A_k^*) - \log R_k^*, \tag{A.16}$$

$$\mathbb{E}_{q^*(\boldsymbol{\pi})}(\log\pi_k) = \boldsymbol{\Psi}(d_k^*) - \boldsymbol{\Psi}\Big(\sum_{k=1}^{K} d_k^*\Big). \tag{A.17}$$

In addition, using the fact that $\mathbb{E}(\mathbf{X}^T\mathbf{X}) = \text{trace}[\text{Var}(\mathbf{X})] + \mathbb{E}(\mathbf{X})^T\mathbb{E}(\mathbf{X})$, we obtain

$$\begin{aligned} & \mathbb{E}_{q^*(\boldsymbol{\phi})}\Big[(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k)^T(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k)\Big] \\ = \; & \text{trace}(\mathbf{B}_i\boldsymbol{\Sigma}_k^*\mathbf{B}_i^T) + (\mathbf{Y}_i - \mathbf{B}_i\mathbf{m}_k^*)^T(\mathbf{Y}_i - \mathbf{B}_i\mathbf{m}_k^*). \end{aligned} \tag{A.18}$$

A.2 ELBO calculation

In this section, we show how to calculate the ELBO, which is the convergence criterion of our proposed VB algorithm and will be updated at the end of each iteration until it converges. Equation (2.4) gives the ELBO:

$$\text{ELBO}(q) = \mathbb{E}_{q^*}[\log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau})] - \mathbb{E}_{q^*}[\log q(\mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau})],$$

where

$$\begin{aligned} \mathbb{E}_{q^*}[\log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau})] = \; & \mathbb{E}_{q^*}[\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau})] + \mathbb{E}_{q^*}[\log p(\mathbf{Z}|\boldsymbol{\pi})] + \\ & \mathbb{E}_{q^*}[\log p(\boldsymbol{\phi})] + \mathbb{E}_{q^*}[\log p(\boldsymbol{\tau})] + \mathbb{E}_{q^*}[\log p(\boldsymbol{\phi})], \end{aligned}$$

and

$$\begin{aligned} \mathbb{E}_{q^*}[\log q(\mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau})] = \; & \mathbb{E}_{q^*}[\log q(\mathbf{Z})] + \mathbb{E}_{q^*}[\log q(\boldsymbol{\phi})] \\ & + \mathbb{E}_{q^*}[\log q(\boldsymbol{\pi})] + \mathbb{E}_{q^*}[\log q(\boldsymbol{\tau})]. \end{aligned}$$

Therefore, we can write the ELBO as the summation of 5 terms:

$$\begin{aligned} \text{ELBO}(q) = \; & \mathbb{E}_{q^*}[\log p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\phi}, \boldsymbol{\tau})] + diff_{\mathbf{Z}} + diff_{\boldsymbol{\phi}} \\ & + diff_{\boldsymbol{\tau}} + diff_{\boldsymbol{\pi}}, \end{aligned} \tag{A.19}$$

where,

$$diff_{\mathbf{Z}} = \mathbb{E}_{q^*}[\log p(\mathbf{Z}|\boldsymbol{\pi})] - \mathbb{E}_{q^*}[\log q(\mathbf{Z})].$$

Specifically,

$$diff_{\mathbf{Z}} \equiv \sum_{i=1}^{N}\sum_{k=1}^{K} p_{ik}^* \mathbb{E}_{q^*(\boldsymbol{\pi})}(\log \pi_k) - \sum_{i=1}^{N}\sum_{k=1}^{K} p_{ik}^* \log p_{ik}^*. \tag{A.20}$$

The other terms in (A.19) are calculated as follows:

$$diff_{\boldsymbol{\phi}} \equiv -\frac{1}{2}\sum_{k=1}^{K} v_k^0\{\text{trace}(\boldsymbol{\Sigma}_k^*) + (\mathbf{m}_k^* - \mathbf{m}_k^0)^T(\mathbf{m}_k^* - \mathbf{m}_k^0)\} + \frac{1}{2}\sum_{k=1}^{K}\log|\boldsymbol{\Sigma}_k^*|,$$

$$\begin{aligned} diff_{\boldsymbol{\tau}} \quad \equiv \quad & \sum_{k=1}^{K}\{(a^0-1)\mathbb{E}_{q^*(\tau_k)}(\log\tau_k) - r^0\mathbb{E}_{q^*(\tau_k)}(\tau_k)\} \\ & - \sum_{k=1}^{K}\{A_k^*(\log R_k^* - 1) - \log\Gamma(A_k^*) \\ & + (A_k^*-1)\mathbb{E}_{q^*(\tau_k)}(\log\tau_k)\}, \end{aligned} \tag{A.21}$$

$$diff_{\boldsymbol{\pi}} \equiv \sum_{k=1}^{K}(d_k^0 - d_k^*)\mathbb{E}_{q^*(\boldsymbol{\pi})}(\log\pi_k),$$

and

$$\begin{aligned} \mathbb{E}_{q^*}[\log p(\mathbf{Y}|\mathbf{Z},\boldsymbol{\pi},\boldsymbol{\phi},\boldsymbol{\tau})] \quad = \quad & \sum_{i=1}^{N}\sum_{k=1}^{K} p_{ik}^*\Big\{\frac{n_i}{2}\mathbb{E}_{q^*(\tau_k)}(\log\tau_k) - \frac{1}{2}\frac{A_k^*}{R_k^*} \\ & \mathbb{E}_{q^*(\boldsymbol{\phi})}\Big[(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k)^T(\mathbf{Y}_i - \mathbf{B}_i\boldsymbol{\phi}_k)\Big]\Big\}. \end{aligned} \tag{A.22}$$

Therefore, at iteration c, we calculate $\text{ELBO}^{(c)}$ using all parameters obtained at the end of iteration c. Convergence of the algorithm is achieved if $\text{ELBO}^{(c)} - \text{ELBO}^{(c-1)}$ is smaller than a given threshold. It is important to note that we use the fact that $\lim_{p_{ik}^*\to 0} p_{ik}^*\log p_{ik}^* = 0$ to avoid numerical issues when calculating (A.20).

Algorithm 4: Clustering functional data via variational inference

Data: N original curves with n_i evaluation points for the ith curve and the $\mathbf{B}_i$ matrix containing the evaluation values of the basis functions, $i = 1, ..., N$; number of clusters K; values of hyperparameters: $\mathbf{d}^0$, $\mathbf{m}_k^0, k = 1, ..., K$, s^0, a^0, r^0; convergence threshold and maximum number of iterations

Result: VB estimated mean curves for each cluster and the cluster index for each original curve

Initialization: initialize R_k^* with arbitrary values (e.g., $R_k^* = r^0$) and p_{ik}^* from k-means, and set $c = 0$;

while $c <$ *maximum number of iterations and difference of ELBO > convergence threshold* **do**

  repeat

    $c = c + 1$;

    update $A_k^{*(c)}$ using $p_{1k}^{*(c-1)}, \ldots, p_{Nk}^{*(c-1)}$ with equation (A.11);

    update $\mathbf{\Sigma}_k^{*(c)}$ using $A_k^{*(c)}$, $R_k^{*(c-1)}$ and $p_{1k}^{*(c-1)}, \ldots, p_{Nk}^{*(c-1)}$ with equations (A.9) and (A.15);

    update $\mathbf{m}_k^{*(c)}$ using $\mathbf{\Sigma}_k^{*(c)}$, $A_k^{*(c)}$, $R_k^{*(c-1)}$ and $p_{1k}^{*(c-1)}, \ldots, p_{Nk}^{*(c-1)}$ with equations (A.10) and (A.15);

    update $R_k^{*(c)}$ using $\mathbf{m}_k^{*(c)}$, $\mathbf{\Sigma}_k^{*(c)}$ and $p_{1k}^{*(c-1)}, \ldots, p_{Nk}^{*(c-1)}$ with equations (A.12) and (A.18);

    update $\boldsymbol{d}^{*(c)}$ using $p_{1k}^{*(c-1)}, \ldots, p_{Nk}^{*(c-1)}$ with equations (A.3) and (A.14);

    update $p_{1k}^{*(c)}, \ldots, p_{Nk}^{*(c)}$ using $R_k^{*(c)}$, $\boldsymbol{d}^{*(c)}$, $\mathbf{m}_k^{*(c)}$ and $\mathbf{\Sigma}_k^{*(c)}$ with equations (A.5), (A.15), (A.16), (A.17) and (A.18);

    calculate the current ELBO, $\text{ELBO}^{(c)}$ using formulas in section A.2;

    calculate the difference of ELBO = $\text{ELBO}^{(c)} - \text{ELBO}^{(c-1)}$;

  until *maximum iteration is achieved or the ELBO converges*;

