Final Report

2025-03-11

# Glossary

AFT - accelerated failure time

CRAN - the Comprehensive R Archive Network

ELBO - evidence based lower bound

GEO - Gene Expression Omnibus

HMC - Hamiltonian Monte Carlo

h-likelihood - hierarchical likelihood

MCMC - Markov Chain Monte Carlo

MSE - mean squared error

VB - variational Bayes

VI - variational inference

# Structured Abstract

## Context & Motivation

The survregVB R package is developed to implement mean-field variational Bayes (VB) algorithms for right-censored log-logistic accelerated failure time (AFT) models with and without frailty as an alternative to Markov Chain Monte Carlo (MCMC) methods. survregVB uses variational inference (VI) to approximate the posterior distributions of model parameters through optimization.

## Research Question & Objectives

How can VB inference be implemented as an R package for parametric survival regression of right-censored log-logistic AFT models with and without frailty? The objectives are to develop **survregVB**, and validate its performance through simulations and application to real-world data.

## Principal Ideas

This research integrates concepts related to Bayesian survival analysis, including VI, AFT models, and shared frailty along with R package development aspects including design, validation, and usability.

## Research Methodology

The research aims to build a prototype system, survregVB, implementing two mean-field VB algorithms for right-censored log-logistic AFT models with and without frailty. Performance is assessed through simulations and real-world applications. survregVB also includes a test suite, example data sets, documentation and a vignette.

## Anticipated Results

The primary outcome is the survregVB package, providing mean-field VB methods for log-logistic AFT models. Performance metrics will be obtained from simulation studies and data applications.

## Anticipated Novelty

While R packages exist for AFT model estimation, no integrated software supports VB methods for AFT models (Xian et al., 2024b). Compared to more traditional Bayesian approaches, the survregVB package offers similar estimation results with much lower computation time.

## Anticipated Impact of Results

Since no existing software provides VB methods for AFT models, the development of survregVB will fill this gap. Using survregVB to evaluate the VB algorithm will address its strengths and limitations and identify areas for future research.

# 1. Introduction

## 1.1 General Background & Problem Description

# 2. Background & Related Work

## 2.1 Survival Analysis

Survival analysis is the branch of statistics concerned with modelling time-to-event data, where the focus is on analyzing survival time, the length of time between an origin and an event of interest. The definition of an event (i.e. death, sickness, accident, bankruptcy, etc.) is highly variable, making survival analysis applicable across many domains, such as medicine, sociology, marketing or economics (Emmert-Streib & Dehmer, 2019). In survival analysis, censored observations occur when the exact survival time is unknown for some individuals as the event of interest does not occur during the duration of a study or data collection. In specific, right censoring occurs when the time taken for event of interest to occur exceeds the observed length of time for an individual (Kobara, 2022).

## 2.2 AFT Models

For situations where explanatory variables may affect survival time, usual survival distribution methods are insufficient. Since survival times are only positive (and therefore rarely normally distributed), and may contain censored observations, survival regression methods are necessary for analyzing survival data for non-homogeneous populations (Kobara, 2022). A commonly used survival regression model is the accelerated failure time (AFT) model, which assumes an accelerative effect of the covariates directly on survival time (Webber et al., 2022). There are several possible distributions for the AFT model, including exponential, Weibull, log-logistic, log-normal or gamma. In particular, the log-logistic distribution is suitable for modelling a wide variety of survival data (Rivas-López et al., 2022). See Appendix A for the log-logistic AFT model specification.

## 2.3 Shared Frailty

Frailty is a multiplicative, latent effect on the baseline hazard function used to account for heterogeneity and random effects. A shared frailty model is a random effects model where the frailties are shared among groups and randomly distributed across groups (Gutierrez, 2002). In cases where correlated survival data arises from clusters of individuals with shared environmental factors, a shared frailty AFT model can be used to account for correlations among survival data (Gorfine & Zucker, 2023; Hanagal, 2011; Hougaard, 1995). Since the choice of distribution is not critical, the log-logistic AFT model can be used (Lambert et al., 2004). See Appendix A for the shared frailty log-logistic AFT model specification.

## 2.4 Bayesian Inference

Bayesian inference is technique used to derive the posterior distribution of parameters based on Bayes’ theorem. For Bayesian models with many parameters, it is challenging to calculate the exact posterior distribution. Therefore, Markov Chain Monte Carlo (MCMC) algorithms are typically used to approximate the posterior (Geman & Geman, 1984). In specific, Bayesian inference MCMC techniques are used for approximating posterior parameters for log-logistic AFT models via numerical approximation by sampling (Wainwright & Jordan, 2007). However, due to high computational costs, research has been done to explore alternative methods such as variational inference (VI) developed from machine learning (Jordan et al., 1999). VI uses optimization to approximate the parameters of a Bayesian model, providing similar estimations to MCMC techniques at a much lower computational cost (Blei et al., 2017). Furthermore, VI can make use of prior results from similar estimation studies to make predictions, and does not rely on asymptotic, making it useful for studies with small sample sizes (Ibrahim et al., 2001).

