Scale=MatchLowercase []Ligatures=TeX,Scale=1

Reliability Estimation in Series Systems
Maximum Likelihood Techniques for Right-Censored and Masked Failure Data

Alex Towell

lex@metafunctor.com

2023-09-30

Abstract

This paper investigates maximum likelihood techniques to estimate component reliability from masked failure data in series systems. A likelihood model accounts for right-censoring and candidate sets indicative of masked failure causes. Extensive simulation studies assess the accuracy and precision of maximum likelihood estimates under varying sample size, masking probability, and right-censoring time for components with Weibull lifetimes. The studies specifically examine the accuracy and precision of estimates, along with the coverage probability and width of BCa confidence intervals. Despite significant masking and censoring, the maximum likelihood estimator demonstrates good overall performance. The bootstrap yields correctly specified confidence intervals even for small sample sizes. Additionally, a likelihood ratio test is developed for selecting between full heterogeneous-shape and reduced homogeneous-shape models, demonstrating proper Type I error control and good power to detect meaningful departures from shape homogeneity. Together, the modeling framework and simulation studies provide rigorous validation of statistical learning from masked reliability data.

Introduction

Quantifying the reliability of individual components in a series system [1] is challenging when only system-level failure data is observable, especially when this data is masked by right-censoring and ambiguity about the cause of failure. This paper develops and validates maximum likelihood techniques to estimate component reliability from right-censored lifetimes and candidate sets indicative of masked failure causes. Specific contributions include:

- Deriving a likelihood model that incorporates right-censoring and candidate sets to enable masked data to be used for parameter estimation.

- Conducting simulation studies for a well-designed series system with component lifetimes following a Weibull distribution. We assess the accuracy and precision of maximum likelihood estimates (MLE) under varying conditions related to sample size, masking probability, and right-censoring. We found that the MLE performs well in the presence of significant masking and censoring even for relatively small samples.

- Evaluating the coverage probability (accuracy) and precision of the BCa confidence intervals (CI) constructed for the MLE. We found that the CIs have good empirical coverage probability even for small sample sizes in the presence of significant masking and censoring, but that the CIs for the shape parameters were less accurate, indicating that the shape parameters are more difficult to estimate than the scale parameters.

The simulation studies focus on three key aspects:

1. The impact of right-censoring on component parameter estimates.
2. How masking probability for the cause of failure affects the estimates.
3. The role of sample size in mitigating challenges related to censoring and masking.

Together, the likelihood framework and simulation methodology enable rigorous validation of inferring component reliability from masked system data. This expands the capability to learn properties of the latent components and perform robust statistical inference given significant data challenges.

Series System Model

Consider a system composed of m components arranged in a series configuration. Each component and system has two possible states, functioning or failed. We have n systems whose lifetimes are independent and identically distributed (i.i.d.). The lifetime of the ith system is denoted by the random variable T_i and the lifetime of its jth component is denoted by the random variable T_{ij}. We assume the component lifetimes in a single system are statistically independent and non-identically distributed. Here, lifetime (or lifespan) is defined as the elapsed time from when the new, functioning component (or system) is put into operation until it fails for the first time. A series system fails when any component fails, thus the lifetime of the ith system is given by the component with the shortest lifetime,

$$T_i = \min\{T_{i1}, T_{i2}, \ldots, T_{im}\}.$$

There are three particularly important distribution functions in reliability analysis: the reliability function, the probability density function (pdf), and the hazard function. The reliability function, $R_{T_i}(t)$, is the probability that the ith system has a lifetime greater than the given duration t,

$$R_{T_i}(t) = \Pr\{T_i > t\}.$$

The pdf of T_i is denoted by $f_{T_i}(t)$ and may be defined as

$$f_{T_i}(t) = -\frac{d}{dt}R_{T_i}(t).$$

Next, we introduce the hazard function. The probability that a failure occurs between the times t and $t+\Delta t$ given that no failure occurs before time t may be written as

$$\Pr\{T_i \le t + \Delta t | T_i > t\} = \frac{\Pr\{t < T_i < t + \Delta t\}}{\Pr\{T_i > t\}}.$$

The failure rate is given by dividing this equation by the length of the time interval, Δt:

$$\frac{\Pr\{t < T_i < t + \Delta t\}}{\Delta t} \frac{1}{\Pr\{T_i > t\}} = -\frac{R_{T_i}(t + \Delta t) - R_{T_i}(t)}{\Delta t} \frac{1}{R_{T_i}(t)}.$$

The hazard function $h_{T_i}(t)$ for T_i is the instantaneous failure rate at time t, which is given by

$$
\begin{aligned}
h_{T_i}(t) = &- \lim_{\Delta t \to 0} \frac{R_{T_i}(t + \Delta t) - R_{T_i}(t)}{\Delta t} \frac{1}{R_{T_i}(t)} \qquad (\#eq: failure-rate)\\
= &\left(-\frac{d}{dt}R_{T_i}(t)\right) \frac{1}{R_{T_i}(t)} = \frac{f_{T_i}(t)}{R_{T_i}(t)}.
\end{aligned}
$$

The lifetime of the jth component is assumed to follow a parametric distribution indexed by a parameter vector $\boldsymbol{\theta_j}$. The parameter vector of the overall system is defined as

$$\boldsymbol{\theta} = (\boldsymbol{\theta_1}, \ldots, \boldsymbol{\theta_m}),$$

where $\boldsymbol{\theta_j}$ is the parameter vector of the jth component.

When a random variable X is parameterized by a particular $\boldsymbol{\theta}$, we denote the reliability function by $R_X(t; \boldsymbol{\theta})$, and the same for the other distribution functions. As a special case, for the components in a series system, we subscript by their labels, e.g, the pdf of the jth component is denoted by $f_j(t; \boldsymbol{\theta_j})$. Two continuous

random variables X and Y have a joint pdf $f_{X,Y}(x, y)$. Given the joint pdf $f_{X,Y}(x, y)$, the marginal pdf of X is given by

$$f_X(x) = \int_{\mathcal{Y}} f_{X,Y}(x, y) dy,$$

where $\mathcal{Y}$ is the support of Y. (If Y is discrete, replace the integration with a summation over $\mathcal{Y}$.)

The conditional pdf of Y given $X = x$, $f_{Y|X}(y|x)$, is defined as

$$f_{X|Y}(y|x) = \frac{f_{X,Y}(x, y)}{f_X(x)}.$$

We may generalize all of the above to more than two random variables, e.g., the joint pdf of $X_1, \ldots, X_m$ is denoted by $f(x_1, \ldots, x_m)$.

Next, we dive deeper into these concepts and provide mathematical derivations for the reliability function, the pdf, and the hazard function of the series system. We begin with the reliability function of the series system, as given by the following theorem.

The series system has a reliability function given by

$$(\#eq : sys - reliability - fn) R_{T_i}(t; \boldsymbol{\theta}) = \prod_{j=1}^{m} R_j(t; \boldsymbol{\theta_j}),$$

where $R_j(t; \boldsymbol{\theta_j})$ is the reliability function of the jth component.

The reliability function is defined as

$$R_{T_i}(t; \boldsymbol{\theta}) = \Pr\{T_i > t\}$$

which may be rewritten as

$$R_{T_i}(t; \boldsymbol{\theta}) = \Pr\{\min\{T_{i1}, \ldots, T_{im}\} > t\}.$$

For the minimum to be larger than t, every component must be larger than t,

$$R_{T_i}(t; \boldsymbol{\theta}) = \Pr\{T_{i1} > t, \ldots, T_{im} > t\}.$$

Since the component lifetimes are independent, by the product rule the above may be rewritten as

$$R_{T_i}(t; \boldsymbol{\theta}) = \Pr\{T_{i1} > t\} \times \cdots \times \Pr\{T_{im} > t\}.$$

By definition, $R_j(t; \boldsymbol{\theta}) = \Pr\{T_{ij} > t\}$. Performing this substitution obtains the result

$$R_{T_i}(t; \boldsymbol{\theta}) = \prod_{j=1}^{m} R_j(t; \boldsymbol{\theta_j}).$$

Theorem @ref(thm:sys-reliability-fn) shows that the system's overall reliability is the product of the reliabilities of its individual components. This is an important relationship in all series systems and will be used in the subsequent derivations. Next, we turn our attention to the pdf of the system lifetime, described in the following theorem.

The series system has a pdf given by

$$(\#eq : sys - pdf) f_{T_i}(t; \boldsymbol{\theta}) = \sum_{j=1}^{m} f_j(t; \boldsymbol{\theta_j}) \prod_{\substack{k=1 \\ k \neq j}}^{m} R_k(t; \boldsymbol{\theta_j}),$$

where $f_j(t; \boldsymbol{\theta_j})$ is the pdf of the jth component and $R_k(t; \boldsymbol{\theta_k})$ is the reliability function of the kth component.

By definition, the pdf may be written as

$$f_{T_i}(t; \boldsymbol{\theta}) = -\frac{d}{dt} \prod_{j=1}^{m} R_j(t; \boldsymbol{\theta_j}).$$

By the product rule, this may be rewritten as

$$f_{T_i}(t;\boldsymbol{\theta}) = -\frac{d}{dt}R_1(t;\boldsymbol{\theta_1})\prod_{j=2}^{m}R_j(t;\boldsymbol{\theta_j}) - R_1(t;\boldsymbol{\theta_1})\frac{d}{dt}\prod_{j=2}^{m}R_j(t;\boldsymbol{\theta_j})$$

$$= f_1(t;\boldsymbol{\theta})\prod_{j=2}^{m}R_j(t;\boldsymbol{\theta_j}) - R_1(t;\boldsymbol{\theta_1})\frac{d}{dt}\prod_{j=2}^{m}R_j(t;\boldsymbol{\theta_j}).$$

Recursively applying the product rule $m-1$ times results in

$$f_{T_i}(t;\boldsymbol{\theta}) = \sum_{j=1}^{m-1}f_j(t;\boldsymbol{\theta_j})\prod_{\substack{k=1\\k\neq j}}^{m}R_k(t;\boldsymbol{\theta_k}) - \prod_{j=1}^{m-1}R_j(t;\boldsymbol{\theta_j})\frac{d}{dt}R_m(t;\boldsymbol{\theta_m}),$$

which simplifies to

$$f_{T_i}(t;\boldsymbol{\theta}) = \sum_{j=1}^{m}f_j(t;\boldsymbol{\theta_j})\prod_{\substack{k=1\\k\neq j}}^{m}R_k(t;\boldsymbol{\theta_k}).$$

Theorem @ref(thm:sys-pdf) shows the pdf of the system lifetime is a function of the pdfs and reliabilities of its components. We continue with the hazard function of the system lifetime, defined in the next theorem.

The series system has a hazard function given by

$$(\#eq:sys-failure-rate)\quad h_{T_i}(t;\boldsymbol{\theta}) = \sum_{j=1}^{m}h_j(t;\boldsymbol{\theta_j}),$$

where $h_j(t;\boldsymbol{\theta_j})$ is the hazard function of the jth component.

By Equation @ref(eq:failure-rate), the ith series system lifetime has a hazard function defined as

$$h_{T_i}(t;\boldsymbol{\theta}) = \frac{f_{T_i}(t;\boldsymbol{\theta})}{R_{T_i}(t;\boldsymbol{\theta})}.$$

Plugging in expressions for these functions results in

$$h_{T_i}(t;\boldsymbol{\theta}) = \frac{\sum_{j=1}^{m}f_j(t;\boldsymbol{\theta_j})\prod_{\substack{k=1\\k\neq j}}^{m}R_k(t;\boldsymbol{\theta_k})}{\prod_{j=1}^{m}R_j(t;\boldsymbol{\theta_j})},$$

which can be simplified to

$$h_{T_i}(t;\boldsymbol{\theta}) = \sum_{j=1}^{m}\frac{f_j(t;\boldsymbol{\theta_j})}{R_j(t;\boldsymbol{\theta_j})} = \sum_{j=1}^{m}h_j(t;\boldsymbol{\theta_j}).$$

Theorem @ref(thm:sys-failure-rate) reveals that the system's hazard function is the sum of the hazard functions of its components. By definition, the hazard function is the ratio of the pdf to the reliability function,

$$h_{T_i}(t;\boldsymbol{\theta}) = \frac{f_{T_i}(t;\boldsymbol{\theta})}{R_{T_i}(t;\boldsymbol{\theta})},$$

and we can rearrange this to get

$$f_{T_i}(t;\boldsymbol{\theta}) = h_{T_i}(t;\boldsymbol{\theta})R_{T_i}(t;\boldsymbol{\theta})$$

$$(\#eq:sys-pdf-2)\qquad\qquad = \left\{\sum_{j=1}^{m}h_j(t;\boldsymbol{\theta_j})\right\}\left\{\prod_{j=1}^{m}R_j(t;\boldsymbol{\theta_j})\right\},$$

which we sometimes find to be a more convenient form than Equation @ref(eq:sys-pdf).

In this section, we derived the mathematical forms for the system's reliability, probability density, and hazard functions. Next, we build upon these concepts to derive distributions related to the component cause of failure.

Component Cause of Failure

Whenever a series system fails, precisely one of the components is the cause. We denote the component cause of failure of the ith series system by the discrete random variable K_i, whose support is given by $\{1, \ldots, m\}$. For example, $K_i = j$ indicates that the component indexed by j failed first, i.e.,

$$T_{ij} < T_{ij'}$$

for every j' in the support of K_i except for j. Since we have series systems, K_i is unique.

The system lifetime and the component cause of failure has a joint distribution given by the following theorem.

The joint pdf of the component cause of failure K_i and the series system lifetime T_i is given by

$$(\#eq: f-k-and-t) \quad f_{K_i,T_i}(j,t;\boldsymbol{\theta}) = h_j(t;\boldsymbol{\theta_j}) \prod_{l=1}^{m} R_l(t;\boldsymbol{\theta_l}),$$

where $h_l(t;\boldsymbol{\theta_l})$ and $R_l(t;\boldsymbol{\theta_l})$ are respectively the hazard and reliability functions of the lth component.

Consider a series system with 3 components. By the assumption that component lifetimes are mutually independent, the joint pdf of T_{i1}, T_{i2}, and T_{i3} is given by

$$f(t_1, t_2, t_3; \boldsymbol{\theta}) = \prod_{j=1}^{3} f_j(t_j; \boldsymbol{\theta_j}),$$

where $f_j(t_j; \boldsymbol{\theta_j})$ is the pdf of the jth component. The first component is the cause of failure at time t if $K_i = 1$ and $T_i = t$, which may be rephrased as the likelihood that $T_{i1} = t$, $T_{i2} > t$, and $T_{i3} > t$. Thus,

$$
\begin{aligned}
f_{K_i,T_i}(j,t;\boldsymbol{\theta}) &= \int_t^\infty \int_t^\infty f_1(t;\boldsymbol{\theta_1}) f_2(t_2;\boldsymbol{\theta_2}) f_3(t_3;\boldsymbol{\theta_3}) dt_3 dt_2 \\
&= \int_t^\infty f_1(t;\boldsymbol{\theta_1}) f_2(t_2;\boldsymbol{\theta_2}) R_3(t;\boldsymbol{\theta_3}) dt_2 \\
&= f_1(t;\boldsymbol{\theta_1}) R_2(t;\boldsymbol{\theta_2}) R_3(t_1;\boldsymbol{\theta_3}).
\end{aligned}
$$

By definition, $f_1(t;\boldsymbol{\theta_1}) = h_1(t;\boldsymbol{\theta_1}) R_1(t;\boldsymbol{\theta_1})$, and when we make this substitution into the above expression for $f_{K_i,T_i}(j,t;\boldsymbol{\theta})$, we obtain the result

$$f_{K_i,T_i}(j,t;\boldsymbol{\theta}) = h_1(t;\boldsymbol{\theta_1}) \prod_{l=1}^{m} R_l(t;\boldsymbol{\theta_l}).$$

Generalizing this result completes the proof.

Theorem @ref(thm:f-k-and-t) shows that the joint pdf of the component cause of failure and system lifetime is a function of the hazard functions and reliability functions of the components. This result will be used in Section @ref(like-model) to derive the likelihood function for the masked data.

The probability that the jth component is the cause of failure is given by the following theorem.

The probability that the jth component is the cause of failure is given by

$$(\#eq: prob-k) \quad \Pr\{K_i = j\} = E_{\boldsymbol{\theta}}\left[\frac{h_j(T_i;\boldsymbol{\theta_j})}{\sum_{l=1}^{m} h_l(T_i;\boldsymbol{\theta_l})} \right],$$

where K_i is the random variable denoting the component cause of failure of the ith system and T_i is the random variable denoting the lifetime of the ith system.