end

B Chapter 2: Plots

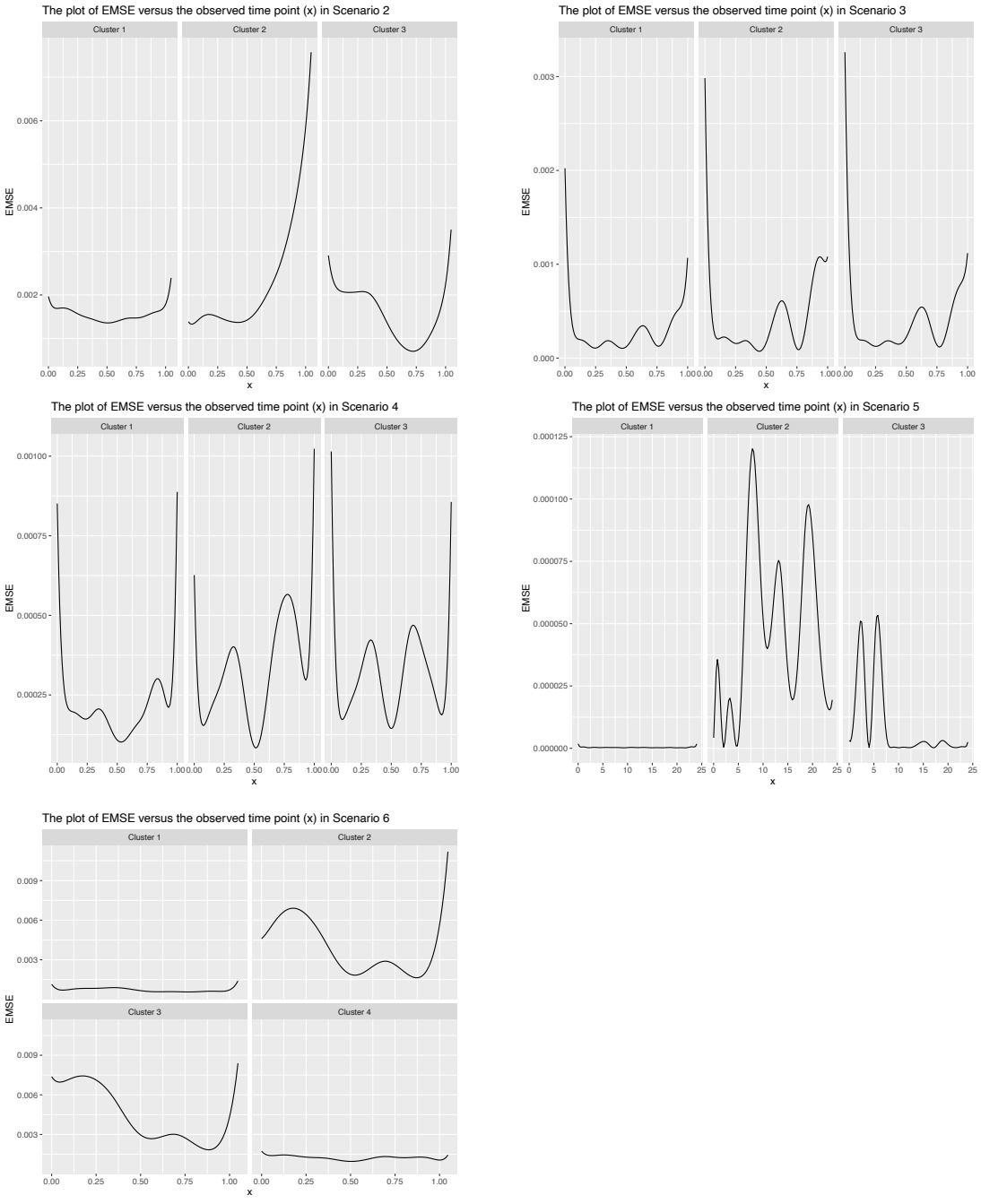


Figure B.1: EMSE versus the observed evaluation point for each cluster in Scenarios 2, 3, 4, 5 and 6. In Scenario 5, the straight line in cluster one does not mean there is no EMSE. This is because compared to cluster two and three, the EMSE in cluster one is very small (the median is 1.41×10^{-11}).

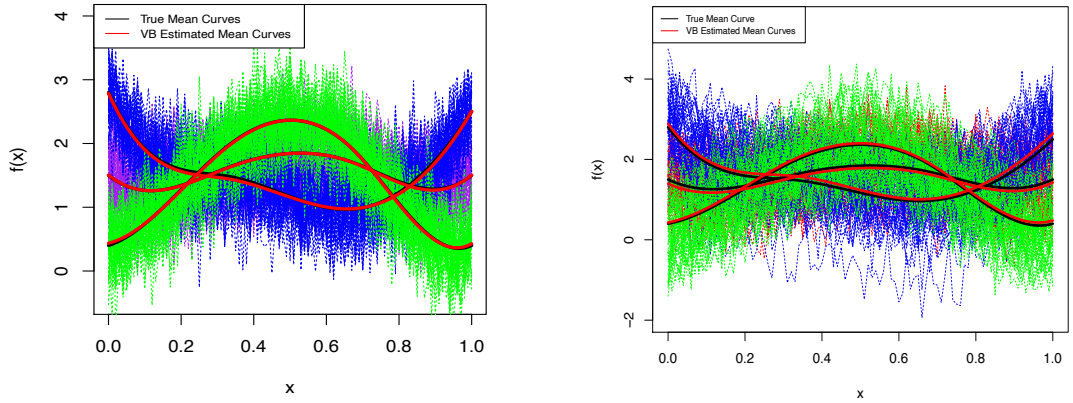


Figure B.2: Example of simulated data under Scenario 10 (left) and Scenario 12 (right) for Model 2. Raw curves (different colors correspond to different clusters), cluster-specific true mean curves (in black) and corresponding estimated mean curves (in red).

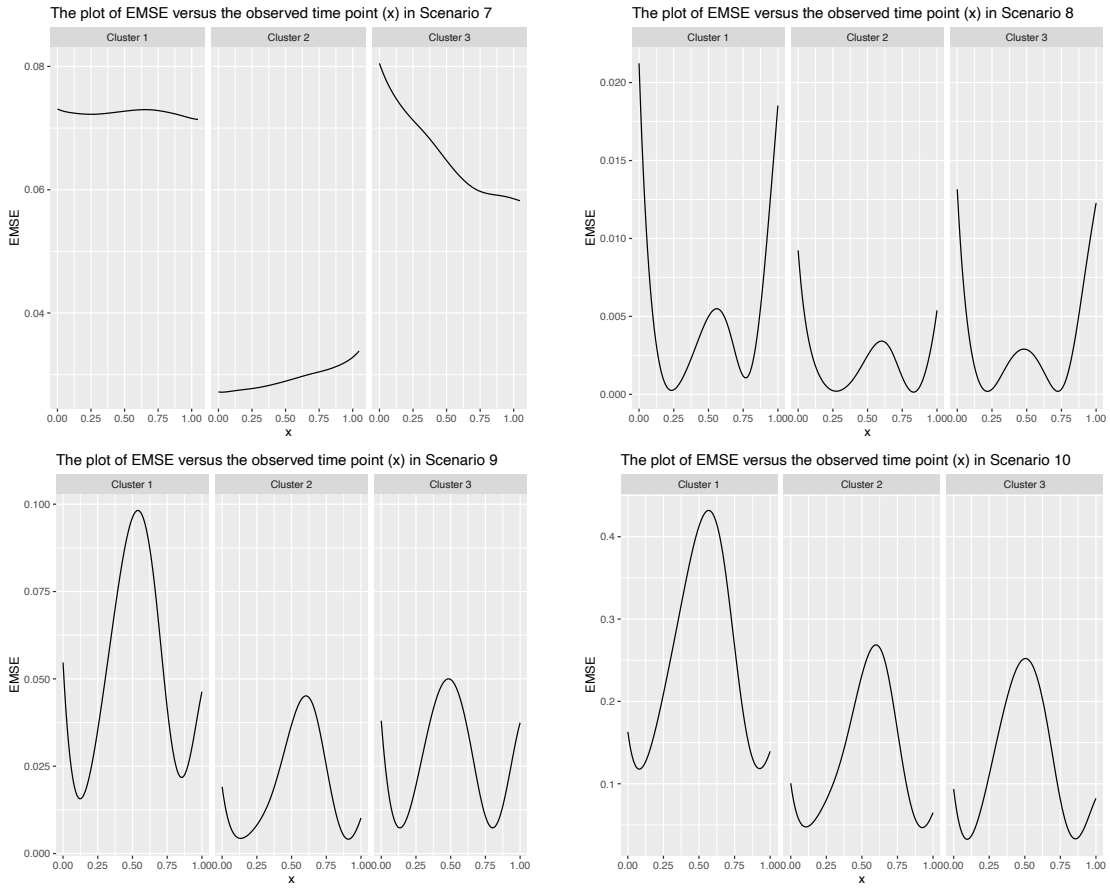


Figure B.3: EMSE versus the observed evaluation point for each cluster in Scenarios 9, 10, 11 and 12.

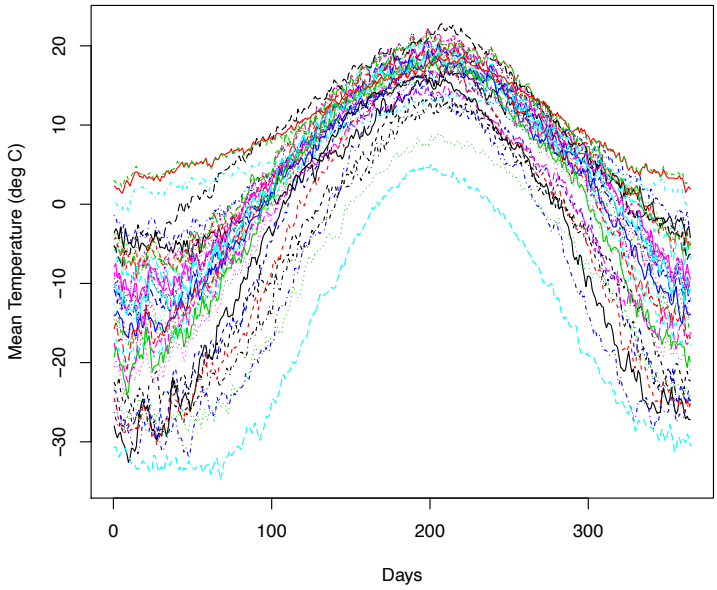


Figure B.4: Raw curves of the Canadian weather data. Different curves have different colors.