## 2.5 The Variational Bayes Algorithm

Mean-field variational Bayes (VB) is a special case of mean-field VI that arises from minimizing Kullback-Leibler (KL) divergence. KL divergence is used to measure the dissimilarity between the approximated and exact posterior densities (Bishop, 2006). Minimizing KL divergence can also be thought of as maximizing the evidence lower-bound (ELBO) (Blei et al., 2017; Jordan et al., 1999). An approximated posterior distribution can be obtained by using the coordinate ascent algorithm under VB to optimize the ELBO, also referred to as coordinate ascent variational inference (CAVI) (Bishop, 2006; Blei et al., 2017). For Bayesian models with several parameters, it is difficult to obtain a closed form of the posterior distribution, so the exact posterior distribution is intractable. In these cases, a piecewise approximation technique is embedded into the VB algorithm to achieve Bayesian conjugacy (Xian et al., 2024b).

## 2.6 Analysis & Research Gap

Several R packages currently exist to provide estimation methods for AFT models, including:

* survreg from survival can be used to fit a likelihood-based parametric survival regression model with a log-logistic distribution (Therneau, 2024).
* rstan provides the MCMC-based Hamiltonian Monte Carlo (HMC) sampling algorithm for survival models (2024b).
* survregBayes from spBayesSurv uses MCMC-based Bayesian inference to estimate shared frailty AFT models (Zhou et al., 2020).
* frailtyHL implements the hierarchical likelihood (h-likelihood) approach for shared frailty AFT models (Ha et al., 2012).

However, VB methods for AFT models are a relatively recent development, and currently no integrated software exists for their practical implementation (Xian et al., 2024b). Compared to more traditional Bayesian approaches, the survregVB package offers the following advantages:

* Significantly reduced computation cost with similar estimation results compared to MCMC, with an average speedup of up to 300 times (Xian et al., 2024b).
* Can outperform likelihood-based methods in terms of empirical mean squared error (MSE) (Xian et al., 2024b).
* Its ability to make use of information from previous studies to make predictions (Ibrahim et al., 2001).
* VI does not rely on asymptotic, making it useful for studies with small sample sizes (Ibrahim et al., 2001).

# 3. Research Objectives

The study aims to:

O1. Develop an R package, survregVB, that implements mean-field VB algorithms for parametric survival regression in log-logistic AFT models.

O2. Provide an accessible and user-friendly interface, ensuring ease of use for survival analysis.

O3. Extend VB inference to models with and without shared frailty to enable analysis of clustered survival data.

O4. Create a comprehensive test suite to ensure the correctness of the package.

O5. Develop thorough function documentation to improve usability and understanding for users.

O6. Create a vignette to provide examples and guidance for users through key functionalities.

O7. Validate the performance of survregVB using simulation studies to assess accuracy and efficiency.

O8. Evaluate survregVB on publicly available survival data sets to compare its results against traditional MCMC-based methods.

O9. Submit survregVB to the Comprehensive R Archive Network (CRAN) to improve accessibility and ensure that it meets R package development standards.

# 4. Methodology

## 4.1 Development Approaches

The research aims to build a prototype system, the survregVB R package, which implements two mean-field VB algorithms for right-censored log-logistic AFT models with and without shared frailty respectively. The performance of survregVB will be assessed through:

* Simulation studies – Generating synthetic survival data sets to evaluate estimation accuracy and computational cost.
* Analysis of publicly available data sets – Applying survregVB to real-world survival data and comparing results with traditional parametric methods.

To ensure usability, accuracy, and compliance with R package development standards, survregVB will include:

* A test suite for verifying function correctness.
* Function documentation to guide users.
* S3 print and summary methods to display results of the fitted model.
* A vignette to provide practical examples and usage instructions.
* Submission to CRAN to improve accessibility and ensure adherence to R package standards.

## 4.2 Tools & Techniques

* The survival Package provides the standard functions for survival modeling in R, ensuring that survregVB is compatible with existing workflows (Therneau, 2024).
* Stan allows MCMC-based validation of VB results, ensuring model accuracy (2024b).
* Package development tools like devtools, usethis, and testthat, are used to maintain code quality in survregVB (Silge et al., 2019).

