Reliability Estimation in Series Systems

Maximum Likelihood Techniques for Right-Censored and Masked Failure Data

Alex Towell

2023-10-12

Reliability Estimation in Series Systems

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

Alex Towell

Context & Motivation

Reliability in **Series Systems** is like a chain's strength – determined by its weakest link.

- Essential for system design and maintenance.

Main Goal: Estimate individual component reliability from *failure data*.

Challenges:

- *Masked* component-level failure data.
- *Right-censoring* system-level failure data.

Our Response:

- Derive techniques to interpret such ambiguous data.
- Aim for precise and accurate reliability estimates for individual components using maximum likelihood estimation (MLE) and confidence intervals (BCa).

Context & Motivation
Reliability in Series Systems is like a chain's strength – determined by its weakest link.
Essential for system design and maintenance.
Main Goal: Estimate individual component reliability from failure data.
Challenges:
Masked component-level failure data.
Right-censoring system-level failure data.
Our Response:
Derive techniques to interpret such ambiguous data.
Aim for precise and accurate reliability estimates for individual components using maximum likelihood estimation (MLE) and confidence intervals (BCa).

- Think of a series system as a **chain**: reliability is determined by its weakest component.
- When any component **fails**, the whole system does.
- So, understanding the **reliability** of each component is needed for the **design** and **maintenance** of these systems.
- So, our **main goal** is to estimate the reliability of each component from failure data.
- But the **data** can pose **challenges**, like right-censoring or masked failures where we don't know which component failed.
- Our **goal** is to use this data to provide accurate reliability estimates for each component, including accurate confidence intervals to quantify the uncertainty in our estimate.
- To obtain good **coverage**, we bootstrap the confidence intervals using the **BCa method**.

Core Contributions

Likelihood Model for **Series Systems**.

- Accounts for *right-censoring* and *masked component failure*.

Specifications of Conditions:

- Assumptions about the masking of component failures.
- Simplifies and makes the model more tractable.

Simulation Studies:

- Components with *Weibull* lifetimes.
- Evaluate MLE and confidence intervals under different scenarios.

R Library: Methods available on GitHub.

- See: www.github.com/queelius/wei.series.md.c1.c2.c3

Core Contributions
Likelihood Model for Series Systems.
Accounts for right-censoring and masked component failure.
Specifications of Conditions:
Assumptions about the masking of component failures.
Simplifies and makes the model more tractable.
Simulation Studies:
Components with Weibull lifetimes.
Evaluate MLE and confidence intervals under different scenarios.
R Library: Methods available on GitHub.
See: www.github.com/queelius/wei.series.md.c1.c2.c3

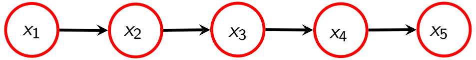
- Our **core contributions** can be broken down into several parts.
- We **derived** a **likelihood model** for series systems that accounts for **Right-censoring** and **masking** of **component failures**.
- We've **clarified** the conditions this model assumes about the masking of component failures. These conditions simplify the model and make it more tractable.
- We've **validated** our model with extensive simulations to gauge its performance under various simulation scenarios.
- The **simulation study** is based on components with **Weibull** lifetimes.
- For those interested, we made our methods available in an **R Library** hosted on GitHub.

Section 1

Series System

Section 1
Series System

Series System



Critical Components: Complex systems often comprise *critical* components. If any component fails, the entire system fails.

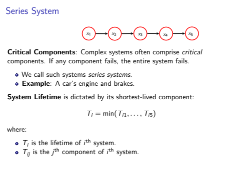
- We call such systems *series systems*.
- **Example**: A car's engine and brakes.

System Lifetime is dictated by its shortest-lived component:

$$T_i = \min(T_{i1}, \ldots, T_{i5})$$

where:

- T_i is the lifetime of i^{th} system.
- T_{ij} is the j^{th} component of i^{th} system.