The probability the jth component is the cause of failure is given by marginalizing the joint pdf of K_i and T_i over T_i,

$$\Pr\{K_i = j\} = \int_0^\infty f_{K_i,T_i}(j,t;\boldsymbol{\theta}) dt.$$

6

By Theorem @ref(thm:f-k-and-t), this is equivalent to

$$\Pr\{K_i = j\} = \int_0^\infty h_j(t; \boldsymbol{\theta_j}) R_{T_i}(t; \boldsymbol{\theta}) dt$$

$$= \int_0^\infty \left(\frac{h_j(t; \boldsymbol{\theta_j})}{h_{T_i}(t; \boldsymbol{\theta})} \right) f_{T_i}(t; \boldsymbol{\theta}) dt$$

$$= E_{\boldsymbol{\theta}} \left[\frac{h_j(T_i; \boldsymbol{\theta_j})}{\sum_{l=1}^m h_l(T_i; \boldsymbol{\theta_l})} \right].$$

If we know the system failure time, then we can simplify the above expression for the probability that the jth component is the cause of failure. This is given by the following theorem.

The probability that the jth component is the cause of system failure given that we know the system failure occurred at time t_i is given by

$$\Pr\{K_i = j | T_i = t_i\} = \frac{h_j(t_i; \boldsymbol{\theta_j})}{\sum_{l=1}^m h_l(t_i; \boldsymbol{\theta_l})}.$$

By the definition of conditional probability,

$$\Pr\{K_i = j | T_i = t_i\} = \frac{f_{K_i, T_i}(j, t_i; \boldsymbol{\theta})}{f_{T_i}(t_i; \boldsymbol{\theta})}$$

$$= \frac{h_j(t_i; \boldsymbol{\theta_j}) R_{T_i}(t_i; \boldsymbol{\theta})}{f_{T_i}(t_i; \boldsymbol{\theta})}.$$

Since $f_{T_i}(t_i; \boldsymbol{\theta}) = h_{T_i}(t_i; \boldsymbol{\theta}) R_{T_i}(t_i; \boldsymbol{\theta})$, we make this substitution and simplify to obtain

$$\Pr\{K_i = j | T_i = t_i\} = \frac{h_j(t_i; \boldsymbol{\theta_j})}{\sum_{l=1}^m h_l(t_i; \boldsymbol{\theta_l})}.$$

Theorems @ref(thm:prob-k) and @ref(thm:prob-k-given-t) are closely related and have similar forms. Theorem @ref(thm:prob-k-given-t) can be seen as a special case of Theorem @ref(thm:prob-k).

System and Component Reliabilities

The reliability of a system is described by its reliability function, which denotes the probability that the system is functioning at a given time, e.g., $R_{T_i}(t'; \boldsymbol{\theta})$ denotes the probability that the ith system is functioning at time t'. If we want a summary measure of the system's reliability, a common measure is the mean time to failure (MTTF), which is the expectation of the system lifetime,

$$(\#mttf - sys)\text{MTTF} = E_{\boldsymbol{\theta}}[T_i],$$

which if certain assumptions are satisfied[1] is equivalent to the integration of the reliability function over its support. While the MTTF provides a summary measure of reliability, it is not a complete description. Depending on the failure characteristics, the MTTF can be misleading. For example, a system that has a high likelihood of failing early in its life may still have a large MTTF if it is fat-tailed.[2]

The reliability of the components in the series system determines the reliability of the system. We denote the MTTF of the jth component by MTTF_j and, according to Theorem @ref(thm:prob-k), the probability that the jth component is the cause of failure is given by $\Pr\{K_i = j\}$. In a **well-designed** series system, there are no components that are much "weaker" than any of the others, e.g., they have similar MTTFs and probabilities of being the component cause of failure. In this paper, we perform a sensitivity analysis of the MLE for a well-designed series system.

---

[1] T_i is non-negative and continuous, $R_{T_i}(t; \boldsymbol{\theta})$ is a well-defined, continuous, and differential function for $t > 0$, and $\int_0^\infty R_{T_i}(t; \boldsymbol{\theta}) dt$ converges.

[2] A "fat-tailed" distribution refers to a probability distribution with tails that decay more slowly than those of the exponential family, such as the case with the Weibull when its shape parameter is less than 1. This means that extreme values are more likely to occur, and the distribution is more prone to "black swan" events or rare occurrences. In the context of reliability, a fat-tailed distribution might imply a higher likelihood of unusually long lifetimes, which can skew measures like the MTTF. [2]

Likelihood Model for Masked Data

We aim to estimate an unknown parameter, $\boldsymbol{\theta}$, using *masked data*. We consider two types of masking: censoring of system failures and masking component causes of failure.

Censoring   We generally encounter two types of censoring: the system failure is observed to occur within some time interval, or the system failure is not observed but we know that it was functioning at least until some point in time. The latter is known as *right-censoring*, which is the type of censoring we consider in this paper.

Component Cause of Failure Masking   In the case of masking the component cause of failure, we may not know the precise component cause of failure, but we may have some indication. A common example is when a diagnostician is able to isolate the cause of failure to a subset of the components. We call this subset the *candidate set*.

Masked Data   In this paper, each system i is put into operation and observed until either it fails or its failure is right-censored after some duration τ_i, so we do not directly observe the system lifetime but rather we observe the right-censored lifetime, S_i, which is given by

$$S_i = \min\{\tau_i, T_i\}.$$

We also observe an event indicator, δ_i, which is given by

$$\delta_i = 1_{T_i < \tau_i},$$

where $1_{\text{condition}}$ is an indicator function that denotes 1 if the condition is true and 0 otherwise. Here, $\delta_i = 1$ indicates the ith system's failure was observed and $\delta_i = 0$ indicates it was right-censored.[3] If a system failure event is observed ($\delta_i = 1$), then we also observe a candidate set that contains the component cause of failure. We denote the candidate set for the ith system by $\mathcal{C}_i$, which is a subset of $\{1, \ldots, m\}$.

In summary, the observed data is assumed to be i.i.d. and is given by $D = \{D_1, \ldots, D_n\}$, where each D_i contains the following elements:

- S_i is the right-censored system lifetime of the ith system.
- δ_i is the event indicator for the ith system.
- $\mathcal{C}_i$ is the set of candidate component causes of failure for the ith system.
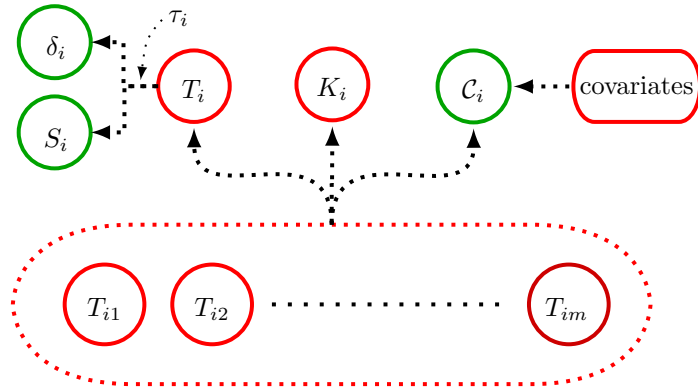

Figure 1: Dependence model for the data generating process for the masked data.

---

[3]In some likelihood models, there may be more than two possible values for δ_i, but in this paper, we only consider the case where δ_i is binary. Future work could consider the case where δ_i is categorical by including more types of censoring events and more types of component cause of failure masking.

In Figure @ref(fig:depmodel), we depict a dependency graph of the data generating process for $D_i = (S_i, \delta_i, \mathcal{C}_i)$. The elements in green are observed in the sample, while the elements in red are unobserved (latent). We see that $\mathcal{C}_i$ is related to both the unobserved component lifetimes $T_{i1}, \ldots, T_{im}$ and other unknown or unobserved covariates, like ambient temperature or the particular diagnostician who generated the candidate set. These two complications for $\mathcal{C}_i$ are why we seek a way to construct a reduced likelihood function in later sections that does not need to model the distribution of $\mathcal{C}_i$.

An example of masked data D with a right-censoring time $\tau = 5$ can be seen in Table @ref(tab:masked-data) for a series system with 3 components.

Table 1: (#tab:masked-data) Right-censored lifetime data with masked component cause of failure.

System	Censored Lifetime (S_i)	Event (δ_i)	Candidates ($\mathcal{C}_i$)
1	1.1	1	$\{1, 2\}$
2	1.3	1	$\{2\}$
4	2.6	1	$\{2, 3\}$
5	3.7	1	$\{1, 2, 3\}$
6	5.0	0	$\emptyset$
3	5.0	0	$\emptyset$

In our model, we assume the data is governed by a pdf, which is determined by a specific parameter, represented as $\boldsymbol{\theta}$ within the parameter space $\boldsymbol{\Omega}$. The joint pdf of the data D can be represented as follows:

$$f(D; \boldsymbol{\theta}) = \prod_{i=1}^{n} f(D_i; \boldsymbol{\theta}) = \prod_{i=1}^{n} f(s_i, \delta_i, c_i; \boldsymbol{\theta}),$$

where s_i is the observed system lifetime, δ_i is the observed event indicator, and c_i is the observed candidate set of the ith system.

This joint pdf tells us how likely we are to observe the particular data, D, given the parameter $\boldsymbol{\theta}$. When we keep the data constant and allow the parameter $\boldsymbol{\theta}$ to vary, we obtain what is called the likelihood function L, defined as

$$L(\boldsymbol{\theta}) = \prod_{i=1}^{n} L_i(\boldsymbol{\theta})$$

where

$$L_i(\boldsymbol{\theta}) = f(D_i; \boldsymbol{\theta})$$

is the likelihood contribution of the ith system.

For each type of data, right-censored data and masked component cause of failure data, we will derive the *likelihood contribution L_i*, which refers to the part of the likelihood function that this particular piece of data contributes to. We present the following theorem for the likelihood contribution model.

The likelihood contribution of the ith system is given by

$$(\#eq:like) L_i(\boldsymbol{\theta}) \propto \prod_{j=1}^{m} R_j(s_i; \boldsymbol{\theta_j}) \left(\sum_{j \in c_i} h_j(s_i; \boldsymbol{\theta_j}) \right)^{\delta_i}$$

where $\delta_i = 0$ indicates the ith system is right-censored at time s_i and $\delta_i = 1$ indicates the ith system is observed to have failed at time s_i with a component cause of failure potentially masked by the candidate set c_i.

In the following subsections, we prove this result for each type of masked data, right-censored system lifetime data ($\delta_i = 0$) and masking of the component cause of failure ($\delta_i = 1$).

Masked Component Cause of Failure

When a system failure occurs, two types of data are observed in our data model: the system's lifetime and a candidate set that is indicative of the component cause of failure without necessarily precisely identifying the failed component. This kind of masking of the true cause of failure is especially prevalent in industrial settings. We will revisit this idea with a real-world example to demonstrate its significance after introducing some specific theoretical conditions.

The key goal of our analysis is to estimate the parameter $\boldsymbol{\theta}$, which maximizes the likelihood of the observed data, and to estimate the precision and accuracy of this estimate using the Bootstrap method. To achieve this, we first need to assess the joint distribution of the system's continuous lifetime, T_i, and the discrete candidate set, $\mathcal{C}_i$, which can be written as

$$f_{T_i,\mathcal{C}_i}(t_i,c_i;\boldsymbol{\theta}) = f_{T_i}(t_i;\boldsymbol{\theta})\Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i|T_i = t_i\},$$

where $f_{T_i}(t_i;\boldsymbol{\theta})$ is the pdf of T_i and $\Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i|T_i = t_i\}$ is the conditional pmf of $\mathcal{C}_i$ given $T_i = t_i$.

We assume the pdf $f_{T_i}(t_i;\boldsymbol{\theta})$ is known, but we do not have knowledge of $\Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i|T_i = t_i\}$, i.e., the data generating process for candidate sets is unknown. However, it is critical that the masked data, $\mathcal{C}_i$, is correlated with the ith system. This way, the conditional distribution of $\mathcal{C}_i$ given $T_i = t_i$ may provide information about $\boldsymbol{\theta}$, despite our statistical interest being primarily in the series system rather than the candidate sets.

To make this problem tractable, we assume a set of conditions that make it unnecessary to estimate the generative processes for candidate sets. The most important way in which $\mathcal{C}_i$ is correlated with the ith system is given by assuming the following condition.

Condition 1.

Condition 1. The candidate set $\mathcal{C}_i$ contains the index of the failed component, i.e.,

$$\Pr_{\boldsymbol{\theta}}\{K_i \in \mathcal{C}_i\} = 1,$$

where K_i is the random variable for the failed component index of the ith system.

Assuming Condition (@ref(cond:c-contains-k)), $\mathcal{C}_i$ must contain the index of the failed component, but we can say little else about what other component indices may appear in $\mathcal{C}_i$. In order to derive the joint distribution of $\mathcal{C}_i$ and T_i assuming Condition , we take the following approach. We notice that $\mathcal{C}_i$ and K_i are statistically dependent. We denote the conditional pmf of $\mathcal{C}_i$ given $T_i = t_i$ and $K_i = j$ as

$$\Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i|T_i = t_i, K_i = j\}.$$

Even though K_i is not observable in our masked data model, we can still consider the joint distribution of T_i, K_i, and $\mathcal{C}_i$. By Theorem @ref(thm:f-k-and-t), the joint pdf of T_i and K_i is given by

$$f_{T_i,K_i}(t_i,j;\boldsymbol{\theta}) = h_j(t_i;\boldsymbol{\theta_j})\prod_{l=1}^{m} R_l(t_i;\boldsymbol{\theta_l}),$$

where $h_j(t_i;\boldsymbol{\theta_j})$ and $R_j(s_i;\boldsymbol{\theta_j})$ are respectively the hazard and reliability functions of the jth component. Thus, the joint pdf of T_i, K_i, and $\mathcal{C}_i$ may be written as

$$f_{T_i,K_i,\mathcal{C}_i}(t_i,j,c_i;\boldsymbol{\theta}) = f_{T_i,K_i}(t_i,k;\boldsymbol{\theta})\Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i|T_i = t_i, K_i = j\}$$

$(\#eq:joint-pdf-t-k-c)$

$$= h_j(t_i;\boldsymbol{\theta_j})\prod_{l=1}^{m} R_l(t_i;\boldsymbol{\theta_l})\Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i|T_i = t_i, K_i = j\}.$$

We are going to need the joint pdf of T_i and $\mathcal{C}_i$, which may be obtained by summing over the support $\{1,\ldots,m\}$ of K_i in Equation @ref(eq:joint-pdf-t-k-c),

$$f_{T_i,\mathcal{C}_i}(t_i,c_i;\boldsymbol{\theta}) = \prod_{l=1}^{m} R_l(t_i;\boldsymbol{\theta_l})\sum_{j=1}^{m}\left\{h_j(t_i;\boldsymbol{\theta_j})\Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i|T_i = t_i, K_i = j\}\right\}.$$

By Condition , $\Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j\} = 0$ when $K_i = j$ and $j \notin c_i$, and so we may rewrite the joint pdf of T_i and $\mathcal{C}_i$ as

$$(\#eq: part1) f_{T_i, \mathcal{C}_i}(t_i, c_i; \boldsymbol{\theta}) = \prod_{l=1}^{m} R_l(t_i; \boldsymbol{\theta_l}) \sum_{j \in c_i} \left\{ h_j(t_i; \boldsymbol{\theta_j}) \Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j\} \right\}.$$

When we try to find an MLE of $\boldsymbol{\theta}$ (see Section @ref(mle)), we solve the simultaneous equations of the MLE and choose a solution $\hat{\boldsymbol{\theta}}$ that is a maximum for the likelihood function. When we do this, we find that $\hat{\boldsymbol{\theta}}$ depends on the unknown conditional pmf $\Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j\}$. So, we are motivated to seek out more conditions (that approximately hold in realistic situations) whose MLEs are independent of the pmf $\Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j\}$.

Condition 2. Given an observed system failure time $T_i = t_i$ and candidate set c_i, the probability of the candidate set is the same when we condition on any component cause of failure in the candidate set. That is,

$$\Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j'\} = \Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j\}$$

for all $j, j' \in c_i$.

Assuming Conditions and 2, $\Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j\}$ may be factored out of the summation in Equation @ref(eq:part1), and thus the joint pdf of T_i and $\mathcal{C}_i$ may be rewritten as

$$f_{T_i, \mathcal{C}_i}(t_i, c_i; \boldsymbol{\theta}) = \Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j'\} \prod_{l=1}^{m} R_l(t_i; \boldsymbol{\theta_l}) \sum_{j \in c_i} h_j(t_i; \boldsymbol{\theta_j})$$

where $j' \in c_i$.

If $\Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j'\}$ is a function of $\boldsymbol{\theta}$, the MLEs are still dependent on the unknown $\Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j'\}$. This is a more tractable problem, but we are primarily interested in the situation where we do not need to know or estimate $\Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j'\}$ to find an MLE of $\boldsymbol{\theta}$. The last condition we assume achieves this result.

Condition 3. The masking probabilities conditioned on failure time T_i and component cause of failure K_i are not functions of $\boldsymbol{\theta}$. In this case, the conditional probability of $\mathcal{C}_i$ given $T_i = t_i$ and $K_i = j'$ is denoted by

$$\beta_i = \Pr\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j'\},$$

where β_i is not a function of $\boldsymbol{\theta}$.

