Bootstrapping confidence intervals (BCa) of the maximum likelihood estimator of components in a series systems from masked failure data

Alex Towell

Abstract

We estimate the parameters of a series system with Weibull component lifetimes from relatively small samples consisting of right-censored system lifetimes and masked component cause of failure. Under a set of conditions that permit us to ignore how the component cause of failures are masked, we assess the bias and variance of the estimator. Then, we assess the accuracy of the boostrapped variance and calibration of the confidence intervals of the MLE under a variety of scenarios.

Contents

1 Introduction **2**

2 Series System Model **2**
2.1 Component Cause of Failure 5

3 Likelihood Model for Masked Data **6**
3.1 Masked Component Cause of Failure 8
3.2 Right-Censored Data 10

4 Maximum Likelihood Estimation **11**

5 Bias-Corrected and Accelerated Bootstrap Confidence Intervals **12**

6 Series System with Weibull Components **13**
6.1 Weibull Series System 14
6.2 Weibull Likelihood Model for Masked Data 15

7 Simulation Study **16**
7.1 Data Generating Process 17
7.2 Convergence to the MLE: Challenges and Issues 17
7.3 Assessing the Impact of Sample Size, Masking, and Censoring on the MLE 18
7.4 Assessing the Impact of Different Component Reliabilities 23

8 Conclusion **31**

Appendix A: R Code For Log-likelihood Function **32**

Appendix B: R Code For Score Function **32**

Appendix C: R Code For Simulation of Scenarios For Assessing Bootstrapped (BCa) Confidence Intervals **33**

Appendix D: Bernoulli Candidate Set Model **36**

1 Introduction

Accurately estimating the reliability of individual components in multi-component systems is an important problem in many engineering domains. However, component lifetimes and failure causes are often not directly observable. In a series system, only the system-level failure time may be recorded along with limited information about which component failed. Such *masked* data poses challenges for estimating component reliability.

In this paper, we develop a maximum likelihood approach to estimate component reliability in series systems using right-censored lifetime data and candidate sets that contain the failed component. The key contributions are:

1. Deriving a likelihood model that accounts for right-censoring and masked failure causes through candidate sets. This allows the available masked data to be used for estimation.

2. Validating the accuracy, precision, and robustness of the maximum likelihood estimator through an extensive simulation study under different sample sizes, masking probabilities, and censoring levels.

3. Demonstrating that bootstrapping provides well-calibrated confidence intervals for the MLEs even with small samples.

Together, these contributions provide a statistically rigorous methodology for learning about latent component properties from series system data. The methods are shown to work well even when failure information is significantly masked. This capability expands the range of applications where component reliability can be quantified from limited observations.

The remainder of this paper is organized as follows. First, we detail the series system and masked data models. Next, we present the likelihood construction and maximum likelihood theory. We then describe the bootstrap approach for variance and confidence interval estimation. Finally, we validate the methods through simulation studies under various data scenarios and sample sizes.

2 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 i^{th} system denoted by the random variable T_i. The lifetime of the j^{th} component in the i^{th} system 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 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 i^{th} 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 survival analysis: the survival function, the probability density function, and the hazard function. The survival function, $R_{T_i}(t)$, is the probability that the i^{th} system has a lifespan larger than a duration t,

$$R_{T_i}(t) = \Pr\{T_i > t\} \tag{2.1}$$

The probability density function (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 t and Δt given that no failure occurs before time t is given by

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

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

$$\frac{\Pr\{t < T < t + \Delta t\}}{\Delta t} \frac{1}{\Pr\{T > t\}} = \frac{R_T(t) - R(t + \Delta t)}{R_T(t)}.$$

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

$$\begin{aligned} h_{T_i}(t) &= \lim_{\Delta t \to 0} \frac{\Pr\{t < T_i < t + \Delta t\}}{\Delta t} \frac{1}{\Pr\{T_i > t\}} \\ &= \frac{f_{T_i}(t)}{R_{T_i}(t)}. \end{aligned} \tag{2.2}$$

\end{definition}

The lifetime of the j^{th} 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

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

When a random variable T is parameterized by a particular $\boldsymbol{\theta}$, we denote the reliability function by $R_T(t; \boldsymbol{\theta})$, and the same for other distribution functions. If it is clear from the context which random variable a distribution function is for, we drop the subscripts, e.g., $R(t)$ instead of $R_T(t)$. As a special case, we denote the pdf of the j^{th} component by $f_j(t; \boldsymbol{\theta_j})$ and its reliability function by $R_j(t; \boldsymbol{\theta_j})$.

Two random variables X and Y have a joint pdf $f_{X,Y}(x, y)$. Given the joint pdf $f(x, y)$, 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, pdf, and hazard function of the series system. We begin with the reliability function of the series system, as given by the following theorem.

Theorem 1. *The series system has a reliability function given by*

$$R(t; \boldsymbol{\theta}) = \prod_{j=1}^{m} R_j(t; \boldsymbol{\theta_j}). \tag{2.3}$$

Proof. The reliability function is defined as

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

which may be rewritten as

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

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

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

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

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

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

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

$\square$

Theorem 1 shows that the system's overall reliability is the product of the reliabilities of its individual components. This property is inherent to 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.

Theorem 2. *The series system has a pdf given by*

$$f(t;\boldsymbol{\theta}) = \sum_{j=1}^{m} f_j(t;\boldsymbol{\theta_j}) \prod_{\substack{k=1\\k\neq j}}^{m} R_k(t;\boldsymbol{\theta_j}). \tag{2.4}$$

Proof. By definition, the pdf may be written as

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

By the product rule, this may be rewritten as

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

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

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

which simplifies to

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

$\square$

Theorem 2 shows the pdf of the system lifetime as 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.

Theorem 3. *The series system has a hazard function given by*

$$h(t;\boldsymbol{\theta}) = \sum_{j=1}^{m} h_j(t;\boldsymbol{\theta_j}). \tag{2.5}$$

Proof. By Equation (2.2), the i^{th} series system lifetime has a hazard function defined as

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

Plugging in expressions for these functions results in

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

which can be simplified to

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

□

Theorem 3 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,

$$h(t;\boldsymbol{\theta}) = \frac{f(t;\boldsymbol{\theta})}{R(t;\boldsymbol{\theta})},$$

and we can rearrange this to get

$$\begin{aligned} f(t;\boldsymbol{\theta}) &= h(t;\boldsymbol{\theta})R(t;\boldsymbol{\theta}) \\ &= \left\{\sum_{j=1}^m h_j(t;\boldsymbol{\theta_j})\right\}\left\{\prod_{j=1}^m R_j(t;\boldsymbol{\theta_j})\right\}, \end{aligned} \tag{2.6}$$

which we sometimes find to be a more convenient form than Equation (2.4).

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

2.1 Component Cause of Failure

Whenever a series system fails, precisely one of the components is the cause. We model the component cause of the series system failure as a random variable.

Definition 1. *The component cause of failure of a series system is denoted by the 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.

Theorem 4. *The joint pdf of the component cause of failure K_i and series system lifetime T_i is given by*

$$f_{K_i,T_i}(j,t;\boldsymbol{\theta}) = h_j(t;\boldsymbol{\theta_j}) \prod_{l=1}^m R_l(t;\boldsymbol{\theta}), \tag{2.7}$$

where $h_j(t;\boldsymbol{\theta_j})$ is the hazard function of the j^{th} component and $R_{T_i}(t;\boldsymbol{\theta})$ is the reliability function of the series system.

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

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

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,

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

Since $h_1(t;\boldsymbol{\theta_1}) = f_1(t;\boldsymbol{\theta_1})/R_1(t;\boldsymbol{\theta_1})$,

$$f_1(t;\boldsymbol{\theta_1}) = h_1(t;\boldsymbol{\theta_1}) R_1(t;\boldsymbol{\theta_1}).$$

Making this substitution into the above expression for $f_{K_i,T_i}(j,t;\boldsymbol{\theta})$ yields

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

Generalizing from this completes the proof. □

The probability that the j^{th} component is the cause of failure is given by

$$\Pr\{K_i = j\} = E_{\boldsymbol{\theta}}\left[\frac{h_j(T_i;\boldsymbol{\theta_j})}{\sum_{l=1}^m h_l(T_i;\boldsymbol{\theta_l})}\right]. \tag{2.8}$$

Proof. The probability the j^{th} component is the cause of failure is given by marginalizing the joint pdf of K_i and T_i over T_i,

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

By Theorem 4, this is equivalent to

$$\begin{aligned}
\Pr\{K_i = j\} &= \int_0^\infty h_j(t;\boldsymbol{\theta_j}) R_{T_i}(t;\boldsymbol{\theta}) dt \\
&= \int_0^\infty h_j(t;\boldsymbol{\theta_j}) / h_{T_i}(t;\boldsymbol{\theta}) f_{T_i}(t;\boldsymbol{\theta}) dt \\
&= E_{\boldsymbol{\theta}}\big[h_j(T_i;\boldsymbol{\theta_j}) / \sum_{l=1}^m h_l(T_i;\boldsymbol{\theta_l})\big].
\end{aligned}$$

□

We use this result in the simulation study to analyze the impact of having a series system with a much "weaker" or "stronger" component.

3 Likelihood Model for Masked Data

The object of interest is the (unknown) parameter value $\boldsymbol{\theta}$. To estimate this $\boldsymbol{\theta}$, we need *data*. In our case, we call it *masked data* because we do not necessarily observe the event of interest, say a system failure, directly. We consider two types of masking: masking the system failure lifetime and masking the component cause of failure.

We generally encounter three types of system failure lifetime masking:

1. A system failure is observed at a particular point in time.
2. A system failure is observed to occur within a particular interval of time.

3. A system failure is not observed, but we know that the system survived at least until a particular point in time. This is known as *right-censoring* and can occur if, for instance, an experiment is terminated while the system is still functioning.

We generally encounter two types of component cause of failure masking:

1. The component cause of failure is observed.
2. The component cause of failure is not observed, but we know that the failed component is in some set of components. This is known as *masking* the component cause of failure.

Thus, the component cause of failure masking will take the form of candidate sets. A candidate set consists of some subset of component labels that plausibly contains the label of the failed component. The sample space of candidate sets are all subsets of $\{1, \ldots, m\}$, thus there are 2^m possible outcomes in the sample space.

In this paper, we limit our focus to observing *right censored* lifetimes and exact lifetimes but with masked component cause of failures. We consider a sample of n i.i.d. series systems, each of which is put into operation at some time and and observed until either it fails or is right-censored. We denote the right-censoring time of the i^{th} system by τ_i. We do not directly observe the system lifetime, T_i, but rather, we observe the right-censored lifetime, S_i, which is given by

$$S_i = \min\{\tau_i, T_i\}, \tag{3.1}$$

We also observe a right-censoring indicator, δ_i, which is given by

$$\delta_i = 1_{T_i < \tau_i} \tag{3.2}$$

where $1_{\text{condition}}$ is an indicator function that outputs 1 if *condition* is true and 0 otherwise. Here, $\delta_i = 1$ indicates the event of interest, a system failure, was observed.

If a system failure lifetime is observed, then we also observe a candidate set that contains the component cause of failure. We denote the candidate set for the i^{th} system by $\mathcal{C}_i$, which is a subset of $\{1, \ldots, m\}$. Since the data generating process for candidate sets may be subject to chance variations, it as a random set.

Consider we have an independent and identically distributed (i.i.d.) random sample of masked data, $D = \{D_1, \ldots, D_n\}$, where each D_i contanis the following:

- S_i, the system lifetime of the i^{th} system.
- δ_i, the right-censoring indicator of the i^{th} system.
- $\mathcal{C}_i$, the set of candidate component causes of failure for the i^{th} system.

The masked data generation process is illustrated by Figure 1.

An example of masked data D for exact, right-censored system failure times with candidate sets that mask the component cause of failure can be seen in Table 1 for a series system with $m = 3$ components.

Table 1: Right-censored lifetime data with masked component cause of failure.