- Many complex systems have **critical components** that are essential to their operation.
- If any of these components fail, the entire system fails. We call these **series systems**.
- Think of a **car** - if the engine or brakes fail, it can't be operated.
- Its **lifetime** is the lifetime of its **shortest-lived** component.
- For reference, we show the some notation we'll use throughout the talk.
- T_i is the system's lifetime and T_{ij} is its j^{th} component's lifetime.

Reliability Function

Reliability Function represents the probability that a system or component functions beyond a specified time.

- Essential for understanding longevity and dependability.

Series System Reliability: Product of the reliability of its components:

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

- Here, $R_{T_i}(t; \boldsymbol{\theta})$ and $R_j(t; \boldsymbol{\theta_j})$ are the reliability functions for the system i and component j, respectively.

Relevance:

- Forms the foundation for most reliability studies.
- Integral to our likelihood model, e.g., right-censoring events.

Reliability Function
Reliability Function represents the probability that a system or component functions beyond a specified time.
Essential for understanding longevity and dependability.
Series System Reliability: Product of the reliability of its components:
$R_{T_i}(t; \boldsymbol{\theta}) = \prod_{j=1}^{m} R_j(t; \boldsymbol{\theta_j}).$
Here, $R_{T_i}(t; \boldsymbol{\theta})$ and $R_j(t; \boldsymbol{\theta_j})$ are the reliability functions for the system i and component j, respectively.
Relevance:
Forms the foundation for most reliability studies.
Integral to our likelihood model, e.g., right-censoring events.

- The **reliability function** tells us the chance a component or system functions past a specific time.
- It's a **key metric** for longevity.
- In a series system, the overall reliability is the **product** of its component reliabilities.
- So, even if **one component** has a low reliability, it can impact the whole system.
- For notation, we denote the reliability function for the system as R_{T_i} and the reliability function for the j^{th} component as R_j.

Hazard Function: Understanding Risks

Hazard Function: Measures the immediate risk of failure at a given time, assuming survival up to that moment.

- Reveals how the risk of failure evolves over time.
- Guides maintenance schedules and interventions.

Series System Hazard Function is the sum of the hazard functions of its components:

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

- Components' risks are additive.

Hazard Function: Understanding Risks

Hazard Function: Measures the immediate risk of failure at a given time, assuming survival up to that moment.

- Reveals how the risk of failure evolves over time.
- Guides maintenance schedules and interventions.

Series System Hazard Function is the sum of the hazard functions of its components:

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

- Components' risks are additive.

- Let's shift focus to the hazard function.
- Essentially, it's a way to gauge the immediate risk of failure, especially if it's been functioning up until that point.
- For notation, we denote the hazard function for the system as h_{T_i} and the hazard function for the j^{th} component as h_j.
- The hazard function for a series system is just the sum of the **component hazards**.
- We see that the component risks are **additive**.

Joint Distribution of Component Failure and System Lifetime

Our likelihood model depends on the **joint distribution** of the system lifetime and the component that caused the failure.

- **Formula**: Product of the failing component's hazard function and the system reliability function:

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

- **Single Point of Failure**: A series system fails due to one component's malfunction.
- **Representation**:
  - K_i: Component causing the i^{th} system's failure.
  - $h_j(t;\boldsymbol{\theta_j})$: Hazard function for the j^{th} component.

Joint Distribution of Component Failure and System Lifetime

Our likelihood model depends on the **joint distribution** of the system lifetime and the component that caused the failure.

- **Formula**: Product of the failing component's hazard function and the system reliability function:

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

- **Single Point of Failure**: A series system fails due to one component's malfunction.
- **Representation**:
  - K_i: Component causing the i^{th} system's failure.
  - $h_j(t;\boldsymbol{\theta_j})$: Hazard function for the j^{th} component.

- In our likelihood model, understanding the **joint distribution** of a system's lifetime and the component that led to its failure is essential.
- It is the **product** of the failed component's hazard function and the system reliability function.
- Here, K_i denotes **failed component**.