Real-World Relevance

According to [3], many industrial problems feature masking due to time constraints and the high costs associated with failure analysis. Crucially, these industrial scenarios often fulfill Conditions , 2, and 3, reinforcing the applicability of the results presented in this paper.

To elucidate, let's consider a diagnostic tool used for identifying failed components in an electronic device comprising three critical components arranged in a series configuration. Two are on a common circuit board (labeled 1 and 2), while the third (labeled 3) is separate. Our diagnostic tool isolates the failure to either the circuit board or the individual component but does not differentiate between components 1 and 2 if the failure is on the shared board. In this case, we have the following conditional probabilities for candidate sets:

$$\Pr\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j\} \begin{cases} 1 & \text{if } c_i = \{1, 2\} \text{ and } j = 1 \text{ or } j = 2, \\ 1 & \text{if } c_i = \{3\} \text{ and } j = 3, \\ 0 & \text{otherwise.} \end{cases}$$

Our diagnostic tool satisfies the conditions as follows:

- **Condition** : The candidate set c_i always contains the failed component j. Our diagnostic tool is able to isolate the failure to either the circuit board or the individual component, and so the candidate set always contains the failed component.

- **Condition 2**: As we vary the cause of failure j, we see that the conditional probability of the given candidate set c_i is the same for all $j \in c_i$. Our diagnostic tool cannot distinguish between components 1 and 2 if the shared circuit board is the cause of failure, and therefore the probability of the candidate set is the same when we condition on either component 1 or 2 being the cause of failure.

- **Condition 3**: The probabilities associated with our diagnostic tool are fixed and do not depend on the system parameter $\boldsymbol{\theta}$.

By emphasizing that these conditions hold both in a general industrial context and a specific real-world example, the paper enhances its applicability and relevance to both theoreticians and practitioners.

Likelihood Contribution

When Conditions , 2, and 3 are satisfied, the joint pdf of T_i and $\mathcal{C}_i$ is given by

$$f_{T_i, \mathcal{C}_i}(t_i, c_i; \boldsymbol{\theta}) = \beta_i \prod_{l=1}^{m} R_l(t_i; \boldsymbol{\theta_l}) \sum_{j \in c_i} h_j(t_i; \boldsymbol{\theta_j}).$$

When we fix the sample and allow $\boldsymbol{\theta}$ to vary, we obtain the contribution to the likelihood L from the ith observation when the system lifetime is exactly known ($\delta_i = 1$) but the component cause of failure is masked by a candidate set c_i:

$$(\#eq: likelihood - contribution - masked) L_i(\boldsymbol{\theta}) \propto \prod_{l=1}^{m} R_l(t_i; \boldsymbol{\theta_l}) \sum_{j \in c_i} h_j(t_i; \boldsymbol{\theta_j}),$$

where we dropped the factor β_i since it is not a function of $\boldsymbol{\theta}$.[4]

To summarize this result, assuming Conditions , 2, and 3, if we observe an exact system failure time for the ith system ($\delta_i = 1$), but the component that failed is masked by a candidate set c_i, then its likelihood contribution is given by Equation @ref(eq:likelihood-contribution-masked).

Right-Censored Data

As described in Section @ref(like-model), we observe realizations of $(S_i, \delta_i, \mathcal{C}_i)$ where $S_i = \min\{T_i, \tau_i\}$ is the right-censored system lifetime, $\delta_i = 1_{T_i < \tau_i}$ is the event indicator, and $\mathcal{C}_i$ is the candidate set.

In the previous section, we discussed the likelihood contribution from an observation of a masked component cause of failure, i.e., $\delta_i = 1$. We now derive the likelihood contribution of a *right-censored* observation, $\delta_i = 0$, in our masked data model.

The likelihood contribution of a right-censored observation ($\delta_i = 0$) is given by

$$L_i(\boldsymbol{\theta}) = \prod_{l=1}^{m} R_l(s_i; \boldsymbol{\theta_l}).$$

When right-censoring occurs, then $S_i = \tau_i = s_i$, and we only know that $T_i > s_i$, and so we integrate over all possible values that it may have obtained,

$$L_i(\boldsymbol{\theta}) = \Pr_{\boldsymbol{\theta}}\{T_i > s_i\}.$$

By definition, this is just the survival or reliability function of the series system evaluated at s_i,

$$L_i(\boldsymbol{\theta}) = R_{T_i}(s_i; \boldsymbol{\theta}) = \prod_{l=1}^{m} R_l(s_i; \boldsymbol{\theta_l}).$$

---

[4]When doing maximum likelihood estimation, we are interested in the parameter values that maximize the likelihood function. Since β_i is not a function of $\boldsymbol{\theta}$, it does not affect the location of the maximum of the likelihood function, and so we can drop it from the likelihood function.

When we combine the two likelihood contributions, we obtain the likelihood contribution for the ith system shown in Theorem ,

$$L_i(\boldsymbol{\theta}) \propto \begin{cases} \prod_{l=1}^m R_l(s_i; \boldsymbol{\theta_l}) & \text{if } \delta_i = 0 \\ \prod_{l=1}^m R_l(s_i; \boldsymbol{\theta_l}) \sum_{j \in c_i} h_j(s_i; \boldsymbol{\theta_j}) & \text{if } \delta_i = 1. \end{cases}$$

We use this result in Section @ref(mle) to derive the maximum likelihood estimator (MLE) of $\boldsymbol{\theta}$.

Identifiability and Convergence Issues

In our likelihood model, masking and right-censoring can lead to issues related to identifiability and flat likelihood regions. Identifiability refers to the unique mapping of the model parameters to the likelihood function, and lack of identifiability can lead to multiple sets of parameters that explain the data equally well, making inference about the true parameters challenging [4], while flat likelihood regions can complicate convergence [5].

In our simulation study, we address these challenges in a pragmatic way. Specifically, failure to converge to a solution within a maximum of 125 iterations is interpreted as evidence of the aforementioned issues, leading to the discarding of the sample.[5] However, in Section @ref(boot), where we discuss the bias-corrected and accelerated (BCa) bootstrap method for constructing confidence intervals, we do not discard any resamples. This strategy helps ensure the robustness of the results, while acknowledging the inherent complexities of likelihood-based estimation in models characterized by masking and right-censoring. In our simulation study, we report the convergence rates, and find that for most scenarios, the convergence rate is greater than 95% (100% once the sample is sufficiently large).

Maximum Likelihood Estimation

In our analysis, we use maximum likelihood estimation (MLE) to estimate the series system parameter $\boldsymbol{\theta}$ from the masked data [6], [7]. The MLE finds parameter values that maximize the likelihood of the observed data under the assumed model. A maximum likelihood estimate, $\hat{\boldsymbol{\theta}}$, is a solution of

$$(\#eq: mle) L(\hat{\boldsymbol{\theta}}) = \max_{\boldsymbol{\theta} \in \boldsymbol{\Omega}} L(\boldsymbol{\theta}),$$

where $L(\boldsymbol{\theta})$ is the likelihood function of the observed data. For computational efficiency and analytical simplicity, we work with the log-likelihood function, denoted as $\ell(\boldsymbol{\theta})$, instead of the likelihood function [7].

The log-likelihood function, $\ell(\boldsymbol{\theta})$, for our masked data model is the sum of the log-likelihoods for each observation,

$$(\#eq: loglike) \ell(\boldsymbol{\theta}) = \sum_{i=1}^n \ell_i(\boldsymbol{\theta}),$$

where $\ell_i(\boldsymbol{\theta})$ is the log-likelihood contribution for the ith observation:

$$\ell_i(\boldsymbol{\theta}) = \sum_{j=1}^m \log R_j(s_i; \boldsymbol{\theta_j}) + \delta_i \log\left(\sum_{j \in c_i} h_j(s_i; \boldsymbol{\theta_j})\right).$$

The log-likelihood function is the logarithm of the likelihood function,

$$\ell(\boldsymbol{\theta}) = \log L(\boldsymbol{\theta}) = \log \prod_{i=1}^n L_i(\boldsymbol{\theta}) = \sum_{i=1}^n \log L_i(\boldsymbol{\theta}).$$

---

[5]The choice of 125 iterations was also made for practical reasons. Since we are generating millions of samples and trying to find an MLE for each in our simulation study, if we did not limit the number of iterations, the simulation study would have taken too long to run.

Substituting $L_i(\boldsymbol{\theta})$ from Equation @ref(eq:like), we consider these two cases of δ_i separately to obtain the result in Theorem @ref(thm:loglike-total). **Case 1**: If the ith system is right-censored ($\delta_i = 0$),

$$\ell_i(\boldsymbol{\theta}) = \log \prod_{l=1}^{m} R_l(s_i; \boldsymbol{\theta_l}) = \sum_{l=1}^{m} \log R_l(s_i; \boldsymbol{\theta_l}).$$

Case 2: If the ith system's component cause of failure is masked but the failure time is known ($\delta_i = 1$),

$$\ell_i(\boldsymbol{\theta}) = \sum_{l=1}^{m} \log R_l(t_i; \boldsymbol{\theta_l}) + \log \beta_i + \log\left(\sum_{j \in c_i} h_j(s_i; \boldsymbol{\theta_j})\right).$$

By Condition 3, we may discard the $\log \beta_i$ term since it does not depend on $\boldsymbol{\theta}$, giving us the result

$$\ell_i(\boldsymbol{\theta}) = \sum_{l=1}^{m} \log R_l(s_i; \boldsymbol{\theta_l}) + \log\left(\sum_{j \in c_i} h_j(s_i; \boldsymbol{\theta_j})\right).$$

Combining these two cases gives us the result in Theorem @ref(thm:loglike-total).

The MLE, $\hat{\boldsymbol{\theta}}$, is often found by solving a system of equations derived from setting the derivative of the log-likelihood function to zero [6], i.e.,

$$(\#eq:mle-eq)\frac{\partial}{\partial \theta_j}\ell(\boldsymbol{\theta}) = 0,$$

for each component θ_j of the parameter $\boldsymbol{\theta}$. When there's no closed-form solution, we resort to numerical methods like the Newton-Raphson method.

Assuming some regularity conditions, such as the likelihood function being identifiable, the MLE has many desirable asymptotic properties that underpin statistical inference, namely that it is an asymptotically unbiased estimator of the parameter $\boldsymbol{\theta}$ and it is normally distributed with a variance given by the inverse of the Fisher Information Matrix (FIM) [7]. However, for smaller samples, these asymptotic properties may not yield accurate approximations. We propose to use the bootstrap method to offer an empirical approach for estimating the sampling distribution of the MLE, in particular for computing confidence intervals.

Bias-Corrected and Accelerated Bootstrap Confidence Intervals

We utilize the non-parametric bootstrap to estimate the sampling distribution of the MLE. In the non-parametric bootstrap, we resample from the observed data with replacement to generate a bootstrap sample. The MLE is then computed for the bootstrap sample. This process is repeated to generate numerous bootstrap replicates of the MLE. The sampling distribution of the MLE is then estimated by the empirical distribution of the bootstrap replicates of the MLE.

Our main focus is on generating confidence intervals for the MLE. The method we use to generate confidence intervals is known as Bias-Corrected and Accelerated Bootstrap Confidence Intervals (BCa) [8], which applies two corrections to the standard bootstrap method:

- **Bias Correction**: This adjusts for bias in the bootstrap distribution itself. This bias is measured as the difference between the mean of the bootstrap distribution and the observed statistic. It works by transforming the percentiles of the bootstrap distribution to correct for these issues.
  This may be a useful adjustment in our case since we are dealing with small samples with two potential sources of bias: right-censoring and masking component cause of failure.

- **Acceleration**: This adjusts for the rate of change of the statistic as a function of the true, unknown parameter. This correction is important when the shape of the statistic's distribution changes with the true parameter. Since we have a number of different shape parameters, $k_1, \ldots, k_m$, we may expect the shape of the distribution of the MLE to change as a function of the true parameter, making this correction potentially useful.

Correctly Specified Confidence Intervals   Since we are primarily interested in generating confidence intervals for small samples for a biased MLE, the BCa method is a reasonable choice for our simulation study. In our simulation study, we evaluate the performance of the BCa confidence intervals by calculating their coverage probability. A *correctly specified* 95% confidence interval contains the true parameter value approximately 95% of the time in repeated sampling.

In this study, we consider a coverage probability above 90% to be satisfactory, as it offers a reasonable trade-off between precision and reliability. A coverage probability below this would signify undue confidence in the precision of the MLE. Conversely, a coverage probability near 100% may indicate excessively wide intervals, thereby diminishing the precision of the MLE. Our objective is to construct confidence intervals that are as narrow as possible while achieving good empirical coverage close to the nominal level of 95%.

Challenges   While the bootstrap method provides a robust and flexible tool for statistical estimation, its effectiveness can be influenced by many factors. A few of these factors are particularly relevant to our study:

- **Convergence**: Instances of non-convergence in our bootstrap samples were observed, which can occur when the estimation method, like the MLE used in our analysis, fails to converge due to the specifics of the resampled data [7]. This issue can potentially introduce bias or reduce the effective sample size of our bootstrap distribution.
- **Small Samples**: The bootstrap's accuracy can be compromised with small sample sizes, as the method relies on the Law of Large Numbers to approximate the true sampling distribution. For small samples, the bootstrap resamples might not adequately represent the true variability in the data, leading to inaccurate results [9].
- **Masking**: Our data involves right censoring and a masking of the component cause of failure when a system failure is observed. These aspects can cause certain data points or trends to be underrepresented or not represented at all in our data, introducing bias in the bootstrap distribution [10].

Despite these challenges, we found the bootstrap method useful in constructing correctly specified confidence intervals.

Series System with Weibull Components

The Weibull distribution, introduced by Waloddi Weibull in 1937, has been instrumental in reliability analysis due to its ability to model a wide range of failure behaviors. Reflecting on its utility, Weibull modestly noted that it "[...] may sometimes render good service." [11] In the context of our study, we model a system as originating from components with Weibull distributed lifetimes arranged in a series configuration, producing a specific form of the likelihood model described in Section @ref(like-model), which deals with challenges such as right censoring and masked component cause of failure.

The jth component of the ith system has a lifetime distribution given by

$$T_{ij} \sim \text{Weibull}(k_j, \lambda_j) \qquad \text{for } i = 1, \ldots, n \text{ and } j = 1, \ldots, m,$$

where $\lambda_j > 0$ is the scale parameter and $k_j > 0$ is the shape parameter. The jth component has a reliability function, pdf, and hazard function given respectively by

$$R_j(t; \lambda_j, k_j) = \exp\left\{ -\left(\frac{t}{\lambda_j} \right)^{k_j} \right\}, \tag{1}$$

$$f_j(t; \lambda_j, k_j) = \frac{k_j}{\lambda_j} \left(\frac{t}{\lambda_j} \right)^{k_j - 1} \exp\left\{ -\left(\frac{t}{\lambda_j} \right)^{k_j} \right\}, \tag{2}$$

$$h_j(t; \lambda_j, k_j)(\#eq : weibull - haz) = \frac{k_j}{\lambda_j} \left(\frac{t}{\lambda_j} \right)^{k_j - 1}. \tag{3}$$

The shape parameter of the Weibull distribution is of particular importance:

- $k_j < 1$ indicates infant mortality. An example of how this might arise is a result of defective components being weeded out early, and the remaining components surviving for a much longer time.
- $k_j = 1$ indicates random failures (independent of age). An example of how this might arise is a result of random shocks to the system, but otherwise the system is age-independent.[6]
- $k_j > 1$ indicates wear-out failures. An example of how this might arise is a result of components wearing as they age.

We show that the lifetime of the series system composed of m Weibull components has a reliability, hazard, and probability density functions given by the following theorem.

The lifetime of a series system composed of m Weibull components has a reliability function, hazard function, and pdf respectively given by

$$(\#eq : sys - weibull - reliability - fn) R_{T_i}(t; \boldsymbol{\theta}) = \exp\left\{ -\sum_{j=1}^{m} \left(\frac{t}{\lambda_j} \right)^{k_j} \right\}, \tag{4}$$

$$(\#eq : sys - weibull - failure - rate - fn) h_{T_i}(t; \boldsymbol{\theta}) = \sum_{j=1}^{m} \frac{k_j}{\lambda_j} \left(\frac{t}{\lambda_j} \right)^{k_j - 1}, \tag{5}$$

$$(\#eq : sys - weibull - pdf) f_{T_i}(t; \boldsymbol{\theta}) = \left\{ \sum_{j=1}^{m} \frac{k_j}{\lambda_j} \left(\frac{t}{\lambda_j} \right)^{k_j - 1} \right\} \exp\left\{ -\sum_{j=1}^{m} (\frac{t}{\lambda_j})^{k_j} \right\}, \tag{6}$$

where $\boldsymbol{\theta} = (k_1, \lambda_1, \ldots, k_m, \lambda_m)$ is the parameter vector of the series system and $\boldsymbol{\theta_j} = (k_j, \lambda_j)$ is the parameter vector of the jth component.