C Chapter 3: Update equations and ELBO calculation

In this appendix, we derive the update equation for each component and the ELBO clacculation in our model. We use $\overset{+}{\approx}$ to denote equality up to a constant additive factor for convenience.

C.1 VB update equations

(1) Update for $q^*(\boldsymbol{\beta})$

$$\log q^*(\boldsymbol{\beta}) \quad \overset{+}{\approx} \quad \mathbb{E}_{q(b)}[\log p(\mathbf{D}\,|\,\boldsymbol{\beta}, b) + \log p(\boldsymbol{\beta})] = \mathbb{E}_{q(b)}[\log p(\mathbf{D}\,|\,\boldsymbol{\beta}, b)] + \mathbb{E}_{q(b)}[\log p(\boldsymbol{\beta})],$$

where

$$\begin{aligned}
& \mathbb{E}_{q(b)}[\log p(\mathbf{D}\,|\,\boldsymbol{\beta}, b)] \\
= \;& \mathbb{E}_{q(b)}\Bigg[- r\log b + \sum_{i=1}^{n}\Bigg(\delta_i \frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b} - (1+\delta_i)\log\Big(1 + \exp\big(\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b}\big)\Big)\Bigg)\Bigg] \\
= \;& -r\mathbb{E}_{q(b)}(\log b) + \sum_{i=1}^{n}\Bigg(\delta_i(y_i - \mathbf{X}_i^T\boldsymbol{\beta})\mathbb{E}_{q(b)}\Big(\frac{1}{b}\Big) \\
& -(1+\delta_i)\mathbb{E}_{q(b)}\Bigg[\log\Big(1 + \exp\big(\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b}\big)\Big)\Bigg]\Bigg). \qquad \text{(C.1)}
\end{aligned}$$

To calculate the last expectation in (C.1) and achieve conjugacy, we then propose and apply a quadratic piecewise approximation of $\log(1 + \exp(x))$ (see Equation (D.1) in Appendix D) to $\log\Big(1 + \exp\big(\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b}\big)\Big)$ obtaining:

$$\begin{aligned}
& \log\Big(1 + \exp\big(\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b}\big)\Big) \\
\approx \;& 0^{\nu_{i1}} \times 0.1696^{\nu_{i2}} \times 0.5^{\nu_{i3}} \times 0.8303^{\nu_{i4}} \times 1^{1-\sum_{j=1}^{4}\nu_{ij}} \frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b} \\
& + 0^{\nu_{i1}} \times 0.0189^{\nu_{i2}} \times 0.1138^{\nu_{i3}} \times 0.0190^{\nu_{i4}} \times 0^{1-\sum_{j=1}^{4}\nu_{ij}} \Bigg(\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b}\Bigg)^2,
\end{aligned}$$

where

$$\nu_{i1} = \begin{cases} 1 & \text{if } \frac{y_i - \mathbf{x}_i^T\boldsymbol{\beta}}{b} \leq -5 \\ 0 & \text{otherwise} \end{cases}, \quad \nu_{i2} = \begin{cases} 1 & \text{if } -5 < \frac{y_i - \mathbf{x}_i^T\boldsymbol{\beta}}{b} \leq -1.7 \\ 0 & \text{otherwise} \end{cases},$$

$$\nu_{i3} = \begin{cases} 1 & \text{if } -1.7 < \frac{y_i - \mathbf{x}_i^T\boldsymbol{\beta}}{b} \leq 1.7 \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad \nu_{i4} = \begin{cases} 1 & \text{if } 1.7 < \frac{y_i - \mathbf{x}_i^T\boldsymbol{\beta}}{b} \leq 5 \\ 0 & \text{otherwise} \end{cases}.$$

Let $\rho_i := 0^{\nu_{i1}} \times 0.1696^{\nu_{i2}} \times 0.5^{\nu_{i3}} \times 0.8303^{\nu_{i4}} \times 1^{1-\sum_{j=1}^4 \nu_{ij}}$ and $\zeta_i := 0^{\nu_{i1}} \times 0.0189^{\nu_{i2}} \times 0.1138^{\nu_{i3}} \times 0.0190^{\nu_{i4}} \times 0^{1-\sum_{j=1}^4 \nu_{ij}}$, we obtain

$$\log\Big(1 + \exp\big(\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b}\big)\Big) \approx \rho_i \frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b} + \zeta_i\Big(\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b}\Big)^2.$$

More details about the proposed quadratic piecewise approximation can be found in Appendix D. Therefore, we can write Equation (C.1) as

$$\begin{aligned}
& \mathbb{E}_{q(b)}[\log p(\mathbf{D} \,|\, \boldsymbol{\beta}, b)] \\
\overset{+}{\approx} \; & -r\mathbb{E}_{q(b)}[\log b] + \sum_{i=1}^{n}\Bigg(\delta_i(y_i - \mathbf{X}_i^T\boldsymbol{\beta})\mathbb{E}_{q(b)}\big(\frac{1}{b}\big) \\
& - (1+\delta_i)\mathbb{E}_{q(b)}\Big[\rho_i\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b} + \zeta_i\Big(\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b}\Big)^2\Big]\Bigg) \\
\overset{+}{\approx} \; & \sum_{i=1}^{n}\Bigg(-\delta_i\mathbf{X}_i^T\boldsymbol{\beta}\,\mathbb{E}_{q(b)}\big(\frac{1}{b}\big) - (1+\delta_i)\Big(-\rho_i\mathbf{X}_i^T\boldsymbol{\beta}\,\mathbb{E}_{q(b)}\big(\frac{1}{b}\big) \\
& + \zeta_i(-2y_i\mathbf{X}_i^T\boldsymbol{\beta} + \boldsymbol{\beta}^T\mathbf{X}_i\mathbf{X}_i^T\boldsymbol{\beta})\mathbb{E}_{q(b)}\big(\frac{1}{b^2}\big)\Big)\Bigg) \\
= \; & \sum_{i=1}^{n}\Bigg(\mathbb{E}_{q(b)}\big(\frac{1}{b}\big)\Big(-\delta_i + (1+\delta_i)\rho_i\Big)\mathbf{X}_i^T + 2\mathbb{E}_{q(b)}\big(\frac{1}{b^2}\big)(1+\delta_i)y_i\zeta_i\mathbf{X}_i^T\Bigg)\boldsymbol{\beta} \\
& - \boldsymbol{\beta}^T\Bigg(\mathbb{E}_{q(b)}\big(\frac{1}{b^2}\big)\sum_{i=1}^{n}(1+\delta_i)\zeta_i\mathbf{X}_i\mathbf{X}_i^T\Bigg)\boldsymbol{\beta},
\end{aligned} \tag{C.2}$$

and note that

$$\mathbb{E}_{q(b)}[\log p(\boldsymbol{\beta})] \overset{+}{\approx} \frac{p}{2}\log v_0 - \frac{1}{2}v_0(\boldsymbol{\beta}-\boldsymbol{\mu}_0)^T(\boldsymbol{\beta}-\boldsymbol{\mu}_0) \overset{+}{\approx} v_0\boldsymbol{\mu}_0^T\boldsymbol{\beta} - \frac{1}{2}v_0\boldsymbol{\beta}^T\boldsymbol{\beta}. \tag{C.3}$$

Combining Equations (C.2) and (C.3), we have

$$\begin{aligned}\log q^*(\beta) \overset{+}{\approx}\ & \left[v_0\boldsymbol{\mu}_0^T + \sum_{i=1}^{n}\left(\mathbb{E}_{q(b)}\Big(\frac{1}{b}\Big)\Big(-\delta_i + (1+\delta_i)\rho_i\Big)\mathbf{X}_i^T + 2\mathbb{E}_{q(b)}\Big(\frac{1}{b^2}\Big)(1+\delta_i)\zeta_i y_i\mathbf{X}_i^T\right)\right]\boldsymbol{\beta} \\ & - \frac{1}{2}\boldsymbol{\beta}^T\left[v_0\mathbf{I} + 2\mathbb{E}_{q(b)}\Big(\frac{1}{b^2}\Big)\sum_{i=1}^{n}(1+\delta_i)\zeta_i\mathbf{X}_i\mathbf{X}_i^T\right]\boldsymbol{\beta}.\end{aligned}$$

Let

$$\Sigma^* := \left[v_0\mathbf{I} + 2\mathbb{E}_{q(b)}\Big(\frac{1}{b^2}\Big)\sum_{i=1}^{n}(1+\delta_i)\zeta_i\mathbf{X}_i\mathbf{X}_i^T\right]^{-1}, \tag{C.4}$$

and

$$\boldsymbol{\mu}^* := \left[\left\{v_0\boldsymbol{\mu}_0^T + \sum_{i=1}^{n}\left(\mathbb{E}_{q(b)}\Big(\frac{1}{b}\Big)\Big(-\delta_i + (1+\delta_i)\rho_i\Big)\mathbf{X}_i^T + 2\mathbb{E}_{q(b)}\Big(\frac{1}{b^2}\Big)(1+\delta_i)y_i\zeta_i\mathbf{X}_i^T\right)\right\}\Sigma^*\right]^T. \tag{C.5}$$

Then, $q^*(\boldsymbol{\beta})$ is $N(\boldsymbol{\mu}^*, \Sigma^*)$. Therefore, we have the conjugate multivariate normal posterior distribution of $\boldsymbol{\beta}$ after applying the piecewise approximation to $\log\left(1+\exp\left(\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b}\right)\right)$.