## 4.3 Algorithms

survregVB implements two mean-field variational Bayes (VB) algorithms for right-censored log-logistic AFT models with and without frailty (Algorithms 1 and 2 respectively in Appendix B) (Xian et al., 2024b, 2024a).

# 5. Results

## 5.1 Contextual Diagram

## 5.2 Technical Work

### 5.2.1 Key Requirements Met

* O1: Created the survregVB package to implement the mean-field VB algorithm in R for parametric survival regression in log-logistic AFT models via the survregVB() function.
* O2: Designed a user-friendly interface similar to survreg for ease of adoption.
* O3: Created the survregVB() function that can be called with or without frailty to fit both standard and shared frailty log-logistic AFT models for clustered survival data.
* O4: Included a test suite with unit tests to cover core functionalities.
* O5: Documented all functions with clear explanations of inputs, outputs, and expected usage, with provided examples.
* O6: Created a vignette with examples for models with and without shared frailty.
* O7: Conducted simulation studies to validate the accuracy and efficiency of survregVB.
* O8: Evaluated the package against publicly available survival data sets and compared results with traditional MCMC methods.
* O9: Prepared survregVB for CRAN submission, ensuring compliance with R package standards.

### 5.2.2 System Design & Architecture

survregVB was based off existing R code from the vbaft repository (<https://github.com/chengqianxian/vbaft/tree/main>), which implements survival regression for right-censored log-logistic AFT models without shared frailty. However, significant modifications have been made to add support for shared frailty, meet CRAN submission standards, and improve usability, performance and maintainability.

#### i) Architectural/Design Patterns Used

The `survregVB` follows the standard R package format (Silge et al., 2019) and includes a test suite, a vignette, function documentation, and simulated and real-world data sets for use in examples and testing. Adherence to R package development best practices ensures usability and maintainability. Refer to Appendix C for the package structure.

#### ii) Component Interfaces

1. survregVB() is the primary user-facing function for performing VB survival regression in an AFT model.

survregVB <- function(  
 formula, # Model formula (e.g., Surv(time, status) ~ predictors)  
 data, # Data frame to interpret the variables.   
 alpha\_0, # Shape hyperparameter of b   
 omega\_0, # Scale hyperparameter of b  
 mu\_0, # Mean hyperparameter of beta  
 v\_0, # Precision hyperparameter of beta  
 lambda\_0, # Shape hyperparameter for frailty component  
 eta\_0, # Scale hyperparameter for frailty component   
 na.action = na.omit, # Handling of missing values  
 cluster, # Optional clustering variable for frailty models  
 max\_iteration = 100, # Maximum number of iterations for the VB algorithm  
 threshold = 0.0001 # Convergence threshold for the ELBO  
)

Its main role is to:

* Parse the formula input (Surv(time, status) ~ predictors).
* Check for missing data, outliers, and non-supported distributions.
* Determine whether to use a frailty model (if cluster is specified).
* Call either survregVB.fit() or survregVB.frailty.fit() based on model type.
* Return a survregVB object with results and metadata.

1. Core fitting functions

* survregVB.fit() implements VB estimation for standard log-logistic AFT models without frailty.
* survregVB.frailty.fit() implements VB estimation for standard log-logistic AFT models with shared frailty when the cluster variable is specified.

1. S3 methods for interpreting/presenting results

* print.survregVB() displays the posterior distributions of the AFT model parameters, and is called automatically when displaying results from survregVB()
* summary.survregVB() provides a detailed model summary including standard deviations and credible intervals for the posterior distributions. It is called automatically by running summary().

#### iii) Quality Attributes

* Performance: survregVB employs VB instead of MCMC for faster convergence and similar estimation results.
* Usability: Syntax mirrors survreg() from the surival package, making adoption easier for users already familiar with survival analysis in R (Therneau, 2024). Clear documentation and vignettes explain how to use survregVB effectively.
* Reliability: Comprehensive unit tests validate correctness through an automated test suite.
* Interoperability: Designed to work with standard R survival analysis workflows (supports **Surv()** notation from the survival package, and is compatible with standard R data frames and tibble formats) (Therneau, 2024).
* Security and Privacy: Since all computations occur in-memory within R, there are no external API calls or data transfers.

### 5.2.3 System Implementation & Testing

#### Algorithms & Techniques

The following tools and technologies were used in developing survregVB:

1. Programming Language

* R: The standard language for statistical computing and survival analysis. It provides built-in methods for data manipulation, model fitting, and visualization (Team, n.d.).