System	Right-censoring time (S_i)	Right censoring indicator (δ_i)	Candidate set ($\mathcal{C}_i$)
1	4.3	1	$\{1, 2\}$
2	1.3	1	$\{2\}$
3	5.4	0	$\emptyset$
4	2.6	1	$\{2, 3\}$
5	3.7	1	$\{1, 2, 3\}$
6	10	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:

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

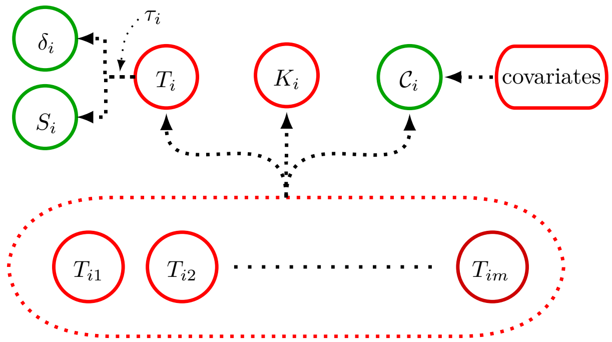

Figure 1: This figure showcases a dependency graph of the generative model for $D_i = (S_i, \delta_i, \mathcal{C}_i)$. The elements in green are observed in the sample, while the elements in red are unobserved (latent). We see that $\mathcal{C}_i$ is related to both the unobserved component lifetimes $T_{i1}, \ldots, T_{im}$ and other unknown and unobserved covariates, like ambient temperature or the particular diagnostician who generated the candidate set. These two complications for $\mathcal{C}_i$ are why seek a way to construct a reduced likelihood function in later sections that is not a function of the distribution of $\mathcal{C}_i$.

where s_i is the observed system lifetime of the i^{th} system, δ_i is the observed right-censoring indicator of the i^{th} system, and c_i is the observed candidate set of the i^{th} 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

$$L_i(\boldsymbol{\theta}) = f(s_i, \delta_i, c_i; \boldsymbol{\theta})$$

is the likelihood contribution of the i^{th} system. In other words, the likelihood function quantifies how likely different parameter values $\boldsymbol{\theta}$ are, given the observed data.

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.

Theorem 5. *The likelihood contribution of the i-th system is given by*

$$L_i(\boldsymbol{\theta}) = R_{T_i}(s_i; \boldsymbol{\theta}) \Bigg(\beta_i \sum_{j \in c_i} h_j(s_i; \boldsymbol{\theta_j}) \Bigg)^{\delta_i} \tag{3.3}$$

where $R_{T_i}(s_i; \boldsymbol{\theta}) = \prod_{j=1}^{m} R_j(s_i; \boldsymbol{\theta_j})$ is the survival function of the series system evaluted at s_i, $\delta_i = 0$ indicates the i^{th} system is right-censored at time s_i, and $\delta_i = 1$ indicates the i^{th} system is observed to have failed at time s_i with a component cause of failure is masked by the candidate set c_i.

In the follow 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 ($\delta_i = 1$).

3.1 Masked Component Cause of Failure

Suppose a diagnostician is unable to identify the precise component cause of the failure, e.g., due to cost considerations he or she replaced multiple components at once, successfully repairing the system but failing to precisely identity the failed component. In this case, the cause of failure is said to be *masked*.

The unobserved component lifetimes may have many covariates, like ambient operating temperature, but the only covariate we observe in our masked data model are the system's lifetime and additional masked data in the form of a candidate set that is somehow correlated with the unobserved component lifetimes.

The key goal of our analysis is to estimate the parameters, $\boldsymbol{\theta}$, which maximize 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 i^{th} 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 i^{th} system is given by assuming the following condition.

Condition 1. *The candidate set $\mathcal{C}_i$ contains the index of the 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 i^{th} system.

Assuming Condition 1, $\mathcal{C}_i$ 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 1, 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 4, the joint pdf of T_i and K_i is given by

$$f_{T_i,K_i}(t_i, j; \boldsymbol{\theta}) = h_j(t_i; \boldsymbol{\theta_j}) R_{T_i}(t_i; \boldsymbol{\theta}),$$

where $h_j(t_i; \boldsymbol{\theta_j})$ is the hazard function for the j^{th} component and $R_{T_i}(t_i; \boldsymbol{\theta})$ is the reliability function of the system. Thus, the joint pdf of T_i, K_i, and $\mathcal{C}_i$ may be written as

$$\begin{aligned} f_{T_i,K_i,\mathcal{C}_i}(t_i, j, c_i; \boldsymbol{\theta}) &= f_{T_i,K_i}(t_i, k; \boldsymbol{\theta}) \Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j\} \\ &= h_j(t_i; \boldsymbol{\theta_j}) R_{T_i}(t_i; \boldsymbol{\theta}) \Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j\}. \end{aligned} \tag{3.4}$$

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 (3.4),

$$f_{T_i,\mathcal{C}_i}(t_i, c_i; \boldsymbol{\theta}) = R_{T_i}(t_i; \boldsymbol{\theta}) \sum_{j=1}^{m} \Bigg\{ h_j(t_i; \boldsymbol{\theta_j}) \Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j\} \Bigg\}.$$

By Condition 1, $\Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j\} = 0$ 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

$$f_{T_i,\mathcal{C}_i}(t_i, c_i; \boldsymbol{\theta}) = R_{T_i}(t_i; \boldsymbol{\theta}) \sum_{j \in c_i} \Bigg\{ h_j(t_i; \boldsymbol{\theta_j}) \Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j\} \Bigg\}. \tag{3.5}$$

When we try to find an MLE of $\boldsymbol{\theta}$ (see Section 4), 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. *Any of the components in the candidate set has an equal probability of being the cause of failure. That is, for a fixed $j \in c_i$,*

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

for all $j' \in c_i$.

According to (Guess et al., 1991), in many industrial problems, masking generally occurred due to time constraints and the expense of failure analysis. In this setting, Condition 2 generally holds.

Assuming Conditions 1 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 (3.5), and thus the joint pdf of T_i and $\mathcal{C}_i$ may be rewritten as

$$f_{T_i,\mathcal{C}_i}(t_i, c_i; \boldsymbol{\theta}) = \Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j'\} R_{T_i}(t_i; \boldsymbol{\theta}) \sum_{j \in c_i} h_j(t_i; \boldsymbol{\theta_j})$$

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 (nor 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 $\boldsymbol{\theta}$.

When Conditions 1, 2, and 3 are satisfied, the joint pdf of T_i and $\mathcal{C}_i$ is given by

$$f_{T_i,\mathcal{C}_i}(t_i, c_i; \boldsymbol{\theta}) = \beta_i R_{T_i}(t_i; \boldsymbol{\theta}) \sum_{j \in c_i} h_j(t_i; \boldsymbol{\theta_j}).$$

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

$$L_i(\boldsymbol{\theta}) = R_{T_i}(t_i; \boldsymbol{\theta}) \sum_{j \in c_i} h_j(t_i; \boldsymbol{\theta_j}). \tag{3.6}$$

To summarize this result, assuming Conditions 1, 2, and 3, if we observe an exact system failure time for the i-th 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 (3.6).

3.2 Right-Censored Data

As described in Section 3, 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 right-censoring 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.

Theorem 6. *The likelihood contribution of a right-censored observation ($\delta_i = 0$) is given by*

$$L_i(\boldsymbol{\theta}) = R_{T_i}(s_i; \boldsymbol{\theta}). \tag{3.7}$$

Proof. When right-censoring occurs, then $S_i = \tau_i$, and we only know that $T_i > \tau_i$, 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,

$$L_i(\boldsymbol{\theta}) = R_{T_i}(s_i; \boldsymbol{\theta}).$$

$\square$

When we combine the two likelihood contributions, we obtain the likelihood contribution for the i^{th} system shown in Theorem 5,

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

We use this result in the next section to derive the maximum likelihood estimator of $\boldsymbol{\theta}$.

4 Maximum Likelihood Estimation

In our analysis, we use maximum likelihood estimation (MLE) to estimate the series system parameter $\boldsymbol{\theta}$ from the masked data (Engelhardt, 1992; Casella and Berger, 2002). 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

$$L(\hat{\boldsymbol{\theta}}) = \max_{\boldsymbol{\theta} \in \boldsymbol{\Omega}} L(\boldsymbol{\theta}), \tag{4.1}$$

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 (Casella and Berger, 2002).

Theorem 7. *The log-likelihood function, $\ell(\boldsymbol{\theta})$, for our masked data model is the sum of the log-likelihoods for each observation,*

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{n} \ell_i(\boldsymbol{\theta}), \tag{4.2}$$

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

$$\ell_i(\boldsymbol{\theta}) = \sum_{j=1}^{m} \log R_j(s_i; \boldsymbol{\theta_j}) + \delta_i \log\Bigl(\sum_{j \in c_i} h_j(s_i; \boldsymbol{\theta_j})\Bigr). \tag{4.3}$$

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

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

Substituting $L_i(\boldsymbol{\theta})$ from Equation (3.3), we consider these two cases of δ_i seperately to obtain the result in Theorem 7.

Case 1: If the i-th system is right-censored ($\delta_i = 0$),

$$\ell_i(\boldsymbol{\theta}) = \log R_{T_i}(s_i; \boldsymbol{\theta}) = \sum_{j=1}^{m} \log R_j(s_i; \boldsymbol{\theta_j}).$$

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

$$\ell_i(\boldsymbol{\theta}) = \log R_{T_i}(s_i; \boldsymbol{\theta}) + \log \beta_i + \log\Bigl(\sum_{j \in c_i} h_j(s_i; \boldsymbol{\theta_j})\Bigr).$$

We replace $R_{T_i}(s_i; \boldsymbol{\theta})$ with its component-wise definition and by Condition 3, we may discard[1] the $\log \beta_i$ term since it does not depend on $\boldsymbol{\theta}$, giving us the result

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

Combining these two cases gives us the result in Theorem 7. □

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, i.e.,

$$\frac{\partial}{\partial \theta_j} \ell(\boldsymbol{\theta}) = 0, \tag{4.4}$$

for each component θ_j of the parameter $\boldsymbol{\theta}$ (Engelhardt, 1992). 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) (Casella and Berger, 2002). 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.

5 Bias-Corrected and Accelerated Bootstrap Confidence Intervals

We utilize the non-parametric bootstrap to approximate 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 B times, giving us B bootstrap replicates of the MLE. The sampling distribution of the MLE is then approximated by the empirical distribution of the bootstrap replicates of the MLE.

The method we use to generate confidence intervals is known as Bias-Corrected and Accelerated Bootstrap Confidence Intervals (BCa), 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 transformation in our case since we are dealing with small samples and we have two potential sources of bias: right-censoring and masking component cause of failure. They seem to have opposing effects on the MLE, but the relationship is difficult to quantify.

- 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.

Since we are primarily interested in generating confidence intervals for small samples (otherwise the inverse FIM would be a good approximation) for a potentially biased MLE for the parameters of Weibull components in a series configuration (see Sections 7.3.1 and 7.3.2), we think the BCa method is a good choice for our analysis. For more details on BCa, see Efron (1987).

In our simulation study, we will assess the performance of the bootstrapped confidence intervals by computing the coverage probability of the confidence intervals. A 95% confidence interval should contain the

[1]Adding or substracting a function by a constant does not change where it obtains a maximum, so we are free to discard such terms from the log-likelihood function.

true value 95% of the time. If the confidence interval is too narrow, it will have a coverage probability less than 95%, which conveys a sort of false confidence in the precision of the MLE. If the confidence interval is too wide, it will have a coverage probability greater than 95%, which conveys a lack of confidence in the precision of the MLE. Thus, we want the confidence interval to be as narrow as possible while still having a coverage probability close to the nominal level, 95%.

Issues with Resampling from the Observed Data

While the bootstrap method provides a robust and flexible tool for statistical estimation, its effectiveness can be influenced by several factors (Efron and Tibshirani, 1994).

Firstly, instances of non-convergence in our bootstrap samples were observed. Such cases can occur when the estimation method, like the MLE used in our analysis, fails to converge due to the specifics of the resampled data (Casella and Berger, 2002). This issue can potentially introduce bias or reduce the effective sample size of our bootstrap distribution.

Secondly, 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 datasets, the bootstrap samples might not adequately represent the true variability in the data, leading to inaccurate results (Efron and Tibshirani, 1994).

Thirdly, 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 (Klein and Moeschberger, 2005).

Despite these challenges, we found the bootstrap method useful in approximating the sampling distribution of the MLE, taking care in interpreting the results, particularly as it relates to coverage probabilities.