(2) Update for $q^*(b)$

$$\log q^*(b) \overset{+}{\approx} \mathbb{E}_{q(\boldsymbol{\beta})}[\log p(\mathbf{D}\,|\,\boldsymbol{\beta}, b) + \log p(b)] = \mathbb{E}_{q(\boldsymbol{\beta})}[\log p(\mathbf{D}\,|\,\boldsymbol{\beta}, b)] + \mathbb{E}_{q(\boldsymbol{\beta})}[\log p(b)].$$

First, we can show that,

$$\begin{aligned}
& \mathbb{E}_{q(\boldsymbol{\beta})}[\log p(\mathbf{D} \mid \boldsymbol{\beta}, b)] \\
= \; & \mathbb{E}_{q(\boldsymbol{\beta})}\left[- r \log b + \sum_{i=1}^{n} \left(\delta_i \frac{y_i - \mathbf{X}_i^T \boldsymbol{\beta}}{b} - (1 + \delta_i) \log \left(1 + \exp \left(\frac{y_i - \mathbf{X}_i^T \boldsymbol{\beta}}{b} \right) \right) \right) \right] \\
= \; & -r \log b + \sum_{i=1}^{n} \mathbb{E}_{q(\boldsymbol{\beta})}\left[\delta_i \frac{y_i - \mathbf{X}_i^T \boldsymbol{\beta}}{b} - (1 + \delta_i) \log \left(1 + \exp \left(\frac{y_i - \mathbf{X}_i^T \boldsymbol{\beta}}{b} \right) \right) \right]. \qquad \text{(C.6)}
\end{aligned}$$

We then propose and apply a linear piecewise approximation of $\log(1 + \exp(x))$ (see Equation (D.1) in Appendix D) to $\log \left(1 + \exp \left(\frac{y_i - \mathbf{X}_i^T \boldsymbol{\beta}}{b} \right) \right)$ obtaining:

$$\begin{aligned}
\log \left(1 + \exp \left(\frac{y_i - \mathbf{X}_i^T \boldsymbol{\beta}}{b} \right) \right) \; \approx \; & 0^{\eta_{i1}} \times 0.0426^{\eta_{i2}} \times 0.3052^{\eta_{i3}} \times 0.6950^{\eta_{i4}} \\
& \times 0.9574^{\eta_{i5}} \times 1^{1 - \sum_{j=1}^{5} \eta_{ij}} \frac{y_i - \mathbf{X}_i^T \boldsymbol{\beta}}{b},
\end{aligned}$$

where

$$\eta_{i1} = \begin{cases} 1 & \text{if } \frac{y_i - \mathbf{X}_i^T \boldsymbol{\beta}}{b} \leq -5 \\ 0 & \text{otherwise} \end{cases}, \; \eta_{i2} = \begin{cases} 1 & \text{if } -5 < \frac{y_i - \mathbf{X}_i^T \boldsymbol{\beta}}{b} \leq -1.701 \\ 0 & \text{otherwise} \end{cases},$$

$$\eta_{i3} = \begin{cases} 1 & \text{if } -1.701 < \frac{y_i - \mathbf{X}_i^T \boldsymbol{\beta}}{b} \leq 0 \\ 0 & \text{otherwise} \end{cases}, \; \eta_{i4} = \begin{cases} 1 & \text{if } 0 < \frac{y_i - \mathbf{X}_i^T \boldsymbol{\beta}}{b} \leq 1.702 \\ 0 & \text{otherwise} \end{cases} \text{ and}$$

$$\eta_{i5} = \begin{cases} 1 & \text{if } 1.702 < \frac{y_i - \mathbf{X}_i^T \boldsymbol{\beta}}{b} \leq 5 \\ 0 & \text{otherwise} \end{cases}.$$

Let

$$\varphi_i := 0^{\eta_{i1}} \times 0.0426^{\eta_{i2}} \times 0.3052^{\eta_{i3}} \times 0.6950^{\eta_{i4}} \times 0.9574^{\eta_{i5}} \times 1^{1-\sum_{j=1}^{5}\eta_{ij}}, \tag{C.7}$$

we obtain

$$\log\Big(1 + \exp\big(\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b}\big)\Big) \approx \varphi_i \frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b}.$$

More details about the proposed linear piecewise approximation can be found in Appendix D. Therefore, we can write Equation (C.6) as

$$\begin{aligned}
\mathbb{E}_{q(\boldsymbol{\beta})}[\log p(\mathbf{D}\,|\,\boldsymbol{\beta}, b)] &\overset{+}{\approx} -r\log b + \sum_{i=1}^{n}\mathbb{E}_{q(\boldsymbol{\beta})}\left[\delta_i\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b} - (1+\delta_i)\varphi_i\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b}\right] \\
&\overset{+}{\approx} -r\log b + \frac{1}{b}\sum_{i=1}^{n}\Big(\delta_i - (1+\delta_i)\varphi_i\Big)\Big(y_i - \mathbf{X}_i^T\mathbb{E}_{q(\boldsymbol{\beta})}(\boldsymbol{\beta})\Big),
\end{aligned} \tag{C.8}$$

and note that

$$\mathbb{E}_{q(\boldsymbol{\beta})}[\log p(b)] \overset{+}{\approx} -(\alpha_0 + 1)\log b - \frac{\omega_0}{b}. \tag{C.9}$$

Combining Equations (C.8) and (C.9), we have

$$\log q^*(b) \overset{+}{\approx} -(\alpha_0 + r + 1)\log b - \frac{1}{b}\Big(\omega_0 - \sum_{i=1}^{n}\Big(\delta_i - (1+\delta_i)\varphi_i\Big)\Big(y_i - \mathbf{X}_i^T\mathbb{E}_{q(\boldsymbol{\beta})}(\boldsymbol{\beta})\Big)\Big).$$

Let

$$\alpha^* = \alpha_0 + r \quad \text{and} \quad \omega^* = \omega_0 - \sum_{i=1}^{n}\Big(\delta_i - (1+\delta_i)\varphi_i\Big)\Big(y_i - \mathbf{X}_i^T\mathbb{E}_{q(\boldsymbol{\beta})}(\boldsymbol{\beta})\Big),$$

then $q^*(b)$ is Inverse-Gamma(α^*, ω^*).

C.2 ELBO calculation

Since our goal is to find $q(\cdot)$ that maximizes the ELBO, the ELBO is used as the convergence criterion of our VB algorithm, which is defined as follows:

$$ELBO(q) = \mathbb{E}_q[\log p(\mathbf{D}, \boldsymbol{\beta}, b)] - \mathbb{E}_q[\log q(\boldsymbol{\beta}, b)],$$

where

$$\log p(\mathbf{D}, \boldsymbol{\beta}, b) = \log p(\mathbf{D} \,|\, \boldsymbol{\beta}, b) + \log p(\boldsymbol{\beta}) + \log p(b) \text{ and } \log q(\boldsymbol{\beta}, b) = \log q(\boldsymbol{\beta}) + \log q(b).$$

Let

$$diff_{\boldsymbol{\beta}} = \mathbb{E}_q[\log p(\boldsymbol{\beta})] - \mathbb{E}_q[\log q(\boldsymbol{\beta})] \text{ and } diff_b = \mathbb{E}_q[\log p(b)] - \mathbb{E}_q[\log q(b)],$$

then we can write the ELBO as

$$ELBO(q) = \mathbb{E}_q[\log p(\mathbf{D} \,|\, \boldsymbol{\beta}, b)] + diff_{\boldsymbol{\beta}} + diff_b. \tag{C.10}$$

We next present how to calculate each term in Equation (C.10) with expectations taken with respect to the approximated variational distributions denoted by $q^*(\cdot)$. When calculating the first term, $\mathbb{E}_{q^*}[\log p(\mathbf{D} \,|\, \boldsymbol{\beta}, b)]$, we apply the linear piecewise approximation to $\log(1 + \exp(x))$ again.

$$
\begin{aligned}
& \mathbb{E}_{q^*}[\log p(\mathbf{D}\,|\,\boldsymbol{\beta}, b)] \\
= \;& \mathbb{E}_{q^*}\Bigg[- r\log b + \sum_{i=1}^{n}\Bigg(\delta_i\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b} - (1+\delta_i)\log\Big(1+\exp\big(\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b}\big)\Big)\Bigg)\Bigg] \\
\approx \;& \mathbb{E}_{q^*}\Bigg[- r\log b + \sum_{i=1}^{n}\Bigg(\delta_i\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b} - (1+\delta_i)\varphi_i\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b}\Bigg)\Bigg] \\
\overset{+}{\approx} \;& -r\mathbb{E}_{q^*(b)}\Big(\log b\Big) + \sum_{i=1}^{n}\Big[\delta_i - (1+\delta_i)\varphi_i\Big]\mathbb{E}_{q^*(b)}\Bigg[\mathbb{E}_{q^*(\boldsymbol{\beta})}\frac{y_i - \mathbf{X}_i^T\boldsymbol{\beta}}{b}\Bigg] \\
= \;& -r\mathbb{E}_{q^*(b)}\Big(\log b\Big) + \mathbb{E}_{q^*(b)}\Big(\frac{1}{b}\Big)\sum_{i=1}^{n}\big(\delta_i - (1+\delta_i)\varphi_i\big)\big(y_i - \mathbf{X}_i^T\mathbb{E}_{q^*(\boldsymbol{\beta})}(\boldsymbol{\beta})\big),
\end{aligned}
$$