1. Libraries and Packages

Survival analysis:

* survival - Core library for survival analysis routines, including definition of Surv objects and parametric AFT models (Therneau, 2024).
* rstan - R interface for stan, used for comparisons against MCMC techniques (2024b).

Package and Development:

* devtools, usethis - For general package development, including functions for automating R package creation and setup, testing and documenting functions, and building the package (Wickham et al., 2022; Wickham, Bryan, et al., 2024).
* testthat - For writing function unit tests to ensure the package works as expected (Wickham, 2011).
* roxygen2 - To generate in-line function documentation in a standardized format (Wickham, Danenberg, et al., 2024).
* knitr, rmarkdown - To create vignettes that provide explanations and examples for the package functions (Allaire et al., 2024; Xie, 2024).
* styler, lintr - To check R code for style and formatting issues to ensure clean and readable code throughout the package (Hester et al., 2025; Müller & Walthert, 2024).
* covr - To check the test coverage of the package (Hester, 2023).

Other libraries:

* invgamma - To sample from the inverse gamma distribution (Kahle & Stamey, 2017).
* stats - For fundamental statistics for model estimation (2024a).
* GEOquery - To access data from the NCBI Gene Expression Omnibus (GEO) for real-world data analysis (Davis & Meltzer, 2007).

1. Algorithms

* survregVB implements Algorithms 1 and 2 in Appendix D through the survregVB.fit() and survregVB.frailty.fit() functions respectively.

1. Development Environment & Version Control

* RStudio - The primary development environment for writing, building and testing R packages (Silge et al., 2019).
* GitHub - Version control and development.

#### Testing

survregVB uses the testthat package to automate testing and improve code structure, and ensure the functions are behaving as expected. Following the conventions of testthat, the test suite is found under the /tests/testthat directory and contains a corresponding test file for each file found under the /R/ directory (Wickham, 2011). The whole test suite is run by executing devtools::test() (Silge et al., 2019).

The covr package was used to measure code coverage for the entire survregVB package, ensuring that all components are covered by test cases (Hester, 2023). The results indicate that 99.79% of the code base is covered by test cases, showing strong test coverage and robust code.

### 5.2.4 System Validation

devtools::check() was used during development for comprehensive package validation to ensure that survregVB meets CRAN standards. This command runs tests, checks documentation, verifies dependencies, and identifies potential issues such as missing imports, documentation inconsistencies, or coding errors (Wickham et al., 2022).

**devtools::check()** was used for comprehensive package validation, ensuring survregVB meets CRAN standards. This command performs the following checks [@devtools]:

* Runs automated tests to detect errors.
* Validates documentation and function consistency.
* Identifies missing dependencies and imports.
* Ensures compliance with R package development best practices.

#### Case Study using dnase

We use the dnase data set included in the survregVB package as a case study of how to use survregVB() to fit an AFT model without frailty, and compare the results to those obtained by the VB algorithm found in vbaft to validate the system. dnase is a processed subset of the rhDNase data set found in survival and contains results of a trial of rhDNase for the treatment of cystic fibrosis (Therneau, 2024).

1. Model Specification

Our goal is to fit it a log-logistic AFT regression model of the form:

where trt () and fev () are the covariates of interest, and the event status indicator is infect (Therneau, 2024; Xian et al., 2024b).

1. Fitting the Model

First, we load the survregVB and survival libraries:

library(survregVB)  
library(survival)

To fit the log-logistic AFT model using survregVB:

fit <- survregVB(  
 formula = Surv(time, infect) ~ trt + fev, data = dnase, alpha\_0 = 501, omega\_0 = 500,  
 mu\_0 = c(4.4, 0.25, 0.04), v\_0 = 1, max\_iteration = 10000, threshold = 0.0005,  
 na.action = na.omit  
)  
print(fit)  
summary(fit)

The distributions of and match those obtained from the VB algorithm in vbaft, validating the implementation.

#### Simulation Studies

We also simulate survival data with and without clustering to validate the performance of survregVB under different scenarios by comparing the results to those obtained by the VB algorithm found in vbaft.

1. Scenario with frailty (simulated\_frailty)

We will show an example using the simulated\_frailty data set included in the package, which contains survival data generated as follows for the subject in the cluster, and :

* with
* Censoring time where , and and to achieve a 15% censoring rate (Xian et al., 2024a).