The proof for the reliability function follows from Theorem @ref(thm:sys-reliability-fn),

$$R_{T_i}(t; \boldsymbol{\theta}) = \prod_{j=1}^{m} R_j(t; \lambda_j, k_j).$$

Plugging in the Weibull component reliability functions obtains the result

$$R_{T_i}(t; \boldsymbol{\theta}) = \prod_{j=1}^{m} \exp\left\{ -\left(\frac{t}{\lambda_j} \right)^{k_j} \right\} = \exp\left\{ -\sum_{j=1}^{m} \left(\frac{t}{\lambda_j} \right)^{k_j} \right\}.$$

The proof for the hazard function follows from Theorem @ref(thm:sys-failure-rate),

$$h_{T_i}(t; \boldsymbol{\theta}) = \sum_{j=1}^{m} h_j(t; \boldsymbol{\theta_j}) = \sum_{j=1}^{m} \frac{k_j}{\lambda_j} \left(\frac{t}{\lambda_j} \right)^{k_j - 1}.$$

The proof for the pdf follows from Theorem @ref(thm:sys-pdf). By definition,

$$f_{T_i}(t; \boldsymbol{\theta}) = h_{T_i}(t; \boldsymbol{\theta}) R_{T_i}(t; \boldsymbol{\theta}).$$

Plugging in the failure rate and reliability functions given respectively by Equations @ref(eq:sys-weibull-reliability-fn) and @ref(eq:sys-weibull-failure-rate-fn) completes the proof.

In Section @ref(reliability), we discussed the concept of reliability, with the MTTF being a common measure of reliability. In the case of Weibull components, the MTTF of the jth component is given by

$$(\#eq : mttf - weibull) \mathrm{MTTF}_j = \lambda_j \Gamma\left(1 + \frac{1}{k_j} \right),$$

where Γ is the gamma function. We mentioned that the MTTF can sometimes be a poor measure of reliability, e.g., the MTTF and the probability of failing early can *both* be large. The Weibull is a good example of this phenomenon. If $k_j > 1$, the lifetime distribution of the jth component is fat-tailed and it can exhibit both a large MTTF and a high probability of failing early. The probability of component failure given by Equation @ref(eq:prob-k) is a particularly useful measure of component reliability relative to the other components in the system.

---

[6]The exponential distribution is a special case of the Weibull distribution when $k_j = 1$.

Likelihood Model

In Section @ref(like-model), we discussed two separate kinds of likelihood contributions, masked component cause of failure data (with exact system failure times) and right-censored data. The likelihood contribution of the ith system is given by the following theorem.

Let δ_i be an indicator variable that is 1 if the ith system fails and 0 (right-censored) otherwise. Then the likelihood contribution of the ith system is given by

$$(\#eq:weibull-likelihood-contribution) L_i(\boldsymbol{\theta}) \propto \begin{cases} \exp\left\{-\sum_{j=1}^{m}\left(\frac{t_i}{\lambda_j}\right)^{k_j}\right\} \sum_{j \in c_i} \frac{k_j}{\lambda_j}\left(\frac{t_i}{\lambda_j}\right)^{k_j-1} & \text{if } \delta_i = 1, \\ \exp\left\{-\sum_{j=1}^{m}\left(\frac{t_i}{\lambda_j}\right)^{k_j}\right\} & \text{if } \delta_i = 0. \end{cases}$$

By Theorem @ref(thm:likelihood-contribution), the likelihood contribution of the ith system is given by

$$L_i(\boldsymbol{\theta}) \propto \begin{cases} R_{T_i}(s_i; \boldsymbol{\theta}) & \text{if } \delta_i = 0 \\ R_{T_i}(s_i; \boldsymbol{\theta}) \sum_{j \in c_i} h_j(s_i; \boldsymbol{\theta_j}) & \text{if } \delta_i = 1. \end{cases}$$

By Equation @ref(eq:sys-weibull-reliability-fn), the system reliability function is given by

$$R_{T_i}(t_i; \boldsymbol{\theta}) = \exp\left\{-\sum_{j=1}^{m}\left(\frac{t_i}{\lambda_j}\right)^{k_j}\right\},$$

where $\boldsymbol{\theta} = (k_1, \lambda_1, \ldots, k_m, \lambda_m)$ is the parameter vector and by Equation @ref(eq:weibull-haz), the hazard function of the jth component is given by

$$h_j(t_i; \boldsymbol{\theta_j}) = \frac{k_j}{\lambda_j}\left(\frac{t_i}{\lambda_j}\right)^{k_j-1},$$

where $\boldsymbol{\theta_j} = (k_j, \lambda_j)$ is the parameter vector of the jth component. Plugging these into the likelihood contribution function obtains the result.

Taking the log of the likelihood contribution function obtains the following result.

The log-likelihood contribution of the ith system is given by

$$(\#eq:weibull-log-likelihood-contribution)\ell_i(\boldsymbol{\theta}) = -\sum_{j=1}^{m}\left(\frac{t_i}{\lambda_j}\right)^{k_j} + \delta_i \log\left(\sum_{j \in c_i} \frac{k_j}{\lambda_j}\left(\frac{t_i}{\lambda_j}\right)^{k_j-1}\right)$$

where we drop any terms that do not depend on $\boldsymbol{\theta}$ since they do not affect the MLE.

See Appendix @ref(app-weibull-loglik-r) for the R code that implements the log-likelihood function for the series system with Weibull components.

We find an MLE by solving @ref(eq:mle-eq), i.e., a point $\hat{\boldsymbol{\theta}} = (\hat{k}_1, \hat{\lambda}_1, \ldots, \hat{k}_m, \hat{\lambda}_m)$ satisfying $\nabla_{\boldsymbol{\theta}} \ell(\hat{\boldsymbol{\theta}}) = \mathbf{0}$, where $\nabla_{\boldsymbol{\theta}} \ell$ is the gradient of the log-likelihood function (score) with respect to $\boldsymbol{\theta}$. To solve this system of equations, we use Newton-like methods, which sometimes require both the gradient and the Hessian of the log-likelihood function. We analytically derive the score but we do not do the same for the Hessian of the log-likelihood function. Our reasoning is based on the following two observations:

- The score function is relatively easy to derive, and it is useful to have for computing gradients efficiently and accurately, which will be useful for accurately numerically approximating the Hessian of the log-likelihood function.

- The Hessian is tedious and error prone to derive, and Newton-like methods often do not require the Hessian to be explicitly computed.

The following theorem derives the score function.

The gradient of the log-likelihood contribution of the ith system is given by

$$(\#eq:weibull-score)\nabla\ell_i(\boldsymbol{\theta}) = \left(\frac{\partial\ell_i(\boldsymbol{\theta})}{\partial k_1}, \frac{\partial\ell_i(\boldsymbol{\theta})}{\partial\lambda_1}, \cdots, \frac{\partial\ell_i(\boldsymbol{\theta})}{\partial k_m}, \frac{\partial\ell_i(\boldsymbol{\theta})}{\partial\lambda_m}\right)',$$

where

$$\frac{\partial\ell_i(\boldsymbol{\theta})}{\partial k_r} = -\left(\frac{t_i}{\lambda_r}\right)^{k_r}\log\left(\frac{t_i}{\lambda_r}\right) + \frac{\frac{1}{t_i}\left(\frac{t_i}{\lambda_r}\right)^{k_r}\left(1 + k_r\log\left(\frac{t_i}{\lambda_r}\right)\right)}{\sum_{j\in c_i}\frac{k_j}{\lambda_j}\left(\frac{t_i}{\lambda_j}\right)^{k_j-1}}1_{\delta_i=1\wedge r\in c_i}$$

and

$$\frac{\partial\ell_i(\boldsymbol{\theta})}{\partial\lambda_r} = \frac{k_r}{\lambda_r}\left(\frac{t_i}{\lambda_r}\right)^{k_r} - \frac{\left(\frac{k_r}{\lambda_r}\right)^2\left(\frac{t_i}{\lambda_r}\right)^{k_r-1}}{\sum_{j\in c_i}\frac{k_j}{\lambda_j}\left(\frac{t_i}{\lambda_j}\right)^{k_j-1}}1_{\delta_i=1\wedge r\in c_i}.$$

The result follows from taking the partial derivatives of the log-likelihood contribution of the ith system given by Equation @ref(eq:weibull-likelihood-contribution). It is a tedious calculation so the proof has been omitted, but the result has been verified by using a very precise numerical approximation of the gradient.

By the linearity of differentiation, the gradient of a sum of functions is the sum of their gradients, and so the score function conditioned on the entire sample is given by

$$(\#eq:weibull-series-score)\nabla\ell(\boldsymbol{\theta}) = \sum_{i=1}^{n}\nabla\ell_i(\boldsymbol{\theta}).$$

Weibull Series System: Homogeneous Shape Parameters

In a series system, the system is only as reliable as its weakest link (weakest component). In a well-designed series system, there is no single component that is much weaker than the others. In the case of components with Weibull lifetimes, this implies the shape parameters are homogenous and the scale parameters are homogenous. The shape parameters are particularly important since they determine the failure behavior of the components.

When the shape parameters are homogenous, the lifetime of the series system with components that are Weibull distributed is also Weibull distributed, as shown in the following theorem.

If the shape parameters of the components are homogenous, then the lifetime series system follows a Weibull distribution with a shape parameter k given by the identical shape parameters of the components and a scale parameter λ given by

$$(\#eq:sys-weibull-scale)\lambda = \left(\sum_{j=1}^{m}\lambda_j^{-k}\right)^{-1/k},$$

where λ_j is the scale parameter of the jth component.

Given m Weibull lifetimes $T_{i1}, \ldots, T_{im}$ with the same shape parameter k and scale parameters $\lambda_1, \ldots, \lambda_m$, the reliability function of the series system is given by

$$R_{T_i}(t;\boldsymbol{\theta}) = \exp\left\{-\sum_{j=1}^{m}\left(\frac{t}{\lambda_j}\right)^{k}\right\}.$$

To show that the series system lifetime is Weibull, we need to find a single scale parameter λ such that

$$R_{T_i}(t;\boldsymbol{\theta}) = \exp\left\{-\left(\frac{t}{\lambda}\right)^{k}\right\},$$

which has the solution

$$\lambda = \frac{1}{\left(\frac{1}{\lambda_1^k} + \ldots + \frac{1}{\lambda_m^k}\right)^{\frac{1}{k}}}.$$

If a series system has Weibull components with homogeneous shape parameters, the component cause of failure is conditionally independent of the system failure time:

$$\Pr\{K_i = j | T_i = t_i\} = \Pr\{K_i = j\} = \frac{\lambda_j^{-k}}{\sum_{l=1}^{m} \lambda_l^{-k}}.$$

By Theorem @ref(thm:prob-k-given-t), the conditional probability of the jth component being the cause of failure given the system failure time is given by

$$\Pr\{K_i = j | T_i = t\} = \frac{f_{K_i, T_i}(j, t; \boldsymbol{\theta})}{f_{T_i}(t; \boldsymbol{\theta})} = \frac{h_j(t; k, \lambda_j) R_{T_i}(t; \boldsymbol{\theta})}{h_{T_i}(t; \boldsymbol{\theta_j}) R_{T_i}(t; \boldsymbol{\theta})}$$

$$= \frac{h_j(t; k, \lambda_j)}{\sum_{l=1}^{m} h_l(t; k, \lambda_l)} = \frac{\frac{k}{\lambda_j} \left(\frac{t}{\lambda_j}\right)^{k-1}}{\sum_{l=1}^{m} \frac{k}{\lambda_l} \left(\frac{t}{\lambda_l}\right)^{k-1}} = \frac{\left(\frac{1}{\lambda_j}\right)^k}{\sum_{l=1}^{m} \left(\frac{1}{\lambda_l}\right)^k}.$$

According to the bias-variance trade-off, we expect the MLE of the homogenous model, which has $m + 1$ parameters (m being the number of components in the series system), to be more biased but have less variance than the MLE of the full model, which has $2m$ parameters.

Simulation Study: Series System with Weibull Components

In this simulation study, we assess the sensitivity of the MLE and BCa confidence intervals to various simulation scenarios for the likelihood model defined in Section @ref(weibull). We begin by specifying the parameters of the series system that will be the central object of our simulation study. We consider the data in [12], in which they study the reliability of a series system with three components. They fit Weibull components in a series configuration to the data, resulting in an MLE with shape and scale estimates given by the first three components in Table 2. To make the model slightly more complex, we add two more components to this series system, with shape and scale parameters given by the last two components in Table 2. We will refer to this system as the **base** system.

In Section @ref(reliability), we defined a well-designed series system as one that consists of components with similar reliabilities, where we define reliability in three ways: the reliability function, MTTF, and probability that a specific component will be the cause of failure (which is a measure of relative reliability of the components). We will use these three measures of reliability to assess the base system. The base system defined in Table 2 satisfies this definition of being a well-designed system since there are no components that are significantly less reliable than any of the others, component 1 being the most reliable and component 3 being the least reliable.

The reliability function, unlike the other two measures of reliability, is not a summary statistic of reliability, but is rather a function of time. Since most of our simulations have the right-censoring time set to the 82.5% quantile of the series system, which we denote here by $\tau_{0.825}$, we can compare the reliability functions of the components at this time. We see that the reliability of the components at this right-censoring time are similar, with component 1 being the most reliable and component 3 being the least reliable. These results are consistent with the previous analysis based on the MTTF and probability of component cause of failure being similar.

The shape parameters for each component is larger than 1, which means each component has a failure characteristic that is more wear-out than infant mortality. The system as a whole is therefore more likely to fail due to wear-out failures than infant mortality. This too is consistent with a well-designed system.

Homogenous Shape Parameters The base system is a well-designed system, and so it is likely that the likelihood model that assumes homogeneous shape parameters described in Section @ref(reduced-weibull) would provide a good fit to any data generated from this system. We performed a preliminary investigation into this by simulating data from the base system, and deviations from the base system that make the system less well-designed, and fitting the homogeneous model to the data. We found that the MLE of the homogeneous model was very close to the true parameter values for slight deviations from the base system, but the MLE was biased for larger deviations from the base system. This is consistent with the bias-variance

trade-off, where the MLE of the homogeneous model is more biased but has less variance than the MLE of the full model. We do not explore this further in this simulation study, but it is an interesting avenue for future research.

Table 2: Weibull Components in Series Configuration

	Shape (k_j)	Scale (λ_j)	MTTF_j	$\Pr\{K_i = j\}$	$R_j(\tau_{0.825}; k_j, \lambda_j)$
Component 1	1.2576	994.3661	924.869	0.169	0.744
Component 2	1.1635	908.9458	862.157	0.207	0.698
Component 3	1.1308	840.1141	803.564	0.234	0.667
Component 4	1.1802	940.1342	888.237	0.196	0.711
Component 5	1.2034	923.1631	867.748	0.195	0.711
Series System	NA	NA	222.884	NA	0.175

Performance Metrics

In this section, we describe the measures we use to assess the performance of the MLE and the BCa confidence intervals in our simulation study. We assess two important properties of the MLE for each simulation scenario:

- **Accuracy (Bias)**: A pivotal metric is the proximity of the MLE's expected value to the true parameter value. Higher accuracy is demonstrated by a closer proximity. We estimate the accuracy in our simulation study by plotting the mean of the MLE.

- **Precision**: Another essential measure is the variability of the MLE across samples. Higher precision is demonstrated by smaller variability. We estimate this in our simulation study by plotting the quantiles of the sampling distribution of the MLE.

In parallel, we assess the following properties of the BCa confidence intervals described in Section @ref(boot):

- **Accuracy (Coverage Probability)**: In Section @ref(boot), we discussed the importance of the coverage probability of the confidence intervals. The coverage probability is the proportion of the computed confidence intervals that contain the true parameter values. Ideally, we would like the CIs to contain the true parameter values around 95% of the time, which is the nominal coverage probability. In this case, the CIs are said to be correctly specified, but we consider anything over 90% satisfactory.

- **Precision**: A useful measure of the precision of the confidence intervals are their widths. Narrower, consistent intervals across samples indicate higher precision. We measure this in our simulation study by plotting the *median-aggregated* CIs, where we aggregate the CIs by taking the medians of the upper and lower bounds. This gives us a sense of the central tendency of the CIs. There is a trade-off between accuracy and precision, where we want the CIs to be as narrow as possible with nominal coverage probability.

If the confidence intervals are not accurate, then we cannot be sure that the intervals contain the true parameters. If the confidence intervals are not precise, then there is significant uncertainty in the locations of the true parameters. In either case, we cannot be confident in the results of the analysis, and we should consider alternative methods for estimating the parameters of the series system. However, we will see that the BCa confidence intervals are accurate and precise for most of the simulation scenarios we consider, providing a reliable method for estimating the parameters of the series system.

Finally, we also consider the convergence rate of the MLE, discussed in Section @ref(identifiability). We take a convergence rate less than 95% to be evidence that the MLE is not a robust or reliable estimator in these cases, due to the insufficient information.