where φ_i is defined as Equation (C.7). Let $\boldsymbol{\Psi}$ be the digamma function defined as $\boldsymbol{\Psi}(x) = \frac{d}{dx}\log\Gamma(x)$, which can be easily calculated via numerical approximation. Then $\mathbb{E}_{q^*(b)}\log b$ can be calculated by $\mathbb{E}_{q^*(b)}\log b = \log(\omega^*) - \boldsymbol{\Psi}(\alpha^*)$. For $diff_{\boldsymbol{\beta}}$, we derive its calculation as follows, using the fact that $\mathbb{E}(\mathbf{X}^T\mathbf{X}) = \text{trace}[\text{Var}(\mathbf{X})] + \mathbb{E}(\mathbf{X})^T\mathbb{E}(\mathbf{X})$ where $\mathbf{X}$ is a column vector:

$$
\begin{aligned}
diff_{\boldsymbol{\beta}} \;=\;& \mathbb{E}_{q^*}[\log p(\boldsymbol{\beta})] - \mathbb{E}_{q^*}[\log q(\boldsymbol{\beta})] \\
\overset{+}{\approx} \;& \mathbb{E}_{q^*}[-\frac{1}{2}v_0(\boldsymbol{\beta}-\boldsymbol{\mu}_0)^T(\boldsymbol{\beta}-\boldsymbol{\mu}_0)] - \mathbb{E}_{q^*}[-\frac{1}{2}\log(|\Sigma^*|) - \frac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\mu}^*)^T(\Sigma^*)^{-1}(\boldsymbol{\beta}-\boldsymbol{\mu}^*)] \\
\overset{+}{\approx} \;& -\frac{1}{2}v_0[\text{trace}(\Sigma^*) + (\boldsymbol{\mu}^*-\boldsymbol{\mu}_0)^T(\boldsymbol{\mu}^*-\boldsymbol{\mu}_0)] + \frac{1}{2}\log(|\Sigma^*|).
\end{aligned}
$$

Note that

$$
\mathbb{E}_{q^*}[\frac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\mu}^*)^T(\Sigma^*)^{-1}(\boldsymbol{\beta}-\boldsymbol{\mu}^*)] = \frac{p}{2},
$$

which is always a constant at each iteration and therefore we ignore it.

For $diff_b$, we have

$$
\begin{aligned}
diff_b &= \mathbb{E}_{q^*}[\log p(b)] - \mathbb{E}_{q^*}[\log q(b)] \\
&\overset{+}{\approx} \mathbb{E}_{q^*}\Big[-(\alpha_0 + 1)\log b - \frac{\omega_0}{b}\Big] - \mathbb{E}_{q^*}\Big[\alpha^* \log \omega^* - \log(\Gamma(\alpha^*)) - (\alpha^* + 1)\log b - \frac{\omega^*}{b}\Big] \\
&= (\alpha^* - \alpha_0)\mathbb{E}_{q^*(b)}(\log b) + (\omega^* - \omega_0)\mathbb{E}_{q^*(b)}(\frac{1}{b}) - \alpha^* \log \omega^*.
\end{aligned}
$$

Since α^* does not change at each iteration, we remove $\log(\Gamma(\alpha^*))$ in the calculation of the ELBO.

D Chapter 3: Piecewise approximations of $\log(1+\exp(x))$

This section presents the idea and details of the piecewise approximations of $\log(1+\exp(x))$. In order to have the conjugacy in our variational Bayes algorithm, we apply piecewise approximations to $\log(1+\exp(x))$, which are used in Section 3.3.1. We know that $\log(1+\exp(x))$ is monotonically increasing in $(-\infty, \infty)$, and when x is approaching $-\infty$, $\log(1+\exp(x))$ approaches 0, while when x is approaching ∞, $\log(1+\exp(x))$ approaches x. Furthermore, when $x \leq -5$, $\log(1+\exp(x)) \approx 0$, and when $x \geq 5$, $\log(1+\exp(x)) \approx x$ since $\log(1+\exp(-5)) = 0.0067$ and $\log(1+\exp(5)) = 5.0067$. Therefore, our goal is to find appropriate piecewise approximations of $\log(1+\exp(x))$ in $[-5, 5]$ whose plot is presented in Figure D.1 Left. To do this, we apply the method introduced by Muggeo (2003) implemented in R with a package called *segmented* which can help find the optimal piecewise linear approximation using regression.

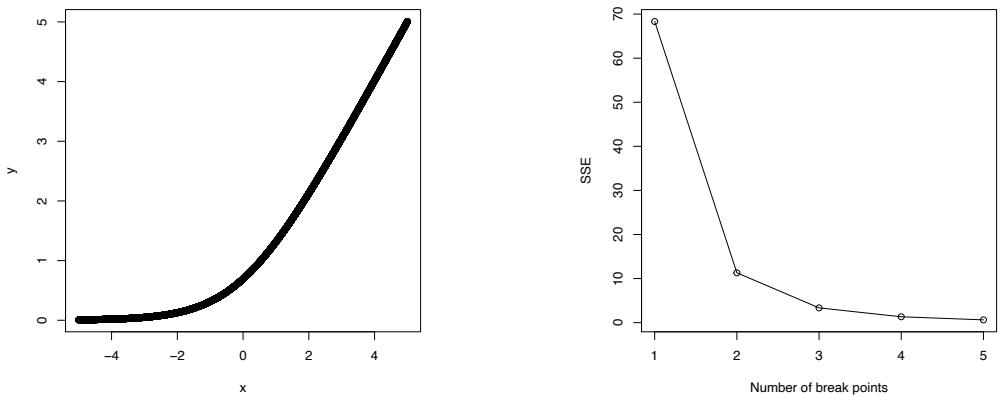


Figure D.1: Left: Plot of $\log(1+\exp(x))$ versus x for $x \in [-5, 5]$. Right: The plot of the sum of squared errors (SSE) versus the number of breakpoints in linear piecewise approximation via regression modelling.

First, we generate $10,000$ data points from $\log(1+\exp(x))$ at equally spaced grid $x_i, i = 1, ..., 10000$ in $[-5, 5]$. One, two, three, four, and five breakpoints are considered, which correspond to two, three, four, five, and six pieces. The sum of squared error (SSE) is used to evaluate the performance of the fitted model on the generated data. Finally, the

optimal number of breakpoints is chosen at the knee of the plot of SSE versus the number of breakpoints. From Figure D.1 Right, the best number of breakpoints is three with an SSE of 3.3527 and an R^2 of 0.9999. A comparison of the fitted lines on the true curves with 2, 3, and 4 breakpoints is shown in Figure D.2. The optimal fitted model with three break points using the *segmented* method proposed by Muggeo (2003) (those three optimal breakpoints are -1.701, 0, and 1.702), $\hat{f}(x)$ is

$$\begin{aligned}\hat{f}(x) &= 0.1938 + 0.0426x + 0.2626(x - (-1.701))_+ \\ &\quad + 0.3898(x - 0)_+ + 0.2624(x - 1.702)_+, \quad x \in [-5, 5],\end{aligned}$$

where $(x - a)_+ := \max(x - a, 0)$ for any $a \in (-\infty, \infty)$.

Therefore, we can approximate $\log(1 + \exp(x))$ in $(-\infty, \infty)$ by

$$\hat{f}(x) = \widehat{\log(1 + \exp(x))} = \begin{cases} 0 & \text{if } x \leq -5, \\ 0.1938 + 0.0426x & \text{if } -5 < x \leq -1.701, \\ 0.6405 + 0.3052x & \text{if } -1.701 < x \leq 0, \\ 0.6405 + 0.6950x & \text{if } 0 < x \leq 1.702, \\ 0.1939 + 0.9574x & \text{if } 1.702 < x \leq 5, \\ x & \text{if } 5 < x. \end{cases} \tag{D.1}$$

We ignore the two minor jumps at $x = -5$ and $x = 5$ since we focus on the approximation of the function, and manually changing the structure of the piecewise approximations will affect the optimum of the approximation.

We construct the quadratic piecewise approximation, Equation (D.2), based on the linear piecewise approximation. We also ignore the discontinuity (minor jump) at each breakpoint. The SSE using quadratic piecewise approximation is 0.1188, and the R^2 of the fitted

models is 1.

$$\hat{f}(x) = \log(1 + \widehat{\exp}(x)) = \begin{cases} 0 & \text{if } x \leq -5, \\ 0.3893 + 0.1696x + 0.0189x^2 & \text{if } -5 < x \leq -1.7, \\ 0.6962 + 0.5000x + 0.1138x^2 & \text{if } -1.7 < x \leq 1.7, \\ 0.3894 + 0.8303x + 0.0190x^2 & \text{if } 1.7 < x \leq 5, \\ x & \text{if } 5 < x. \end{cases} \tag{D.2}$$