Component Failure & Well-Designed Series Systems

We can use the joint distribution to calculate the probability of component cause of failure.

- Helps predict the cause of failure.
- **Derivation**: Marginalize the joint distribution over the system lifetime:

$$\Pr\{K_i = j\} = E_{\boldsymbol{\theta}}\left[\frac{h_j(T_i; \boldsymbol{\theta_j})}{h_{T_i}(T_i; \boldsymbol{\theta_l})}\right].$$

- **Well-Designed Series System**: Components exhibit comparable chances of causing system failures.
- **Relevance**: Our simulation study employs a (reasonably) well-designed series system.

Component Failure & Well-Designed Series Systems
We can use the joint distribution to calculate the probability of component cause of failure.
- Helps predict the cause of failure.
- **Derivation**: Marginalize the joint distribution over the system lifetime:
$$\Pr\{K_i = j\} = E_{\boldsymbol{\theta}}\left[\frac{h_j(T_i; \boldsymbol{\theta_j})}{h_{T_i}(T_i; \boldsymbol{\theta_l})}\right].$$
- **Well-Designed Series System**: Components exhibit comparable chances of causing system failures.
- **Relevance**: Our simulation study employs a (reasonably) well-designed series system.

- We can use the **joint distribution** of the system lifetime and the failed component to calculate the probability of each component causing the failure.
- This helps us **predict** the cause of failure.
- It is **derived** by marginalizing the joint distribution over the system lifetime.
- When we do so, we find that it is the **expected value** of the ratio of component and system hazard functions.
- We say that a series system is **well-designed** if each components has a **comparable** chance of failing.
- Our simulation study is **based** on a reasonably well-designed series system.

Section 2

Likelihood Function

Section 2
Likelihood Function

Likelihood Function

Likelihood: Measures how well model explains the data. Each system contributes to *total likelihood* via its *likelihood contribution*:

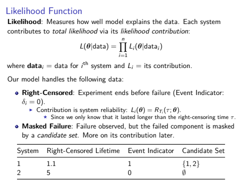
$$L(\boldsymbol{\theta}|\text{data}) = \prod_{i=1}^{n} L_i(\boldsymbol{\theta}|\text{data}_i)$$

where $\textbf{data}_i$ = data for i^{th} system and L_i = its contribution.

Our model handles the following data:

- **Right-Censored**: Experiment ends before failure (Event Indicator: $\delta_i = 0$).
  - Contribution is system reliability: $L_i(\boldsymbol{\theta}) = R_{T_i}(\tau; \boldsymbol{\theta})$.
    - Since we only know that it lasted longer than the right-censoring time τ.
- **Masked Failure**: Failure observed, but the failed component is masked by a *candidate set*. More on its contribution later.

System	Right-Censored Lifetime	Event Indicator	Candidate Set
1	1.1	1	$\{1, 2\}$
2	5	0	$\emptyset$

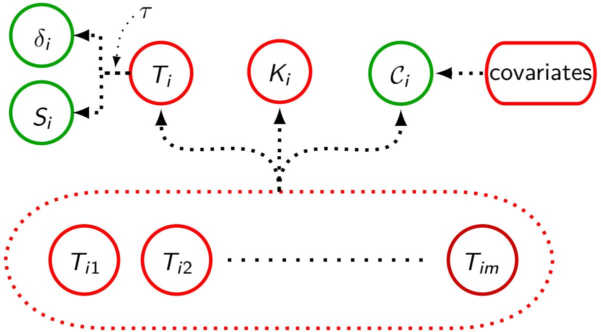


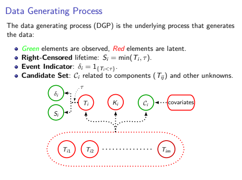
- Let's talk about the **likelihood function**, which is a way of **measuring** how well our model explains the data.
- The **total likelihood** is the product of the likelihood contributions of each system.
- Our model deals two kinds of **contributions**, right-censoring and masked failures.
- **Right-censoring**, for instance, occurs when the experiment ends before the system fails. Its contribution is just the system reliability, since we only know that it lasted longer than the right-censoring time.
- **Masking** occurs when we observe a failure but we don't know which component failed. Instead, we see a set of components that mask the failure. More on this later.

