# 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

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 $i^{\text{th}}$ system denoted by the random variable $T_i$. The lifetime of the $j^{\text{th}}$ component in the $i^{\text{th}}$ 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 $i^{\text{th}}$ system is given by the component with the shortest lifetime,

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

There are three particularly important distribution functions in reliability analysis: the reliability function, the probability density function, and the hazard function. The reliability function, $R_{T_i}(t)$, is the probability that the $i^{\text{th}}$ system has a lifespan larger than a duration $t$,

$$R_{T_i}(t) = \Pr\{T_i > t\} \tag{1}$$

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 $\Delta t$ given that no failure occurs before time $t$ is given by

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

The failure rate is given by the dividing this equation by the length of the time interval, $\Delta t$:

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

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

$$\begin{aligned} h_{T_i}(t) &= \lim_{\Delta t \to 0} \frac{\Pr\{t < T_i < t + \Delta t\}}{\Delta t} \frac{1}{\Pr\{T_i > t\}} \\ &= \frac{f_{T_i}(t)}{R_{T_i}(t)}. \end{aligned} \tag{2}$$

\end{definition}

The lifetime of the $j^{\text{th}}$ 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 $X$ is parameterized by a particular $\boldsymbol{\theta}$, we denote the reliability function by $R_X(t; \boldsymbol{\theta})$, and the same for the other distribution functions. As a special case, for the components in the series system, we subscript by their labels, e.g, the $j^{\text{th}}$ component's pdf is denoted by $f_j(t; \boldsymbol{\theta_j})$.

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 2.1.** *The series system has a reliability function given by*

$$R_{T_i}(t;\boldsymbol{\theta}) = \prod_{j=1}^{m} R_j(t;\boldsymbol{\theta_j}). \tag{3}$$

*Proof.* The reliability function is defined as

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

which may be rewritten as

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

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

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

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

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

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

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

□

Theorem 2.1 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 2.2.** *The series system has a pdf given by*

$$f_{T_i}(t;\boldsymbol{\theta}) = \sum_{j=1}^{m} f_j(t;\boldsymbol{\theta_j}) \prod_{\substack{k=1\\k\neq j}}^{m} R_k(t;\boldsymbol{\theta_j}). \tag{4}$$

*Proof.* By definition, the pdf may be written as

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

By the product rule, this may be rewritten as

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

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

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

which simplifies to

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

□

Theorem 2.2 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 2.3.** *The series system has a hazard function given by*

$$h_{T_i}(t;\boldsymbol{\theta}) = \sum_{j=1}^{m} h_j(t;\boldsymbol{\theta_j}). \tag{5}$$

*Proof.* By Equation (2), the $i^{\text{th}}$ series system lifetime has a hazard function defined as

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

Plugging in expressions for these functions results in

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

which can be simplified to

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

□

Theorem 2.3 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,

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

and we can rearrange this to get

$$\begin{aligned} f_{T_i}(t;\boldsymbol{\theta}) &= h_{T_i}(t;\boldsymbol{\theta}) R_{T_i}(t;\boldsymbol{\theta}) \\ &= \left\{\sum_{j=1}^{m} h_j(t;\boldsymbol{\theta_j})\right\}\left\{\prod_{j=1}^{m} R_j(t;\boldsymbol{\theta_j})\right\}, \end{aligned} \tag{6}$$

which we sometimes find to be a more convenient form than Equation (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 2.1.** 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 2.4.** *The joint pdf of the component cause of failure $K_i$ and series system lifetime $T_i$ is given by*

$$f_{K_i,T_i}(j,t;\boldsymbol{\theta}) = h_j(t;\boldsymbol{\theta_j}) \prod_{l=1}^{m} R_l(t;\boldsymbol{\theta}), \tag{7}$$

*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 $j^{\text{th}}$ component is the cause of failure is given by

$$\Pr\{K_i = j\} = E_{\boldsymbol{\theta}}\left[\frac{h_j(T_i;\boldsymbol{\theta_j})}{\sum_{l=1}^{m} h_l(T_i;\boldsymbol{\theta_l})}\right]. \tag{8}$$

*Proof.* The probability the $j^{\text{th}}$ 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 2.4, this is equivalent to

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

□

## 2.2 System and Component Reliabilities

A common measure of reliability is mean time to failure (MTTF). The MTTF is defined as the expectation of the lifetime,

$$\text{MTTF} = E_{\boldsymbol{\theta}}\{T_i\}, \tag{9}$$

which if certain assumptions are satisfied[1] is equivalent to the integration of the reliability function over its support.

While the MTTF provides a summary measure of reliability, it is not a complete description. Depending on the failure characteristics, MTTF can be misleading. For example, a system that has a high likelihood of failing early in its life may still have a large MTTF if it is fat-tailed.[2]

The reliability of the components in the series system determines the reliability of the system. We denote the MTTF of the $j^{\text{th}}$ component by $\text{MTTF}_j$ and, according to Equation (8), the probability that the $j^{\text{th}}$ component is the cause of failure is given by $\Pr\{K_i = j\}$. In a well-designed series system, there is no component that is the "weakest link" that either has a much shorter MTTF or a much higher probability of being the component cause of failure than any of the other components, e.g., $\Pr\{K_i = j\} \approx \Pr\{K_i = k\}$ and and $\text{MTTF}_j \approx \text{MTTF}_k$ for all $j$ and $k$. This just means that the components should have similar reliabilities and failure characteristics.

We use these results in the simulation study in Section 8, where we assess the sensitivity of the MLE with respect to varying the reliability of one of the Weibull components. We vary its reliability in two different ways:

1. We vary its shape parameter (keeping its scale parameter constant), which determines the failure characteristics of the component and also affects its MTTF.

2. We vary its scale parameter (keeping its shape parameter constant), which scales its MTTF while retaining the same failure characteristics.

# 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.

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

[2] A "fat-tailed" distribution refers to a probability distribution with tails that decay more slowly than those of the exponential family, such as the case with the Weibull when its shape parameter is greater than 1. This means that extreme values are more likely to occur, and the distribution is more prone to "black swan" events or rare occurrences. In the context of reliability, a fat-tailed distribution might imply a higher likelihood of unusually long lifetimes, which can skew measures like the MTTF. See, for example, Taleb (2007).

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 $i^{\text{th}}$ system by $\tau_i$. We do not directly observe the system lifetime, $T_i$, but rather, we observe the right-censored lifetime, $S_i$, which is given by

$$S_i = \min\{\tau_i, T_i\}, \tag{10}$$

We also observe a right-censoring indicator, $\delta_i$, which is given by

$$\delta_i = 1_{T_i < \tau_i} \tag{11}$$

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

The masked data generation process is illustrated by Figure 1.

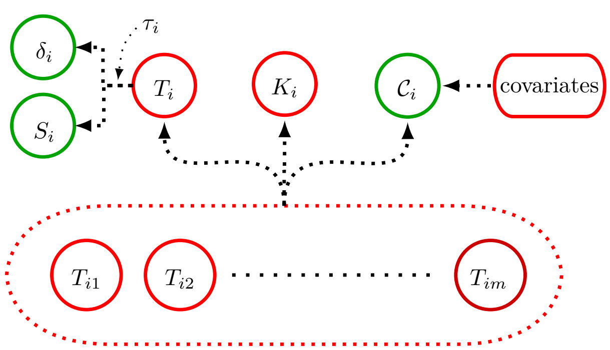


Figure 1: This figure showcases a dependency graph of the generative model for $D_i = (S_i, \delta_i, \mathcal{C}_i)$. The elements in green are observed in the sample, while the elements in red are unobserved (latent). We see that $\mathcal{C}_i$ is related to both the unobserved component lifetimes $T_{i1}, \ldots, T_{im}$ 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 $\mathcal{C}_i$.