The following fits the model with non-informative priors:

fit2 <- survregVB(  
 formula = Surv(T.15, delta.15) ~ x1 + x2, data = simulation\_frailty, alpha\_0 = 3,  
 omega\_0 = 2, mu\_0 = c(0, 0, 0), v\_0 = 0.1, lambda\_0 = 3, eta\_0 = 2,  
 cluster = cluster, max\_iteration = 100, threshold = 0.01  
)  
print(fit2)  
summary(fit2)

These distributions of , (scale) and (intercept) match those obtained from the shared frailty VB algorithm from Xian et al. (2024a), validating the implementation.

1. Scenario without frailty (simulated\_nofrailty)

Similar simulation studies were performed using unclustered survival data found in the simulated\_nofrailty data set included in survregVB. The results from these studies match those from the VB algorithm found in vbaft, validating the implementation.

## 5.3 Novelty of Results

While VB methods have been explored in statistical literature, no existing software provides a dedicated and user-friendly implementation of VB for AFT models. survregVB fills this gap by offering an efficient alternative to traditional MCMC-based Bayesian methods for survival analysis data that integrates seamlessly with existing R survival workflows, pre-processes data and handles missing values. These features make Bayesian inference more accessible to applied researchers in various fields, including medicine, engineering, and social sciences.

### Comparative Analysis to Other Work

To establish the novelty of survregVB, we compare its performance on two real-world data sets with:

* Likelihood-based estimation using survreg() from survival.
* HMC sampling from rstan.

For each of the following data sets, we fit a log-logistic AFT model, use historically motivated priors for Bayesian estimation, and compare parameter estimates, runtime, and model fit.

#### GSE102287 Data set

This data set, publicly available in the NCBI Gene Expression Omnibus (GEO), contains clinical and gene expression data from African American and European American patients with non-small cell lung cancer (NSCLC) (Mitchell et al., 2017). A processed subset of 60 African American patients referred to as lung\_cancer is included in survregVB, focusing on factors influencing survival, such as age, cancer stage, gender, and smoking status . Previous research shows that the log-logistic model is the best fit for the data as compared to other parametric models (Kumar et al., 2019).

We choose priors based off historical analysis on this type of data (Kumar et al., 2019). We choose the log of half the follow-up period length, as the mean of the intercept. To summarize, we consider the following prior distributions for the model parameters (Xian et al., 2024b):

* ), with and
* with and

The ELBO convergence threshold is set as which is the default recommendation (Yao et al., 2018). To fit this model using survregVB, we use the following:

fit <- survregVB(formula = Surv(time, status) ~ age + Stage + gender + factor(smoking),  
 data = lung\_cancer, alpha\_0 = 11, omega\_0 = 10,  
 mu\_0 = c(6.8, 0, 0, 0, 0, 0), v\_0 = 1, threshold = 0.01)

#### Lung Data set

The lung data set in the survival package contains survival data from 228 patients with advanced lung cancer from the North Central Cancer Treatment Group (Therneau, 2024). We have assessed that this data set is suitable for the log-logistic distribution.

We choose the log of the median follow-up period length, as the mean of the intercept and as the prior means for the other covariates based of previous studies (Ando et al., 2001; Kumar et al., 2019). To summarize, we consider the following prior distributions for the model parameters (Xian et al., 2024b):

* ), with and
* with and

We fit the model as follows:

fit2 <- survregVB(formula = Surv(time, status == 2) ~ age + factor(sex) +  
 factor(ph.ecog), data = lung, alpha\_0 = 501, omega\_0 = 500,  
 mu\_0 = c(6.7, 0, 0, 0, 0, 0), v\_0 = 1, threshold = 0.01)

#### Comparative Results

The estimation results from survregVB, survreg and rstan for GSE102287 and lung are shown in Table 1 and Table 2 in Appendix D. The convergence of the MCMC algorithm was well assessed and checked by the trace plot and autocorrection plot (“Comparison of Accelerated Failure Time Models,” 2021). The results demonstrate that survregVB results align well with likelihood and MCMC-based estimates. Furthermore, the credible intervals from survregVB overlap with those from MCMC for both data sets, showing that it captures parameter uncertainty well.

For GSE102287, survregVB achieves a 960x speedup (41.5504 sec → 0.0431 sec), and for lung, survregVB achieves a 160x speedup (12.5573 sec → 0.0765 sec). This significant reduction in runtime confirms that VB provides a stable and computationally efficient alternative to MCMC-based inference. Furthermore, unlike rstan, survregVB automates preprocessing (handling missing data, categorical variables, etc.), making Bayesian inference accessible to applied researchers.

