

Likelihood Models for Series Systems with Masked Component Failure Data:

An R Package for Maximum Likelihood Estimation

Alexander Towell
lex@metafunctor.com
ORCID: 0000-0001-6443-9897

December 3, 2025

Abstract

This technical report introduces the `likelihood.model.series.md` R package for maximum likelihood estimation in series systems with masked component cause of failure data. The package provides a unified framework for exponential and Weibull series systems, implementing log-likelihood functions, score vectors, and Hessian matrices under specific masking conditions (C1, C2, C3). We describe the mathematical foundation, software architecture, and integration with the broader `likelihood.model` ecosystem. The package enables practitioners to perform parameter estimation, construct confidence intervals, and conduct hypothesis tests for series system reliability problems where component failure causes are only partially observable.

1 Introduction

Series systems are prevalent in reliability engineering, where system failure occurs when any single component fails. A fundamental challenge in series system reliability analysis is that often only the system failure time is observable, while the specific component that caused the failure may be unknown or only partially identified through a *candidate set*—a subset of components that plausibly contain the failed component.

This situation arises in many practical contexts:

- Field failure data where diagnostic information is incomplete
- Systems where post-failure inspection is infeasible or costly
- Warranty data where failure cause is self-reported with uncertainty
- Accelerated life testing where failure modes may be ambiguous

The `likelihood.model.series.md` package provides tools for maximum likelihood estimation (MLE) from such *masked* failure data. This report describes the mathematical foundation, software design, and usage of the package.

2 Mathematical Framework

2.1 Series System Model

Consider a series system with m components. Let T_j denote the lifetime of component j , for $j = 1, \dots, m$. The system lifetime is

$$T = \min(T_1, \dots, T_m), \quad (1)$$

and the component that causes system failure is

$$K = \arg \min_j T_j. \quad (2)$$

We assume component lifetimes T_1, \dots, T_m are independent with distribution functions $F_j(t; \theta_j)$ parameterized by θ_j . Let $\boldsymbol{\theta} = (\theta_1, \dots, \theta_m)$ denote the full parameter vector.

2.2 Data Structure

For each observation $i = 1, \dots, n$, we observe:

- t_i : The system lifetime (possibly right-censored)
- δ_i : Right-censoring indicator ($\delta_i = 1$ if exact, $\delta_i = 0$ if right-censored)
- $C_i \subseteq \{1, \dots, m\}$: Candidate set of components that may have caused failure

The candidate set provides partial information about the failed component. When $|C_i| = 1$, the failure cause is exactly known; when $C_i = \{1, \dots, m\}$, no information about failure cause is available.

2.3 Masking Conditions

The likelihood function depends on assumptions about the masking mechanism. We consider three conditions:

Condition 1 (C1: Candidate Set Validity). The failed component is always included in the candidate set:

$$\Pr(K_i \in C_i) = 1. \quad (3)$$

Condition 2 (C2: Symmetric Masking). The probability of observing candidate set c is the same regardless of which component in c actually failed:

$$\Pr(C_i = c \mid K_i = j, T_i = t) = \Pr(C_i = c \mid K_i = j', T_i = t) \quad (4)$$

for any $j, j' \in c$.

Condition 3 (C3: Parameter Independence). The masking probabilities do not depend on the system parameters $\boldsymbol{\theta}$:

$$\Pr(C_i = c \mid K_i, T_i) \text{ is independent of } \boldsymbol{\theta}. \quad (5)$$

Under conditions C1, C2, and C3, the likelihood function simplifies considerably, allowing the masking probabilities to be factored out and ignored for parameter estimation.

2.4 Likelihood Function

Under conditions C1, C2, C3, the likelihood contribution from observation i is:

$$L_i(\boldsymbol{\theta}) = \begin{cases} S(t_i; \boldsymbol{\theta}) \cdot \sum_{j \in C_i} h_j(t_i; \theta_j) & \text{if } \delta_i = 1 \\ S(t_i; \boldsymbol{\theta}) & \text{if } \delta_i = 0 \end{cases} \quad (6)$$

where $S(t; \boldsymbol{\theta}) = \prod_{j=1}^m S_j(t; \theta_j)$ is the system survival function and $h_j(t; \theta_j) = f_j(t; \theta_j)/S_j(t; \theta_j)$ is the hazard function for component j .