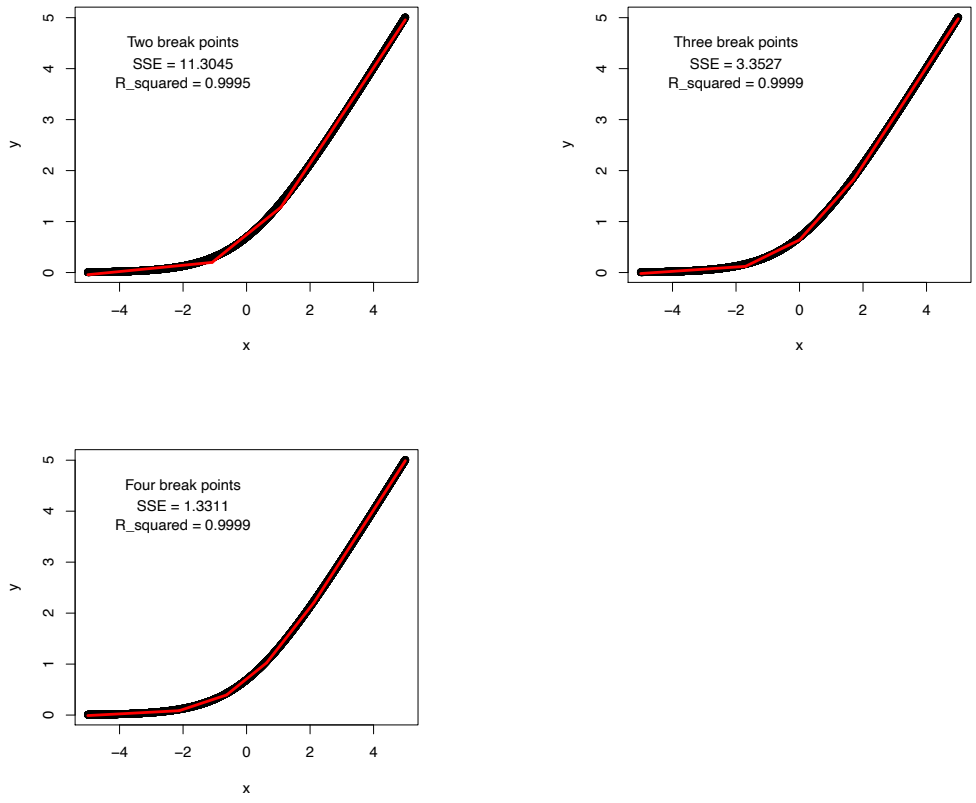


Figure D.2: A comparison of the fitted lines on the true curves using 2, 3, and 4 break points with sum of squared errors (SSE) and R squared added to the plots.

E Chapter 4: Update equations and ELBO calculation

In this appendix, we derive the update equation for each component and the ELBO clacu-lation in the shared frailty log-logistic AFT model.

E.1 VB update equations

(1) Update equation for $q^*(\boldsymbol{\beta})$

$$\log q^*(\boldsymbol{\beta}) \overset{+}{\approx} \mathbb{E}_{\_\boldsymbol{\beta}}[\log p(\mathbf{D}\,|\,\boldsymbol{\beta},\boldsymbol{\gamma},b)] + \mathbb{E}_{\_\boldsymbol{\beta}}[\log p(\boldsymbol{\beta})],$$

where

$$\begin{aligned} & \mathbb{E}_{\_\boldsymbol{\beta}}[\log p(\mathbf{D}\,|\,\boldsymbol{\beta},\boldsymbol{\gamma},b)] \\ = \; & \mathbb{E}_{\_\boldsymbol{\beta}}\Bigg[-\delta \log b + \sum_{i=1}^{K}\sum_{j=1}^{n_i}\Bigg\{\delta_{ij}\frac{y_{ij}-\mathbf{X}_{ij}^T\boldsymbol{\beta}-\gamma_i}{b} - (1+\delta_{ij}) \\ & \log\Big\{1+\exp\big(\frac{y_{ij}-\mathbf{X}_{ij}^T\boldsymbol{\beta}-\gamma_i}{b}\big)\Big\}\Bigg\}\Bigg], \end{aligned} \tag{E.1}$$

with $\delta = \sum_{i=1}^{K}\sum_{j=1}^{n_i}\delta_{ij}$ being the number of observed survival times, and

$$\begin{aligned} \mathbb{E}_{\_\boldsymbol{\beta}}[\log p(\boldsymbol{\beta})] \; & \overset{+}{\approx} \; \frac{p}{2}\log v_0 - \frac{1}{2}v_0(\boldsymbol{\beta}-\boldsymbol{\mu}_0)^T(\boldsymbol{\beta}-\boldsymbol{\mu}_0) \\ & \overset{+}{\approx} \; -\frac{1}{2}v_0\left[\boldsymbol{\beta}^T\boldsymbol{\beta} - 2\boldsymbol{\mu}_0^T\boldsymbol{\beta}\right] = v_0\,\boldsymbol{\mu}_0^T\boldsymbol{\beta} - \frac{1}{2}v_0\boldsymbol{\beta}^T\boldsymbol{\beta}. \end{aligned}$$

In Equation (E.1), we use the quadratic piece-wise approximation proposed by Xian et al. (2024) to approximate $1+\exp\big(\frac{y_{ij}-\mathbf{X}_{ij}^T\boldsymbol{\beta}-\gamma_i}{b}\big)$:

$$1+\exp\big(\frac{y_{ij}-\mathbf{X}_{ij}^T\boldsymbol{\beta}-\gamma_i}{b}\big) \approx \rho_{ij}\frac{y_{ij}-\mathbf{X}_{ij}^T\boldsymbol{\beta}-\gamma_i}{b} + \zeta_{ij}\big(\frac{y_{ij}-\mathbf{X}_{ij}^T\boldsymbol{\beta}-\gamma_i}{b}\big)^2, \tag{E.2}$$

where $\rho_{ij} := 0^{\nu_{ij1}} \times 0.1696^{\nu_{ij2}} \times 0.5^{\nu_{ij3}} \times 0.8303^{\nu_{ij4}} \times 1^{1-\sum_{k=1}^{4}\nu_{ijk}}$ and $\zeta_{ij} := 0^{\nu_{ij1}} \times 0.0189^{\nu_{ij2}} \times 0.1138^{\nu_{ij3}} \times 0.0190^{\nu_{ij4}} \times 0^{1-\sum_{k=1}^{4}\nu_{ijk}}$ with $\nu_{ij1} = \mathbb{1}(\frac{y_{ij}-\mathbf{x}_{ij}^T\boldsymbol{\beta}-\gamma_i}{b} \leq -5)$, $\nu_{ij2} = \mathbb{1}(-5 < \frac{y_{ij}-\mathbf{x}_{ij}^T\boldsymbol{\beta}-\gamma_i}{b} \leq -1.7)$, $\nu_{ij3} = \mathbb{1}(-1.7 < \frac{y_{ij}-\mathbf{x}_{ij}^T\boldsymbol{\beta}-\gamma_i}{b} \leq 1.7)$, and $\nu_{ij4} = \mathbb{1}(1.7 < \frac{y_{ij}-\mathbf{x}_{ij}^T\boldsymbol{\beta}-\gamma_i}{b} \leq 5)$. Then, we obtain,

$$\begin{aligned}\log q^*(\boldsymbol{\beta}) \overset{+}{\approx} & \left[v_0\boldsymbol{\mu}_0^T + \Big(\sum_{i=1}^{K}\sum_{j=1}^{n_i}\Big(\mathbb{E}_{q(b)}(\frac{1}{b})\Big(-\delta_{ij} + (1+\delta_{ij})\rho_{ij}\Big)\right. \\ & \left. +2\mathbb{E}_{q(b)}(\frac{1}{b^2})(1+\delta_{ij})\zeta_{ij}(y_{ij} - \mathbb{E}_{q(\gamma_i)}(\gamma_i))\Big)\mathbf{X}_{ij}^T\Big)\right]\boldsymbol{\beta} \\ & -\frac{1}{2}\boldsymbol{\beta}^T\Big(v_0\mathbf{I} + 2\mathbb{E}_{q(b)}(\frac{1}{b^2})\sum_{i=1}^{K}\sum_{j=1}^{n_i}(1+\delta_{ij})\zeta_{ij}\mathbf{X}_{ij}\mathbf{X}_{ij}^T\Big)\boldsymbol{\beta}.\end{aligned}$$

Let

$$\Sigma^* := \left[v_0\mathbf{I} + 2\mathbb{E}_{q(b)}(\frac{1}{b^2})\sum_{i=1}^{K}\sum_{j=1}^{n_i}(1+\delta_{ij})\zeta_{ij}\mathbf{X}_{ij}\mathbf{X}_{ij}^T\right]^{-1},$$

and

$$\begin{aligned}\boldsymbol{\mu}^* := & \left[\Big\{v_0\boldsymbol{\mu}_0^T + \sum_{i=1}^{K}\sum_{j=1}^{n_i}\Big(\mathbb{E}_{q(b)}(\frac{1}{b})\Big(-\delta_{ij} + (1+\delta_{ij})\rho_{ij}\Big)\right. \\ & \left. +2\mathbb{E}_{q(b)}(\frac{1}{b^2})(1+\delta_{ij})\zeta_{ij}(y_{ij} - \mathbb{E}_{q(\gamma_i)}(\gamma_i))\Big)\mathbf{X}_{ij}^T\Big\}\Sigma^*\right]^T.\end{aligned}$$

Then, $q^*(\boldsymbol{\beta})$ is $N_p(\boldsymbol{\mu}^*, \Sigma^*)$.