Data Generating Process

In this section, we describe the data generating process for our simulation studies. It consists of three parts: the series system, the candidate set model, and the right-censoring model.

Series System Lifetime   We generate data from a series system with $m = 5$ components with Weibull lifetimes. As described in Section @ref(weibull), the jth component of the ith system has a lifetime distribution given by

$$T_{ij} \sim \text{Weibull}(k_j, \lambda_j)$$

and the lifetime of the series system composed of m Weibull components is defined as

$$T_i = \min\{T_{i1}, \ldots, T_{im}\}.$$

To generate a data set, we first generate the m component failure times, by efficiently sampling from their respective distributions, and we then set the failure time t_i of the system to the minimum of the component failure times.

Right-Censoring Model   We employ a simple right-censoring model, where the right-censoring time is fixed at some known value, e.g., an experiment is run for a fixed amount of time τ, and all systems that have not failed by the end of the experiment are right-censored. The censoring time S_i of the ith system is thus given by

$$S_i = \min\{T_i, \tau\}.$$

So, after we generate the system failure time T_i, we generate the censoring time S_i by taking the minimum of T_i and τ.

In the simulation study, we parameterize instead by a right-censoring quantile q for the series system, where q is the proportion of systems that are expected to fail before the right-censoring time, denoted by τ_q. This is given by

$$\tau_q = F_{T_i}^{-1}(q; \boldsymbol{\theta}),$$

where $F_{T_i}^{-1}$ is the inverse CDF of the series system. To solve for the right-censoring time τ_q of the series system, we define a function g as

$$g(\tau_q) = F_{T_i}(\tau_q; \boldsymbol{\theta}) - q$$

and find its root using the Newton's method. See Appendix @ref(app-series-quantile) for the R code that implements this procedure.

For instance, if $q = 0.825$ (which is what we set it to in some of simulation scenarios), then 82.5% of the series systems are expected to fail before the corresponding right-censoring time τ_q and 17.5% of the systems are expected to be right-censored.

Masking Model for Component Cause of Failure   We must generate data that satisfies the masking conditions described in Section @ref(candmod). There are many ways to satisfy the masking conditions. We choose a simple method, which we call the *Bernoulli masking model*. In this model, we satisfy Conditions @ref(cond:c-contains-k), @ref(cond:equal-prob-failure-cause), and @ref(cond:masked-indept-theta) by generating a candidate set c_i for each system i as follows:

- If the jth component fails, it is deterministically placed in the candidate set. This satisfies Condition @ref(cond:c-contains-k), $\Pr\{K_i \in \mathcal{C}_i\} = 1$.

- For each of the $m - 1$ components that did not fail, we generate a Bernoulli random variable X_j with probability p of being 1, where p is a fixed probability. If $X_j = 1$, the jth component is placed in the candidate set, which satisfies Condition @ref(cond:masked-indept-theta), since the Bernoulli random variables are not a function of $\boldsymbol{\theta}$.

21

- Condition @ref(cond:equal-prob-failure-cause) may be the least intuitive of the three conditions. It states that
$$\Pr\{\mathcal{C}_i = c_i | K_i = j, T_i = t_i\} = \Pr\{\mathcal{C}_i = c_i | K_i = j', T_i = t_i\}$$
for all $j, j' \in c_i$. In words, this means that the probability of the candidate set $\mathcal{C}_i$ being equal to some set c_i is the same when conditioned on any component in c_i being the cause of failure and the system failure time t_i. This is satisfied by the Bernoulli masking model since, first, it is independent of the system failure time t_i, and second, the probability that a non-failed component is in the candidate set is fixed at some constant p for all components. To see this, consider the following example. Suppose we have a system with $m = 3$ components, and the first component is the cause of failure. Then the probability of the candidate set being equal to some set c_i is given by

$$\Pr\{\mathcal{C}_i = c_i | K_i = 1, T_i = t_i\} = \begin{cases} (1-p)^2 & \text{if } c_i = \{1\} \\ p(1-p) & \text{if } c_i = \{1,2\} \\ (1-p)p & \text{if } c_i = \{1,3\} \\ p^2 & \text{if } c_i = \{1,2,3\}, \end{cases} \tag{7}$$

where a non-failed component is in the candidate set with probability p and otherwise it is not placed in the candidate set with probability $1 - p$. Now, let's change the component cause of failure to the second component:

$$\Pr\{\mathcal{C}_i = c_i | K_i = 2, T_i = t_i\} = \begin{cases} (1-p)^2 & \text{if } c_i = \{2\} \\ p(1-p) & \text{if } c_i = \{1,2\} \\ (1-p)p & \text{if } c_i = \{2,3\} \\ p^2 & \text{if } c_i = \{1,2,3\}. \end{cases} \tag{8}$$

The same pattern holds for the third component. We see that the probability of the candidate set being equal to some set c_i is the same for all components in that c_i and for all system failure times t_i, e.g., if the probability that $c_i = \{1,2\}$ is $p(1-p)$ when we condition in either component 1 or component 2 being the cause of failure, which satisfies Condition @ref(cond:equal-prob-failure-cause).

There are many more ways to satisfy the masking conditions, but we choose the Bernoulli masking model because it is simple to understand and allows us to easily vary the masking probability p for our simulation study.

See Appendix @ref(app-cand-model-r) for the R code that implements this model.

Overview of Simulations

We define a simulation scenario to be some combination of n (sample size), p (masking probability in our Bernoulli masking model), and q (right-censoring quantile). We are interested in choosing a small number of scenarios that are representative of real-world scenarios and that are interesting to analyze. For how we run a simulation scenario, see Appendix @ref(app-sim-study-r), but here is an outline of the process:

1. **Parameter Initialization**: Fix a combination of simulation parameters to some value, and vary the remaining parameters. For example, if we want to assess how the sampling distribution of the MLE changes with respect to sample size, we might choose some particular values for p and q and then vary the sample size n over the desired range.

2. **Data Generation**: Simulate $R \geq 300$ datasets from the Data Generating Process (DGP) described in Section @ref(data-gen-proc).

3. **MLE Computation**: Compute an MLE for each of the R datasets.

4. **Statistical Evaluation**: For each of these R MLEs, compute some function of the MLE, like the BCa confidence intervals. This will give us R statistics as a Monte-carlo estimate of the sampling distribution of the statistic.

5. **Distribution Property Estimation**: Use the R statistics to estimate some property of the sampling distribution of the statistic, e.g., the mean of the MLE or the coverage probability of the BCa confidence intervals, with respect to the parameter we are varying in the scenario, e.g., assess how the coverage probability of the BCa confidence intervals changes with respect to sample size.

In this study, we are focusing on three distinct scenarios. Section @ref(effect-censoring) explores how varying the right-censoring affects the estimator with the masking and sample size fixed. Section @ref(p-vs-mttf) explores how varying the masking affects the estimator with the right-censoring and sample size fixed. Section @ref(effect-samp-size) explores how varying the sample size affects the estimator with the right-censoring and masking fixed. Each scenario aims to provide insights into how these parameters influence the behavior of MLEs, which is crucial for understanding their performance in real-world applications.

Scenario: Assessing the Impact of Right-Censoring

In this scenario, we use the well-designed series system described in Table 2 and we use the following simulation scenario values:

- We vary the right-censoring quantile (q) from 60% to 100% (no right-censoring). We denote the corresponding right-censoring time by τ_q.
- We fix the Bernoulli masking probability p to 21.5% based on estimates from masked data.[7]
- We fix the sample size n to 90, which is small enough to show the impact of right-censoring on the MLE, but large enough to obtain reasonable convergence rates.

In Figure 2, we show the effect of right-censoring on the MLEs for the shape and scale parameters. The top four plots only show the effect on the MLEs for the shape and scale parameters of components 1 and 3. We chose these components because they are the most and least reliable components, respectively, and so we expect them to be the most and least sensitive to right-censoring. The bottom two plots show the coverage probabilities for all parameters.

Background

When a right-censoring event occurs, in order to increase the likelihood of the data, the MLE is nudged in a direction that increases the probability of a right-censoring event at time τ, which is given by $R_{T_i}(t; \boldsymbol{\theta})$, representing a source of bias in the estimate.

To increase $R_{T_i}(\tau_q)$, we move in the direction (gradient) of these partial derivatives. The partial derivatives of $R_{T_i}(\tau_q)$ are given by

$$\frac{\partial R_{T_i}(\tau_q)}{\partial \lambda_j} = R_{T_i}(\tau_q; \boldsymbol{\theta}) \left(\frac{\tau}{\lambda_j} \right)^{k_j} \frac{k_j}{\lambda_j},$$

$$\frac{\partial R_{T_i}(\tau_q)}{\partial k_j} = R_{T_i}(\tau_q; \boldsymbol{\theta}) \left(\frac{\tau}{\lambda_j} \right)^{k_j} (\log \lambda_j - \log \tau_q),$$

for $j = 1, \ldots, m$. We see that these partial derivatives are related to the score of a right-censored likelihood contribution in Theorem @ref(thm:weibull-score). Let us analyze the implications these partial derivatives have on the MLE:

- **Effect of Increasing Right-Censoring Quantile**: As the right-censoring quantile q increases (τ_q increases), $R_{T_i}(\tau_q; \boldsymbol{\theta})$ decreases, reducing the impact of right-censoring on the MLE. This behavior is evident in Figure 2.

- **Positive Bias in Scale Parameters**: The partial derivatives with respect to the scale parameters are always positive. This means that right-censoring introduces a positive bias in the scale parameter estimates, making right-censoring events more likely. The extent of this bias is related to the amount of right-censoring $(1 - q)$, as seen in Figure 2.

---

[7]See 'Table 2: Example Data for a Series System' in [12] for details on the masked data.
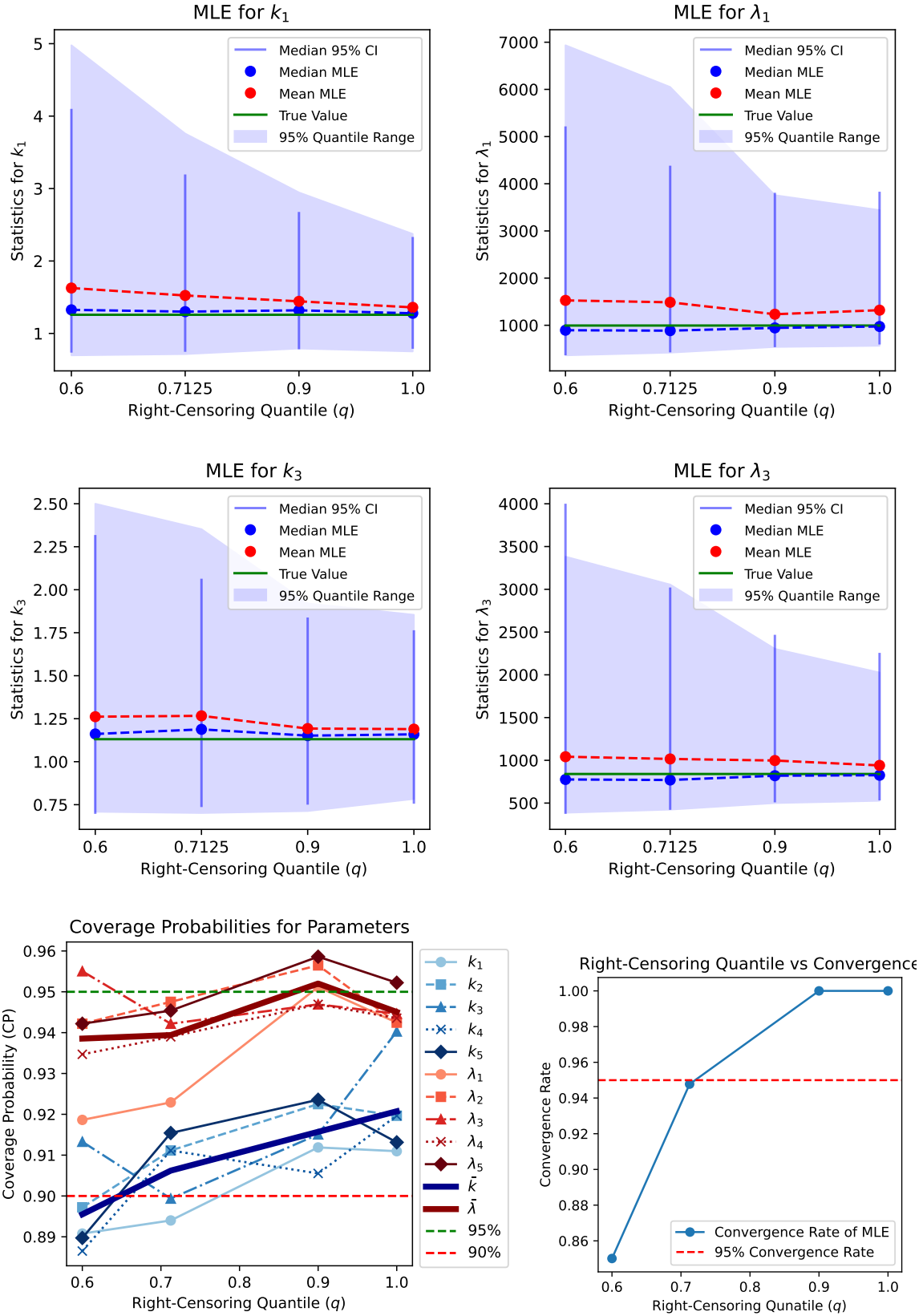
Figure 2: Right-Censoring Quantile (q) vs MLE ($p = 0.215, n = 90$)

- **Conditional Bias in Shape Parameters**: The partial derivative with respect to the shape parameter of the jth component, k_j, is non-negative if $\lambda_j \geq \tau_q$ and otherwise negative. In our well-designed series system, the scale parameters are large compared to most of the right-censoring times τ_q, so the MLE nudges the shape parameter estimates in a positive direction to increase the probability $R_{T_i}(\tau_q)$ of a right-censoring event at time τ_q. We see this in Figure 2, where the shape parameter estimates are positively biased for most of the quantiles q.

Key Observations

Coverage Probability (CP) The confidence intervals are generally correctly specified, obtaining coverages above 90% for most of the parameters across the entire range of right-censoring quantiles, and they are converging to the nominal 95% level as the right-censoring quantile increases. This suggests that the bootstrapped CIs will contain the true value of the parameters with the specified confidence level with high probability. The CIs are neither too wide nor too narrow.

However, the scale parameters are better calibrated than the shape parameters. The scale parameters are consistently around the nominal 95% level for all right-censoring quantiles, but the shape parameters are consistently less correctly specified, suggesting that the shape parameters are more difficult to estimate than the scale parameters.

We also see that the coverage probabilities for k_1 and λ_1 generally have worse coverage than the other parameters. This is likely due to the fact that component 1 is the most reliable component, and so it is less likely to fail before the right-censoring time τ_q. This means that the likelihood contribution of component 1 is less informative than the other components. Conversely, we see that k_3 and λ_3 generally have better coverage than the other parameters. This is likely due to the fact that component 3 is the least reliable component, and so it is more likely to fail before the right-censoring time. This means that the likelihood contribution of component 3 is more informative than the other components in the presence of right-censoring.

Dispersion of MLEs The shaded regions representing the 95% probability range of the MLEs get narrower as the right-censoring quantile increases. This is an indicator of the increased precision in the estimates as more data is available due to decreased censoring.

We see that the dispersion of the MLEs for k_1 and λ_1 are much larger than the dispersion of the MLEs for k_3 and λ_3. This is consistent with the previous analysis for the coverage probabilities.

Median-Aggregated CIs The median CI (vertical blue bars) decreases in length as the right-censoring quantile increases. This suggests that the bootstrapped CIs become more consistent and centered around a tighter range as the right-censoring quantile increases, while maintaining a good coverage probability. As right-censoring events become less likely, the bootstrapped CIs gravitate closer to each other and the true parameter values.

For small right-censoring quantiles, the CIs are quite large, which was necessary to maintain good coverage. The estimator is sensitive to the data, and so the bootstrapped CIs are wide to account for this sensitivity when the sample contains insufficient information due to censoring. Again, we see that the median-aggregated CIs for k_1 and λ_1 are much wider than the median-aggregated CIs for k_3 and λ_3.

Bias of MLEs The red dashed line indicating the mean of MLEs initially is quite biased for the shape parameters, but quickly diminishes to negligible levels as the right-censoring quantile increases. The bias for the shape parameters never reach zero, but this is potentially due to masking.

The bias for the scale parameters is quite small and remains stable across different right-censoring quantiles, suggesting that the scale MLEs are reasonably unbiased. It could be the case that masking or other factors are counteracting the bias due to right-censoring.

Again, we see that the bias for k_1 and λ_1 is much greater than the bias for k_3 and λ_3, which is consistent with the previous analysis.

Convergence Rate The convergence rate increases as the right-censoring quantile q increases. This is consistent with the expectation that more censoring reduces the information in a sample, making the likelihood function less informative and more difficult to identify the parameters that maximize it. We see

that the convergence rate is around 95% or greater for right-censoring quantiles $q \geq 0.7$, but drops below 95% for $q < 0.7$.