The full log-likelihood is

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^n \left[\log S(t_i; \boldsymbol{\theta}) + \delta_i \cdot \log \left(\sum_{j \in C_i} h_j(t_i; \theta_j) \right) \right]. \quad (7)$$

2.5 Exponential Series Systems

For exponential component lifetimes with rate parameters $\lambda_1, \dots, \lambda_m$, we have:

- $S_j(t; \lambda_j) = e^{-\lambda_j t}$
- $h_j(t; \lambda_j) = \lambda_j$

The log-likelihood simplifies to:

$$\ell(\boldsymbol{\lambda}) = - \left(\sum_{i=1}^n t_i \right) \cdot \sum_{j=1}^m \lambda_j + \sum_{i: \delta_i=1} \log \left(\sum_{j \in C_i} \lambda_j \right). \quad (8)$$

The score vector and Hessian matrix have closed-form expressions:

$$\frac{\partial \ell}{\partial \lambda_j} = - \sum_{i=1}^n t_i + \sum_{i: \delta_i=1} \frac{\mathbf{1}(j \in C_i)}{\sum_{k \in C_i} \lambda_k} \quad (9)$$

$$\frac{\partial^2 \ell}{\partial \lambda_j \partial \lambda_k} = - \sum_{i: \delta_i=1} \frac{\mathbf{1}(j \in C_i) \cdot \mathbf{1}(k \in C_i)}{\left(\sum_{l \in C_i} \lambda_l \right)^2} \quad (10)$$

2.6 Weibull Series Systems

For Weibull component lifetimes with shape parameters $\alpha_1, \dots, \alpha_m$ and scale parameters β_1, \dots, β_m , we have:

- $S_j(t; \alpha_j, \beta_j) = \exp \left(- \left(\frac{t}{\beta_j} \right)^{\alpha_j} \right)$
- $h_j(t; \alpha_j, \beta_j) = \frac{\alpha_j}{\beta_j} \left(\frac{t}{\beta_j} \right)^{\alpha_j - 1}$

The parameter vector is $\boldsymbol{\theta} = (\alpha_1, \beta_1, \dots, \alpha_m, \beta_m)$ with $2m$ parameters. The log-likelihood is:

$$\ell(\boldsymbol{\theta}) = - \sum_{i=1}^n \sum_{j=1}^m \left(\frac{t_i}{\beta_j} \right)^{\alpha_j} + \sum_{i: \delta_i=1} \log \left(\sum_{j \in C_i} h_j(t_i; \alpha_j, \beta_j) \right). \quad (11)$$

Analytical score expressions are provided in the package; the Hessian is computed numerically via the Jacobian of the score.

2.7 Homogeneous Shape Weibull Model (Reduced Model)

For well-designed series systems where components have similar wear-out characteristics, it is reasonable to assume a common shape parameter k across all components while retaining individual scale parameters β_1, \dots, β_m . This *reduced model* has $m + 1$ parameters instead of $2m$.

A key property: under homogeneous shapes, the series system lifetime is itself Weibull distributed:

$$T \sim \text{Weibull}(k, \beta_s), \quad \text{where } \beta_s = \left(\sum_{j=1}^m \beta_j^{-k} \right)^{-1/k}. \quad (12)$$

The log-likelihood simplifies to:

$$\ell(k, \beta_1, \dots, \beta_m) = - \sum_{i=1}^n \sum_{j=1}^m \left(\frac{t_i}{\beta_j} \right)^k + \sum_{i:\delta_i=1} \log \left(\sum_{j \in C_i} \frac{k}{\beta_j} \left(\frac{t_i}{\beta_j} \right)^{k-1} \right). \quad (13)$$

This reduced model offers several advantages:

- Fewer parameters ($m+1$ vs $2m$) leads to lower estimator variance
- System lifetime has closed-form Weibull distribution
- Interpretable as a single failure mode with component-specific scales
- Model selection via likelihood ratio test: $\Lambda = -2(\ell_R - \ell_F) \sim \chi^2_{m-1}$

3 Package Architecture

3.1 Design Philosophy

The `likelihood.model.series.md` package follows several design principles:

1. **Generic Interface:** Implements S3 methods conforming to the `likelihood.model` package API, enabling use with generic MLE fitting functions.
2. **Composability:** Separates concerns—data generation, masking, likelihood specification, and fitting are independent operations that can be composed.
3. **Extensibility:** New component lifetime distributions can be added by implementing the required S3 methods.

3.2 Core Classes

The package provides three main likelihood model classes:

- `exp_series_md_c1_c2_c3`: Exponential series system model (m parameters)
- `wei_series_md_c1_c2_c3`: Full Weibull series system model ($2m$ parameters)
- `wei_series_homogeneous_md_c1_c2_c3`: Reduced Weibull model with common shape ($m + 1$ parameters)

Each class implements the following S3 methods from the `likelihood.model` interface:

Method	Description
<code>loglik()</code>	Log-likelihood function generator
<code>score()</code>	Score (gradient) function generator
<code>hess_loglik()</code>	Hessian matrix function generator
<code>assumptions()</code>	Model assumptions

3.3 Data Format

The package expects data frames with the following structure:

- `t`: System lifetime column
- `delta`: Right-censoring indicator (1 = exact, 0 = censored)
- `x1, x2, ..., xm`: Boolean candidate set indicators

For backwards compatibility, if the `delta` column is absent, censoring is inferred from empty candidate sets (all `FALSE`).