(2) Update equation for $q^*(\gamma_i)$

$$\log q^*(\gamma_i) \overset{+}{\approx} \mathbb{E}_{-\gamma_i}\Big[\log p(\mathbf{D}\,|\,\boldsymbol{\beta}, \boldsymbol{\gamma}, b)\Big] + \mathbb{E}_{-\gamma_i}\Big[\sum_{i=1}^{K}\log p(\gamma_i\,|\sigma_\gamma^2)\Big],$$

where

$$\begin{aligned}
&\mathbb{E}_{-\gamma_i}\Big[\log p(\mathbf{D}\,|\boldsymbol{\beta},\boldsymbol{\gamma},b)\Big] \\
\overset{+}{\approx}\ &\mathbb{E}_{-\gamma_i}\Bigg[\sum_{j=1}^{n_i}\Big\{\delta_{ij}\frac{y_{ij}-\mathbf{X}_{ij}^T\boldsymbol{\beta}-\gamma_i}{b}-(1+\delta_{ij})\log\Big(1+\exp\big(\frac{y_{ij}-\mathbf{X}_{ij}^T\boldsymbol{\beta}-\gamma_i}{b}\big)\Big)\Big\}\Bigg], \quad \text{(E.3)}
\end{aligned}$$

and

$$\mathbb{E}_{-\gamma_i}\Big[\sum_{i=1}^{K}\log p(\gamma_i\,|\sigma_\gamma^2)\Big]\overset{+}{\approx}-\frac{1}{2}\gamma_i^2\mathbb{E}_{q(\sigma_\gamma^2)}(\frac{1}{\sigma_\gamma^2}).$$

We again apply the quadratic approximation in (E.2) to (E.3), and obtain

$$\begin{aligned}
\log q^*(\gamma_i)\ \overset{+}{\approx}\ &\gamma_i\Bigg(\sum_{j=1}^{n_i}\Big[\mathbb{E}_{q(b)}(\frac{1}{b})(-\delta_{ij}+(1+\delta_{ij})\rho_{ij})+2\mathbb{E}_{q(b)}(\frac{1}{b^2})(1+\delta_{ij})\zeta_{ij}(y_{ij}-\mathbf{X}_{ij}^T\mathbb{E}_{q(\boldsymbol{\beta})}\boldsymbol{\beta})\Big]\Bigg) \\
&-\frac{1}{2}\gamma_i^2\Big(\mathbb{E}_{q(\sigma_\gamma^2)}(\frac{1}{\sigma_\gamma^2})+2\mathbb{E}_{q(b)}(\frac{1}{b^2})\sum_{j=1}^{n_i}(1+\delta_{ij})\zeta_{ij}\Big).
\end{aligned}$$

Let

$$\sigma_i^{2*}=\big[\mathbb{E}_{q(\sigma_\gamma^2)}(\frac{1}{\sigma_\gamma^2})+2\mathbb{E}_{q(b)}(\frac{1}{b^2})\sum_{j=1}^{n_i}(1+\delta_{ij})\zeta_{ij}\big]^{-1},$$

and

$$\tau_i^*=\sigma_i^{2*}\sum_{j=1}^{n_i}\Big[\mathbb{E}_{q(b)}(\frac{1}{b})(-\delta_{ij}+(1+\delta_{ij})\rho_{ij})+2\mathbb{E}_{q(b)}(\frac{1}{b^2})(1+\delta_{ij})\zeta_{ij}(y_{ij}-\mathbf{X}_{ij}^T\mathbb{E}_{q(\boldsymbol{\beta})}\boldsymbol{\beta})\Big].$$

Then, $q^*(\gamma_i)$ is $N_l(\tau_i^*,\sigma_i^{2*})$.

(3) Update equation for $q^*(b)$

$$\log q^*(b)\overset{+}{\approx}\mathbb{E}_{-b}[\log p(\mathbf{D}\,|\boldsymbol{\beta},\boldsymbol{\gamma},b)]+\mathbb{E}_{-b}[\log p(b)],$$

where

$$\begin{aligned} & \mathbb{E}_{-b}[\log p(\mathbf{D} \mid \boldsymbol{\beta}, \boldsymbol{\gamma}, b)] \\ = \; & \mathbb{E}_{-b}\Bigg[- \delta \log b + \sum_{i=1}^{K} \sum_{j=1}^{n_i} \Bigg\{ \delta_{ij} \frac{y_{ij} - \mathbf{X}_{ij}^T \boldsymbol{\beta} - \gamma_i}{b} - (1 + \delta_{ij}) \\ & \log \Big\{ 1 + \exp\big(\frac{y_{ij} - \mathbf{X}_{ij}^T \boldsymbol{\beta} - \gamma_i}{b}\big) \Big\} \Bigg\} \Bigg], \end{aligned} \tag{E.4}$$

and $\mathbb{E}_{-b}[\log p(b)] \overset{+}{\approx} -(\alpha_0+1)\log b - \omega_0/b$ since we assume the prior, $b \sim$ Inverse-Gamma (α_0, ω_0). In Equation (E.4), we apply the linear piece-wise approximation proposed by Xian et al. (2024) to approximate $1 + \exp\big(\frac{y_{ij} - \mathbf{X}_{ij}^T \boldsymbol{\beta} - \gamma_i}{b}\big)$ to achieve conjugacy:

$$1 + \exp\big(\frac{y_{ij} - \mathbf{X}_{ij}^T \boldsymbol{\beta} - \gamma_i}{b}\big) \approx \varphi_{ij} \frac{y_{ij} - \mathbf{X}_{ij}^T \boldsymbol{\beta} - \gamma_i}{b},$$

where $\varphi_{ij} = 0^{\eta_{ij1}} \times 0.0426^{\eta_{ij2}} \times 0.3052^{\eta_{ij3}} \times 0.6950^{\eta_{ij4}} \times 0.9574^{\eta_{ij5}} \times 1^{1-\sum_{k=1}^{5} \eta_{ijk}}$ with $\eta_{ij1} = \mathbb{1}(\frac{y_{ij} - \mathbf{X}_{ij}^T \boldsymbol{\beta} - \gamma_i}{b} \leq -5)$, $\eta_{ij2} = \mathbb{1}(-5 < \frac{y_{ij} - \mathbf{X}_{ij}^T \boldsymbol{\beta} - \gamma_i}{b} \leq -1.701)$, $\eta_{ij3} = \mathbb{1}(-1.701 < \frac{y_{ij} - \mathbf{X}_{ij}^T \boldsymbol{\beta} - \gamma_i}{b} \leq 0)$, $\eta_{ij4} = \mathbb{1}(0 < \frac{y_{ij} - \mathbf{X}_{ij}^T \boldsymbol{\beta} - \gamma_i}{b} \leq 1.702)$, and $\eta_{ij5} = \mathbb{1}(1.702 < \frac{y_{ij} - \mathbf{X}_{ij}^T \boldsymbol{\beta} - \gamma_i}{b} \leq 5)$.

Then, we obtain

$$\log q^*(b) \overset{+}{\approx} -(\alpha_0 + \delta + 1)\log b - \frac{1}{b}\Big[\omega_0 - \sum_{i=1}^{K} \sum_{j=1}^{n_i} \Big(\delta_{ij} - (1 + \delta_{ij})\varphi_{ij}\Big)\Big(y_{ij} - \mathbf{X}_{ij}^T \mathbb{E}_{q(\boldsymbol{\beta})}(\boldsymbol{\beta}) - \mathbb{E}_{q(\gamma_i)}\gamma_i\Big)\Big].$$

Let $\alpha^* = \alpha_0 + \delta$ and

$$\omega^* = \omega_0 - \sum_{i=1}^{K} \sum_{j=1}^{n_i} \Big(\delta_{ij} - (1 + \delta_{ij})\varphi_{ij}\Big)\Big(y_{ij} - \mathbf{X}_{ij}^T \mathbb{E}_{q(\boldsymbol{\beta})}(\boldsymbol{\beta}) - \mathbb{E}_{q(\gamma_i)}\gamma_i\Big),$$

we have $q^*(b)$ is Inverse-Gamma (α^*, ω^*).

(4) Update equation for $q^*(\sigma_\gamma^2)$

$$\log q^*(\sigma_\gamma^2) \overset{+}{\approx} \mathbb{E}_{-\sigma_\gamma^2}\Big[\sum_{i=1}^{K} \log p(\gamma_i\,|\sigma_\gamma^2)\Big] + \mathbb{E}_{-\sigma_\gamma^2}\Big[\log p(\sigma_\gamma^2)\Big],$$

where

$$\begin{aligned}\mathbb{E}_{-\sigma_\gamma^2}\Big[\sum_{i=1}^{K} \log p(\gamma_i\,|\sigma_\gamma^2)\Big] &= \mathbb{E}_{-\sigma_\gamma^2}\Big[\sum_{i=1}^{K} \log\Big(\frac{1}{\sqrt{2\pi\sigma_\gamma^2}}\exp(-\frac{1}{2\sigma_\gamma^2}\gamma_i^2)\Big)\Big] \\ &\overset{+}{\approx} -\frac{K}{2}\log\sigma_\gamma^2 - \frac{1}{2\sigma_\gamma^2}\sum_{i=1}^{K}\mathbb{E}_{q(\gamma_i)}\gamma_i^2,\end{aligned}$$

and $\mathbb{E}_{-\sigma_\gamma^2}[\log p(\sigma_\gamma^2)] = -(\lambda_0+1)\log\sigma_\gamma^2 - \eta_0/\sigma_\gamma^2$.

Therefore,

$$\log q^*(\sigma_\gamma^2) \overset{+}{\approx} -(\lambda_0 + \frac{K}{2} + 1)\log\sigma_\gamma^2 - [\eta_0 + \frac{1}{2}\sum_{i=1}^{K}\mathbb{E}_{q(\gamma_i)}\gamma_i^2]/\sigma_\gamma^2.$$

Let $\lambda^* = \lambda_0 + K/2$ and

$$\eta^* = \eta_0 + \frac{1}{2}\sum_{i=1}^{K}\mathbb{E}_{q(\gamma_i)}\gamma_i^2,$$

$q^*(\sigma_\gamma^2)$ is an Inverse-Gamma (λ^*, η^*).