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 ($\delta_i$) | 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:

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

where $s_i$ is the observed system lifetime of the $i^{\text{th}}$ system, $\delta_i$ is the observed right-censoring indicator of the $i^{\text{th}}$ system, and $c_i$ is the observed candidate set of the $i^{\text{th}}$ 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 $i^{\text{th}}$ 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 3.1.** *The likelihood contribution of the i-th system is given by*

$$L_i(\boldsymbol{\theta}) = R_{T_i}(s_i; \boldsymbol{\theta}) \Big( \beta_i \sum_{j \in c_i} h_j(s_i; \boldsymbol{\theta_j}) \Big)^{\delta_i} \tag{12}$$

*where* $R_{T_i}(s_i; \boldsymbol{\theta}) = \prod_{j=1}^{m} R_j(s_i; \boldsymbol{\theta_j})$ *is the reliability 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 $i^{\text{th}}$ 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 $i^{\text{th}}$ 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 $i^{\text{th}}$ 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 2.4, 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 $j^{\text{th}}$ 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

$$\begin{aligned} f_{T_i,K_i,\mathcal{C}_i}(t_i, j, c_i; \boldsymbol{\theta}) &= f_{T_i,K_i}(t_i, k; \boldsymbol{\theta}) \Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j\} \\ &= h_j(t_i; \boldsymbol{\theta_j}) R_{T_i}(t_i; \boldsymbol{\theta}) \Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j\}. \end{aligned} \tag{13}$$

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 (13),

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

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

$$f_{T_i,\mathcal{C}_i}(t_i, c_i; \boldsymbol{\theta}) = R_{T_i}(t_i; \boldsymbol{\theta}) \sum_{j \in c_i} \Biggl\{ h_j(t_i; \boldsymbol{\theta_j}) \Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j\} \Biggr\}. \tag{14}$$

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 (14), 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 $\beta_i$ 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 $i^{\text{th}}$ 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 $c_i$:

$$L_i(\boldsymbol{\theta}) = R_{T_i}(t_i; \boldsymbol{\theta}) \sum_{j \in c_i} h_j(t_i; \boldsymbol{\theta_j}). \tag{15}$$

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 (15).

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

$$L_i(\boldsymbol{\theta}) = R_{T_i}(s_i; \boldsymbol{\theta}). \tag{16}$$

*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 $s_i$,

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

□

When we combine the two likelihood contributions, we obtain the likelihood contribution for the $i^{\text{th}}$ system shown in Theorem 3.1,

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

We use this result in Section 4 to derive the maximum likelihood estimator (MLE) of $\boldsymbol{\theta}$.

### 3.3 Identifiability and Convergence Issues

In our likelihood model, masking and right-censoring can lead to issues related to identifiability and flat likelihood regions. Identifiability refers to the unique mapping of the model parameters to the likelihood function, and lack of identifiability can lead to multiple sets of parameters that explain the data equally well, making inference about the true parameters challenging (Lehmann and Casella, 1998), while flat likelihood regions can complicate convergence (Wu, 1983).

In our simulation study, we address these challenges in a pragmatic way. Specifically, failure to converge to a solution within a maximum of 125 iterations is interpreted as evidence of the aforementioned issues, leading to the discarding of the sample, with the process then repeated with a new synthetic sample. Note, however, that in Section 5 where we discuss the bias-corrected and accelerated (BCa) bootstrap method for constructing confidence intervals, we do not discard any resamples.

This strategy helps ensure the robustness of the results, while acknowledging the inherent complexities of likelihood-based estimation in models characterized by masking and right-censoring.

## 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 (Bain and 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

$$L(\hat{\boldsymbol{\theta}}) = \max_{\boldsymbol{\theta}\in\boldsymbol{\Omega}} L(\boldsymbol{\theta}), \tag{17}$$

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

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{n} \ell_i(\boldsymbol{\theta}), \tag{18}$$

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

$$\ell_i(\boldsymbol{\theta}) = \sum_{j=1}^{m} \log R_j(s_i;\boldsymbol{\theta_j}) + \delta_i \log\Big(\sum_{j\in c_i} h_j(s_i;\boldsymbol{\theta_j})\Big). \tag{19}$$

*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 (12), we consider these two cases of $\delta_i$ seperately to obtain the result in Theorem 4.1.

**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 ($\delta_i = 1$),

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

We replace $R_{T_i}(s_i; \boldsymbol{\theta})$ with its component-wise definition and by Condition 3, we may discard[3] 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 4.1. $\square$

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.,

$$\frac{\partial}{\partial \theta_j} \ell(\boldsymbol{\theta}) = 0, \tag{20}$$

for each component $\theta_j$ of the parameter $\boldsymbol{\theta}$ (Bain and 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.

[3]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.

- 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 for a potentially biased MLE, the BCa method may be 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 well-calibrated 95% confidence interval contains the true value around 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. We want confidence intervals to be as narrow as possible while still having a coverage probability close to the nominal level, 95%.

### 5.1 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

The Weibull distribution, introduced by Waloddi Weibull in 1937, has been instrumental in reliability analysis due to its ability to model a wide range of failure behaviors. Reflecting on its utility, Weibull modestly noted that it "[...] may sometimes render good service." (Source: "New Weibull Handbook"). In the context of our study, we utilize the Weibull to model a system as originating from Weibull components in a series configuration, producing a specific form of the likelihood model described in Section 3, which deals with challenges such as right censoring and masked component cause of failure. In Section 8, we conduct a simulation study to assess the sensivity of this model to variations in masking probabilities, sample size, and differentiated component reliabilities.

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

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

where $\lambda_j > 0$ is the scale parameter and $k_j > 0$ is the shape parameter. The $j^{\text{th}}$ component has a reliability function, pdf, and hazard function given respectively by

$$R_j(t;\lambda_j,k_j) = \exp\biggl\{-\biggl(\frac{t}{\lambda_j}\biggr)^{k_j}\biggr\}, \tag{21}$$

$$f_j(t;\lambda_j,k_j) = \frac{k_j}{\lambda_j}\biggl(\frac{t}{\lambda_j}\biggr)^{k_j-1}\exp\biggl\{-\biggl(\frac{t}{\lambda_j}\biggr)^{k_j}\biggr\}, \tag{22}$$

$$h_j(t;\lambda_j,k_j) = \frac{k_j}{\lambda_j}\biggl(\frac{t}{\lambda_j}\biggr)^{k_j-1}. \tag{23}$$

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

- $k_j < 1$ indicates infant mortality. An example of how this might arise is a result of defective components being weeded out early, and the remaining components surviving for a much longer time.
- $k_j = 1$ indicates random failures (independent of age). An example of how this might arise is a result of random shocks to the system, but otherwise the system is age-independent.[4]
- $k_j > 1$ indicates wear-out failures. An example of how this might arise is a result of components wearing as they age

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

$$R_{T_i}(t;\boldsymbol{\theta}) = \exp\biggl\{-\sum_{j=1}^{m}\biggl(\frac{t}{\lambda_j}\biggr)^{k_j}\biggr\}. \tag{24}$$

*Proof.* By Theorem 2.1,

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

Plugging in the Weibull component reliability functions obtains the result

$$\begin{aligned}R_{T_i}(t;\boldsymbol{\theta}) &= \prod_{j=1}^{m}\exp\biggl\{-\biggl(\frac{t}{\lambda_j}\biggr)^{k_j}\biggr\}\\ &= \exp\biggl\{-\sum_{j=1}^{m}\biggl(\frac{t}{\lambda_j}\biggr)^{k_j}\biggr\}.\end{aligned}$$

□

The Weibull series system's hazard function is given by

$$h_{T_i}(t;\boldsymbol{\theta}) = \sum_{j=1}^{m}\frac{k_j}{\lambda_j}\biggl(\frac{t}{\lambda_j}\biggr)^{k_j-1}, \tag{25}$$

whose proof follows from Theorem 2.3. The pdf of the series system is given by

$$f_{T_i}(t;\boldsymbol{\theta}) = \biggl\{\sum_{j=1}^{m}\frac{k_j}{\lambda_j}\biggl(\frac{t}{\lambda_j}\biggr)^{k_j-1}\biggr\}\exp\biggl\{-\sum_{j=1}^{m}\bigl(\frac{t}{\lambda_j}\bigr)^{k_j}\biggr\}. \tag{26}$$

*Proof.* By definition,

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

Plugging in the failure rate and reliability functions given respectively by Equations (24) and (25) completes the proof. □

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

### 6.1 Special Case: Series System That Is Weibull

A series system composed of Weibull components is not generally Weibull unless the shape parameters of the components are homogeneous.

**Theorem 6.1.** *If the shape parameters of the components are identical, then the series system is Weibull with a shape parameter $k$ given by the shape parameter of the components and a scale parameter $\lambda$ given by*

$$\lambda = \left(\sum_{j=1}^{m} \lambda_j^{-k}\right)^{-1/k}, \tag{27}$$

where $\lambda_j$ is the scale parameter of the $j^{\text{th}}$ component.