6 Series System with Weibull Components

In the real world, systems are quite complex:

1. They are not perfect series systems.
2. The components in a system are not independent.
3. The lifetimes of the components are not precisely modeled by any named probability distributions.
4. The components may depend on many other unobserved factors.

With these caveats in mind, we model the data as coming from a Weibull series system of $m = 5$ components, and other factors, like ambient temperature, are either negligible (on the distribution of component lifetimes) or are more or less constant.

The j^{th} component of the i^{th} has a lifetime distribution given by

$$T_{ij} \sim \text{WEI}(\boldsymbol{\theta_j})$$

where $\boldsymbol{\theta_j} = (k_j, \lambda_j)$ for $j = 1, \ldots, m$. Thus, $\boldsymbol{\theta} = (\boldsymbol{\theta_1}, \ldots, \boldsymbol{\theta_m})' = \bigl(k_1, \lambda_1, \ldots, k_m, \lambda_m\bigr)$. The random variable T_{ij} has a reliability function, pdf, and hazard function given respectively by

$$R_j(t; \lambda_j, k_j) = \exp\biggl\{-\left(\frac{t}{\lambda_j}\right)^{k_j}\biggr\}, \tag{6.1}$$

$$f_j(t; \lambda_j, k_j) = \frac{k_j}{\lambda_j}\left(\frac{t}{\lambda_j}\right)^{k_j - 1} \exp\biggl\{-\left(\frac{t}{\lambda_j}\right)^{k_j}\biggr\}, \tag{6.2}$$

$$h_j(t; \lambda_j, k_j) = \frac{k_j}{\lambda_j}\left(\frac{t}{\lambda_j}\right)^{k_j - 1} \tag{6.3}$$

where $t > 0$ is the lifetime, $\lambda_j > 0$ is the scale parameter and $k_j > 0$ is the shape parameter.

The mean-time-to-failure (MTTF) of the j^{th} Weibull component is given by

$$E[T_{ij}] = \lambda_j \Gamma\left(1 + \frac{1}{k_j}\right).$$

The lifetime of the series system composed of m Weibull components has a reliability function given by

$$R(t;\boldsymbol{\theta}) = \exp\left\{-\sum_{j=1}^{m}\left(\frac{t}{\lambda_j}\right)^{k_j}\right\}. \tag{6.4}$$

Proof. By Theorem 1,

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

Plugging in the Weibull component reliability functions obtains the result

$$\begin{aligned} R(t;\boldsymbol{\theta}) &= \prod_{j=1}^{m} \exp\left\{-\left(\frac{t}{\lambda_j}\right)^{k_j}\right\} \\ &= \exp\left\{-\sum_{j=1}^{m}\left(\frac{t}{\lambda_j}\right)^{k_j}\right\}. \end{aligned}$$

□

The Weibull series system's hazard function is given by

$$h(t;\boldsymbol{\theta}) = \sum_{j=1}^{m} \frac{k_j}{\lambda_j}\left(\frac{t}{\lambda_j}\right)^{k_j-1}, \tag{6.5}$$

whose proof follows from Theorem 3. The pdf of the series system is given by

$$f(t;\boldsymbol{\theta}) = \left\{\sum_{j=1}^{m} \frac{k_j}{\lambda_j}\left(\frac{t}{\lambda_j}\right)^{k_j-1}\right\} \exp\left\{-\sum_{j=1}^{m}\left(\frac{t}{\lambda_j}\right)^{k_j}\right\}. \tag{6.6}$$

Proof. By definition,

$$f(t;\boldsymbol{\theta}) = h(t;\boldsymbol{\theta})R(t;\boldsymbol{\theta}).$$

Plugging in the failure rate and reliability functions given respectively by Equations (6.4) and (6.5) completes the proof. □

6.1 Weibull Series System

A series system composed of m Weibull components is not in general a Weibull system unless they all have the shape.

Theorem 8. *If the shape parameters of the components are identical, then the system is a Weibull with a shape parameter given by the shape parameter of the components and a scale parameter given by*

$$\lambda = \left(\sum_{j=1}^{m} \lambda_j^{-k}\right)^{-1/k}. \tag{6.7}$$

Proof. Given m Weibull lifetimes $T_{i1}, \ldots, T_{im}$ with the same shape parameter $k > 0$ and scale parameters $\lambda_1, \ldots, \lambda_m$, the survival function of the series system is

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

To make this a Weibull system, we need to find a single scale parameter λ such that

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

which has the solution

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

□

In a well-designed series system, the components are chosen such that the failure characteristcs of the components are similar, which suggests the shape parameters of the components are similar. In this case, the system is approximately Weibull. The MTTF of the system is approximately given by

$$E[T_i] = \lambda\Gamma\left(1 + \frac{1}{\bar{k}}\right), \tag{6.8}$$

where λ is given by Equation (6.7) and $\bar{k}$ is the average shape parameter of the components.

The component cause of failure also simplifies to

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

which is independent of the system failure time t_i.

6.2 Weibull Likelihood Model for Masked Data

In Section 3, 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 i^{th} system is given by the following theorem.

Theorem 9. *Let δ_i be an indicator variable that is 1 if the i^{th} system fails and 0 (right-censored) otherwise. Then the likelihood contribution of the i^{th} system is given by*

$$L_i(\boldsymbol{\theta}) = \begin{cases} \exp\left\{-\sum_{j=1}^m \left(\frac{t_i}{\lambda_j}\right)^{k_j}\right\} \beta_i \sum_{j\in c_i} \frac{k_j}{\lambda_j}\left(\frac{t_i}{\lambda_j}\right)^{k_j-1} & \text{if } \delta_i = 1, \\ \exp\left\{-\sum_{j=1}^m \left(\frac{t_i}{\lambda_j}\right)^{k_j}\right\} & \text{if } \delta_i = 0. \end{cases} \tag{6.9}$$

Proof. By Theorem 5, the likelihood contribution of the i-th system is given by

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

By Equation (6.4), the system reliability function R_{T_i} is given by

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

and by Equation (6.3), the Weibull component hazard function h_j is given by

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

Plugging these into the likelihood contribution function obtains the result. □

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

Corollary 1. *The log-likelihood contribution of the i-th system is given by*

$$\ell_i(\boldsymbol{\theta}) = -\sum_{j=1}^{m} \left(\frac{t_i}{\lambda_j}\right)^{k_j} + \delta_i \log\left(\sum_{j \in c_i} \frac{k_j}{\lambda_j} \left(\frac{t_i}{\lambda_j}\right)^{k_j - 1}\right) \tag{6.10}$$

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

We find an MLE by solving (4.4), i.e., a point $\hat{\boldsymbol{\theta}} = (\hat{k}_1, \hat{\lambda}_1, \ldots, \hat{k}_m, \hat{\lambda}_m)$ satisfying $\nabla_\theta \ell(\hat{\boldsymbol{\theta}}) = \mathbf{0}$, where $\nabla_{\boldsymbol{\theta}}$ is the gradient of the log-likelihood function (score) with respect to $\boldsymbol{\theta}$.

To solve this system of equations, we use the Newton-Raphson method, which requires the score and the Hessian of the log-likelihood function. We analytically derive the score since it is useful to have for the Newton-Raphson method, but we do not do the same for the Hessian of the log-likelihood for the following reasons:

1. The gradient is relatively easy to derive, and it is useful to have for computing gradients efficiently and accurately, which will be useful for numerically approximating the Hessian.
2. 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.

Theorem 10. *The score function of the log-likelihood contribution of the i-th Weibull series system is given by*

$$\nabla \ell_i(\boldsymbol{\theta}) = \left(\frac{\partial \ell_i(\boldsymbol{\theta})}{\partial k_1}, \frac{\partial \ell_i(\boldsymbol{\theta})}{\partial \lambda_1}, \cdots, \frac{\partial \ell_i(\boldsymbol{\theta})}{\partial k_m}, \frac{\partial \ell_i(\boldsymbol{\theta})}{\partial \lambda_m}\right)', \tag{6.11}$$

where

$$\frac{\partial \ell_i(\boldsymbol{\theta})}{\partial k_r} = -\left(\frac{t_i}{\lambda_r}\right)^{k_r} \log\left(\frac{t_i}{\lambda_r}\right) + \frac{\frac{1}{t_i}\left(\frac{t_i}{\lambda_r}\right)^{k_r}\left(1 + k_r \log\left(\frac{t_i}{\lambda_r}\right)\right)}{\sum_{j \in c_i} \frac{k_j}{\lambda_j}\left(\frac{t_i}{\lambda_j}\right)^{k_j - 1}} 1_{\delta_i = 1 \wedge r \in c_i} \tag{6.12}$$

and

$$\frac{\partial \ell_i(\boldsymbol{\theta})}{\partial \lambda_r} = \frac{k_r}{\lambda_r}\left(\frac{t_i}{\lambda_r}\right)^{k_r} - \frac{\left(\frac{k_r}{\lambda_r}\right)^2 \left(\frac{t_i}{\lambda_r}\right)^{k_r - 1}}{\sum_{j \in c_i} \frac{k_j}{\lambda_j}\left(\frac{t_i}{\lambda_j}\right)^{k_j - 1}} 1_{\delta_i = 1 \wedge r \in c_i} \tag{6.13}$$

The result follows from taking the partial derivatives of the log-likelihood contribution of the i-th system given by Equation (6.9). 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

$$\nabla \ell(\boldsymbol{\theta}) = \sum_{i=1}^{n} \nabla \ell_i(\boldsymbol{\theta}). \tag{6.14}$$

7 Simulation Study

We derived the likelihood model for masked data for the Weibull series system in Section 6. A series system is only as reliable as its least reliable component. In our simulation study, we consider two kinds of scenarios:

1. The components are all approximately equally reliable, which is the case for many real-world systems that are designed to be reliable.

   In this part of the paper, we assess the accuracy and precision of the MLE and the BCa bootstrap method for computing 95% confidence intervals for a

2. One of the components is significantly less reliable or more reliable than the others. This is a pathological case, but it is useful to consider since it poses a challenge to our likelihood model.

   In this part of the paper, we assess sensitivity of the MLE with respect to changing the reliability of a single component. The reliability of the component varies from being much more reliable to being much less reliable than the other components, where we define the reliability of a component as the probability it is is the cause of a system failure.

In both scenarios, we assess the accuracy and precision of the MLE and the BCa bootstrap method for computing 95% confidence intervals for a series system with $m = 5$ Weibull components.

7.1 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.

Weibull Series System Lifetime

We generate data from a Weibull series system with m components. As described in Section 6, the j^{th} component of the i^{th} system has a lifetime distribution given by

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

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 very simple right-censoring model, where the right-censoring time τ 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 i^{th} 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.

Masking Model for Component Cause of Failure

We must generate data that satisfies the masking conditions described in Section 3.1. There are many ways to satisfying the masking conditions. We choose the simplest method, which we call the *Bernoulli candidate set model*. In this model, each non-failed component is included in the candidate set with a fixed probability p, independently of all other components and independently of $\boldsymbol{\theta}$, and the failed component is always included in the candidate set. See Appendic G for the R code that implements this model.

7.2 Convergence to the MLE: Challenges and Issues

Estimating latent components' parameters can be challenging. [2] The surface of the log-likelihood function may have multiple local maxima and ridges, which make it more challengnig to find the MLE, and in some cases, there is no unique MLE due to insufficient data in the presence of masking and right-censoring. We seek to explore some of these issues in this section.

[2] In our simulation study, we began the optimization at the known true parameter value $\boldsymbol{\theta}$. In real-world scenarios, the true value is unknown, requiring an initial guess. A poor initial guess may further complicate the process, neccessitating the use of global optimization methods such as simulated annealing to find a good initial guess (McLachlan and Krishnan, 2007).

1. **Candidate Sets Construction**: The candidate sets are constructed such that component 1 is present if and only if component 2 is present, it becomes impossible to estimate their parameters separately. This may occur if an analyst can only identify a larger failed component without specifying the smaller components within it.[3]

   In our Bernoulli candidate set model, this is something that can arise only by chance. In Section 7.3.2, we explore this issue by assessing the effect of varying the masking probability p in the Bernoulli candidate set model on the MLE.

2. **Least Reliable Component**: If a series system has a significantly less reliable component that causes every system failure, the data may only contain information about that component's parameters.