# 6. Discussion

## 6.1 Threats to the Validity of the Results

## 6.2 Implications of the Research Results

Since no existing software provides VB methods for AFT models, the development of survregVB will fill this gap to offer an efficient alternative to MCMC-based methods. Applying VB inference to a variety of data sets, such as ones with small sample sizes, many covariates or differing cluster sizes, provides new insight into its robustness and practicality. Comparing results with traditional techniques provides further information on the computational cost and accuracy of the VB methods. This evaluation will help address the strengths and weaknesses of the current VB algorithms, and identify areas for future research for Bayesian inference in survival models.

As survival analysis is an important field of study across multiple domains, survregVB will benefit researchers involved in biology, medicine, engineering, marketing, social sciences or behavioral sciences (Emmert-Streib & Dehmer, 2019). The package will be freely available through CRAN, making VB inference widely accessible to end users. Function documentation will be available in the package, and a vignette will serve as an in-depth guide to improve usage. Furthermore, the package will automatically handle missing values, perform one-hot encoding, and generate summary statistics, streamlining the analysis process.

## 6.3 Limitations of the Results

## 6.4 Generalisability of the Results

# 10. Appendix

## A. Log-logistic AFT Model Formulas

**(1) Log-logistic AFT Model Formula**

For survival time , censoring time and censoring indicator for the subject in the sample, , the log-logistic AFT model is represented as follows:

where is column vector of covariates and a constant one (i.e. ), is a vector of coefficients for the covariates, is a random variable following a standard logistic distribution and is a scale parameter (Xian et al., 2024b).

**(2) Shared Frailty Log-logistic AFT Model Formula**

or survival time , censoring time and censoring indicator for the subject from the cluster in a sample with clusters and subjects, the shared frailty log-logistic AFT model is represented as follows:

where is column vector of covariates and a constant one (i.e. ), is a vector of coefficients for the covariates, is a random intercept for the cluster, is a variable following a standard logistic distribution and is a scale parameter (Xian et al., 2024a).

## B. VB Algorithms

**Algorithm 1.** Variational Bayes Inference of Survival Data using a Log-logistic AFT Model (Xian et al., 2024b)

**Data**: a sample of independent log observed time , their corresponding covariate vectors and the right censoring indicator where is the sample size; values of hyperparameters: and ; convergence threshold and maximum number of iterations

**Result**: posterior distributions of and , and their hyperparameters

1 **Initialization**: initialize and , set and ;

2 **Calculation**: obtain by with ;

3 **while** iteration and difference of **do**

4 ;

5 ;

6 ;

7 ;

8 calculate the current ;

9 calculate the current difference of , ;

10 **end**

**Algorithm 2**: Variational Bayes Inference of Survival Data using a Shared Frailty Log-logistic AFT Model (Xian et al., 2024a)

**Data**: a sample of independent log observed time , their corresponding covariate vectors and the right censoring indicator for the observation from the group; values of hyperparameters: and ; convergence threshold and maximum number of iterations

**Results**: posterior distributions of and , and their parameters

1 **Initialization**: initialize and , set and ;

2 **Calculation**: obtain by with and by ;

3 **while** iteration and difference of **do**

4 ;

5 ;

6 ;

7 ;

8 ;

9 ;

10 ;

11 calculate the current ;

12 calculate the current difference of , ;

13 **end**

## C. Components of survregVB

**(1) survregVB** **Package Structure**

survregVB/  
├── R/  
│ ├── survregVB.R # Primary user-facing function   
│ ├── survregVB.frailty.fit.R # Implements VB with shared frailty   
│ ├── survregVB.fit.R # Implements VB without shared frailty   
│ ├── summary.survregVB.R # Summary method for survregVB  
│ ├── print.survregVB.R # Print method for survregVB  
│ ├── print.summary.survregVB.R # Print method for summary.survregVB  
│ ├── parameters.R # Parameter update calculations for VB   
│ ├── ELBO.R # Convergence criteria calculations for VB  
│ ├── data.R # Dataset documentation  
│ ├── credible\_intervals.R # Computes Bayesian credible intervals   
│  
├── tests/ # Unit tests for functions found in R/   
│ ├── testthat/  
│  
├── vignettes/   
│ ├── survregVB.Rmd # User guide and examples  
│  
├── man/ # Documentation  
│  
├── data/ # Example data sets   
│ ├── simulation\_nofrailty.rda # Simulated data set without clusters  
│ ├── simulation\_frailty.rda # Simulated data set with clusters  
│ ├── lung\_cancer.rda # Subset of the GSE102287 data set  
│ ├── dnase.rda # Subset of the rhDNase data set  
├── data-raw/ # Clean raw data before saving in data/  
│  
├── DESCRIPTION # Package metadata  
├── NAMESPACE # Function exports  
├── LICENSE # License file  
├── LICENSE.md # Readable version of license file   
└── README.md # Installation guide