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

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

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

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

which has the solution

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

□

**Theorem 6.2.** *If a series system has Weibull components with identical shape parameters, the component cause of failure is conditionally independent of the system failure time:*

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

*Proof.* The conditional probability of the $j^{\text{th}}$ component being the cause of failure given the system failure time is given by

$$\begin{aligned}
\Pr\{K_i = j | T_i = t\} &= \frac{f_{K_i,T_i}(j, t; \boldsymbol{\theta})}{f_{T_i}(t; \boldsymbol{\theta})} = \frac{h_j(t; k, \lambda_j) R_{T_i}(t; \boldsymbol{\theta})}{h_{T_i}(t; \boldsymbol{\theta_j}) R_{T_i}(t; \boldsymbol{\theta})} \\
&= \frac{h_j(t; k, \lambda_j)}{\sum_{l=1}^{m} h_l(t; k, \lambda_l)} = \frac{\frac{k}{\lambda_j}\left(\frac{t}{\lambda_j}\right)^{k-1}}{\sum_{l=1}^{m} \frac{k}{\lambda_l}\left(\frac{t}{\lambda_l}\right)^{k-1}} = \frac{\left(\frac{1}{\lambda_j}\right)^k}{\sum_{l=1}^{m}\left(\frac{1}{\lambda_l}\right)^k}.
\end{aligned}$$

□

In Section 2.2, we discussed the concept of reliability. In the case of Weibull components, the MTTF of the $j^{\text{th}}$ component is given by

$$\text{MTTF}_j = \lambda_j \Gamma\left(1 + \frac{1}{k_j}\right), \tag{28}$$

where $\Gamma$ is the gamma function.

We mentioned that the MTTF can sometimes be a poor measure of reliability, e.g., the MTTF and the probability of failing early can be large. The Weibull is a good example of this phenomenon. If $k > 1$, the Weibull is a fat-tailed distribution, and it will exhibit both a large MTTF and a high

Components may have similar MTTFs, but some components may be more likely to fail early and others may be more likely to fail late, depending upon their shapes. A good measure of the *relative* reliability of the components is given by the probability of component failure given by Equation (8).

In a well-designed series system, the component failure characteristics are similar: they have a similar MTTF and a similar probability of being the component cause of failure, i.e., they have similar shapes and scales, so that system failures are not dominated by some subset of components.

As we discussed in the previous section, if the shapes are similar, the series system is approximately Weibull. In our simulation study, we analyze a series system with Weibull components that have similar shape parameters (except in those studies where we vary the shape parameters to test the sensitivity of the MLE to different component failure characteristics).

## 6.2 Likelihood Model

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 $i^{\text{th}}$ system is given by the following theorem.

**Theorem 6.3.** *Let $\delta_i$ 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*

$$L_i(\boldsymbol{\theta}) = \begin{cases} \exp\Bigl\{-\sum_{j=1}^m \bigl(\frac{t_i}{\lambda_j}\bigr)^{k_j}\Bigr\} \beta_i \sum_{j \in c_i} \frac{k_j}{\lambda_j}\bigl(\frac{t_i}{\lambda_j}\bigr)^{k_j - 1} & \text{if } \delta_i = 1, \\ \exp\bigl\{-\sum_{j=1}^m \bigl(\frac{t_i}{\lambda_j}\bigr)^{k_j}\bigr\} & \text{if } \delta_i = 0. \end{cases} \tag{29}$$

*Proof.* By Theorem 3.1, 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 (24), the system reliability function $R_{T_i}$ is given by

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

and by Equation (23), 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 6.1.** *The log-likelihood contribution of the $i$-th system is given by*

$$\ell_i(\boldsymbol{\theta}) = -\sum_{j=1}^m \left(\frac{t_i}{\lambda_j}\right)^{k_j} + \delta_i \log\left(\sum_{j \in c_i} \frac{k_j}{\lambda_j}\left(\frac{t_i}{\lambda_j}\right)^{k_j - 1}\right) \tag{30}$$

*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 (20), 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 6.4.** *The score function of the log-likelihood contribution of the $i$-th Weibull series system is given by*

$$\nabla \ell_i(\boldsymbol{\theta}) = \left( \frac{\partial \ell_i(\boldsymbol{\theta})}{\partial k_1}, \frac{\partial \ell_i(\boldsymbol{\theta})}{\partial \lambda_1}, \cdots, \frac{\partial \ell_i(\boldsymbol{\theta})}{\partial k_m}, \frac{\partial \ell_i(\boldsymbol{\theta})}{\partial \lambda_m} \right)', \tag{31}$$

*where*

$$\frac{\partial \ell_i(\boldsymbol{\theta})}{\partial k_r} = -\left(\frac{t_i}{\lambda_r}\right)^{k_r} \log\left(\frac{t_i}{\lambda_r}\right) + \frac{\frac{1}{t_i}\left(\frac{t_i}{\lambda_r}\right)^{k_r}\left(1 + k_r \log\left(\frac{t_i}{\lambda_r}\right)\right)}{\sum_{j \in c_i} \frac{k_j}{\lambda_j}\left(\frac{t_i}{\lambda_j}\right)^{k_j - 1}} 1_{\delta_i = 1 \wedge r \in c_i} \tag{32}$$

*and*

$$\frac{\partial \ell_i(\boldsymbol{\theta})}{\partial \lambda_r} = \frac{k_r}{\lambda_r}\left(\frac{t_i}{\lambda_r}\right)^{k_r} - \frac{\left(\frac{k_r}{\lambda_r}\right)^2 \left(\frac{t_i}{\lambda_r}\right)^{k_r - 1}}{\sum_{j \in c_i} \frac{k_j}{\lambda_j}\left(\frac{t_i}{\lambda_j}\right)^{k_j - 1}} 1_{\delta_i = 1 \wedge r \in c_i} \tag{33}$$

The result follows from taking the partial derivatives of the log-likelihood contribution of the $i$-th system given by Equation (29). 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

$$\nabla \ell(\boldsymbol{\theta}) = \sum_{i=1}^{n} \nabla \ell_i(\boldsymbol{\theta}). \tag{34}$$

# 7 Weibull Series Model with Homogeneous Shape Parameters

Before condcuting a simulation study for the full Weibull series model, where no assumptions about the shape parameters of the components are made, we first consider the special case where the shape parameters are homogeneous, i.e., $k_1 = \cdots = k_m = k$. This reduces the series system to a known distribution, the Weibull distribution, which is particularly useful for analysis and interpretation, especially in the context of small samples, since instead of estimating $2m$ parameters we only need to estimate $m + 1$ parameters, one shape parameter and $m$ scale parameters.

The log-likelihood function for the reduced series system, $\ell_R$, the series system with homogeneous shape parameters, is given by

$$\ell_R(k, \lambda_1, \lambda_2, \cdots, \lambda_m) = \ell(k, \lambda_1, k, \lambda_2, \ldots, k, \lambda_m),$$

i.e., we just plug the same shape parameter $k$ into each component of the series system, and the same is done for the score and hessian of the log-likelihood functions.

We wish to conduct a hypothesis test to determine if there is statistically significant evidence that the shape parameters are homogeneous:

$$H_0 : \text{all shape parameters equal.}$$
$$H_1 : \text{not all shape parameters are equal.}$$

We will use the likelihood ratio test (LRT) to test this hypothesis. The LRT statistic is given by

$$\Lambda = -2 \log \frac{\ell_R}{\ell_F},$$

where $\ell_R$ is the log-likelihood function of the null (reduced) model and $\ell_F$ is the log-likelihood function of the full model. Under the null model, the LRT statistic is asymptotically distributed chi-squared with $m-1$ degrees of freedom, where $m$ is the number of components in the series system,

$$\Lambda \sim \chi^2_{m-1}.$$

If the LRT statistic is greater than the critical value of the chi-squared distribution with $m-1$ degrees of freedom, $\chi^2_{m-1,1-\alpha}$, then we find the data to be incompatible with the null hypothesis $H_0$, otherwise we find the data to be compatible with the null hypothesis $H_0$.

## 7.1 Results of the Comparison

The comparison between the full and reduced models yields a chi-squared value of [Insert Value], with a corresponding p-value of [Insert Value]. The interpretation of these results indicates that [Insert Interpretation, e.g., the data supports/rejects the Weibull homogeneity in the reduced model]. This finding has implications for [Insert specific implications relevant to your study].

#### Conclusion

This analysis further illuminates the properties of the reduced model in relation to the full model. By examining Weibull homogeneity through the LRT, we gain insights that enrich the overall understanding of [Insert broader context or subject matter of the paper].