   We explore this issue in Section **??** by assessing the effect of varying the reliability of a single component on the MLE.

3. **Aggressive Right-Censoring**: If the right-censoring time τ is too short, the data may not contain enough information to estimate the parameters of any of the components.

   We do not explore this issue, but it is something to keep in mind when designing experiments. We do comment on the expected effect of right-censoring on the MLE in Section 7.3.1.

To We largely ignored identifiability issues in our simulation study, with the exception that we discarded any data sets that did not converge to a solution after 125 iterations.[4] A log-likelihood function that is flat can cause our convergence criteria to take a long time to reach a solution. Therefore, a failure to converge within 125 iterations could be seen as evidence of potential identifiability issues.

Nonetheless, such scenarios occurred infrequently. During the bootstrapping of confidence intervals, we included all MLEs, even those that did not converge. This worst-case analysis approach was adopted because our main objective was to assess the performance of the BCa confidence intervals. We were concerned that if we took any additional steps, we may unintentionally bias the results in favor of producing narrow BCa confidence intervals with good coverage probabilities.

7.3 Assessing the Impact of Sample Size, Masking, and Censoring on the MLE

In this section, we focus our attention on well-engineered systems in which the component reliabilities are approximately identical. We consider the data from Guo et al. (2013), which includes a study of the reliability of a series system with three Weibull components with shape and scale parameters given by

$$\begin{aligned} k_1 &= 1.2576 \quad \lambda_1 = 994.3661 \\ k_2 &= 1.1635 \quad \lambda_2 = 908.9458 \\ k_3 &= 1.1308 \quad \lambda_3 = 840.1141. \end{aligned} \tag{7.1}$$

Our approach is to extend this system to a five component system by adding two more components with shape and scale parameters given by

$$\begin{aligned} k_4 &= 1.1802 \quad \lambda_4 = 940.1342 \\ k_5 &= 1.2034 \quad \lambda_5 = 923.1631. \end{aligned} \tag{7.2}$$

As shown by Table 2, there are no components that are significantly less reliable than any of the others. Before we begin our simulation study, we first explain what we expect the simulations to demonstrate and offer an explanation for why we expect this to be the case.

[3]In this case, we may want to combine the components into a single component and estimate the parameters of the reduced system.

[4]The choice of 125 iterations was driven by the computational demands of the simulation study combined with the subsequent bootstrapping of the confidence intervals.

Table 2: Mean Time To Failure (MTTF) and Probability of Component Failure of Weibull Components in Series Configuration

	MTTF	Failure Probability
Component 1	924.869	0.169
Component 2	862.157	0.207
Component 3	803.564	0.234
Component 4	888.237	0.196
Component 5	867.748	0.195
Series System	222.884	NA

7.3.1 Effect of Right-Censoring

In all of our simulation studies, we use a fixed right-censoring time $\tau = 377.71$, which is the 82.5% quantile of the series system. This means that 82.5% of the series systems are expected to fail before time τ and 17.5% of the series are expected to be right-censored. To solve for the 82.5% quantile of the series system, we solve for τ in the equation

$$F_{T_i}(\tau; \boldsymbol{\theta}) = 0.825 \tag{7.3}$$

using the Newton-Raphson method to find the root of $g(\tau) = F_{T_i}(\tau; \boldsymbol{\theta}) - 0.825$, which has the unique solution $\tau = 377.71$.

Having a fixed right-censoring lifetime τ represents a situation in which an experiment is run for a fixed amount of time τ, and all systems under observation that have not failed by the end of the experiment are right-censored.

Right-censoring introduces a source of bias in the MLE. Right-censoring has the effect of pushing the MLE to estimate a larger MTTF for each of the components, so that the series system has a larger MTTF. This is because when we observe a right-censoring event, we know that the system failed after the censoring time, but we do not know precisely when it will fail. This uncertainty has the effect of pushing the MLE to estimate a larger MTTF for the system so that it is more likely to fail after the censoring time. See Klein and Moeschberger (2005) for more information on this phenomenon.

To increase the MTTF of a series system, the MTTF of each component is increased. The mean time to failure (MTTF) for the j^{th} component in a Weibull distribution is given by

$$\text{MTTF}_j = \lambda_j \Gamma(1 + 1/k_j), \tag{7.4}$$

therefore, in order to increase the MTTF of the components, lower values for the shape parameters are chosen and higher values for the scale parameters are chosen.

7.3.2 Effect of Masking the Component Cause of Failure

When we observe a system failure, we know that one of the components in the candidate set caused the system to fail, but we do not know which one. This uncertainty has the effect of pushing the MLE to estimate a smaller MTTF for each of the components in the candidate set. For components that are frequently in candidate sets but proportionally not more likely to be a component cause of failure, the effect is more pronounced, which may introduce a source of bias in the MLE for such components.

In our Bernoulli candidate set model, the masking probability p determines how commonly each non-failed component is in the candidate set, and so we expect that as p increases, this will become a more pronoucned source of bias.[5] However, note that the effect of masking, which pushes the MLE to estimate a smaller MTTF, has opposite effect to that of right-censoring, which pushes the MLE to estimate a larger MTTF. As these two sources of bias compete with each other, it is not clear which one will dominate

In what follows, we explain how the bias induced by masking the component cause of failure effects the MLE for the shape and scale parameters of a Weibull component. Assessing Equation (7.4), we see that the

[5]In a more complicated candidate set model, it is possible that masking could introduce a significant source of bias for some components, and none at all for others.

MTTF of a Weibull component is proportional to its scale parameter λ_j, which means when we decrease the scale parameter λ_j (keeping the shape parameter k_j constant), the MTTF decreases. Therefore, if the j^{th} component is in the candidate set, to make it more likely to appear in the candidate set, its scale parameter should be decreased, potentially biasing the MLE for the scale parameter downwards.

Conversely, we see that the MTTF decreases as we increase the shape parameter k_j. Therefore, if the j^{th} component is in the candidate set, to make it more likely to appear in in the candidate set, its shape parameter should be increased, potentially biasing the MLE for the shape parameter upwards.

> NOTE TO SELF: The right-censoring has an effect best seen by the MTTF of the series system, and consequently the components. The masking probability, on the other hand, has an effect best seen by the probability of component failure. Additionally, when looking at the plots in the sim study where we vary k_3, we see that the effect of masking is more pronounced for ... finish these thoughts after looking at the plots.

7.3.3 Assessing the Bootstrapped Confidence Intervals

Our primary interest is in assessing the performance of the BCa confidence intervals for the MLE. We will assess the performance of the BCa confidence intervals by computing the coverage probability of the confidence intervals. Under a variety of scenarios, we will bootstrap a 95%-confidence interval for $\boldsymbol{\theta}$ using the BCa method, and we will evaluate its calibration by computing the coverage probability and its precision by assessing the width of the confidence interval.

The coverage probability is defined as the proportion of times that the true value of $\boldsymbol{\theta}$ falls within the confidence interval. We will compute the coverage probability by generating R datasets from the Data Generating Process (DGP) and computing the coverage probability for each dataset. We will then aggregate this information across all R datasets to estimate the coverage probability.

7.3.4 Simulation Scenarios

We used a fixed right sensoring time of τ, which we defined to be the 82.5% quantile of the series system such that 82.5% of the sytems are expected to fail before time $\tau$$ and 17.5% of the series systems are expected to be right-censored.

We define a simulation scenario to be some comination of n and p. We are interested in choosing a small number of scenarios that are representative of real-world scenarios and that are interesting to analyze.

Here is an outline of the simulation study analysis:

1. Choose a sample size n and a masking probability p (with τ fixed) and simulate R datasets from the Data Generating Process (DGP) described in Section **??**.

2. For each of these R datasets, compute the MLE.

3. For each of these R datasets, perform bootstrap resampling B times to create a set of bootstrap samples.

4. Calculate the MLE for each of these bootstrap samples. This generates an empirical distribution of the MLE, which is used to construct a confidence interval for the MLE.

5. For each dataset, determine whether the true parameter value falls within the computed CI. Aggregate this information across all R datasets to estimate the coverage probability of the CI.

6. Interpret the results and discuss the performance of the MLE estimator under the chosen scenarios.

For how we generate a scenario, see Appendix A. Now, we will discuss the results of the simulation study.

7.3.5 Coverage Probability vs Sample Size

In this simulation study, we have generated many different synthetic samples of different sizes (n) from a data generating process (DGP) that is compatible with the assumptions our likelihood model makes about the data. In particular, right-censored series system lifetimes with a fixed right-censoring time for the system and five components with Weibull lifetimes, each with a different shape and scale parameter. For each observation, we then mask the component cause of failure with candidate sets that satisfy the three primary conditions of the likelihood model, e.g., the failed component is always in the candidate set. For each synthetic data set, we then compute the MLEs of the shape and scale parameters of the Weibull distribution. We then use the MLEs to compute the BCa bootstrapped 95% CIs.

In what follows, we analyze the performance of the BCa bootstrapped CIs for the shape and scale parameters under different masking conditions (p) for the component cause of failure. We will focus on the following statistics:

- *Coverage Probability (CP)*: The CP is the proportion of the bootstrapped CIs that contain the true value of the parameter. The CP is a good indicator of the reliability of the estimates as previously discussed.
- *Dispersion of MLEs*: The shaded regions representing the 95% probability range of the MLEs get narrower as the sample size increases. This is an indicator of the increased precision in the estimates as more data is available. We call it a *Confidence Band*, but it is actually an estimate of the quantile range of the MLEs. The shaded region provides insight into the distribution of the MLEs.
- *IQR of Bootstrapped CIs*: The vertical blue bars represent the Interquartile Range (IQR) of the actual bootstrapped Confidence Intervals (CIs). Since in practice we only have one sample and consequently one MLE, we use bootstrapping to resample and compute multiple CIs. The IQR then represents the middle 50% range of these bootstrapped CIs.
- *Mean of the MLEs*: The mean of the MLEs is a good indicator of the bias in the estimates. If the mean of the MLEs is close to the true value, then the MLEs are, on average, unbiased.

The distinction between the shaded region (95% range of MLEs) and the blue vertical bars (IQR of bootstrapped CIs) is important. The shaded region provides insight into the distribution of the MLEs, whereas the blue vertical bars provide information about the variation in the bootstrapped CIs. Both are relevant for understanding the behavior of the estimations.

Scale Parameters Figure 2 shows the distribution of the MLEs for the shape parameters of the first three components and the bootstrapped CIs for different sample sizes with a component cause of failrue masking probaility of $p = 0.215$ (each non-failed component is in the candidate set with a 21.5% probabiltiy).

The distinction between the shaded region (95% range of MLEs) and the blue vertical bars (IQR of bootstrapped CIs) is important. The shaded region provides insight into the distribution of the MLEs, whereas the blue vertical bars provide information about the variation in the bootstrapped CIs. Both are relevant for understanding the behavior of the estimations. Here are several key observations:

- *Coverage Probability (CP)*: The CP is well-calibrated, obtaining a value near the nominal 95% level across different sample sizes. This suggests that the bootstrapped CIs will contain the true value of the shape parameter with the specified confidence level. The CIs are neither too wide nor too narrow.
- *Dispersion of MLEs*: The shaded regions representing the 95% probability range of the MLEs get narrower as the sample size increases. This is an indicator of the increased precision in the estimates as more data is available.
- *IQR of Bootstrapped CIs*: The IQR (vertical blue bars) reduces with an increase in sample size. This suggests that the bootstrapped CIs are getting more consistent and focused around a narrower range with larger samples while maintaining a good coverage probability. As we get more data, the bootstrapped CIs are more likely to be closer to each other and the true value of the scale parameter. For small sample sizes, they are quite large, but to maintain well-calibrated CIs, this was necessary. The estimator is quite sensitive to the data, and so the bootstrapped CIs are quite wide to account for this sensitivity when the sample size is small and not necessarily representative of the true distribution.
- *Mean of MLEs*: The red dashed line indicating the mean of MLEs remains stable across different sample sizes and close to the true value, suggesting that the MLEs are, on average, reasonably unbiased.