Data Generating Process

The data generating process (DGP) is the underlying process that generates the data:

- *Green* elements are observed, *Red* elements are latent.
- **Right-Censored** lifetime: $S_i = \min(T_i, \tau)$.
- **Event Indicator**: $\delta_i = 1_{\{T_i < \tau\}}$.
- **Candidate Set**: $\mathcal{C}_i$ related to components (T_{ij}) and other unknowns.





- Let's discuss the **data generating process** to motivate our model.
- Here's the graph: **green** is observed and **red** is latent.
- We don't get to see the red elements, but we can **infer** most of them from the green elements.
- So, let's focus on the **green** elements, the **observed** data.
- The **right-censored** time is the minimum of the system lifetime and the right-censoring time τ.
- The **event** indicator is 1 if the system fails before the right-censoring time, 0 otherwise.
- The candidate set is related to the component lifetimes and many other factors.
- The candidate sets are difficult **difficult** to model, so we seek a simple model that is valid under certain assumptions, which we discuss next.

Likelihood Contribution: Masked Failure Conditions

The candidate sets that **mask** the failed component are assumed to satisfy the following conditions:

- **Candidate Set Contains Failed Component**: The candidate set includes the failed component.
- **Masking Probabilities Uniform Across Candidate Sets**: The probability of of the candidate set is constant across different components within it.
- **Masking Probabilities Independent of Parameters**: The masking probabilities when conditioned on the system lifetime and the failed component aren't functions of the system parameter.

Likelihood Contribution: Masked Failure Conditions

The candidate sets that **mask** the failed component are assumed to satisfy the following conditions:

- **Candidate Set Contains Failed Component**: The candidate set includes the failed component.
- **Masking Probabilities Uniform Across Candidate Sets**: The probability of of the candidate set is constant across different components within it.
- **Masking Probabilities Independent of Parameters**: The masking probabilities when conditioned on the system lifetime and the failed component aren't functions of the system parameter.

- The right-censoring contribution is straightforward. The **masked failure contribution** is more complex.
- **Masking** occurs when a system fails but the precise failed component is masked by a candidate set.
- To make the problem **tractable**, we introduce certain conditions or assumptions about the candidate sets.
- In **Condition 1**, the candidate set always includes the failed component.
- In **Condition 2**, the probability of the candidate set is constant across different components within it.
- In **Condition 3**, the masking probabilities when conditioned on the system lifetime and the failed component aren't functions of the system parameter.
- These conditions are often **reasonable** in industrial settings.

Likelihood Contribution: Derivation for Masked Failures

Take the **joint distribution** of T_i, K_i, and $\mathcal{C}_i$ and marginalize over K_i:

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

Apply **Condition 1** to get a sum over candidate set:

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

Apply **Condition 2** to move probability outside the sum:

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

Apply **Condition 3** to remove the probability's dependence on $\boldsymbol{\theta}$:

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

Result: $L_i(\boldsymbol{\theta}) \propto \sum_{j \in c_i} f_{T_i,K_i}(t_i, j; \boldsymbol{\theta}) = R_{T_i}(t_i; \boldsymbol{\theta}) \sum_{j \in c_i} h_j(t_i; \boldsymbol{\theta_j})$.