# 8 Simulation Study

In this section, we conduct a simulation study to assess the performance of the MLE for the likelihood model defined in Section 6. In this simulation study, we assess the sensitivity of the MLE to various simulation scenarios. In particular, we assess two important properties of the MLE with respect to a scenario:

1. Accuracy (Bias): How close is the expected value of the MLE to the true parameter values? If the expected value of the MLE is close to the true parameter values, the accuracy is high.

2. Precision: How much does the MLE vary from sample to sample? We measure this by assessing the 95% confidence intervals (BCa, Bias-Corrected and accelerated). If the confidence intervals are both small and have good coverage probability (the proportion of confidence intervals that contain the true parameter values), then the MLE is precise.

We begin by specifying the parameters of the series system that will be the central object of our simulation study. We consider the data in Guo et al. (2013), in which they study the reliability of a series system with three components. They fit Weibull components in a series configuration to the data, resulting in an MLE with shape and scale estimates given by the first three components in Table 2. To make the model slightly more complex, we add two more components to this series system, with shape and scale parameters given by the last two components in Table 2. We will refer to this system as the **base** system.

In Section 2.2, we defined a well-designed series system as one that consists of components with similar reliabilities, where we define reliability in two ways, the mean time to failure (MTTF) and the probability that a specific component will be the cause of failure. All things else being equal, components with long MTTFs and with near uniform probability of being the component cause of failure is preferrable, otherwise we have a weak link in the system.

The base system defined in Table 2 satisfies this definition of being a well-designed system. We see that there are no components that are significantly less reliable than any of the others, component 1 being the most reliable and component 3 being the least reliable. This is a result of the scales and shapes being similar for each component. In addition, the shapes are larger than 1, which means components are unlikely to fail early.

Table 2: Weibull Components in Series Configuration

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

## 8.1 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 $j^{\text{th}}$ component of the $i^{\text{th}}$ 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 simple right-censoring model, where the right-censoring time $\tau$ is fixed at some known value, e.g., an experiment is run for a fixed amount of time $\tau$, and all systems that have not failed by the end of the experiment are right-censored. The censoring time $S_i$ of the $i^{\text{th}}$ 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 $\tau$.

In our simulation study, we parmaterize the right-censoring time $\tau$ by the quantile $q = 0.825$ of the series system,

$$\tau = F_{T_i}^{-1}(q).$$

This means that 82.5% of the series systems are expected to fail before time $\tau$ and 17.5% of the series are expected to be right-censored. To solve for the 82.5% quantile of the series system, we define the function $g$ as

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

and find its root using the Newton-Raphson method. See Appendix 13 for the R code that implements this procedure.

**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 Appendix **??** for the R code that implements this model.

## 8.2 Impacts on the MLE

Before we present the results of our simulation studies, we discuss some of the potential impacts of the candidate set model and right-censoring on the MLE of the series system.

#### Effect of Right-Censoring

In all of our simulation studies, we use a fixed right-censoring quantile of 82.5% for the series system. So, in our simulation studies, we do not isolate the effect of right-censoring. However, we can still discuss the effect of right-censoring on the MLE of the series system. Right-censoring introduces a source of bias in the MLE.

> Dr. Agustin: I'm reworking this section. It's generally true for our "well-designed system", which approximately the same shape and scale parameters, with the shapes larger than 1, but when one or more components have a shape less than 1, early deaths are more common but the MTTF may be significantly larger than the censoring time and the other components. In these cases, the MLE for that component's shape parameter is nudged upwards, which decreases MTTF but decreases the likelihood of early deaths before the right-censoring time.

In the case of our well-designed system with components that have approximately the same shape and scale, and each system wears out with age (shapes larger than 1), 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. By Equation (**??**), the mean time to failure (MTTF) for the $j^{\text{th}}$ Weibull component is given by

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

therefore, in order to increase the MTTF of the components, higher values for the scale parameters are chosen. The effect on the shape parameter is more nuanced, as if the MTTF is increased by decreasing the shape, it also increases the probability of early failures. Therefore, we speculate that the scale parameter in our well-designed system is more likely to be affected by right-censoring than the shape parameter.

#### 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 either a smaller MTTF or a higher infant mortality rate for each of the components in the candidate set, depending upon the failure characteristics.

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 (**??**), we see that the MTTF of a Weibull component is proportional to its scale parameter $\lambda_j$, which means when we decrease the scale parameter $\lambda_j$ (keeping the shape parameter $k_j$ constant), the MTTF decreases. Therefore, if the $j^{\text{th}}$

[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.

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.

The shape parameter is more complicated, since it determines the failure characteristics of the components. In our well-designed system, the shape parameters are larger than 1, and in this case, since the MTTF decreases as we increase the shape parameter $k_j$, if the $j^{\text{th}}$ component is in the candidate set, to make it more likely to appear in in the candidate set, its shape parameter is nudged upwards, potentially positively biasing the MLE for the shape parameter. However, we can also decrease the shape parameter, which will increase the MTTF, but also increase the infant morality rate, which is another way to increase the probability of a component appearing in the candidate set. It is not clear which of these two effects will dominate in general.

#### Effect of Differentiated Scales

...

#### Effect of Differentiated Shapes

...

### 8.3 Simulation Scenarios

We define a simulation scenario to be some combination of $n$, $p$, $k_3$, and $\lambda_3$. 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. Fix a combination of simulation parameters to some value, and vary the remaining parameter, e.g., fix $p$, $k_3$, $\lambda_3$, and vary the sample size $n$.
2. Simulate $R$ datasets from the Data Generating Process (DGP) described in Section 8.1.
3. For each of these $R = 500$ datasets, compute the MLE.
4. For each of these $R$ datasets, perform bootstrap resampling $B = 1000$ times to create a set of bootstrap samples (replicates).
5. 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.
6. 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 and the bias of the MLE at that particular combination of simulation parameters.

Once we generate the data for a simulation scenario, we plot the results and discuss the performance of the MLE estimator under the chosen scenario. For how we run a simulation scenario, see Appendix **??**.

#### Scenario #1: Assessing the Impact of Sample Size on the MLE

In this scenario, we use the well-designed series system described in Section 8, Table 2. We fix the masking probability to $p = 0.215$ (small masking), we fix the right-censoring quantile to $q = 0.825$, and we vary the sample size $n$ from 50 to 1000.

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** For the components 2, 3, and 4, Figure 2 shows the distribution of the MLEs for the scale parameters 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) and a right-censoring quantile of $q = 0.825$.

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 scale MLEs are, on average, reasonably unbiased.

**Shape Parameters** For components 2, 3, and 4, Figure 3 shows the distribution of the MLEs for the shape parameters 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) and a right-censoring quantile of $q = 0.825$.