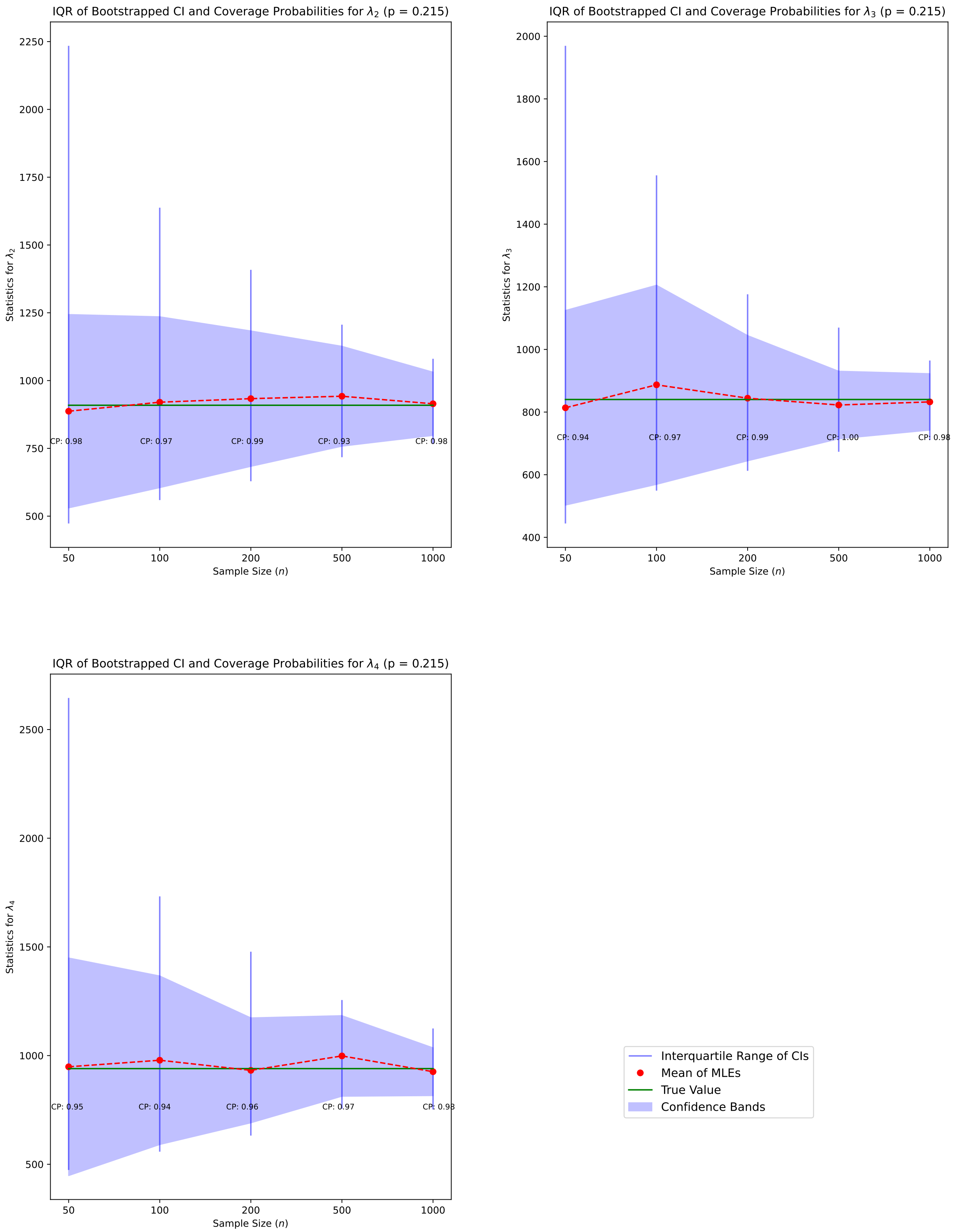

Figure 2: Sample Size vs Bootstrapped Scale CI Statistics (p = 0.215)

7.3.6 Masking Probability for Component Cause of Failure

In this scenario, we fix the sample size at $n = 90$ and we fix the right-censoring quantile fixed $q = .825$, and we vary the masking probability p from $p = 0$ (no masking the component cause of failure) to $p = 0.45$ (significant masking of the component cause of failure).

Scale Parameter In Figures 3, we show the effect of the masking probability p on the MLE and the bootstrapped BCa confidence intervals for the scale parameters. At this sample size, we see that the MLE is relatively unbiased for small p, but as p increases, the MLE becomes increasingly biased. We also see the confidence interval width seems stable until the masking becomes significant at $p = 0.45$. The confidence intervals appear to be well-calibrated at all masking probabilities, even for $p = 0.45$, although it exhibits the worst coverage at around 90% for 95% confidence intervals.

Shape Parameter In Figures 4, we show the effect of the masking probability p on the shape parameters. Here, we see that as we increase the masking probability, the confidence interval widths increase fairly significantly, as does the bias. In other words, the shape parameters appear to be more sensitive to masking.

Overall, for both parameter types, we see that as the masking probability increases, the IQR of the bootstrapped CIs, the dispersion of the MLEs,a the bias increases, which indicates that the masking probability effects the precision and accuracy of the estimates. As the masking probability increases, we have less certainty about the component cause of failure, and thus less certainty about the estimates for the component parameters.

7.4 Assessing the Impact of Different Component Reliabilities

We consider two definitions of reliability, the mean time to failure (MTTF) and the probability of component cause of failure. The MTTF for the j^{th} component is given by its expected value,

$$\text{MTTF}_j = E(T_{ij})$$

which for the Weibull distribution has the closed-form expression

$$\text{MTTF}_j = \lambda_j \, \Gamma(1 + 1/k_j),$$

where Γ is the gamma function. If the components have the same shape parameter, then the MTTF is proportional to the scale parameter, $\text{MTTF}_j \propto \lambda_j$, and the component with the smallest scale parameter is the least reliable by both definitions, the MTTF and the probability of component cause of failure.

The same is true if the components have the same scale parameter, then the MTTF increases as the shape parameter decreases, $\text{MTTF}_j \propto \Gamma(1 + 1/k_j)$, and the component with the largest shape parameter is the least reliable by both definitions.

However, the shape parameter k of the Weibull distribtion is of particular importance:

$k_j < 1$ The hazard function decreases with respect to time. For instance, this may occur as a result of defective components being weeded out early.

$k_j = 1$ The hazard function is constant with respect to time. This is an idealized case that is rarely observed in practice, but may be useful for modeling purposes.

$k_j > 1$ The hazard function increases with respect to time. For instance, this may occur as a result of components wearing out or aging.

A system with a mixture of components, some with decreasing and some with increasing failure rates is more difficult to understand and model.

In Figure 5, the top row shows two components with roughly the same shape and scale parameters, and the bottom row shows two components with different shape and scale parameters. Every component has the same MTTF, but they have vastly different hazard and survival functions (and vastly different probabilities of being the cause of a system failure). The top system is well-designed, with no weak link in the chain, while the bottom system is poorly designed and difficult to understand.