## D. Comparative Results

**Table 1**: Results from analysis on NCBI lung cancer data: Posterior means (Mean) with posterior standard deviations (SD), and 95% credible intervals (95% Cred. Int.) from the VB algorithm implemented by survregVB and MCMC, respectively. Point estimates (Est.) with standard errors (SE), and 95% confidence interval (95% Conf. Int.) from survreg in the R package survival

**Table 2**: Results from analysis on lung cancer data from the R package survival: Posterior means (Mean) with posterior standard deviations (SD), and 95% credible intervals (95% Cred. Int.) from the VB algorithm implemented by survregVB and MCMC, respectively. Point estimates (Est.) with standard errors (SE), and 95% confidence interval (95% Conf. Int.) from survreg in the R package survival

# References

Allaire, J., Xie, Y., Dervieux, C., McPherson, J., Luraschi, J., Ushey, K., Atkins, A., Wickham, H., Cheng, J., Chang, W., & Iannone, R. (2024). *Rmarkdown: Dynamic documents for r*. <https://github.com/rstudio/rmarkdown>

Ando, M., Ando, Y., Hasegawa, Y., Shimokata, K., Minami, H., Wakai, K., Ohno, Y., & Sakai, S. (2001). Prognostic value of performance status assessed by patients themselves, nurses and oncologists in advanced non-small cell lung cancer. *British Journal of Cancer*, *85*(11), 1634–1639. <https://doi.org/10.1054/bjoc.2001.2162>

Bishop, C. (2006). *Pattern recognition and machine learning*. Springer New York.

Blei, D. M., Kucukelbir, A., & McAuliffe, J. D. (2017). Variational Inference: A Review for Statisticians. *Journal of the American Statistical Association*, *112*(518), 859–877. <https://doi.org/10.1080/01621459.2017.1285773>

Comparison of accelerated failure time models: A bayesian study on head and neck cancer data. (2021). *Journal of Statistics Applications & Probability*, *10*(3), 715–738. <https://doi.org/10.18576/jsap/100311>

Davis, S., & Meltzer, P. (2007). *GEOquery: A bridge between the gene expression omnibus (GEO) and BioConductor*. *14*. <https://doi.org/10.1093/bioinformatics/btm254>

Emmert-Streib, F., & Dehmer, M. (2019). Introduction to Survival Analysis in Practice. *Machine Learning and Knowledge Extraction*, *1*(3), 1013–1038. <https://doi.org/10.3390/make1030058>

Geman, S., & Geman, D. (1984). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *PAMI-6*(6), 721–741. <https://doi.org/10.1109/tpami.1984.4767596>

Gorfine, M., & Zucker, D. M. (2023). Shared Frailty Methods for Complex Survival Data: A Review of Recent Advances. *Annual Review of Statistics and Its Application*, *10*(1), 51–73. <https://doi.org/10.1146/annurev-statistics-032921-021310>

Gutierrez, R. G. (2002). Parametric Frailty and Shared Frailty Survival Models. *The Stata Journal: Promoting Communications on Statistics and Stata*, *2*(1), 22–44. <https://doi.org/10.1177/1536867x0200200102>

Ha, I., Noh, M., & Lee, Y. (2012). frailtyHL: A Package for Fitting Frailty Models with H-likelihood. *The R Journal*, *4*(2), 28. <https://doi.org/10.32614/rj-2012-010>

Hanagal, D. D. (2011). *Modeling Survival Data Using Frailty Models*. Chapman; Hall/CRC. <https://doi.org/10.1201/b10510>

Hester, J. (2023). *Covr: Test coverage for packages*. <https://CRAN.R-project.org/package=covr>

Hester, J., Angly, F., Hyde, R., Chirico, M., Ren, K., Rosenstock, A., & Patil, I. (2025). *Lintr: A ’linter’ for r code*. <https://CRAN.R-project.org/package=lintr>

Hougaard, P. (1995). Frailty models for survival data. *Lifetime Data Analysis*, *1*(3), 255–273. <https://doi.org/10.1007/bf00985760>

Ibrahim, J. G., Chen, M.-H., & Sinha, D. (2001). *Bayesian Survival Analysis*. Springer New York. <https://doi.org/10.1007/978-1-4757-3447-8>

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., & Saul, L. K. (1999). *Machine Learning*, *37*(2), 183–233. <https://doi.org/10.1023/a:1007665907178>

Kahle, D., & Stamey, J. (2017). *Invgamma: The inverse gamma distribution*. <https://CRAN.R-project.org/package=invgamma>

Kobara, Y. M. (2022). *Statistical applications to the management of intensive care and step-down units* [PhD thesis]. The University of Western Ontario (Canada).