Summary

In this scenario, we find that right-censoring significantly influences the MLEs of shape and scale parameters. As the degree of right-censoring decreases, the precision of these estimates improves, and the bias in the shape parameters diminishes, although it never fully disappears, possibly due to masking effects. Additionally, Bootstrapped (BCa) confidence intervals generally show good coverage probabilities, particularly for scale parameters, and become more focused as right-censoring decreases. As expected, the estimates for the most reliable component are more sensitive to right-censoring than the least reliable component. We also find that the convergence rate of the MLE decreases as the degree of right-censoring increases, suggesting that the MLE is less reliable in these cases.

These insights set the stage for the subsequent scenario on masking probability, where we will explore how masking adds another layer of complexity to parameter estimation and how these effects might be mitigated.

Scenario: Assessing the Impact of Masking Probability for Component Cause of Failure

In this scenario, we use the well-designed series system described in Table 2 and we use the following simulation scenario values:

- We vary the Bernoulli masking probability p from 10% to 70%.[8]
- We fix the right-censoring quantile q to 82.5%, which means the probability of a right-censoring event is 17.5%. This is less than the masking probability, making the masking probability the dominant source of uncertainty.
- We fix the sample size to n to 90, which is small enough to show the impact of masking on the MLE, but large enough to obtain reasonable convergence rates.

In Figure 3, we show the effect of the masking probability p on the MLE for the shape and scale parameters. The top four plots only show the effect on the MLE for the shape and scale parameters of components 1 and 3, since the rest demonstrated similar results. The bottom left plot shows the coverage probabilities for all parameters and the bottom right plot shows the convergence rate of the MLE.

Background

Masking introduces a layer of complexity that is different from right-censoring. While right-censoring deals with the uncertainty in the timing of failure, masking adds ambiguity in identifying which component actually failed. In the Bernoulli masking model, the failed component is guaranteed to be in the candidate set and each non-failed component is included with a probability p. This has the following implications on the MLE:

- **Ambiguity**: A higher p makes larger candidate sets more probable, making it less clear which parameter estimates should be adjusted to make the data more likely in the MLE. This is particularly problematic for components that are not the cause of failure, since the MLE will adjust their parameters to make them more likely to be the cause of failure, which is not necessarily correct.

- **Bias**: In an ideal scenario, knowing which component failed would allow the MLE to make that component's failure more likely and a right-censoring effect would be applied to the non-failed components. However, a larger masking probability p introduces uncertainty, causing the MLE to adjust the estimates for the parameters of the non-failed components to be more likely to fail at the observed failure time while simultaneously applying a right-censoring effect to the other components, including the failed component. This introduces a bias similar to the bias introduced by right-censoring, and the greater the masking probability p, the greater the bias.

---

[8]The probability that the component cause of failure is masked is the probability that at least one functional component will be in the candidate set, which is given by $1 - (1-p)^{m-1}$, where m denotes the number of components in the system and p is the Bernoulli masking probability. So, for $m = 5$ and p ranges from 10% to 70%, the probability that the component cause of failure is masked ranges from 34.4% to 99.2%.
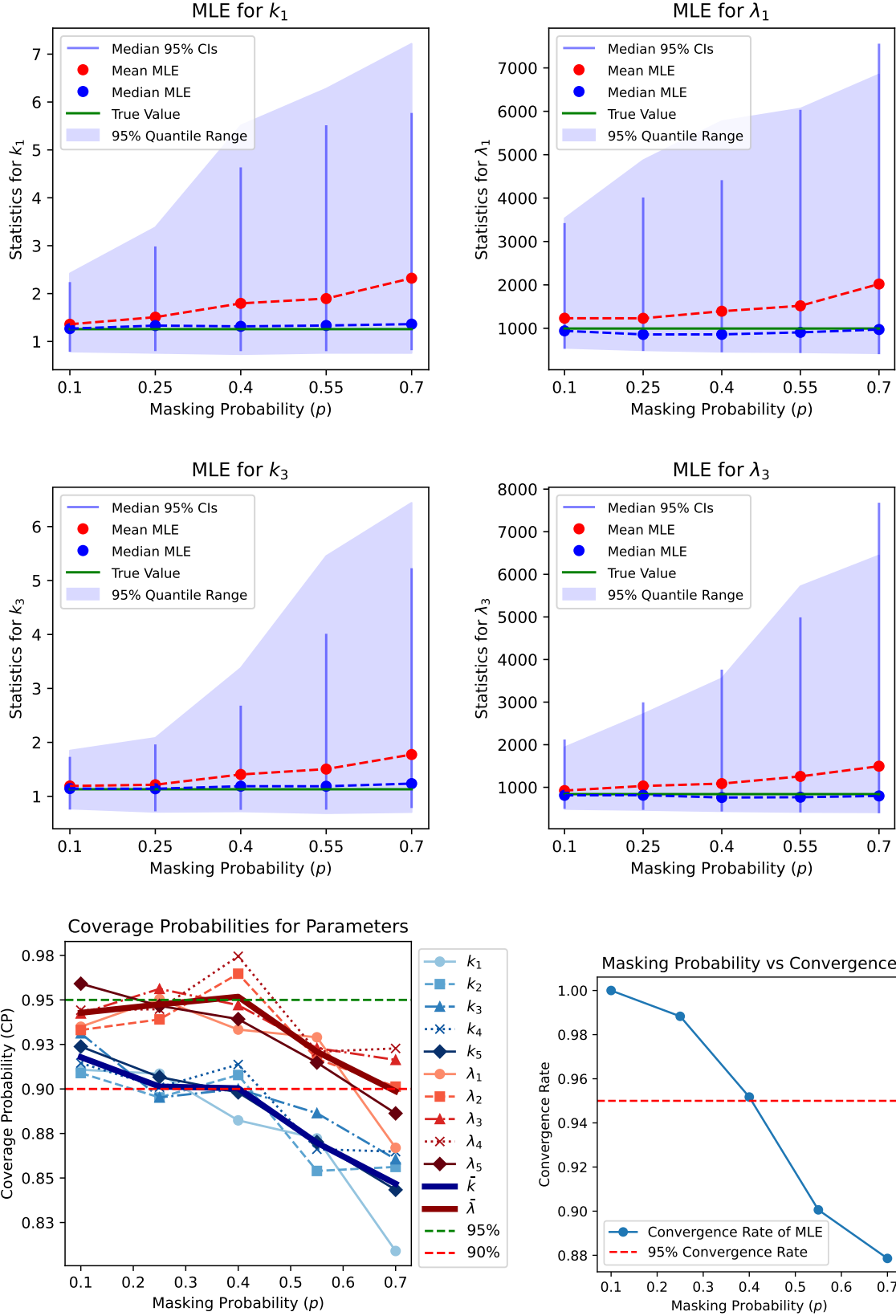
Figure 3: Component Cause of Failure Masking (p) vs MLE $(q = 0.825, n = 90)$

- **Precision**: As the masking probability p increases, the likelihood function becomes less informative, reducing the precision of the estimates.

Key Observations

Coverage Probability (CP) For the scale parameters, the 95% CI is correctly specified for Bernoulli masking probabilities up to $p = 0.7$, which is quite significant, obtaining coverages over 90%. For the shape parameters, the 95% CI is correctly specified for masking probabilities only up to $p = 0.4$, which is still large. At $p = 0.4$, the coverage probability is around 90, but continues to drop after that point well below 90%. This suggests that the shape parameters are more difficult to estimate than the scale parameters, which is consistent with the previous scenario where we varied the amount of right-censoring.

The BCa confidence intervals are correctly specified for most realistic masking probabilities, constructing CIs that are neither too wide nor too narrow, but when the masking is severe and the sample size is small, one should take the CIs with a grain of salt.

Dispersion of MLEs The shaded regions representing the 95% quantile of the MLEs become wider as the masking probability increases. This is an indicator of the decreased precision in the estimates when provided with more ambiguous data about the component cause of failure.

Median-Aggregated CIs The median-aggregated CIs (vertical dark blue bars) show that the BCa CIs are becoming more spread out as the masking probability increases. They are also asymmetric, with the lower bound being more spread out than the upper-bound, which is consistent with the behavior of the dispersion of the MLEs. The width of the CIs consistently increase as the masking probability increases, which we intuitively expected given the increased uncertainty about the component cause of failure.

Bias of MLEs The mean of the MLE (red dashed lines) demonstrates a steadily increasing positive bias as the masking probability increases. This is consistent with the expectation that the MLE will apply an increasing right-censoring effect to the estimates as the masking probability increases.

Convergence Rate The convergence rate decreases as the Bernoulli masking probability p increases. This is consistent with the expectation that a higher masking probability decreases the information the sample contains about the parameters. We see that the convergence rate remains above 95% for masking probabilities up to $p = 0.4$, but drops below 95% for $p > 0.4$. This is consistent with behavior of the CPs, which drop below 90% for $p > 0.4$.

Summary

In this scenario, we examine the influence of masking probability p on the MLE, keeping the sample size and right-censoring constant. As the masking probability increases, the precision of the MLEs decreases, and the coverage probability of the CIs begins to drop. However, even at fairly significant Bernoulli masking probabilities, particularly for the scale parameters, the CIs have good coverage. These observations highlight the challenges of parameter estimation under varying degrees of masking and set the stage for the subsequent scenario on sample size, which shows how increasing the sample size can mitigate the effects of both right-censoring and masking.

Scenario: Assessing the Impact of Sample Size

In this scenario, we want to assess how the sample size can mitigate the affects of right-censoring and masking previously discussed. We use the well-designed series system described in Table 2 and we use the following simulation scenario values:

- We vary the sample size n from 50 (small sample size) to 500 (large sample size).
- We fix the right-censoring quantile q to 82.5%.
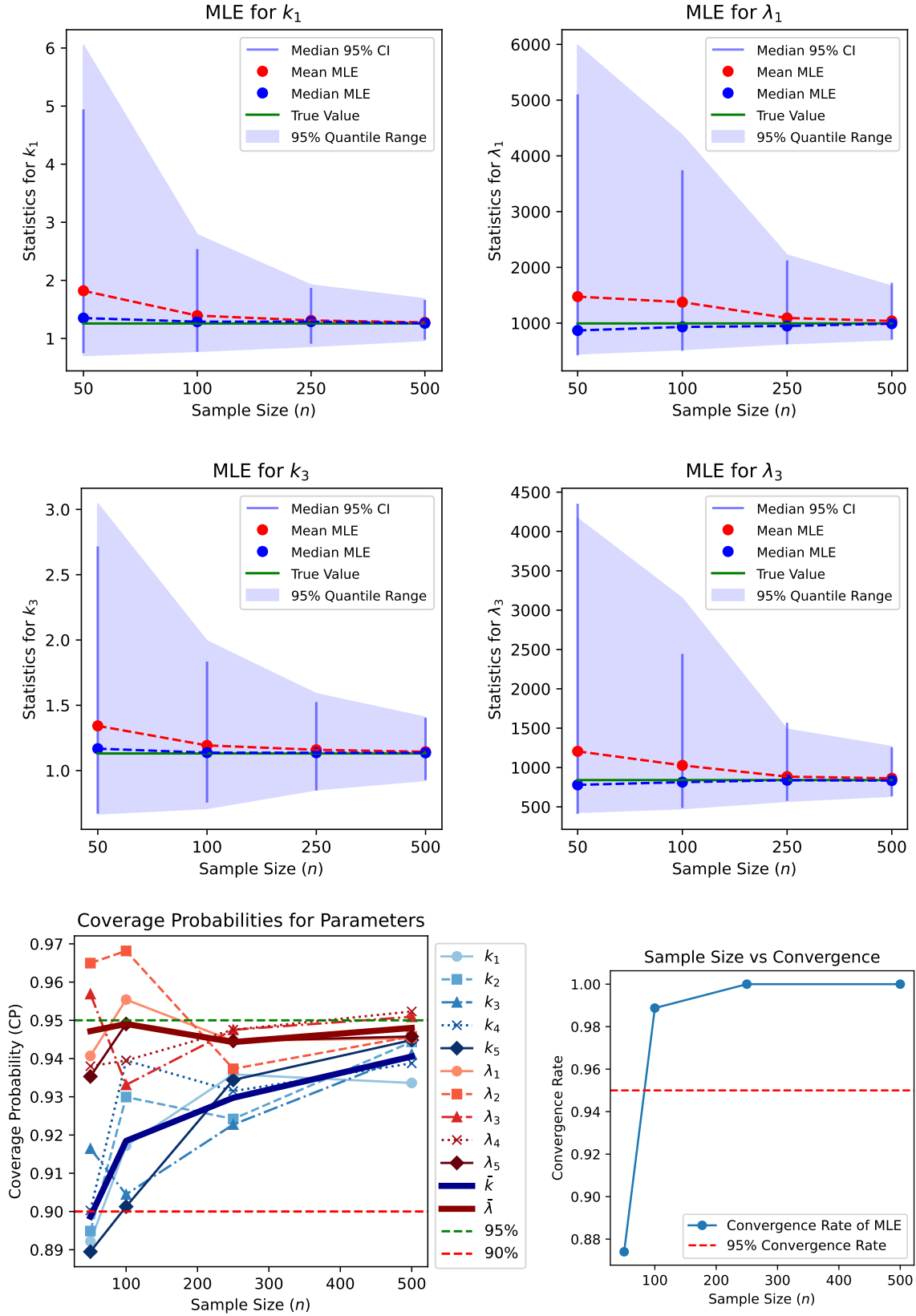- We fix the masking probability p to 21.5%.

Figure 4: Sample Size (n) vs MLEs $(p = 0.215, q = 0.825)$

In Figure 4, we show the effect of the sample size n on the MLEs for the shape and scale parameters. The top four plots only show the effect on the MLEs for the shape and scale parameters of components 1 and 3, component 1 being the most reliable component in the system and component 3 being the least reliable component. The bottom left plot shows the coverage probabilities of the confidence intervals for all parameters and the bottom right plot shows the convergence rate of the MLE.

Key Observations

Coverage Probability (CP) The confidence intervals are correctly specified, obtaining coverages above 90% for most of the parameters across the entire sample size range, and they are converging to the nominal 95% level as the sample size increases. The BCa CIs contain the true value of the parameters with the specified confidence level, and the CIs neither too wide nor too narrow.

However, as in the previous scenario where we varied the right-censoring amount, the scale parameters have better coverage than the shape parameters. The scale parameters are consistently around the nominal 95% level for all sample sizes, but the shape parameters lag behind, suggesting that the shape parameters are more difficult to estimate than the scale parameters.

Dispersion of MLEs The shaded regions representing the 95% quantile range of the MLEs become more narrow as the sample size increases. This is an indicator of the increased precision in the estimates when provided with more data.

We also see that the dispersion of the MLEs for k_1 and λ_1 are much larger than the dispersion of the MLEs for k_3 and λ_3. This is consistent with the previous analysis in the right-censoring scenario.

Median-Aggregated CIs The median-aggregated CIs (vertical dark blue bars) show that the CIs are becoming less spread out as the sample size increases, indicating that they are becoming more consistent and centered around a tighter range while maintaining good coverage probability. The end result is that we can construct more precise and accurate CIs with larger samples and thus we can make more confident inferences about the true parameter value.

The estimator is quite sensitive to the data, and so the CIs are wide to account for this sensitivity when the sample size is small and not necessarily representative of the true distribution.

We also observe that the upper bound of the CI is more spread out than the lower bound. This is consistent with the behavior of the dispersion of the MLEs, which have a positive bias. Thus, the CIs are accounting for this bias by being more spread out in the direction of the bias.

Bias of MLEs The mean of the MLE (red dashed lines) initially has a large positive bias, but diminishes to negligible levels as the sample size increases. In the previous right-censoring scenario, the bias never reached zero, but we see that in this scenario, at around a sample size of 250, the estimator is essentially unbiased, suggesting that there is enough information in the sample to overcome the bias from the right-censoring ($q = 0.825$) and masking ($p = 0.215$) effects.

Convergence Rate The convergence rate increases as the sample size n increases. This is consistent with the expectation that more data provides more information about the parameters, making the likelihood function more informative and easier to identify the parameters that maximize it. We see that the convergence rate is 95% or greater for sample sizes $n \geq 100$ given moderate right-censoring and masking. Given how quickly the convergence rate increased, we anticipate that even for extreme censoring and masking, the convergence rate would likely rapidly increase to over 95% as the sample size increases. However, for sample sizes $n < 100$, at least for series systems with $m = 5$ components with Weibull lifetimes, the convergence rate is low. For small samples, any estimates should be taken with a grain of salt.

Summary

In this scenario, we delve into the mitigating effects of sample size on the challenges identified in previous scenarios concerning right-censoring and masking. The precision and accuracy of the MLE rapidly improves with a bias approaching zero. This is consistent with statistical theory and suggests that increasing the

sample size can mitigate the effects of right-censoring and masking. The BCa CIs also narrow and become more reliable (with coverage probabilities approaching the nominal 95% level) with larger samples, reinforcing the role of sample size in achieving robust estimates.