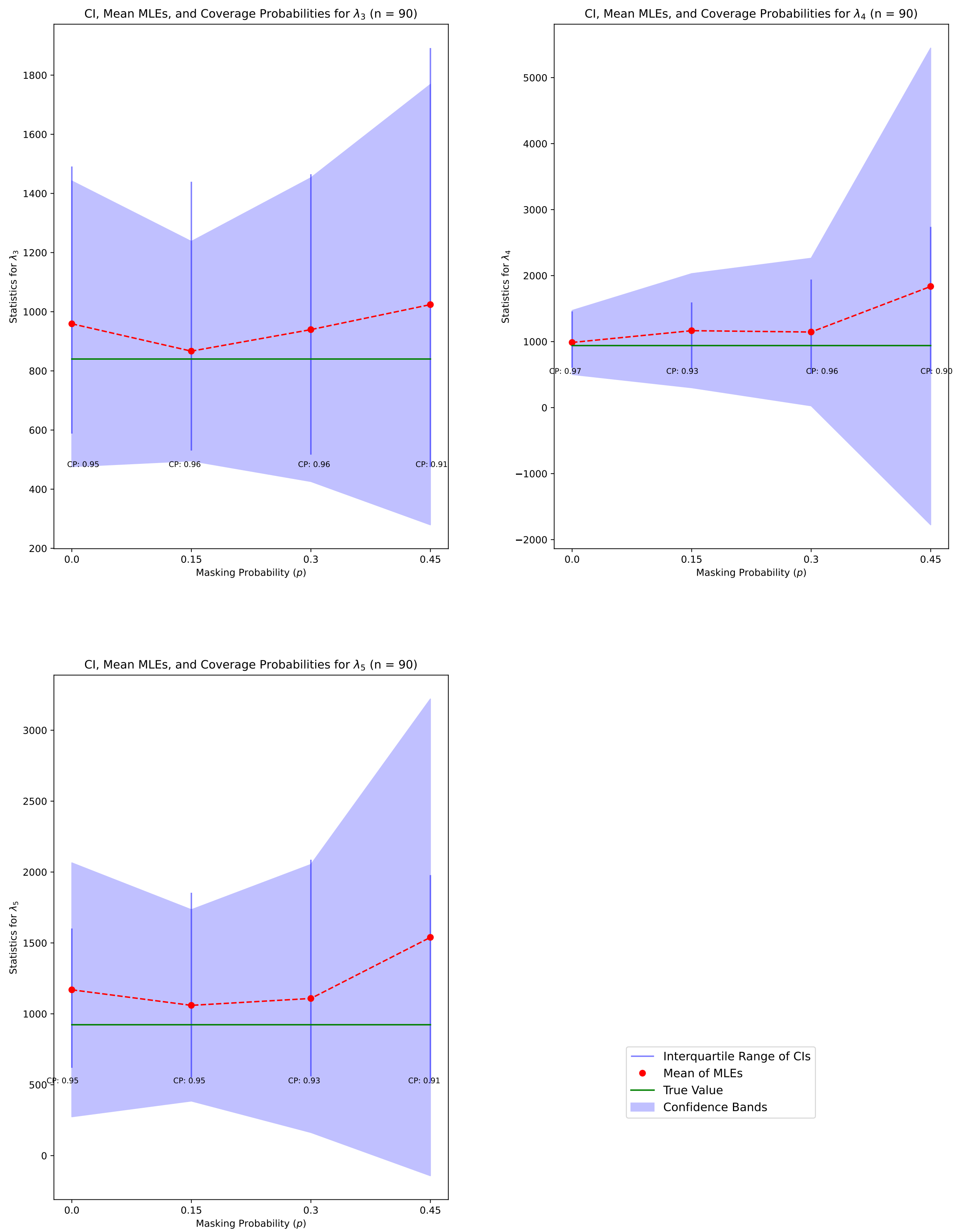

Figure 3: Component Cause of Failure Masking (p) vs Scale CI Statistics

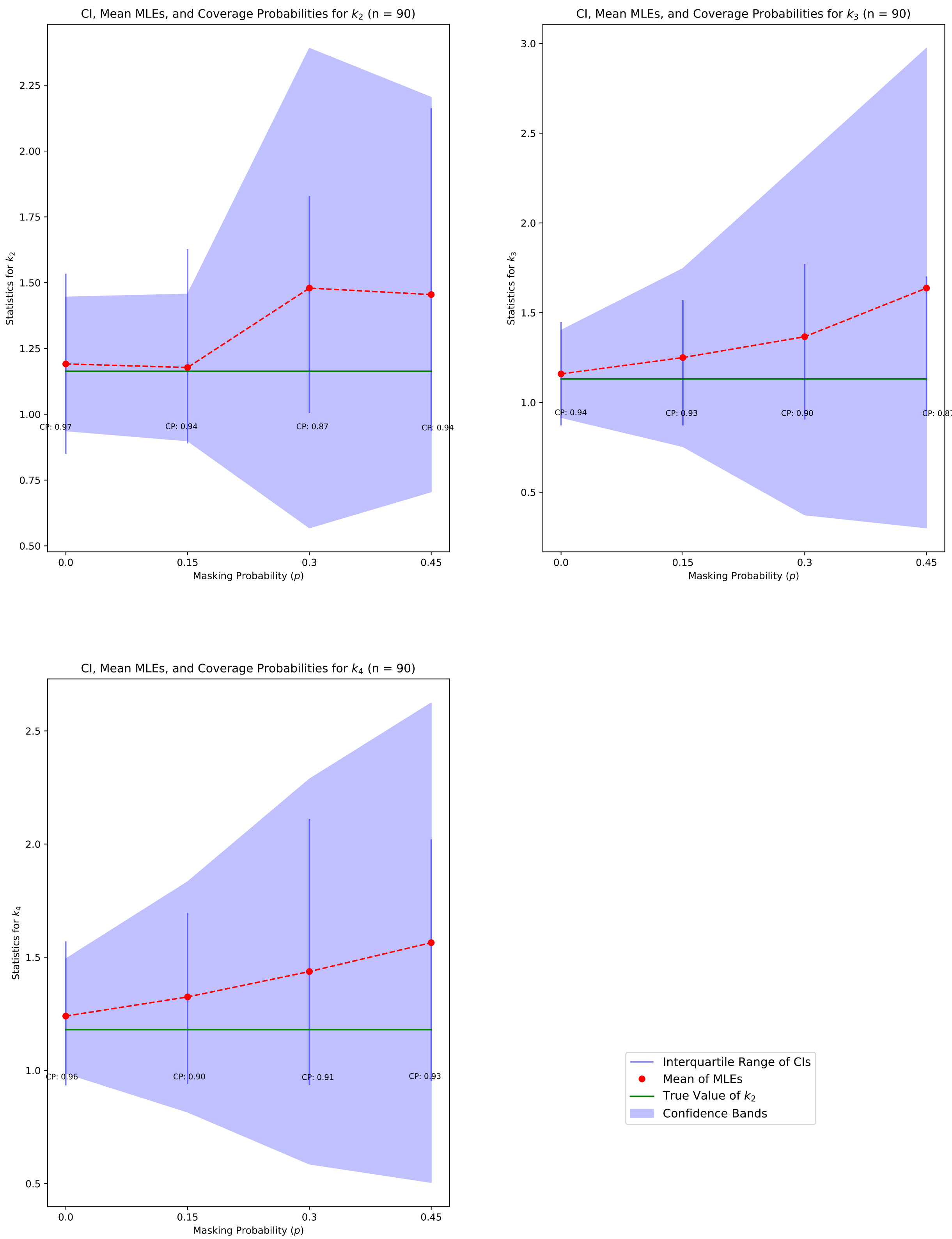

Figure 4: Component Cause of Failure Masking (p) vs Shape CI Statistics

We focus our attention on systems in which all components have increasing, or all components have decreasing, failure rates, and then we vary the scale or shape parameter of one of the components to examine the effect this has on the MLE and its sampling distribution.

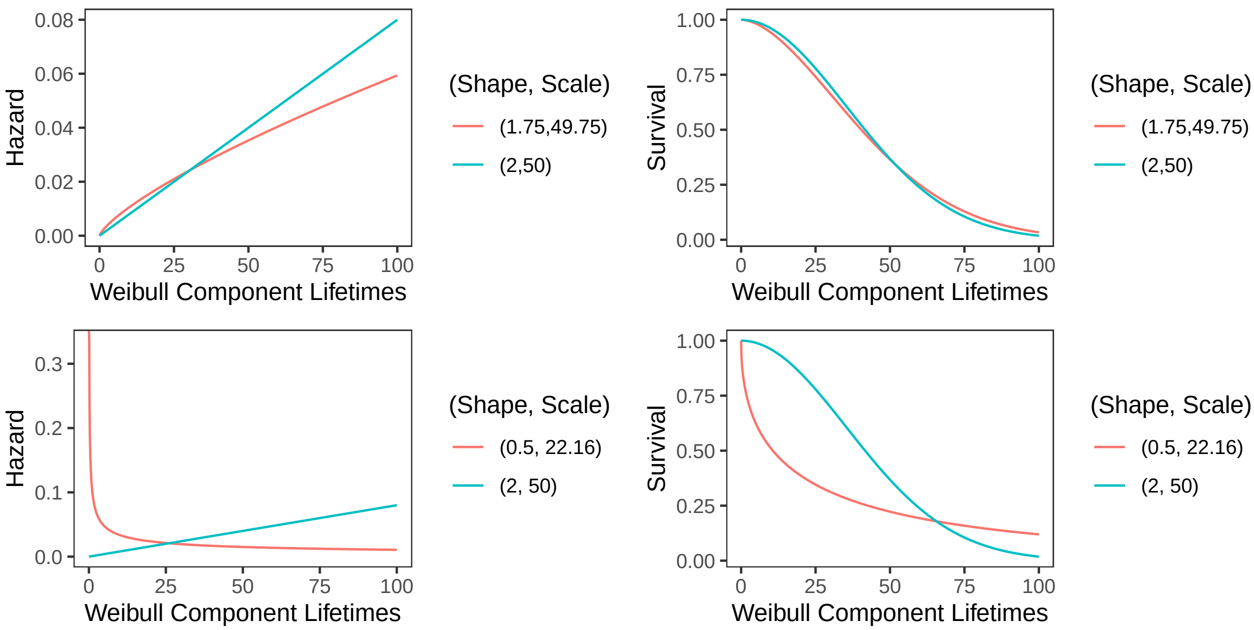

Figure 5: Two components (in series configuration). All components have the same MTTF. On the top, both components have a similiar aging process. On the bottom, the red component has a burn-in process and the blue component has an aging process.

7.4.1 Effect of Changing the MTTF of Component 3 By Changing Its Scale

We see that the MTTF is proportional to the scale parameter λ_j, which means when we decrease the scale parameter of a component, we proportionally decrease the MTTF. In this section, we will explore this phenomenon in more detail by manipulating the MTTF of component 3 and observing the effect it has on the MLE and the bootstrapped confidence intervals for component 3 and component 1.[6]

In Figure 6, we show the effect of the MTTF of component 3 on the MLE and the bootstrapped confidence intervals for the shape and scale parameters for components 1 and 3 (the component we are varying). We simulate samples with a sample size of $n = 100$, a right-censoring quantile of $q = 0.825$, and a masking probability of $p = 0.215$. (Note that while q is fixed, τ varies as we change the MTTF of component 3.) The MTTF of component 3 varies from around 300 to 1500 and MTTF of the other components, including component 2, is around 900. There are several interesting observations that we can make about Figure 6:

1. When the MTTF of component 3 is much smaller than the other components, the estimate of parameters of component 3 is precise (narrow CIs with high Probability coverage) and accurate (the MLE is close to the true value). This is because component 3 is the component cause of failure in nearly every system failure, and so the data is very informative about the parameters of component 3. Conversely, the estimates of the parameters of the other components is quite poor, with wide CIs and large positive bias. Nonetheless, the coverage probability of the CIs for the other components is still well-calibrated, which means that the CIs will contain the true value of the parameter with a probability around the specified confidence level. So, while we may not have a good point estimates for the parameters, we can still be confident that CIs contain them. That is to say, we have properly quantified our uncertainty about the parameters of the other components.

[6] Since the other components had a similiar MTTF, we will arbitrarily choose component 1 to represent the other components.

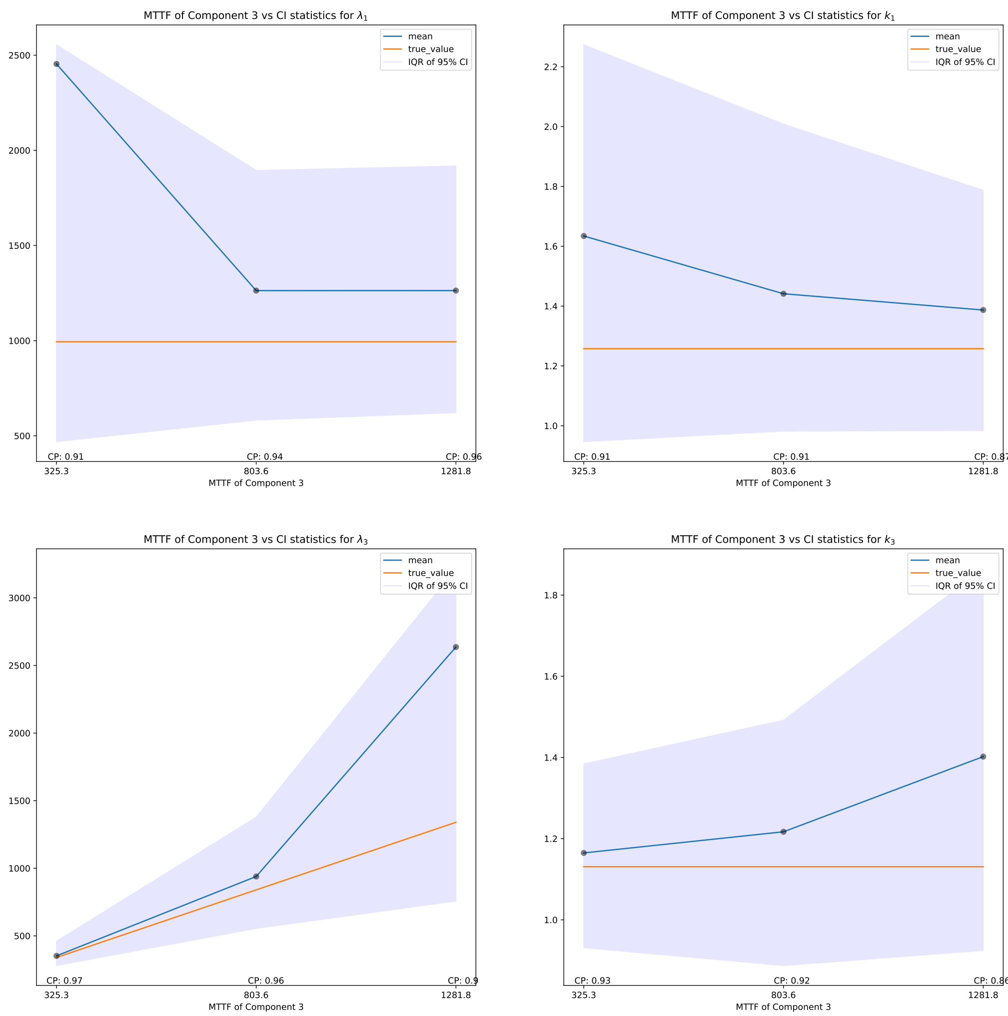

Figure 6: MTTF vs Parameter Statistics

Table 3: Mean Time To Failure (MTTF) and Probability of Component Failure of Weibull Components in Series Configuration

	MTTF	Failure Probability
Component 1	120	0.4687278
Component 2	2	0.2656353
Component 3	2	0.2656353

2. When the MTTF of component 3 is much larger than the MTTF of the other components, then component 3 is much less likely to be the component cause of failure, and with a masking probability of $p = 0.215$, it will be in the candidate set with approximately 21.5% probability, but it will generally be a false candidate. The end result is that the estimates of the parameters of component 3 are quite poor, with wide CIs and large positive bias. However, the estimates of the parameters of the other components are quite good, with narrow CIs and small positive bias. The coverage probability of the CIs for the other components are, in comparison, quite good. As the MTTF of component 3 increases and it becomes less likely to be the component cause of failure, the estimates of the parameters of the other components become more precise and accurate.

   We also see that the bias is positive for both parameters of component 3. We had not necessarily expected this, but we knew there would be a complex relationship given the presence of right-censoring and masking. When a system is right-censored, or the exact time of failure is observed but the component cause of failure is masked and component 3 is not in the candidate set, then to make component 3 more likely to not be the component cause of failure, its failure rate at that observed time is pushed down and its MTTF is pushed to the right by the MLE. Thus, $\hat{\lambda}_3$ being positively biased is expected. However, k_3 being positively biased is not necessarily expected, but the fact is, decreasing k_3 only has a small impact on the MTTF compared to the scale parameter λ_3, and the shape parameter may be more particular about when the failures occur. For example, if the shape parameter is large, then the failures may be more likely to occur at the beginning of the lifetime, which would cause the MTTF to be pushed to the right. This is

   anticipated this: from our preliminary analysis, we had expected that the bias would be positive for the scale parameter and negative for the shape parameter. We believed this because if component 3 is not the component cause of failure, then the system is more likely to fail due to the failure of one of the other components, which would cause the system to fail sooner. This would cause

7.4.2 Effect of Varying Probability of Component Failure Through the Shape Parameter

The shape parameter of the Weibull distribution has a non-linear relationship to its MTTF. The non-linearity complicates any analysis, and so instead we use a more relevant definition of reliability for a series system, the probability of a component being the cause of failure. The probability that the j^{th} component is the cause of failure is given by

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

We consider $m = 3$ Weibull components in a series configuration. We will vary the shape parameter of component 1, k_1, and observe the bias in the scale and shape parameters for components 1 and 2. The shape and scale parameters are given by

$$\begin{aligned} 0.1 \leq k_1 < 0.55 \qquad & \lambda_1 = 1 \\ k_2 = 0.5 \qquad & \lambda_2 = 1 \\ k_3 = 0.5 \qquad & \lambda_3 = 1. \end{aligned} \tag{7.5}$$

As shown by Table 3, the MTTF of component 1 can be significantly larger than the MTTF of the other components, but the probability of component 1 being the cause of failure is significantly larger. In fact, as the MTTF of component 1 increases as we decrease its shape parameter, the probability of component 1 being the cause of failure increases. This is because the shape parameter of component 1 is less than 1, and so the hazard function decreases with respect to time. This means that component 1 is more likely to fail early in its lifetime (high infant mortality) and less likely to fail later in its lifetime (low aging). This means that component 1 is more likely to be the cause of failure than the other components, even though it has a larger MTTF.