Kumar, M., Sonker, P. Kr., Saroj, A., Jain, A., Bhattacharjee, A., & Saroj, R. Kr. (2019). Parametric survival analysis using R: Illustration with lung cancer data. *Cancer Reports*, *3*(4). <https://doi.org/10.1002/cnr2.1210>

Lambert, P., Collett, D., Kimber, A., & Johnson, R. (2004). Parametric accelerated failure time models with random effects and an application to kidney transplant survival. *Statistics in Medicine*, *23*(20), 3177–3192. <https://doi.org/10.1002/sim.1876>

Mitchell, K. A., Zingone, A., Toulabi, L., Boeckelman, J., & Ryan, B. M. (2017). Comparative Transcriptome Profiling Reveals Coding and Noncoding RNA Differences in NSCLC from African Americans and European Americans. *Clinical Cancer Research*, *23*(23), 7412–7425. <https://doi.org/10.1158/1078-0432.ccr-17-0527>

Müller, K., & Walthert, L. (2024). *Styler: Non-invasive pretty printing of r code*. <https://CRAN.R-project.org/package=styler>

R Core Team. (2024a). *R: A language and environment for statistical computing*. <https://www.R-project.org/>

Rivas-López, M. J., Martín-Martín, R., & García-Camacha Gutiérrez, I. (2022). Recent Advances in Robust Design for Accelerated Failure Time Models with Type I Censoring. *Mathematics*, *10*(3), 379. <https://doi.org/10.3390/math10030379>

Silge, J., Nash, John,C., & Graves, S. (2019). Navigating the R Package Universe. *The R Journal*, *10*(2), 558. <https://doi.org/10.32614/rj-2018-058>

Stan Development Team. (2024b). *RStan: The r interface to stan*. <https://mc-stan.org/>

Team, R. C. (n.d.). *The r project for statistical computing*. <https://www.r-project.org/about.html>

Therneau, T. M. (2024). *A package for survival analysis in r*. <https://CRAN.R-project.org/package=survival>

Wainwright, M. J., & Jordan, M. I. (2007). Graphical Models, Exponential Families, and Variational Inference. *Foundations and Trends® in Machine Learning*, *1*(12), 1–305. <https://doi.org/10.1561/2200000001>

Webber, C., Brundage, M., Hanna, T. P., Booth, C. M., Kennedy, E., Kong, W., Peng, Y., Whitehead, M., & Groome, P. A. (2022). Explaining regional variations in colon cancer survival in Ontario, Canada: a population-based retrospective cohort study. *BMJ Open*, *12*(9), e059597. <https://doi.org/10.1136/bmjopen-2021-059597>

Wickham, H. (2011). *Testthat: Get started with testing*. *3*. <https://journal.r-project.org/archive/2011-1/RJournal_2011-1_Wickham.pdf>

Wickham, H., Bryan, J., Barrett, M., & Teucher, A. (2024). *Usethis: Automate package and project setup*. <https://CRAN.R-project.org/package=usethis>

Wickham, H., Danenberg, P., Csárdi, G., & Eugster, M. (2024). *roxygen2: In-line documentation for r*. <https://CRAN.R-project.org/package=roxygen2>

Wickham, H., Hester, J., Chang, W., & Bryan, J. (2022). *Devtools: Tools to make developing r packages easier*. <https://CRAN.R-project.org/package=devtools>

Xian, C., Souza, C. P. E. de, He, W., Rodrigues, F. F., & Tian, R. (2024a). *Fast variational bayesian inference for correlated survival data: An application to invasive mechanical ventilation duration analysis*. <https://doi.org/10.48550/ARXIV.2408.00177>

Xian, C., Souza, C. P. E. de, He, W., Rodrigues, F. F., & Tian, R. (2024b). Variational Bayesian analysis of survival data using a log-logistic accelerated failure time model. *Statistics and Computing*, *34*(2). <https://doi.org/10.1007/s11222-023-10365-6>

Xie, Y. (2024). *Knitr: A general-purpose package for dynamic report generation in r*. <https://yihui.org/knitr/>

Yao, Y., Vehtari, A., Simpson, D., & Gelman, A. (2018). Yes, but did it work?: Evaluating variational inference. *International Conference on Machine Learning*, 5581–5590.

Zhou, H., Hanson, T., & Zhang, J. (2020). *spBayesSurv: Fitting bayesian spatial survival models using r*. *92*. <https://doi.org/10.18637/jss.v092.i09>