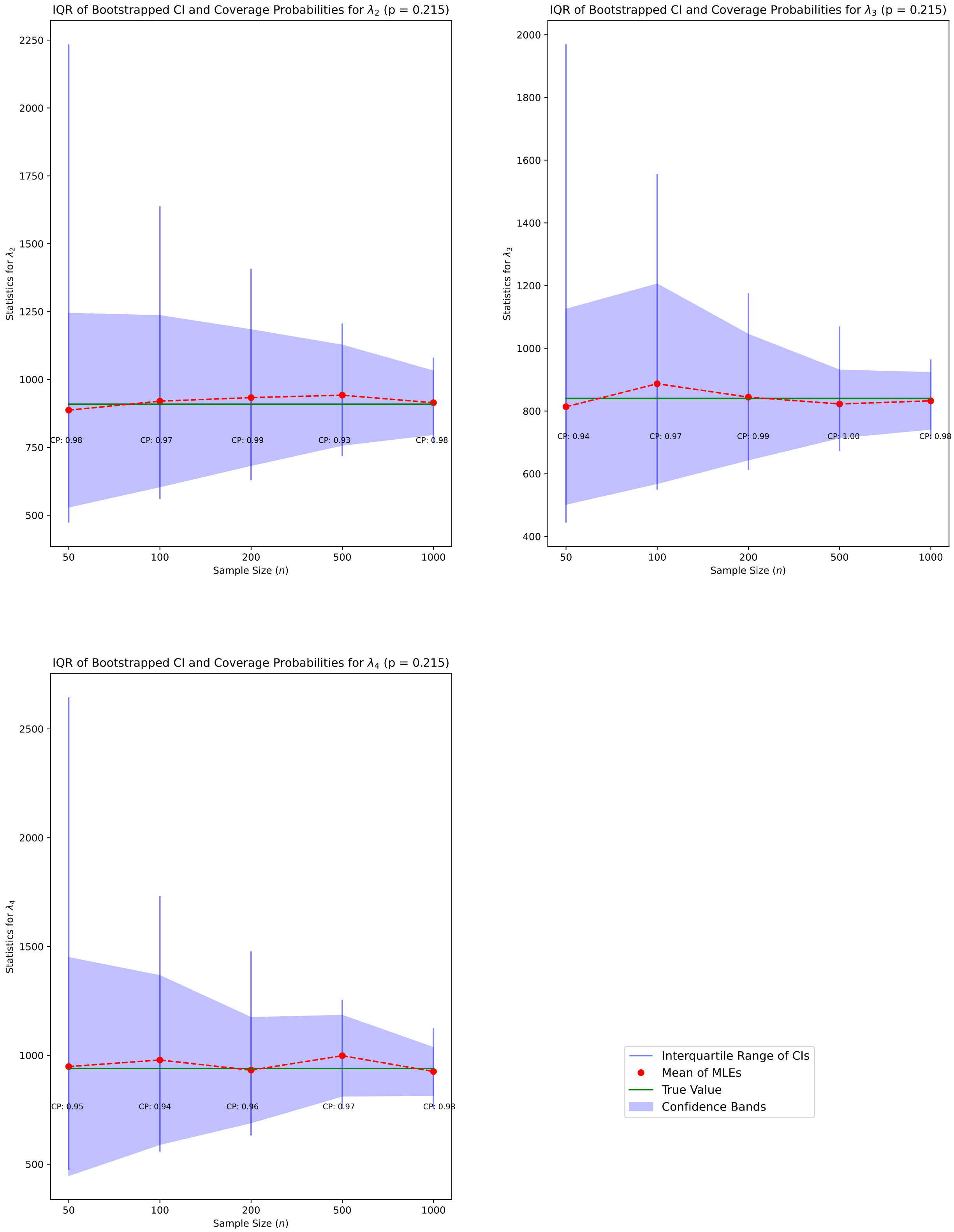


Figure 2: Sample Size vs Bootstrapped Scale CI Statistics ($p = 0.215, q = 0.825$)

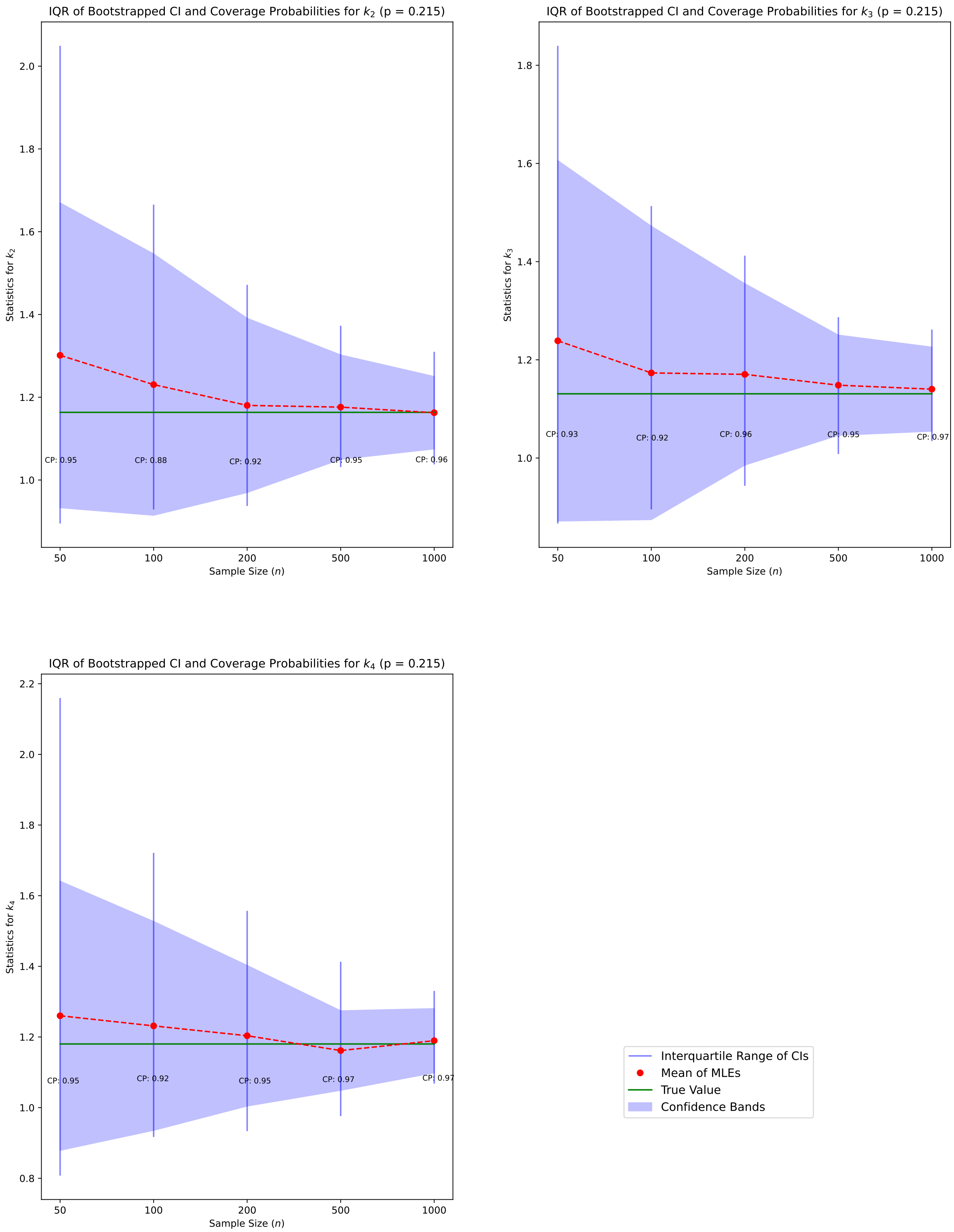


Figure 3: Sample Size vs Bootstrapped Scale CI Statistics ($p = 0.215, q = 0.825$)

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 is the mean of shape MLEs. Unlike the scale MLEs, we see that for small samples, particularly less than 200, we observe a significant amount of positive bias for shape MLEs. The MLE for the shape parameters in this scenario appear to be more sensitive to the data than the scale parameters. We speculate that this is

#### Scenario #2: Assessing the Impact of Masking Probability for Component Cause of Failure

In this scenario, we use the well-designed series system described in Section 8, Table 2. We fix the sample size to $n = 90$ (reasonable sample size) and we fix the right-censoring quantile to $q = 0.825$, and we vary the masking probability from $p$ from 0.1 (very slight masking the component cause of failure) to 0.85 (extreme masking of the component cause of failure).

Since the effect was identical on each of the components, we only show the results for the shape and scale parameters of component 1.

**Scale Parameter** In Figures 4, we show the effect of the masking probability $p$ on the MLE and the bootstrapped BCa confidence intervals for the scale parameter of component 1. At this sample size, we see that the MLE is relatively unbiased for small $p$, but as $p$ increases, the MLE becomes increasingly positively biased. We also see the confidence interval width seems stable until the masking becomes significant at $p > 0.4$. The confidence intervals were well-calibrated at all masking probabilities except $p = 0.85$, at which point the coverage probability was only 77% despite the very wide CIs.

This Figure is consistent with the our analysis in Section 8.2, where we indicated we expected a positive bias on the scale parameter to push the component's MTTF estimate upwards.

**Shape Parameters** In Figures 5, we show the effect of the masking probability $p$ on the MLE and bootstrapped BCa confidence intervals for the shape parameter of component 1. We see almost the exact same pattern as we did for the scale parameter. The MLE is relatively unbiased for small $p$, although there is a slight positive bias (potentially due to the right-censoring effect). As $p$ increases past $p = 0.55$, the MLE becomes increasingly positively biased. Again, the confidence intervals are well-calibrated at all masking probabilities except $p = 0.85$, at which point the coverage probability was around 65% despite the very wide CIs.

This Figure is also consistent with our analysis in Section 8.2, where we indicated we expected a positive bias on the shape parameter to push the component's MTTF estimate upwards in the case of the well-designed series system. Note, however, that we might expect a different result if a component had a different failure distribution (e.g., a Weibull with a shape parameter less than 1).

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.

**Scenario #3: Assessing the Impact of Changing the Scale Parameter of a Component**

We see that the MTTF is proportional to the scale parameter $\lambda_j$, 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, $\tau$ 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.