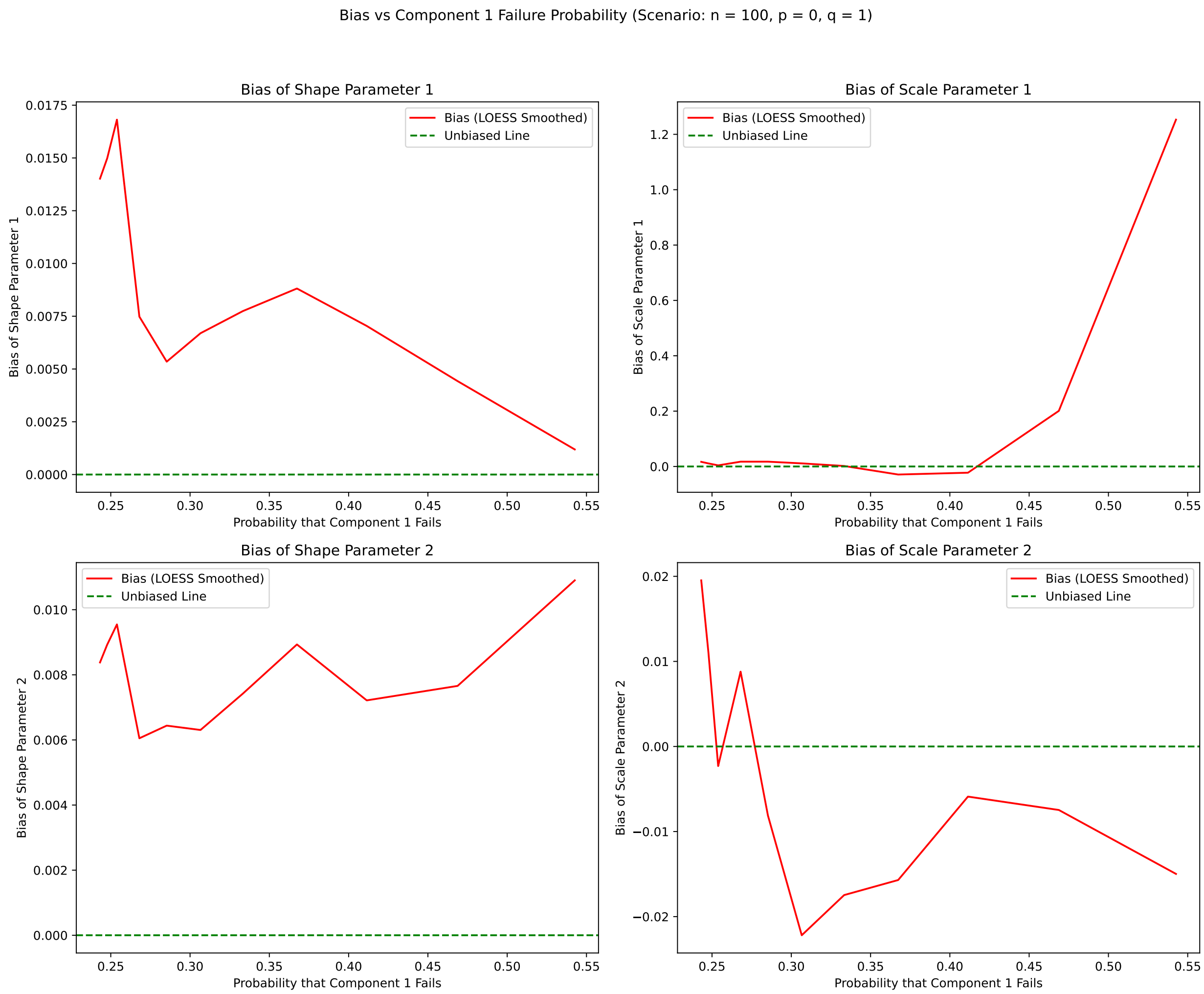

Figure 7: Probability of Component 1 Failure vs Bias (p = 0, q = 1)

7.4.2.1 Ideal Case: No Right-Censoring and No Masking of Component Cause of Failure

Figure 7 depict the bias of the scale and shape parameters of components 1 and 2 in relation to the failure probability of component 1 in the series system when we have a sample with no right censoring and no masking of the component cause of failure. This represents an ideal scenario. The green dashed line represents the true value of the parameters, and the red line shows the bias.

The bias (red line) shows a non-linear behavior as the probability of component 1 failure changes. For both the scale and shape parameters, the bias fluctuates and does not remain constant.

It is difficult to understand the pattern of the bias for the shape and scale parameters as the probability of component 1 failure changes. We expected the MLE of the shape parameter to become more positively biased as the probability of component 1 failure increases in order to nudge its MTTF down. Similarly, we expected the MLE of the scale parameter to become more negatively biased as the probability of component

Table 4: Probability of Component Failures and Mean Time To Failures As We Vary the Shape Parameter of Component 1

k_1	P_1	P_2	P_3	MTTF_1	MTTF_2	MTTF_3	System MTTF
0.1	0.54	0.23	0.23	3628800.00	2	2	0.19
0.2	0.47	0.27	0.27	120.00	2	2	0.20
0.3	0.41	0.29	0.29	9.26	2	2	0.21
0.4	0.37	0.32	0.32	3.32	2	2	0.22
0.5	0.33	0.33	0.33	2.00	2	2	0.22
0.6	0.31	0.35	0.35	1.50	2	2	0.23
0.7	0.29	0.36	0.36	1.27	2	2	0.23
0.8	0.27	0.37	0.37	1.13	2	2	0.24
0.9	0.25	0.37	0.37	1.05	2	2	0.24
1.0	0.24	0.38	0.38	1.00	2	2	0.24

1 failure increases in order to nudge its MTTF down. However, we observe the exact opposite. The bias of the shape and scale parameters for component 2 are more in line with what we expected for component 1. For the same reason, for component 2, we expected to see the bias of the shape parameter become more negative and bias of the scale parameter to become more positive as the probability of component 1 failure increases in order to nudge the MTTF of component 2 upwards, but we see the exact opposite.

To understand these graphs, we must better understand the relationship between the probability of component 1 failure and the shape parameter of the components.

In the following analysis, we fix the shape parameters of components 1 and 2 to 0.5 ($k_2 = k_3 = 0.5$) and vary the shape parameter of component 1 (k_1) to observe how the probability of component failure changes in relation to the shape parameter of component 1 (k_1). We also show the MTTF of the components and the system.

Let P_j denote the probability that the j^{th} component is the cause of the system failure ($\Pr\{K_i = j\}$), where we use the notation K_i to denote the component cause of failure for the i^{th} system. Then,

$$P_j = \int_0^\infty f_{T_i,K_i}(t, j; \boldsymbol{\theta}) dt = E_{\boldsymbol{\theta}}\left\{\frac{h_j(T_i; \boldsymbol{\theta_j})}{h_{T_i}(T_i; \boldsymbol{\theta})}\right\}.$$

Since this is a complex integral, we use numerical integration to compute the probability of component failure for different values of k_1.

In Table 4, we show these results for different values of k_1:

Here are some key observations:

- As k_1 increases, $\Pr\{K_i = 3\}$ decreases. This may seem unexpected, as it *decreases* the MTTF of component j. However, this is because the shape parameter of component 1 is less than 1, and so the hazard function decreases with respect to time. This means that component 1 is more likely to fail early in its lifetime (high infant mortality) and less likely to fail later in its lifetime (low aging). This means that component 1 is more likely to be the cause of failure than the other components, even though it has a larger MTTF.

- In order to make it more likely to see component 1 as the cause of system failure, therefore we must decrease k_1. In Figure 7, the MLE nudges k_1 down. This is what we see in the the figure, as the bias decreases (the MLE for k_1 is nudged down).

- Conversely, if you want to nudge the probability of component 2 and 3 not being a cause of component failure, you should increase k_2 and k_3. This is what we see in Figure 7 for the bias of k_2, where the MLE is nudging its bias upwards. Alternatively, you can keep it the same while only decreasing the shape parameter for component 1. This may indicate that the MLE is more sensitive to the shape parameter of component 1 than the shape parameter of component 2.

Table 5: Probability of Component Failures and Mean Time To Failures As We Vary the Scale Parameter of Component 1

λ_1	P_1	P_2	P_3	MTTF_1	MTTF_2	MTTF_3	System MTTF
1	0.33	0.33	0.33	2	2	2	0.22
2	0.26	0.37	0.37	4	2	2	0.27
3	0.22	0.39	0.39	6	2	2	0.30
4	0.20	0.40	0.40	8	2	2	0.32

Table 6: Probability of Component Failures and Mean Time To Failures As We Vary the Scale Parameter of Component 1 and the Shape Parameter of Component 1

k_1	λ_1	P_1	P_2	P_3	MTTF_1	MTTF_2	MTTF_3	System MTTF
0.25	1	0.44	0.28	0.28	24.00	2	2	0.21
0.50	1	0.33	0.33	0.33	2.00	2	2	0.22
0.75	1	0.28	0.36	0.36	1.19	2	2	0.23
0.25	2	0.39	0.31	0.31	48.00	2	2	0.24
0.50	2	0.26	0.37	0.37	4.00	2	2	0.27
0.75	2	0.20	0.40	0.40	2.38	2	2	0.30
0.25	3	0.36	0.32	0.32	72.00	2	2	0.25
0.50	3	0.22	0.39	0.39	6.00	2	2	0.30
0.75	3	0.16	0.42	0.42	3.57	2	2	0.33

To analyze the behavior of the scale parameter bias, we show the effect of varying the scale parameter of component 1 on the probability of component failure and the MTTF of the components and the system in Table 5.

Here are some key observations about Table 5:

- The relationship is more linear and intuitive when we vary the scale parameter of component 1. As the scale parameter of component 1 increases, the probability of component 1 being the cause of failure decreases, and the probability of component 2 and 3 being the cause of failure increases. This is also in alignment with the reasoning that, in this case, increasing the MTTF of a component decreases its probability of being the cause of failure. However, when we vary the shape parameter of component 1, the relationship is more complex, as previously discussed.
- To increase the probability of component 1 being the cause of failure, we must decrease the scale parameter of component 1. This is not, however, what we see in Figure 7. The MLE nudges the scale parameter of component 1 upwards (increasing bias). It could be due to a conflict between nudging the scale parameter and nudging the shape parameter. It may be a complex linear relationship the merits further investigation. This is what we see in the the figure, as the bias decreases (the MLE for the scale parameter of component 1 is nudged down).

To examine this further, we show the effect of varying the scale parameter of component 1 and the shape parameter of component 1 on the probability of component failure and the MTTF of the components and the system in Table 6.

Here are some key observations:

- For a given shape parameter, if we increase the scale parameter,

8 Conclusion

In this paper, we have presented a likelihood model for a series system with masked component...

Appendix A: R Code For Log-likelihood Function

The following code is the log-likelihood function for the Weibull series system with a likelihood model that includes masked component cause of failure and right-censoring. It is implemented in the R library `wei.series.md.c1.c2.c3` and is available on GitHub.