Scenario: Model Selection via Likelihood Ratio Test

The previous simulation scenarios focused on estimating parameters under the full heterogeneous model. In practice, analysts must also decide whether the reduced homogeneous-shape model is appropriate. Section established that when components share a common shape parameter, the series system lifetime is itself Weibull-distributed, simplifying inference. Here we investigate the likelihood ratio test (LRT) as a formal procedure for selecting between these models.

Test Setup

The LRT compares the full model $\mathcal{M}_F$ with $2m$ parameters (separate shapes $k_1, \ldots, k_m$ and scales $\lambda_1, \ldots, \lambda_m$) against the reduced model $\mathcal{M}_R$ with $m+1$ parameters (common shape k and scales $\lambda_1, \ldots, \lambda_m$). The test statistic is

$$\Lambda = 2\big[\ell(\hat{\boldsymbol{\theta}}_F) - \ell(\hat{\boldsymbol{\theta}}_R)\big],$$

where $\ell(\cdot)$ denotes the log-likelihood and $\hat{\boldsymbol{\theta}}_F$, $\hat{\boldsymbol{\theta}}_R$ are the MLEs under each model. Under the null hypothesis $H_0 : k_1 = \cdots = k_m$, the statistic Λ asymptotically follows a chi-squared distribution with $m - 1$ degrees of freedom [17].

For the simulations in this section, we use the well-designed series system described in Table 2 with parameters:

- Sample sizes $n \in \{100, 500, 1000, 5000, 10000\}$

- Right-censoring quantile $q = 0.825$ (approximately 17.5% censored)

- Bernoulli masking probability $p = 0.215$

- Number of Monte Carlo replicates $R = 500$

Divergence from Homogeneity

To assess both Type I error control and power, we conducted simulations across varying levels of shape heterogeneity, quantified by the coefficient of variation (CV) of the shape parameters:

$$\text{CV} = \frac{\sigma_k}{\bar{k}} = \frac{\sqrt{\frac{1}{m}\sum_{j=1}^{m}(k_j - \bar{k})^2}}{\frac{1}{m}\sum_{j=1}^{m} k_j},$$

where $\bar{k} = \frac{1}{m}\sum_{j=1}^{m} k_j$ is the mean shape parameter. At CV $= 0$, all shapes are identical (H_0 true), and the rejection rate should equal the nominal significance level $\alpha = 0.05$. At CV > 0, the shapes differ (H_0 false), and the rejection rate measures power.

Figure 5 displays the LRT rejection rate as a function of shape heterogeneity for various sample sizes. The results reveal several important patterns:

Type I Error Control At CV $= 0$ (perfect homogeneity), the rejection rates across all sample sizes fall within 95% confidence intervals of the nominal $\alpha = 0.05$, confirming that the LRT maintains proper Type I error control. For example, at $n = 1000$, the rejection rate is exactly 0.050, while at $n = 100$ it is 0.066 with a 95% CI of [0.044, 0.088] that contains 0.05.

Power Analysis The test exhibits excellent power to detect heterogeneity, with power increasing in both the degree of heterogeneity (CV) and sample size. Table 3 summarizes the minimum CV required to achieve 80% power at each sample size. With $n = 5000$ observations, even modest heterogeneity (CV $\approx 8\%$) is detected with high probability. For smaller samples ($n = 100$), substantial heterogeneity (CV $\approx 41\%$) is required for reliable detection.
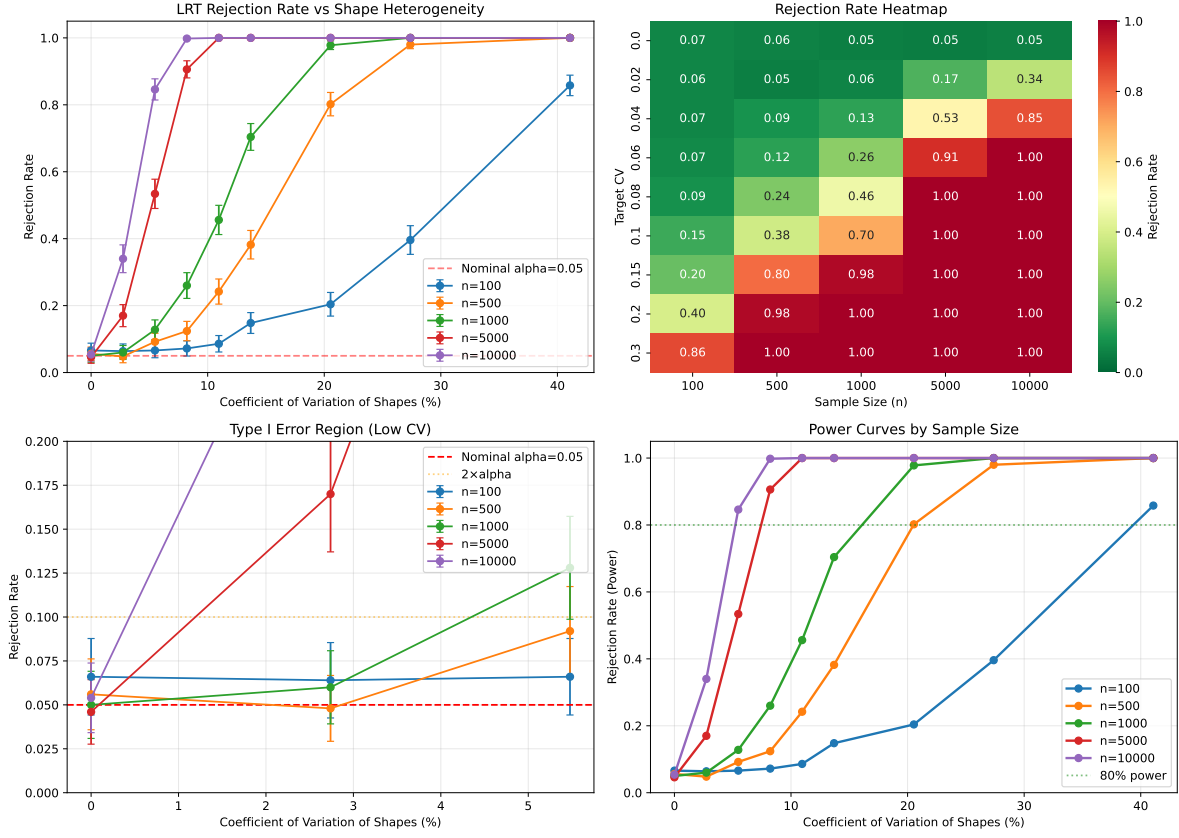
Figure 5: LRT rejection rate versus shape heterogeneity (CV of shape parameters). Top left: Power curves showing rejection rate versus CV for each sample size. Top right: Heatmap of rejection rates. Bottom left: Type I error region (low CV). Bottom right: Power curves with 80% power threshold.

Table 3: Minimum coefficient of variation (CV) of shape parameters required to achieve 80% power at each sample size.

Sample Size (n)	Minimum CV for 80% Power
100	41.1%
500	20.5%
1,000	20.5%
5,000	8.2%
10,000	5.5%

Robustness to Data Quality Parameters

We also examined whether masking probability, censoring level, and system complexity affect the LRT's Type I error control. For each scenario, we simulated data under H_0 (homogeneous shapes) and computed rejection rates.

Masking Probability   Varying the masking probability p from 0.05 to 0.60 (corresponding to average candidate set sizes from 1.2 to 3.4 components), the LRT maintains Type I error near the nominal level across sample sizes. Higher masking reduces information about the true failure cause, but does not systematically inflate or deflate the Type I error rate.

Censoring Level   Similarly, varying the censoring quantile q from 0.50 to 1.00 (corresponding to 0% to 50% censored observations), the test remains well-calibrated. More censoring reduces the effective sample size but does not bias the test.

Number of Components   Increasing the number of components m from 2 to 8 changes the degrees of freedom from 1 to 7. The LRT adapts appropriately, with Type I error remaining near 0.05 across system complexities.

   Table 4 summarizes representative Type I error rates across these robustness scenarios at $n = 1000$, demonstrating that the test maintains appropriate calibration under varied conditions.

Table 4: LRT Type I error rates under various data quality conditions ($n = 1000$, $\alpha = 0.05$). Values in brackets are 95% confidence intervals.

Scenario	Parameter	Rejection Rate
Divergence at CV = 0 (perfect homogeneity)		
	$n = 100$	0.066 [0.044, 0.088]
	$n = 1000$	0.050 [0.031, 0.069]
	$n = 10000$	0.054 [0.034, 0.074]
Varying masking (n = 1000)		
	$p = 0.05$	0.118 [0.090, 0.146]
	$p = 0.215$	0.086 [0.061, 0.111]
	$p = 0.60$	0.060 [0.039, 0.081]
Varying censoring (n = 1000)		
	$q = 0.50$	0.054 [0.034, 0.074]
	$q = 0.825$	0.106 [0.079, 0.133]
	$q = 1.00$	0.098 [0.072, 0.124]
Varying components (n = 1000)		
	$m = 3$	0.178 [0.145, 0.211]
	$m = 5$	0.102 [0.075, 0.129]
	$m = 8$	0.078 [0.055, 0.101]

   These robustness results indicate that the chi-squared approximation for the LRT is reliable across a range of practical conditions, supporting its use as a model selection tool for masked series system data.

Summary

The likelihood ratio test provides a statistically rigorous approach to model selection between full and reduced Weibull series system models. The test maintains correct Type I error control under the null hypothesis and has good power to detect meaningful departures from shape homogeneity, particularly for moderate to large sample sizes. For practitioners, this means that if the LRT fails to reject the homogeneous model, one can proceed with the simpler reduced model with confidence. Conversely, a significant test result indicates that the full heterogeneous model is needed to adequately describe the data.

Future Work

This paper developed maximum likelihood techniques and simulation studies to estimate component reliability from masked failure data in series systems. The key results were:

- The likelihood model enabled rigorous inference from masked data via right-censoring and candidate sets.
- Despite masking and censoring, the MLE demonstrated accurate and robust performance in simulation studies.
- BCa confidence intervals had good coverage probability even for small samples.
- Estimation of shape parameters was more challenging than scale parameters.
- The likelihood ratio test provides a well-calibrated procedure for selecting between full and reduced models, with good power to detect meaningful departures from shape homogeneity.

Building on these findings, next we consider promising areas for future work.

Relaxing Masking Conditions To further generalize the likelihood model, we can relax conditions @ref(cond:c-contains-k), @ref(cond:equal-prob-failure-cause), and @ref(cond:masked-indept-theta) on the masking model. We could perform a sensitivity analysis to violations of these conditions, which would provide insights into the robustness of the likelihood model, or we could develop alternative likelihood models that are less stringent.

Deviations from Well-Designed Series Systems Assess the sensitivity of the estimator to deviations from well-designed series systems, in which there are no components that are significantly more likely to fail than others. We could vary the probabilities of component failure (while keeping the system reliability constant) to assess how the estimator behaves when the system deviates from the well-designed system criteria. We did some preliminary investigation of this, and we found that the estimator was quite sensitive to deviations in system design.

Model Selection Extensions Section established that the likelihood ratio test provides a well-calibrated procedure for selecting between the full and reduced models. Future work could extend this by comparing the LRT with information criteria such as AIC and BIC, which penalize model complexity differently and may be preferred in certain applications. Additionally, the bias-variance trade-off between models could be quantified more precisely by examining estimation accuracy for specific reliability functions under each model choice.

Semi-Parametric Bootstrap We used the non-parametric bootstrap to construct confidence intervals, but we could also investigate the semi-parametric bootstrap. In the semi-parametric bootstrap, instead of resampling from the original data, we sample component lifetimes from the parametric distribution fitted to the original data and sample candidate sets from the conditional empirical distribution of the candidate sets in the original data. This is a compromise between the non-parametric bootstrap and the fully parametric bootstrap.[9]

Regularization Methods Investigate regularization methods like data augmentation and penalized likelihood to improve parameter estimates, in particular shape parameter estimates.

Extending the Likelihood Model with Predictors Our research is based on a general likelihood model. A straightforward extension is to integrate predictors, allowing each observation's hazard function to be influenced by specific variables, such as in the Cox proportional hazards model [13], but any function that satisfies the fundamental mathematical principles of a hazard function could be used. This would allow the likelihood model to be more flexible and adaptable to a wider range of applications.

---

[9]The fully parametric bootstrap is not appropriate for our likelihood model because we do not assume a parametric form for the distribution of the candidate sets.

Bootstrapped Prediction Intervals In this paper, we applied the bootstrap method to construct confidence intervals for the parameter $\boldsymbol{\theta}$, which generated correctly specified confidence intervals in our simulation study. We could do a similar analysis for other statistics, in particular prediction intervals, which are useful for predicting the reliability of a system or a component at a future time.

The current results provide a solid foundation for extensions like these that can further refine the methods and expand their applicability. By leveraging the rigorous likelihood framework and simulation techniques validated in this study, future work can continue advancing the capability for statistical learning from masked reliability data.

Conclusion

This work presented maximum likelihood techniques to estimate component reliability from masked failure data in series systems. The methods demonstrated accurate and robust performance despite significant challenges introduced by masking and right-censoring.

Simulation studies revealed that for our well-designed series system with Weibull component lifetimes, right-censoring and masking positively biased the estimates and the more reliable components were more sensitive to these effects. Additionally, the shape parameters were more difficult to estimate than the scale parameters. However, with sufficiently large sample sizes, these difficulties were overcome, suggesting enough information existed in the data to mitigate censoring and masking effects.

Despite the challenges, the bootstrapped bias-corrected and accelerated confidence intervals were generally correctly specified with good overall coverage probabilities, even for relatively small sample sizes. This good empirical coverage demonstrates the reliability of these intervals for statistical inference, striking a balance between precision and robustness. The modeling framework offers a statistically rigorous method for estimating latent component properties based on limited observational data concerning system reliability. The simulation studies validate these techniques and provide practical insights into their efficacy under diverse real-world scenarios. This enhances our capacity for statistical learning from obscured system failure data.

Appendices

(APPENDIX) Appendix

Series System with Weibull Component Lifetimes

These functions are implemented in the R library `wei.series.md.c1.c2.c3`, which is available on GitHub at `github.com/queelius/wei.series.md.c1.c2.c3` [14]. They are for series systems with Weibull component lifetimes with masked data described in Section @ref(like-model). For clarity and brevity, we removed some of the functionality and safeguards in the actual code, but we provide the full code in the R package.