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 $\lambda_3$, 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

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

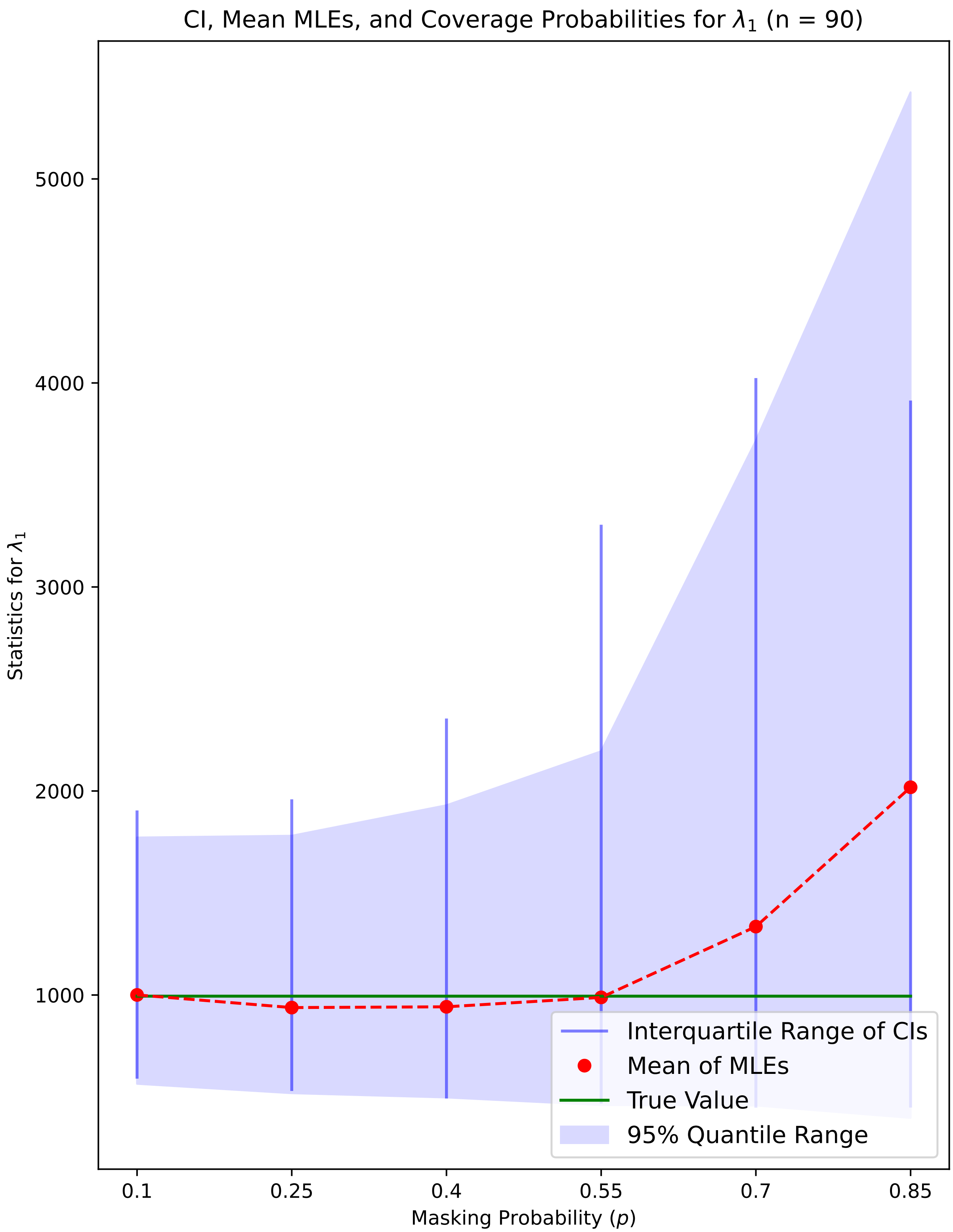


Figure 4: Component Cause of Failure Masking (p) vs Scale CI Statistics

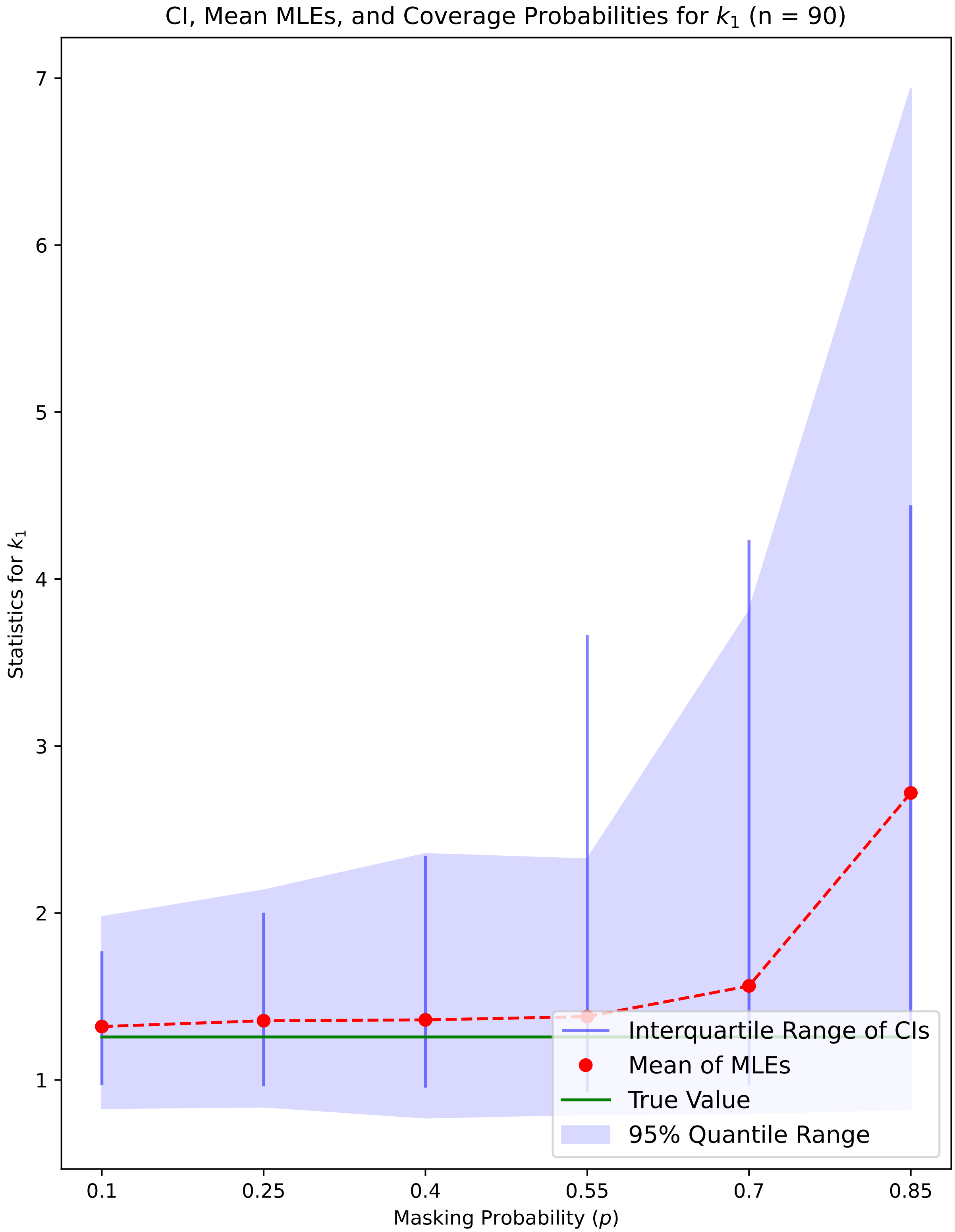


Figure 5: Component Cause of Failure Masking (p) vs Shape CI Statistics

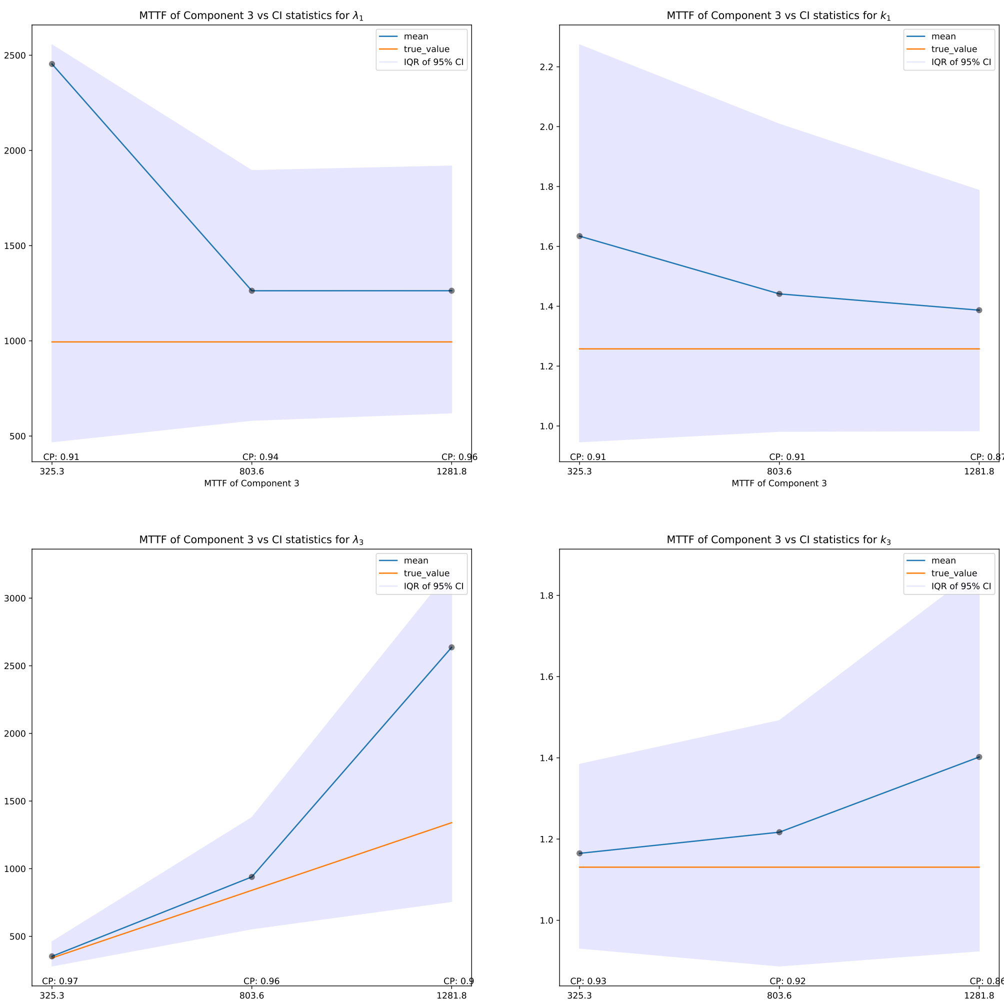


Figure 6: MTTF vs Parameter Statistics

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

### Scenario #4: Assessing the Impact of Changing the Shape Parameter of a Component

We vary the shape paramenter of component 3 from 0.1 to 1.9 and observe the effect it has on the MLE and the bootstrapped confidence intervals (BCa). The shape parameter determines the failure characteristics of component 3.

When $k_3 < 1$, this indicates infant mortality, with a decreasing failure rate over time, so even though it has a high failure rate at the beginning of its lifetime, it has a low failure rate at the end of its lifetime and its MTTF is much higher than the other components even though it has a higher probability of failing first.

When $k_3 > 1$, this indicates wear-out failures, with an increasing failure rate over time, so even though it has a low failure rate at the beginning of its lifetime, it has a high failure rate at the end of its lifetime and it has a lower probability of failing first.

We analyze the effect of component 3's shape parameter on the MLE and the bootstrapped confidence intervals for the shape and scale parameters of components 1 and 3 (the component we are varying). First, we look at the effect on the scale parameter.

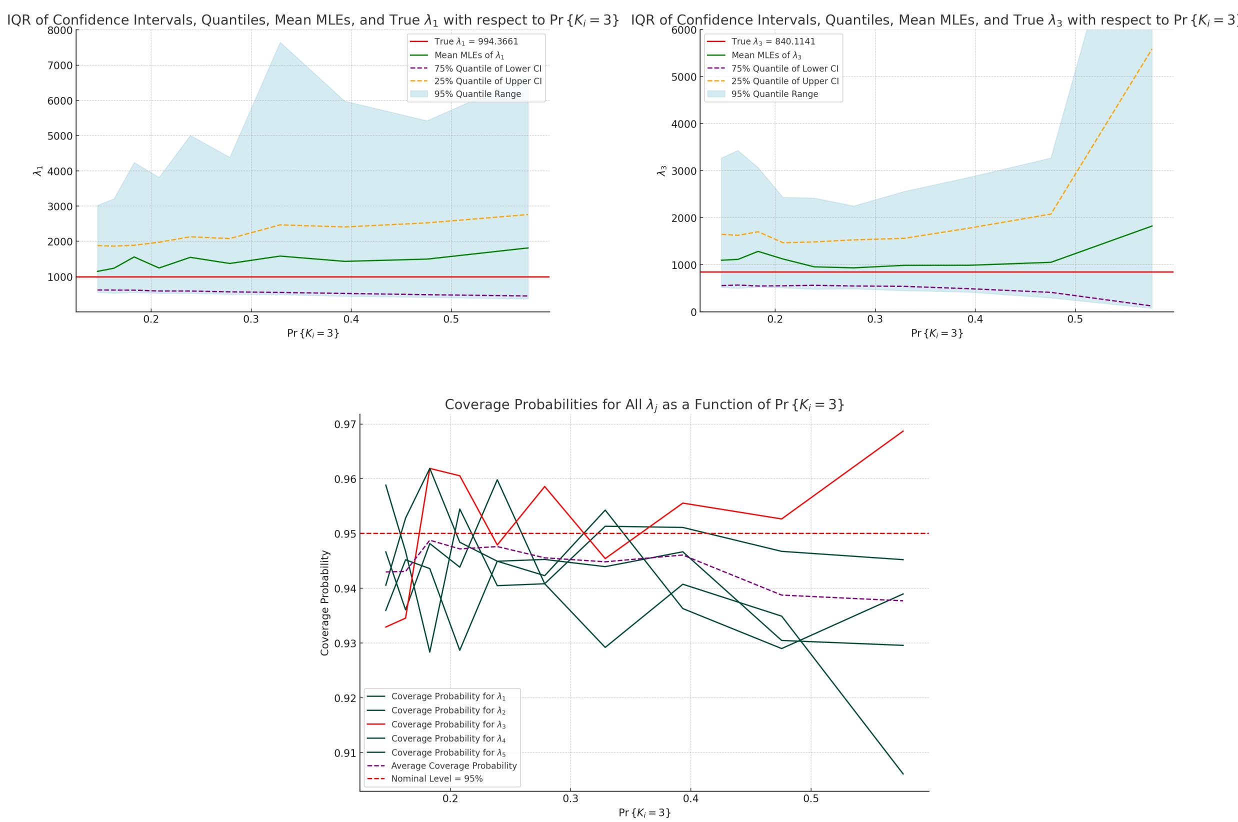


Figure 7: Probability of Component 3 Failure vs Scale Statistics