3.4 Dependencies

The package integrates with several related packages:

- `likelihood.model`: Provides the generic likelihood model interface
- `algebraic.mle`: MLE objects with rich method support (`confint`, `vcov`, etc.)
- `md.tools`: Utilities for encoding/decoding masked data matrices
- `numDeriv`: Numerical differentiation for Weibull Hessian

4 Usage Example

4.1 Creating a Likelihood Model

```

1 library(likelihood.model.series.md)
2
3 # Create exponential series model
4 model_exp <- exp_series_md_c1_c2_c3()
5
6 # Create Weibull series model (full, 2m parameters)
7 model_wei <- wei_series_md_c1_c2_c3()
8
9 # Create reduced Weibull model (homogeneous shape, m+1 parameters)
10 model_hom <- wei_series_homogeneous_md_c1_c2_c3()

```

4.2 Evaluating the Log-Likelihood

```
1 # Get log-likelihood function
2 ll_fn <- loglik(model_exp)
3
4 # Evaluate at parameter values
5 # For 3-component system with rates (0.5, 0.3, 0.2)
6 ll_value <- ll_fn(data, par = c(0.5, 0.3, 0.2))
```

4.3 Maximum Likelihood Estimation

```
1 # Using optim directly
2 ll_fn <- loglik(model_exp)
3 result <- optim(
4     par = c(1, 1, 1), # initial values
5     fn = function(theta) -ll_fn(data, theta),
6     method = "L-BFGS-B",
7     lower = rep(1e-6, 3)
8 )
9 mle <- result$par
10
11 # Or using the likelihood.model framework
12 library(likelihood.model)
13 solver <- fit(model_exp)
14 mle_result <- solver(data, par = c(1, 1, 1))
```

4.4 Generating Simulated Data

```
1 library(dplyr)
2
3 # Generate component lifetimes
4 n <- 100
5 df <- data.frame(
6     t1 = rexp(n, 0.5),
7     t2 = rexp(n, 0.3),
8     t3 = rexp(n, 0.2)
9 )
10
11 # Apply right-censoring at time tau
12 df <- md_series_lifetime_right_censoring(df, tau = 10)
13
14 # Generate candidate sets (p = masking probability)
15 df <- md_bernoulli_cand_c1_c2_c3(df, p = 0.3)
16 df <- md_cand_sampler(df)
17
18 # Result has columns: t, delta, x1, x2, x3
```

5 Theoretical Properties

5.1 Identifiability

Under conditions C1, C2, C3 and with sufficient variation in candidate sets, the parameters are identifiable. However, severe masking (all candidate sets equal $\{1, \dots, m\}$) or extreme censoring can lead to practical non-identifiability.

5.2 Asymptotic Properties

Under standard regularity conditions, the MLE $\hat{\boldsymbol{\theta}}_n$ satisfies:

$$\sqrt{n}(\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0) \xrightarrow{d} N(0, I(\boldsymbol{\theta}_0)^{-1}) \quad (14)$$

where $I(\boldsymbol{\theta}_0)$ is the Fisher information matrix.

The observed information matrix $-H(\hat{\boldsymbol{\theta}}_n)$ (negative Hessian at the MLE) provides a consistent estimator of the Fisher information, enabling construction of confidence intervals and hypothesis tests.

5.3 Simulation Study Results

Extensive simulation studies (see package vignette) demonstrate that:

1. The MLE performs well even with significant masking and censoring
2. Bootstrap confidence intervals achieve nominal coverage
3. Performance degrades gracefully as masking probability increases
4. The estimator is robust to moderate right-censoring

6 Conclusion

The `likelihood.model.series.md` package provides a principled, well-documented implementation for likelihood-based inference in series systems with masked component failure data. By integrating with the broader `likelihood.model` ecosystem, the package enables practitioners to leverage sophisticated MLE infrastructure while focusing on their specific reliability analysis problems.

Future extensions may include:

- Additional component lifetime distributions (log-normal, gamma)
- Relaxation of masking conditions (non-symmetric masking)
- Support for interval censoring
- Bayesian inference methods

The package is available at <https://github.com/queelius/likelihood.model.series.md> with documentation at <https://queelius.github.io/likelihood.model.series.md/>.

Acknowledgments

This work builds on the theoretical framework developed in the author's Master's thesis on reliability estimation in series systems.