For clarity and brevity, we removed some of the functionality that is not relevant to the analysis in this paper.

```r
#' Generates a log-likelihood function for a Weibull series system with respect
#' to parameter `theta` (shape, scale) for masked data with candidate sets
#' that satisfy conditions C1, C2, and C3 and right-censored data.
#'
#' @param df (masked) data frame
#' @param theta parameter vector (shape1, scale1, ..., shapem, scalem)
#' @returns Log-likelihood with respect to `theta` given `df`
loglik_wei_series_md_c1_c2_c3 <- function(df, theta) {
    n <- nrow(df)
    C <- md_decode_matrix(df, candset)
    m <- ncol(C)
    delta <- df[[right_censoring_indicator]]
    t <- df[[lifetime]]
    k <- length(theta)
    shapes <- theta[seq(1, k, 2)]
    scales <- theta[seq(2, k, 2)]

    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)))
        }
    }
    s
}
```

Appendix B: R Code For Score Function

The following code is the score function (gradient of the log-likelihood function with respect to $\boldsymbol{\theta}$) for the Weibull series system with a likelihood model that includes masked component cause of failure and right-censoring. It is implemented in the R library `wei.series.md.c1.c2.c3` and is available on GitHub.

For clarity and brevity, we removed some of the functionality that is not relevant to the analysis in this paper.

```r
#' Computes the score function (gradient of the log-likelihood function) for a
#' Weibull series system with respect to parameter `theta` (shape, scale) for masked
#' data with candidate sets that satisfy conditions C1, C2, and C3 and right-censored
#' data.
#'
#' @param df (masked) data frame
#' @param theta parameter vector (shape1, scale1, ..., shapem, scalem)
#' @returns Score with respect to `theta` given `df`
score_wei_series_md_c1_c2_c3 <- function(df, theta) {
```

```r
    n <- nrow(df)
    C <- md_decode_matrix(df, candset)
    m <- ncol(C)
    delta <- df[[right_censoring_indicator]]
    t <- df[[lifetime]]
    shapes <- theta[seq(1, length(theta), 2)]
    scales <- theta[seq(2, length(theta), 2)]
    shape_scores <- rep(0, m)
    scale_scores <- rep(0, m)

    for (i in 1:n) {
        rt.term.shapes <- -(t[i]/scales)^shapes * log(t[i]/scales)
        rt.term.scales <- (shapes/scales) * (t[i]/scales)^shapes

        # Initialize mask terms to 0
        mask.term.shapes <- rep(0, m)
        mask.term.scales <- rep(0, m)

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

            numer.shapes <- 1/t[i] * (t[i]/scales[cindex])^shapes[cindex] *
                (1 + shapes[cindex] * log(t[i]/scales[cindex]))
            mask.term.shapes[cindex] <- numer.shapes/denom

            numer.scales <- (shapes[cindex]/scales[cindex])^2 * (t[i]/scales[cindex])^(shapes[cindex] -
                1)
            mask.term.scales[cindex] <- numer.scales/denom
        }

        shape_scores <- shape_scores + rt.term.shapes + mask.term.shapes
        scale_scores <- scale_scores + rt.term.scales - mask.term.scales
    }

    scr <- rep(0, length(theta))
    scr[seq(1, length(theta), 2)] <- shape_scores
    scr[seq(2, length(theta), 2)] <- scale_scores
    scr
}
```

Appendix C: R Code For Simulation of Scenarios For Assessing Bootstrapped (BCa) Confidence Intervals

The following code is the Monte-carlo simulation code for estimating the confidence intervals of the MLE using the bootstrap method.

```r
#### Setup simulation parameters here ####
theta <- c(shape1 = 1.2576, scale1 = 994.3661, shape2 = 1.1635, scale2 = 908.9458,
    shape3 = NA, scale3 = 840.1141, shape4 = 1.1802, scale4 = 940.1342,
    shape5 = 1.2034, scale5 = 923.1631)
```

```r
shapes3 <- c(1.1308)  # shape 3 true parameter values to simulate
N <- c(30, 60, 100)  # sample sizes to simulate
P <- c(0.215, 0.333)  # masking probabilities to simulate
Q <- c(0.825)  # right censoring probabilities to simulate
R <- 100  # number of simulations per scenario
B <- 1000L  # number of bootstrap samples
max_iter <- 125L  # max iterations for MLE
max_boot_iter <- 125L  # max iterations for bootstrap MLE
n_cores <- detectCores() - 1  # number of cores to use for parallel processing
filename <- "data-boot-tau-fixed-bca-p-vs-ci"  # filename prefix for output files
ci_method <- "bca"  # bootstrap CI method. See ?boot::boot.ci for details.
ci_level <- 0.95  # confidence interval level

#### Simulation code below here ####
library(tidyverse)
library(parallel)
library(boot)
library(algebraic.mle)  # for `mle_boot`
library(wei.series.md.c1.c2.c3)  # for `mle_lbfgsb_wei_series_md_c1_c2_c3` etc

file.meta <- paste0(filename, ".txt")
file.csv <- paste0(filename, ".csv")
if (file.exists(file.meta)) {
    stop("File already exists: ", file.meta)
}
if (file.exists(file.csv)) {
    stop("File already exists: ", file.csv)
}

shapes <- theta[seq(1, length(theta), 2)]
scales <- theta[seq(2, length(theta), 2)]
m <- length(shapes)

sink(file.meta)
cat("boostrap of confidence intervals:\n")
cat("   simulated on: ", Sys.time(), "\n")
cat("   type: ", ci_method, "\n")
cat("weibull series system:\n")
cat("   number of components: ", m, "\n")
cat("   scale parameters: ", scales, "\n")
cat("   shape parameters: ", shapes, "\n")
cat("simulation parameters:\n")
cat("   shapes3: ", shapes3, "\n")
cat("   N: ", N, "\n")
cat("   P: ", P, "\n")
cat("   Q: ", Q, "\n")
cat("   R: ", R, "\n")
cat("   B: ", B, "\n")
cat("   max_iter: ", max_iter, "\n")
cat("   max_boot_iter: ", max_boot_iter, "\n")
cat("   n_cores: ", n_cores, "\n")
sink()
```

```r
for (shape3 in shapes3) {
    for (n in N) {
        for (p in P) {
            for (q in Q) {
                shapes[3] <- shape3
                theta["shape3"] <- shape3

                cat("[starting scenario: shape3 = ", shape3, ", n = ",
                  n, ", p = ", p, ", q = ", q, "]\n")
                tau <- qwei_series(p = q, scales = scales, shapes = shapes)

                # we compute R MLEs for each scenario
                shapes.mle <- matrix(NA, nrow = R, ncol = m)
                scales.mle <- matrix(NA, nrow = R, ncol = m)
                shapes.lower <- matrix(NA, nrow = R, ncol = m)
                shapes.upper <- matrix(NA, nrow = R, ncol = m)
                scales.lower <- matrix(NA, nrow = R, ncol = m)
                scales.upper <- matrix(NA, nrow = R, ncol = m)
                logliks <- rep(0, R)

                iter <- 0L
                repeat {
                  retry <- FALSE
                  tryCatch({
                    repeat {
                      df <- generate_guo_weibull_table_2_data(shapes = shapes,
                        scales = scales, n = n, p = p, tau = tau)

                      sol <- mle_lbfgsb_wei_series_md_c1_c2_c3(theta0 = theta,
                        df = df, hessian = FALSE, control = list(maxit = max_iter,
                          parscale = theta))
                      if (sol$convergence == 0) {
                        break
                      }
                      cat("[", iter, "] MLE did not converge, retrying.\n")
                    }

                    mle_solver <- function(df, i) {
                      mle_lbfgsb_wei_series_md_c1_c2_c3(theta0 = sol$par,
                        df = df[i, ], hessian = FALSE, control = list(maxit = max_boot_iter,
                          parscale = sol$par))$par
                    }

                    # do the non-parametric bootstrap
                    sol.boot <- boot(df, mle_solver, R = B, parallel = "multicore",
                      ncpus = n_cores)
                  }, error = function(e) {
                    cat("[error] ", conditionMessage(e), "\n")
                    cat("[retrying scenario: n = ", n, ", p = ", p,
                      ", q = ", q, "\n")
                    retry <<- TRUE
                  })
                  if (retry) {
```

```r
                    next
                  }
                  iter <- iter + 1L
                  shapes.mle[iter, ] <- sol$par[seq(1, length(theta),
                    2)]
                  scales.mle[iter, ] <- sol$par[seq(2, length(theta),
                    2)]
                  logliks[iter] <- sol$value

                  tryCatch({
                    ci <- confint(mle_boot(sol.boot), type = ci_method,
                      level = ci_level)
                    shapes.ci <- ci[seq(1, length(theta), 2), ]
                    scales.ci <- ci[seq(2, length(theta), 2), ]
                    shapes.lower[iter, ] <- shapes.ci[, 1]
                    shapes.upper[iter, ] <- shapes.ci[, 2]
                    scales.lower[iter, ] <- scales.ci[, 1]
                    scales.upper[iter, ] <- scales.ci[, 2]
                  }, error = function(e) {
                    cat("[error] ", conditionMessage(e), "\n")
                  })
                  if (iter%%10 == 0) {
                    cat("[iteration ", iter, "] shapes = ", shapes.mle[iter,
                      ], "scales = ", scales.mle[iter, ], "\n")
                  }

                  if (iter == R) {
                    break
                  }
                }

                df <- data.frame(n = rep(n, R), rep(shape3, R), p = rep(p,
                  R), q = rep(q, R), tau = rep(tau, R), B = rep(B,
                  R), shapes = shapes.mle, scales = scales.mle, shapes.lower = shapes.lower,
                  shapes.upper = shapes.upper, scales.lower = scales.lower,
                  scales.upper = scales.upper, logliks = logliks)

                write.table(df, file = file.csv, sep = ",", row.names = FALSE,
                  col.names = !file.exists(file.csv), append = TRUE)
            }
        }
    }
}
```

Appendix D: Bernoulli Candidate Set Model

```r
#' Bernoulli candidate set model is a particular type of *uninformed* model.
#' This model satisfies conditions C1, C2, and C3.
#' The failed component will be in the corresponding candidate set with
#' probability 1, and the remaining components will be in the candidate set
#' with probability `p` (the same probability for each component). `p`
#' may be different for each system, but it is assumed to be the same for
```

```r
#' each component within a system, so `p` can be a vector such that the
#' length of `p` is the number of systems in the data set (with recycling
#' if necessary).
#'
#' @param df masked data.
#' @param p a vector of probabilities (p[j] is the probability that the jth
#'          system will include a non-failed component in its candidate set,
#'          assuming the jth system is not right-censored).
#' @importFrom md.tools md_decode_matrix md_encode_matrix
#' @importFrom dplyr %>% bind_cols
#' @export
md_bernoulli_cand_c1_c2_c3 <- function(df, p) {
    n <- nrow(df)
    p <- rep(p, length.out = n)
    Tm <- md_decode_matrix(df, comp)
    m <- ncol(Tm)
    Q <- matrix(p, nrow = n, ncol = m)
    Q[cbind(1:n, apply(Tm, 1, which.min))] <- 1
    Q[!df[[right_censoring_indicator]], ] <- 0
    df %>%
        bind_cols(md_encode_matrix(Q, prob))
}
```

References

Casella, G. and Berger, R. L. (2002). *Statistical Inference*. Duxbury Advanced Series.

Efron, B. (1987). Better bootstrap confidence intervals. *Journal of the American Statistical Association*, 82(397):171–185.

Efron, B. and Tibshirani, R. J. (1994). *An introduction to the bootstrap*. CRC press.

Engelhardt, B. (1992). Introduction to probability and mathematical statistics. page 294.

Guess, F. M., Hodgson, T. J., and Usher, J. S. (1991). Estimating system and component reliabilities under partial information on cause of failure. *Journal of Statistical Planning and Inference*, 29:75–85.

Guo, H., Niu, P., and Szidarovszky, F. (2013). Estimating component reliabilities from incomplete system failure data. *Annual Reliability and Maintainability Symposium (RAMS)*, pages 1–6.

Klein, J. P. and Moeschberger, M. L. (2005). *Survival analysis: techniques for censored and truncated data*. Springer Science & Business Media.

McLachlan, G. and Krishnan, T. (2007). *The EM Algorithm and Extensions*. John Wiley & Sons.