**8.3.0.1 Scales** In Figure 8, we show the effect of the shape parameter of component 3 on the MLE and the bootstrapped confidence intervals for the shape parameters of components 1 and 3.

We see that the mean MLE, in green, is relatively close to the true value, in red, for the scale parameter of both components. There is a slight positive bias, which may be due to the fact that the data is right-censored with moderate masking of the component cause of failure. We see that as the probability of component 3 being the cause of failure increases, the bootstrapped confidence intervals generally increase in width, with the exception of when $k_3 < 1$ which causes $\Pr\{K_i = 3\}$ to be very small and as $\Pr\{K_i = 3\}$ approaches 0.2, all of the components are approximately equally like to be the component cause of failure, and so the CIs seem to be fairly small for all scale parameters.

However, for $\Pr\{K_i = 3\} > 0.5$, we see that the the mean MLE begins to increase significantly for $\lambda_3$. This is somewhat unexpecte; we might think that, because its probabilty of being the component cause of failure is higher, that we would estimate $\lambda_3$ to be lower to proportionately decrase its MTTF. However, the fact is that the shape parameter has a much bigger impact.

Also, the coverage probabilites of the confidence intervals for the scale parameters decreases for the scale parameter of components other than 3 as $\Pr\{K_i = 3\}$ increases, but the coverage probability for the scale parameter of component 3 increases. This may be because, as $\Pr\{K_i = 3\}$ increases, we are more likely to observe a failure of component 3, and so we have more information about its parameters and are able to estimate them more accurately.

**8.3.0.2 Shapes** Now, we look at the effect of the shape parameter of component 3 on the MLE and the bootstrapped confidence intervals for the shape parameters of components 1 and 3.