E.2 ELBO calculation

We now present details of calculating $diff_{\boldsymbol{\beta}} + diff_{\boldsymbol{\gamma}} + diff_b + diff_{\sigma^2_\gamma}$ in the ELBO defined in (4.5).

$$\begin{aligned} diff_{\boldsymbol{\beta}} &= \mathbb{E}_q[\log p(\boldsymbol{\beta})] - \mathbb{E}_q[\log q(\boldsymbol{\beta})] \\ &\overset{+}{\approx} \mathbb{E}_q[-\frac{1}{2}v_0(\boldsymbol{\beta}-\boldsymbol{\mu}_0)^T(\boldsymbol{\beta}-\boldsymbol{\mu}_0)] - \mathbb{E}_q[-\frac{1}{2}\log(|\Sigma^*|) - \frac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\mu}^*)^T(\Sigma^*)^{-1}(\boldsymbol{\beta}-\boldsymbol{\mu}^*)] \\ &\overset{+}{\approx} -\frac{1}{2}v_0[\text{trace}(\Sigma^*) + (\boldsymbol{\mu}^*-\boldsymbol{\mu}_0)^T(\boldsymbol{\mu}^*-\boldsymbol{\mu}_0)] + \frac{1}{2}\log(|\Sigma^*|). \end{aligned}$$

Note that

$$\mathbb{E}_q[\frac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\mu}^*)^T(\Sigma^*)^{-1}(\boldsymbol{\beta}-\boldsymbol{\mu}^*)] = \frac{p}{2},$$

which is always a constant at each iteration and therefore we ignore it.

$$\begin{aligned} diff_{\boldsymbol{\gamma}} &= \mathbb{E}_q[\sum_{i=1}^K \log p(\gamma_i|\sigma^2_\gamma)] - \mathbb{E}_q[\sum_{i=1}^K \log q(\gamma_i)] \\ &\overset{+}{\approx} -\frac{K}{2}\mathbb{E}_{q(\sigma^2_\gamma)}(\log\sigma^2_\gamma) - \frac{1}{2}\mathbb{E}_{q(\sigma^2_\gamma)}(\frac{1}{\sigma^2_\gamma})\sum_{i=1}^K \mathbb{E}_{q(\gamma_i)}(\gamma_i^2) \\ &\quad -\frac{1}{2}\sum_{i=1}^K [\log\sigma^{2*}_i - \frac{1}{\sigma^{2*}_i}(\mathbb{E}_{q(\gamma_i)}(\gamma_i^2) - \mu_i^{*2})] \\ &\overset{+}{\approx} -\frac{K}{2}\mathbb{E}_{q(\sigma^2_\gamma)}(\log\sigma^2_\gamma) - \frac{1}{2}\mathbb{E}_{q(\sigma^2_\gamma)}(\frac{1}{\sigma^2_\gamma})\sum_{i=1}^K \mathbb{E}_{q(\gamma_i)}(\gamma_i^2) - \frac{1}{2}\sum_{i=1}^K(\log\sigma^{2*}_i). \end{aligned}$$

Since $(\mathbb{E}_{q(\gamma_i)}(\gamma_i^2) - \mu_i^{*2})/\sigma^{2*}_i = 1$ for $i = 1, ...K$, and therefore,

$$\Big(\sum_{i=1}^K (\mathbb{E}_{q(\gamma_i)}(\gamma_i^2) - \mu_i^{*2})\Big)/(2\sigma^{2*}_i) = K/2,$$

which is a constant, and we can ignore it.

$$
\begin{aligned}
diff_b &= \mathbb{E}_q[\log p(b)] - \mathbb{E}_q[\log q(b)] \\
&\overset{+}{\approx} \mathbb{E}_q\Big[-(\alpha_0 + 1)\log b - \frac{\omega_0}{b}\Big] \\
&\quad -\mathbb{E}_q\Big[\alpha^* \log \omega^* - \log(\Gamma(\alpha^*)) - (\alpha^* + 1)\log b - \frac{\omega^*}{b}\Big] \\
&= (\alpha^* - \alpha_0)\mathbb{E}_{q(b)}(\log b) + (\omega^* - \omega_0)\mathbb{E}_{q(b)}(\frac{1}{b}) - \alpha^* \log \omega^*.
\end{aligned}
$$

Since α^* does not change at each iteration, we remove $\log(\Gamma(\alpha^*))$ in the calculation of the ELBO.

$$
\begin{aligned}
diff_{\sigma^2_\gamma} &= \mathbb{E}_q[\log p(\sigma^2_\gamma)] - \mathbb{E}_q[\log q(\sigma^2_\gamma)] \\
&\overset{+}{\approx} \mathbb{E}_q\Big[-(\lambda_0 + 1)\log \sigma^2_\gamma - \frac{\eta_0}{\sigma^2_\gamma}\Big] \\
&\quad -\mathbb{E}_q\Big[\lambda^* \log \eta^* - \log(\Gamma(\lambda^*)) - (\lambda^* + 1)\log \sigma^2_\gamma - \frac{\eta^*}{\sigma^2_\gamma}\Big] \\
&= (\lambda^* - \lambda_0)\mathbb{E}_{q(\sigma^2_\gamma)}(\log \sigma^2_\gamma) + (\eta^* - \eta_0)\mathbb{E}_{q(\sigma^2_\gamma)}(\frac{1}{\sigma^2_\gamma}) - \lambda^* \log \eta^*.
\end{aligned}
$$

Since λ^* does not change at each iteration, we remove $\log(\Gamma(\lambda^*))$ in the calculation of the ELBO.

Curriculum Vitae

Name:	Chengqian Xian
Post-Secondary Education and Degrees:	University of Western Ontario London, Ontario, Canada 2020 - 2024 Ph.D. in Statistics University of Western Ontario London, Ontario, Canada 2019 - 2020 M.Sc. in Statistics University of Western Ontario London, Ontario, Canada 2018 - 2019 Visiting student in Statistics South China University of Technology Guangzhou, Guangdong, China 2015 - 2019 B.Sc. in Mathematics and Applied Mathematics 2015 - 2019 B.A. in English
Honours and Awards:	Western Graduate Research Scholarship, 2019 - 2024 Student travel award at 2024 ISBA World Meeting, International Society for Bayesian Analysis, Venice, Italy
Related Work Experience:	Teaching Assistant and Research Assistant University of Western Ontario, 2019 - 2024 Data Consultant and R Workshop Instructor Western Data Science Solutions, 2020 - 2023

Publications:

[1] **Chengqian Xian***, Camila P.E. de Souza, Felipe F. Rodrigues, Health Outcome Predictive Modelling in Intensive Care Units, *Operations Research for Health Care* (2023), doi: https://doi.org/10.1016/j.orhc.2023.100409

[2] **Chengqian Xian***, Camila P.E. de Souza, Wenqing He, Felipe F. Rodrigues, Renfang Tian, Variational Bayesian Analysis of Survival Data Using a Log-logistic Accelerated Failure Time Model, *Statistics and Computing* (2024) 34, 67. https://doi.org/10.1007/s11222-023-10365-6

[3] **Chengqian Xian***, Camila de Souza, John Jewell, Ronaldo Dias, Clustering Functional Data via Variational Inference, *Advances in Data Analysis and Classification* (2024). https://doi.org/10.1007/s11634-024-00590-w

[4] Huang, Jinbao, Marshall Shuai Yang, **Chengqian Xian**, James Joseph Noël, Yolanda Susanne Hedberg, Jian Chen, Ubong Eduok, Ivan Barker, Jeffrey Daniel Henderson, Haiping Zhang, and et al., Synergistic Effect of Nanoclay and Barium Sulfate Fillers on the Corrosion Resistance of Polyester Powder Coatings, *Coatings* (2023) 13, no. 10: 1680. https://doi.org/10.3390/coatings13101680

[5] Yang, Marshall Shuai, Jinbao Huang, Hui Zhang, James Joseph Noël, Yolanda Susanne Hedberg, Jian Chen, Ubong Eduok, Ivan Barker, Jeffrey Daniel Henderson, **Chengqian Xian**, and et al., Study on the Self-Repairing Effect of Nanoclay in Powder Coatings for Corrosion Protection, *Coatings* (2023), no. 7: 1220. https://doi.org/10.3390/coatings13071220

[6] Ryu Lien, Joyla A. Furlano, Suzanne T. Witt, **Chengqian Xian**, Lindsay S. Nagamatsu, The effects of a six-month exercise intervention on white matter microstructure in older adults at risk for diabetes, *Cerebral Circulation - Cognition and Behavior* (2024). https://doi.org/10.1016/j.cccb.2024.100369

* corresponding author

Research presentations:

[1] Chengqian Xian, Camila de Souza, John Jewell, Ronaldo Dias, Clustering Functional Data via Variational Inference, the Annual Meeting of the Statistical Society of Canada, June 7-11, 2021. (Oral presentation via Zoom)

[2] Chengqian Xian, Camila de Souza, John Jewell, Ronaldo Dias, Clustering Functional Data via Variational Inference, National Symposium on Probability and Statistics in Brazil, July 31 - August 5, 2022. (Oral presentation via Zoom)

[3] Chengqian Xian, Camila de Souza, John Jewell, Ronaldo Dias, Clustering Functional Data via Variational Inference, the 3rd Waterloo Student Conference of Statistics, Actuarial Science and Finance, October 14 - 15, 2022. (Oral presentation)

[4] Chengqian Xian, Camila de Souza, Wenqing He, Felipe Rodrigues, Renfang Tian, Variational Bayes Inference of Survival Data Using a Log-logistic AFT Model, the Annual Meeting of the Statistical Society of Canada, May 28 - 31, 2023. (Oral presentation)

[5] Chengqian Xian, Camila de Souza, Wenqing He, Felipe Rodrigues, Renfang Tian, Variational Bayesian Analysis of Survival Data Using a Log-logistic Accelerated Failure Time Model, the 2024 World Meeting of the International Society for Bayesian Analysis , July 1 -7, 2024. (Poster presentation)