Likelihood Contribution: Derivation for Masked Failures
Take the **joint distribution** of T_i, K_i, and $\mathcal{C}_i$ and marginalize over K_i:
$$f_{T_i,\mathcal{C}_i}(t_i, c_i; \boldsymbol{\theta}) = \sum_{j=1}^{m} f_{T_i,K_i}(t_i, j; \boldsymbol{\theta}) \Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j\}.$$
Apply **Condition 1** to get a sum over candidate set:
$$f_{T_i,\mathcal{C}_i}(t_i, c_i; \boldsymbol{\theta}) = \sum_{j \in c_i} f_{T_i,K_i}(t_i, j; \boldsymbol{\theta}) \Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j\}.$$
Apply **Condition 2** to move probability outside the sum:
$$f_{T_i,\mathcal{C}_i}(t_i, c_i; \boldsymbol{\theta}) = \Pr_{\boldsymbol{\theta}}\{\mathcal{C}_i = c_i | T_i = t_i, K_i = j'\} \sum_{j \in c_i} f_{T_i,K_i}(t_i, j; \boldsymbol{\theta}).$$
Apply **Condition 3** to remove the probability's dependence on $\boldsymbol{\theta}$:
$$f_{T_i,\mathcal{C}_i}(t_i, c_i; \boldsymbol{\theta}) = \beta_i \sum_{j \in c_i} f_{T_i,K_i}(t_i, j; \boldsymbol{\theta}).$$
Result: $L_i(\boldsymbol{\theta}) \propto \sum_{j \in c_i} f_{T_i,K_i}(t_i, j; \boldsymbol{\theta}) = R_{T_i}(t_i; \boldsymbol{\theta}) \sum_{j \in c_i} h_j(t_i; \boldsymbol{\theta_j})$.

- Here, we **derive** the likelihood contribution for masked failures.
- To start, we use the **joint distribution** of the system lifetime, the failed component, and the candidate set.
- Then, we **marginalize** over the failed component, since we don't know which component failed.
- We apply **condition 1** to get a **sum** over the **candidate set** instead.
- We apply **condition 2** to move the probability **outside** the sum.
- We apply **condition 3** to **remove** the probability's dependence on the system parameter.
- And we **end up** with a likelihood contribution proportional to the product of the system reliability and the sum of the component hazards in the candidate set.

Bootstrap Confidence Intervals (CIs)

Confidence Intervals (CI) help capture the *uncertainty* in our estimate.

- **Normal** assumption for constructing CIs may not be accurate.
  - *Masking* and *censoring*.
- **Bootstrapped CIs**: Resample data and obtain MLE for each.
  - Use **percentiles** of bootstrapped MLEs for CIs.
- **Coverage Probability**: Probability the interval covers the true parameter value.
  - **Challenge**: Actual coverage may deviate to bias and skew in MLEs.
- **BCa** adjusts the CIs to counteract bias and skew in the MLEs.

Bootstrap Confidence Intervals (CIs)

Confidence Intervals (CI) help capture the *uncertainty* in our estimate.

- **Normal** assumption for constructing CIs may not be accurate.
  - *Masking* and *censoring*.
- **Bootstrapped CIs**: Resample data and obtain MLE for each.
  - Use **percentiles** of bootstrapped MLEs for CIs.
- **Coverage Probability**: Probability the interval covers the true parameter value.
  - **Challenge**: Actual coverage may deviate to bias and skew in MLEs.
- **BCa** adjusts the CIs to counteract bias and skew in the MLEs.

- We need to measure the **uncertainty** in our estimate.
- **Confidence intervals** are a popular choice and help us pin down the likely range of values for our parameters.
- Due to masking and censoring, the **normal** approximation for constructing CIs may be inaccurate.
- So, we've chosen to **bootstrap** the intervals instead, which isn't as sensitive to these issues.
- **Coverage probability** is the probability the interval covers the true parameter value.
- Due to bias and skew in the MLE, the coverage probability may be too low/high, indicating over/under confidence.
- We use the **BCa method** to adjust the confidence intervals to counteract bias and skew.

Challenges with MLE on Masked Data

We discovered some challenges with the MLE.