Log-likelihood Function

The log-likelihood function is the sum of the log-likelihood contributions for each system. For our series system with Weibull component lifetimes, we analytically derived the log-likelihood function in Theorem @ref(thm:weibull-likelihood-contribution) and implemented it in the `loglik_wei_series_md_c1_c2_c3` R function below.

```
#' Generates a log-likelihood function for a series system with Weibull
#' component lifetimes for masked data.
#'
#' @param df masked data frame
#' @param theta parameter vector
#' @returns log-likelihood function
```

```r
loglik_wei_series_md_c1_c2_c3 <- function(df, theta) {
  n <- nrow(df)
  C <- md_decode_matrix(df, "x")
  m <- ncol(C)
  delta <- df[["delta"]]
  shapes <- theta[seq(1, length(theta), 2)]
  scales <- theta[seq(2, length(theta), 2)]
  t <- df[[lifetime]]
  s <- 0
  for (i in 1:n) {
    s <- s - sum((t[i] / scales)^shapes)
    if (delta[i]) {
      s <- s + log(sum(shapes[C[i, ]] / scales[C[i, ]] *
        (t[i] / scales[C[i, ]])^(shapes[C[i, ]] - 1)))
    }
  }
  return(s)
}
```

Score Function

The score function is the gradient of the log-likelihood function. For our series system with Weibull component lifetimes, we analytically derived the score function in Theorem @ref(thm:weibull-score) and implemented it in the `score_wei_series_md_c1_c2_c3` R function below.

```r
#' Computes the score function for a series system with Weibull component
#' lifetimes for masked data.
#'
#' @param df masked data frame
#' @param theta parameter vector
#' @returns score function
score_wei_series_md_c1_c2_c3 <- function(df, theta) {
  n <- nrow(df)
  C <- md_decode_matrix(df, "x")
  m <- ncol(C)
  delta <- df[["delta"]]
  t <- df[[lifetime]]
  shapes <- theta[seq(1, length(theta), 2)]
  scales <- theta[seq(2, length(theta), 2)]
  shapes.scr <- scales.scr <- rep(0, m)

  for (i in 1:n) {
    shapes.rt <- -(t[i] / scales)^shapes * log(t[i] / scales)
    scales.rt <- (shapes / scales) * (t[i] / scales)^shapes
    shapes.trm <- scales.trm <- rep(0, m)

    if (delta[i]) {
      c <- C[i, ]
      denom <- sum(shapes[c] / scales[c] *
        (t[i] / scales[c])^(shapes[c] - 1))

      shapes.num <- (t[i] / scales[c])^shapes[c] / t[i] *
        (1 + shapes[c] * log(t[i] / scales[c]))
```

```r
      shapes.trm[c] <- shapes.num / denom

      scales.num <- (shapes[c] / scales[c])^2 *
        (t[i] / scales[c])^(shapes[c] - 1)
      scales.trm[c] <- scales.num / denom
    }

    shapes.scr <- shapes.scr + shapes.rt + shapes.trm
    scales.scr <- scales.scr + scales.rt - scales.trm
  }

  scr <- rep(0, length(theta))
  scr[seq(1, length(theta), 2)] <- shapes.scr
  scr[seq(2, length(theta), 2)] <- scales.scr
  return(scr)
}
```

Quantile Function

For our series system with Weibull component lifetimes, the quantile function is the inverse of the cdf F_{T_i}. By definition, the quantile p for the strictly monotonically increasing cdf F_{T_i} is the value t that satisfies $F_{T_i}(t; \boldsymbol{\theta}) - p = 0$, and so we solve for t using Newton's method, in which the kth iteration is given by

$$t^{(k+1)} = t^{(k)} - \frac{F_{T_i}(t^{(k)}; \boldsymbol{\theta}) - p}{f_{T_i}(t^{(k)}; \boldsymbol{\theta})}.$$

We have derived a slightly more efficient method in the `qwei_series` R function below.

```r
#' Quantile function for a series system with Weibull component lifetimes.
#'
#' @param p quantile
#' @param shapes shape parameters
#' @param scales scale parameters
#' @returns p-th quantile
qwei_series <- function(p, shapes, scales) {
  t0 <- 1
  repeat {
    t1 <- t0 - sum((t0 / scales)^shapes) + log(1 - p) /
      sum(shapes * t0^(shapes - 1) / scales^shapes)
    if (abs(t1 - t0) < tol) {
      break
    }
    t0 <- t1
  }
  return(t1)
}
```

Maximum Likelihood Estimation

We use the Newton-Raphson method for Maximum Likelihood Estimation (MLE) in a series system with Weibull component lifetimes. Numerical optimization is carried out using R's `optim` package and the **L-BFGS-B** method [15]. This quasi-Newton method approximates the Hessian using the gradient of the log-likelihood function (see Appendices @ref(app-weibull-loglik-r) and @ref(app-score-fn-r)). Bound constraints are applied to maintain positive shape and scale parameters.

```
#' L-BFGS-B solver for the series system with Weibull component lifetimes
#' given masked data.
#'
#' @param df masked data frame
#' @param theta0 initial guess
#' @return MLE solution
mle_lbfgsb_wei_series_md_c1_c2_c3 <- function(df, theta0) {
  optim(theta0,
    fn = function(theta) loglik_wei_series_md_c1_c2_c3(df,
      theta[seq(1, length(theta), 2)], theta[seq(2, length(theta), 2)]),
    gr = function(theta) score_wei_series_md_c1_c2_c3(df,
      theta[seq(1, length(theta), 2)], theta[seq(2, length(theta), 2)]),
    lower = rep(1e-9, length(theta0)),
    method = "L-BFGS-B",
    control = list(fnscale = -1, maxit = 125L))
}
```

Bernoulli Masking Model

In the Bernoulli masking model, the failed component is guaranteed to be in the candidate set and each non-failed component is included with some fixed probability p.

```
#' Bernoulli masking model is a particular type of *uninformed* model.
#' Note that we do not generate candidate sets with this function. See
#' `md_cand_sampler` for that.
#'
#' @param df masked data frame.
#' @param p Bernoulli masking probability
#' @returns masked data frame with Bernoulli candidate set probabilities
md_bernoulli_cand_c1_c2_c3 <- function(df, p) {
  n <- nrow(df)
  p <- rep(p, length.out = n)
  Tm <- md_decode_matrix(df, "t")
  m <- ncol(Tm)
  Q <- matrix(p, nrow = n, ncol = m)
  Q[cbind(1:n, apply(Tm, 1, which.min))] <- 1
  Q[!df[["delta"]], ] <- 0
  df %>% bind_cols(md_encode_matrix(Q, prob))
}
```

Simulation

This appendix showcases select Python and R code instrumental in generating this paper. The entire source code can be found on GitHub: github.com/queelius/reliability-estimation-in-series-systems [16]. For clarity and brevity, certain functionalities and safeguards from the original code have been omitted. However, the full, unedited code, as well as the methods to reproduce this paper, are available in the GitHub repository.

Scenario Simulation

Below, you'll find the R code for the Monte-Carlo simulation, tailored to run the scenarios discussed in Section @ref(sim-study).

```r
# A function to simulate the effects of various parameters on the sampling
# distribution of the MLE
simulate_scenario <- function(
    csv_file,              # Destination file to save simulation results
    N,                     # Vector of sample sizes
    P,                     # Vector of masking probabilities
    Q,                     # Vector of quantiles of the series system
    R,                     # Number of simulation replicates
    B,                     # Number of bootstrap samples
    theta                  # True parameters for the Weibull distribution
) {
  shapes <- theta[seq(1, length(theta), 2)]
  scales <- theta[seq(2, length(theta), 2)]
  m <- length(scales)
  cname <- c("n", "p", "q", "tau", "B",
             paste0("shape.", 1:m), paste0("scale.", 1:m),
             paste0("shape.mle.", 1:m), paste0("scale.mle.", 1:m),
             paste0("shape.lower.", 1:m), paste0("shape.upper.", 1:m),
             paste0("scale.lower.", 1:m), paste0("scale.upper.", 1:m),
             "convergence", "loglik")

  for (n in N) {
    for (p in P) {
      for (q in Q) {
        tau <- qwei_series(p = q, scales = scales, shapes = shapes)
        for (iter in 1:R) {
          df <- generate_guo_weibull_table_2_data(shapes, scales, n, p, tau)
          sol <- mle_lbfgsb_wei_series_md_c1_c2_c3(theta0 = theta, df = df)

          mle_solver <- function(df, i) {
            mle_lbfgsb_wei_series_md_c1_c2_c3(theta0 = sol$par, df = df[i, ])
          }
          sol.boot <- boot::boot(df, mle_solver, R = B)
          ci <- confint(mle_boot(sol.boot))

          result <- c(n, p, q, tau, B, shapes, scales,
                      sol$par[seq(1, length(theta), 2)],
                      sol$par[seq(2, length(theta), 2)],
                      ci[seq(1, length(theta), 2), 1:2],
                      ci[seq(2, length(theta), 2), 1:2],
                      sol$convergence, sol$value)

          result_df <- setNames(data.frame(t(result)), cname)
          write.table(result_df, file = csv_file, sep = ",", append = TRUE)
        }
      }
    }
  }
}
```

For example, to run the simulation study for the effect of masking probability, we would run the following code:

```r
source("sim-scenario.R")
library(wei.series.md.c1.c2.c3)
simulate_scenario(
    theta = alex_weibull_series$theta, # the well-designed system
    N = c(90L), # sample size
    P = c(0.1, 0.215, 0.4, 0.55, 0.7), # vary masking probability
    Q = c(0.825), # right-censoring quantile
    R = 400L, B = 1000L,
    max_iter = 125L, max_boot_iter = 125L,
    n_cores = 2L, csv_file = "masking-prob-sim.csv",
    ci_method = "bca", ci_level = .95)
```

Please see the GitHub repository for the full code, since the code as presented will not run without supporting functions.

Plot Generation Code

The following Python code generates the plots for the simulation study, `plot_cp` for the coverage probabilities and `plot_mle` for the the sampling distribution of the MLE, both with respect to some simulation parameter, like sample size (n) or masking probability (p).

```python
def plot_cp(data, x_col, x_col_label):
  rel_cols = [x_col] + [f'shape.{i}' for i in range(1, 6)] + \
    [f'scale.{i}' for i in range(1, 6)] + \
    [f'shape.lower.{i}' for i in range(1, 6)] + \
    [f'shape.upper.{i}' for i in range(1, 6)] + \
    [f'scale.lower.{i}' for i in range(1, 6)] + \
    [f'scale.upper.{i}' for i in range(1, 6)]
  rel_data = data[rel_cols].copy()

  def compute_coverage(row, j):
    shape_in = row[f'shape.lower.{j}'] <= row[f'shape.{j}'] <= row[f'shape.upper.{j}']
    scale_in = row[f'scale.lower.{j}'] <= row[f'scale.{j}'] <= row[f'scale.upper.{j}']
    return pd.Series([shape_in, scale_in], index=[f'shape.cov.{j}', f'scale.cov.{j}'])

  for j in range(1, 6):
    rel_data[[f'shape.cov.{j}', f'scale.cov.{j}']] = \
      rel_data.apply(lambda row: compute_coverage(row, j), axis=1)

  cols.cov = [f'shape.cov.{j}' for j in range(1, 6)] + \
             [f'scale.cov.{j}' for j in range(1, 6)]
  cps = rel_data.groupby(x_col)[cols.cov].mean().reset_index()

  # Mean coverage probabilities
  mean_shape_cp = cps[[f'shape.cov.{j}' for j in range(1, 6)]].mean(axis=1)
  mean_scale_cp = cps[[f'scale.cov.{j}' for j in range(1, 6)]].mean(axis=1)
  cps['mean_shape_cp'] = mean_shape_cp
  cps['mean_scale_cp'] = mean_scale_cp

  plt.figure(figsize=[5, 4])
  shape_cmap = plt.get_cmap('Blues')
  scale_cmap = plt.get_cmap('Reds')

  shape.ls = ['-', '--', '-.', ':', '-']
```

```python
    shape.mk = ['o', 's', '^', 'x', 'D']

    for j, color, ls, mk in zip(range(1, 6), \
        shape_cmap(np.linspace(0.4, 1, 5)), shape.ls, shape.mk):
      plt.plot(cps[x_col], cps[f'shape.cov.{j}'], label=f'$k_{j}$', \
        color=color, linestyle=ls, marker=mk)

    scale.ls = ['-', '--', '-.', ':', '-']
    scale.mk = ['o', 's', '^', 'x', 'D']

    for j, color, ls, mk in zip(range(1, 6), \
        scale_cmap(np.linspace(0.4, 1, 5)), scale.ls, scale.mk):
      plt.plot(cps[x_col], cps[f'scale.cov.{j}'], label=f'$\lambda_{j}$', \
        color=color, linestyle=ls, marker=mk)

    plt.plot(cps[x_col], cps['mean_shape_cp'], color='darkblue', linewidth=4, \
      linestyle='-', label='$\\bar{k}$')
    plt.plot(cps[x_col], cps['mean_scale_cp'], color='darkred', linewidth=4, \
      linestyle='-', label='$\\bar{\lambda}$')

    plt.axhline(y=0.95, color='green', linestyle='--', label='$95\\%$')
    plt.axhline(y=0.90, color='red', linestyle='--', label='$90\\%$')
    plt.xlabel(f'{x_col_label} (${x_col}$)')
    plt.ylabel('Coverage Probability (CP)')
    plt.gca().yaxis.set_major_formatter(
        FuncFormatter(lambda y, _: '{:.2f}'.format(y)))
    plt.title('Coverage Probabilities for Parameters')
    plt.legend(loc='best', bbox_to_anchor=(1, 1))
    plt.tight_layout()
    plt.savefig(f'plot-{x_col}-vs-cp.pdf')

def plot_mle(raw_data, x_col, par, par_label, label, k=100, loc='upper left'):
  ps = par.split('.')
  par_low = f'{ps[0]}.lower.{ps[1]}'
  par_up = f'{ps[0]}.upper.{ps[1]}'
  par_mle = f'{ps[0]}.mle.{ps[1]}'
  x_vals = sorted(raw_data[x_col].unique())

  median_mles = []
  true_vals = []
  mean_mles = []
  low_q = []
  up_q = []
  plt.figure(figsize=(4, 4))

  for i, x in enumerate(x_vals):
    data = raw_data[raw_data[x_col] == x]
    low_med, up_med = np.percentile(
      data[par_low], 50), np.percentile(data[par_up], 50)
    mean_mle = data[par_mle].mean()
    true_val = data[par].mean()
    median_mle = data[par_mle].median()
    mean_mles.append(mean_mle)
```

41

```
    median_mles.append(median_mle)
    low_q.append(np.percentile(data[par_mle], 2.5))
    up_q.append(np.percentile(data[par_mle], 97.5))
    true_vals.append(true_val)

    plt.vlines(i, low_med, up_med, color='blue',
      label='Median 95% CI' if i == 0 else "")
    plt.plot(i, mean_mle, 'ro', label='Mean MLE' if i == 0 else "")
    plt.plot(i, median_mle, 'bo', label='Median MLE' if i == 0 else "")

  plt.plot(np.arange(len(x_vals)), mean_mles, 'r--')
  plt.plot(np.arange(len(x_vals)), median_mles, 'b--')
  plt.plot(np.arange(len(x_vals)), true_vals,
    'g-', label='True Value', linewidth=2)

  plt.fill_between(np.arange(len(x_vals)), low_q, up_q,
    color='blue', alpha=0.15, label='95% Quantile Range')

  plt.xticks(np.arange(len(x_vals)), x_vals)
  plt.xlabel(f'{label} (${x_col}$)')
  plt.ylabel(f'Statistics for {par_label}')
  plt.title(f'MLE for {par_label}')
  plt.legend(loc=loc)

  leg = plt.gca().get_legend()
  for text in leg.get_texts():
    plt.setp(text, fontsize='small')

  plt.tight_layout(h_pad=4.0, w_pad=2.5)
  plt.savefig(f'plot-{x_col}-vs-{par}-mle.pdf')
```

For example, to generate a plot of the coverage probabilities with respect to sample size (n) for the CSV file `data.csv`, we would run the following code:

```
from plot_utils import plot_cp
import pandas as pd
plot_cp(pd.read_csv("data.csv"), 'n', 'Sample Size')
```

Please see the GitHub repository for the full code, since the code as presented will not run without supporting functions.

Acknowledgements

[1] M. Agustin, "Systems in series," in *Wiley encyclopedia of operations research and management science*, John Wiley & Sons, Ltd, 2011. doi: https://doi.org/10.1002/9780470400531.eorms0866. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470400531.eorms0866

[2] N. N. Taleb, *The black swan: The impact of the highly improbable.* Random House, 2007.

[3] F. M. Guess, T. J. Hodgson, and J. S. Usher, "Estimating system and component reliabilities under partial information on cause of failure," *Journal of Statistical Planning and Inference*, vol. 29, nos. 1-2, pp. 75–85, Sep. 1991, doi: 10.1016/0378-3758(92)90123-a

[4] E. L. Lehmann and G. Casella, *Theory of point estimation.* Springer Science & Business Media, 1998.

[5] C. F. J. Wu, "On the convergence properties of the em algorithm," *The Annals of Statistics*, vol. 11, no. 1, pp. 95–103, 1983.

[6] L. J. Bain and M. Engelhardt, *Introduction to probability and mathematical statistics*, Second. Duxbury Press, 1992, p. 644.

[7] G. Casella and R. L. Berger, *Statistical inference.* Duxbury Advanced Series, 2002.

[8] B. Efron, "Better bootstrap confidence intervals," *Journal of the American Statistical Association*, vol. 82, no. 397, pp. 171–185, 1987.

[9] B. Efron and R. J. Tibshirani, *An introduction to the bootstrap.* CRC press, 1994.

[10] J. P. Klein and M. L. Moeschberger, *Survival analysis: Techniques for censored and truncated data.* Springer Science & Business Media, 2005.

[11] R. B. Abernethy, *New weibull handbook*, 5th ed. Abernethy, 2006.

[12] H. Guo, P. Niu, and F. Szidarovszky, "Estimating component reliabilities from incomplete system failure data," *Annual Reliability and Maintainability Symposium (RAMS)*, pp. 1–6, Jan. 2013, doi: 10.1109/rams.2013.6517765

[13] D. R. Cox, "Regression models and life-tables," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 34, no. 2, pp. 187–202, 1972, doi: https://doi.org/10.1111/j.2517-6161.1972.tb00899.x. Available: https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.2517-6161.1972.tb00899.x

[14] A. Towell, *Wei.series.md.c1.c2.c3: Estimating reliability of weibull components in series from masked data.* 2023. Available: https://queelius.github.io/wei.series.md.c1.c2.c3/

[15] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, "A limited memory algorithm for bound constrained optimization," *SIAM Journal on Scientific Computing*, vol. 16, no. 5, pp. 1190–1208, 1995.

[16] A. Towell, "Reliability estimation in series systems: Maximum likelihood techniques for right-censored and masked failure data." 2023. Available: https://github.com/queelius/reliability-estimation-in-series-systems

[17] S. S. Wilks, "The large-sample distribution of the likelihood ratio for testing composite hypotheses," *The Annals of Mathematical Statistics*, vol. 9, no. 1, pp. 60–62, 1938.