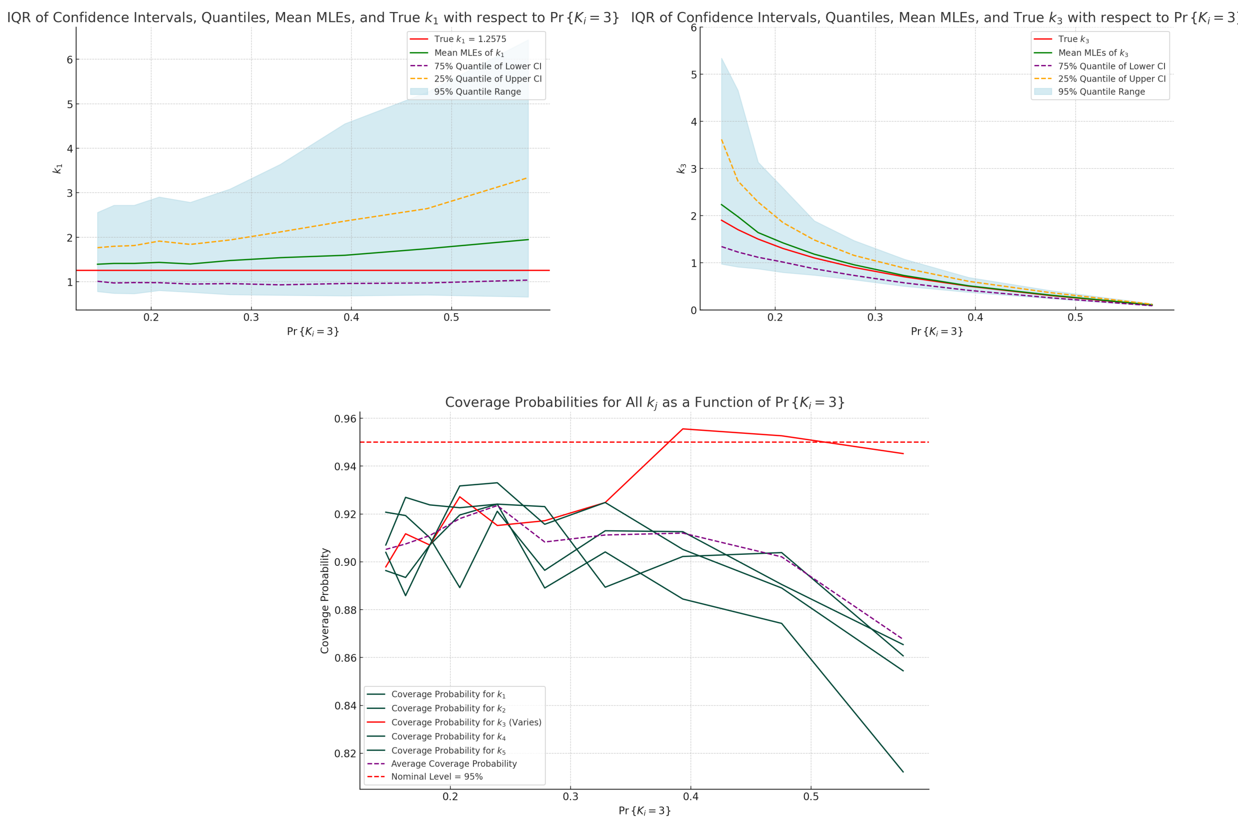


Figure 8: Probability of Component 3 Failure vs Shape Statistics

In Figure 8, we show the effect of the shape parameter of component 3 on the MLE and the bootstrapped confidence intervals for the shape parameters of components 1 and 3.

We see that the bias for $k_1$ slowly increases (positive bias) as $\Pr\{K_i = 3\}$ increases, and the bias for $k_3$ slowly decreases (positive bias) to 0 as $\Pr\{K_i = 3\}$ increases. This makes sense, as a larger positive bias for $k_1$ means that the MLE is nudging the shape parameter of component 1 to be larger so that component 1 is less likely to be the cause of failure. Similarly, a smaller positive bias for $k_3$ means that the MLE is nudging the shape parameter of component 3 to be smaller so that component 3 is more likely to be the cause of failure.

The confidence intervals for $k_1$ also become quite wide as $\Pr\{K_i = 3\}$ increases, which is expected since we observe fewer failures of component 1 as $\Pr\{K_i = 3\}$ increases, and so we have less information about its parameters and are less able to estimate them accurately. Conversely, the confidence intervals for $k_3$ become narrower as $\Pr\{K_i = 3\}$ increases, which is also expected, since we observe more failures of component 3 as $\Pr\{K_i = 3\}$ increases, and so we have more information about its parameters and are more able to estimate them accurately. The CI widths for $k_3$ becomes extremely small for $\Pr\{K_i = 3\} > 0.3$.

The coverage probabilities are generally less well-caliibrated for the shape parameters compared to the scale parameters, but they are still reasonably well-calibrated for $\Pr\{K_i = 3\} < 0.4$. For $k_3$, the coverage probabilities are very well-calibrated for all values of $\Pr\{K_i = 3\}$, but improve as $\Pr\{K_i = 3\}$ increases due to the fact that we observe more failures of component 3 as $\Pr\{K_i = 3\}$ increases and thus have more informationa bout $k_3$ for our estimate.

# 9 Conclusion

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

# 10 Appendix A: R Code For Log-likelihood Function {-# app:loglike-code #}

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
}
```

# 11 Appendix B: R Code For Score Function {-# app:score-code #}

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) {
    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[cinde
            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
}
```

# 12 Appendix C: R Code For Simulation of Scenarios {-# app:sim-code #}

The following code is the Monte-carlo simulation code for running the scenarios described in Section 8.

```
#### 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
)

shapes3 <- c(1.1308) # shape 3 true parameter values to simulate
scales3 <- c(840.1141) # scale 3 true parameter values to simulate
N <- c(30, 60, 100) # sample sizes to simulate
P <- c(.215) # masking probabilities to simulate
Q <- c(.825) # right censoring probabilities to simulate
R <- 1000L # 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" # filename prefix for output files

#### 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)
```

```r
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("   scales3: ", scales3, "\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()

for (scale3 in scales3) {
    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: scale3 = ", scale3, ",
                        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)

                    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)
```

```r
                    )
                    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 =
            },
            error = function(e) {
                cat("[error] ", conditionMessage(e), "\n")
                cat("[retrying scenario: n = ", n, ", p = ", p, ", q = ", q, "\n")
                retry <<- TRUE
            }
        )
        if (retry) {
            next
        }
        iter <- iter + 1L
        shapes.mle[iter, ] <- sol$par[seq(1, length(theta), 2)]
        scales.mle[iter, ] <- sol$par[seq(2, length(theta), 2)]

        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 %% 5 == 0) {
            cat("[iteration ", iter, "] shapes = ", shapes.mle[iter, ], "scales = ", sca
        }

        if (iter == R) {
            break
        }
    }
```

```
                    df <- data.frame(
                        n = rep(n, R), rep(scale3, 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
                    )

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

# 13 Appendix D: Bernoulli Candidate Set Model {-# app:cand-model #}

```
#' 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
#' 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).
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))
}
```

# Appendix E: Series System Quantile Function

```r
#' Quantile function (inverse of the cdf).
#' By definition, the quantile `p` * 100% for a strictly monotonically increasing
#' cdf `F` is the value `t` that satisfies `F(t) - p = 0`.
#' We solve for `t` using newton's method.
#'
#' @param p vector of probabilities.
#' @param shapes vector of weibull shape parameters for weibull lifetime
#'               components
#' @param scales vector of weibull scale parameters for weibull lifetime
#'               components
qwei_series <- function(p, shapes, scales) {
    t0 <- 1
    repeat {
        t1 <- t0 - sum((t0/scales)^shapes) + log(1-p) /
            sum(shapes*t0^(shapes-1)/scales^shapes)
        if (abs(t1 - t0) < tol) {
            break
        }
        t0 <- t1
    }
    return(t1)
}
```

## References


Bain, L. and Engelhardt, M. (1992). *Introduction to Probability and Mathematical Statistics*. Duxbury Press, second edition.

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.

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.

Lehmann, E. L. and Casella, G. (1998). *Theory of Point Estimation*. Springer Science & Business Media.

Taleb, N. N. (2007). *The Black Swan: The Impact of the Highly Improbable*. Random House.

Wu, C. F. J. (1983). On the convergence properties of the em algorithm. *The Annals of Statistics*, 11(1):95–103.