- **Convergence Issues**: Flat likelihood regions observed.
  - Ambiguity in masked data with small samples.
- **Bootstrap Issues**: Bootstrap relies on the Law of Large Numbers.
  - It might not represent the true variability for small samples.
  - Censoring and masking reduces effective sample size.
- **Mitigation**: In simulation study, discard non-convergent samples for the MLE on original data, but keep all resamples for the BCa CIs.
  - Helps ensure robustness of the results, while acknowledging the inherent complexities introduced by masking and censoring.
  - We report convergence rates in our simulation study.

Challenges with MLE on Masked Data

We discovered some challenges with the MLE.

- **Convergence Issues**: Flat likelihood regions observed.
  - Ambiguity in masked data with small samples.
- **Bootstrap Issues**: Bootstrap relies on the Law of Large Numbers.
  - It might not represent the true variability for small samples.
  - Censoring and masking reduces effective sample size.
- **Mitigation**: In simulation study, discard non-convergent samples for the MLE on original data, but keep all resamples for the BCa CIs.
  - Helps ensure robustness of the results, while acknowledging the inherent complexities introduced by masking and censoring.
  - We report convergence rates in our simulation study.

- We use the standard **maximum likelihood approach** to estimate the parameters.
- Like any model, ours has its challenges. **Masking** and **censoring**, combined with small sample sizes, can cause flat likelihood regions, which can lead to convergence issues.
- For **small samples**, bootstrapping may not always capture the true variability in the data.
- To **deal** with these issues in our simulation study, we **discard** samples that did not converge within 125 iterations for the MLE on original data.
- However, we **retain** all MLEs for the resampled data in the bootstrap for the confidence intervals.
- This helps ensure the **robustness** of our results while acknowledging the inherent complexities introduced by masking and censoring.

Section 3

Simulation Study

Section 3
Simulation Study

Series System: Weibull Components

The lifetime of the j^{th} component in the i^{th} system:

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

- λ_j is the **scale** parameter
- k_j is the **shape** parameter:
  - $k_j < 1$: Indicates infant mortality.
  - $k_j = 1$: Indicates random failures (exponential distribution).
  - $k_j > 1$: Indicates wear-out failures.

Weibull has well known *reliability* and *hazard* functions.

Series System: Weibull Components

The lifetime of the j^{th} component in the i^{th} system:

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

- λ_j is the **scale** parameter
- k_j is the **shape** parameter:
  - $k_j < 1$: Indicates infant mortality.
  - $k_j = 1$: Indicates random failures (exponential distribution).
  - $k_j > 1$: Indicates wear-out failures.

Weibull has well known *reliability* and *hazard* functions.

- In our **simulation study**, we analyze a series system with **Weibull** components.
- The Weibull has two parameters: the **scale** and **shape**.
- **Shape** parameter tells us a lot about the failure characteristics.
- When its **greater** than **one**, think of it as wearing-out over time.
- If it's **less** than one, that usually signals some early-life challenges.
- If it's **equal** to one, it has random failures.

Well-Designed Series System

Simulation study centered on a series system with these Weibull components:

Component	Shape	Scale	$\Pr\{K_i\}$
1	1.26	994.37	0.17
2	1.16	908.95	0.21
3	1.13	840.11	0.23
4	1.18	940.13	0.20
5	1.20	923.16	0.20

- Based on (Guo, Niu, and Szidarovszky 2013) study of 3 components.
  - We added components 4 and 5.
  - **Shapes** greater than 1, indicating wear-outs.
- **Probabilities** are comparable: it is *reasonably well-designed*.
  - **Reliability**: Components 1 and 3 *most* and *least* reliable, respectively.
  - **Simulation Study**: Only show estimates for these two components.

Well-Designed Series System

Simulation study centered on a series system with these Weibull components:

Component	Shape	Scale	$\Pr\{K_i\}$
1	1.26	994.37	0.17
2	1.16	908.95	0.21
3	1.13	840.11	0.23
4	1.18	940.13	0.20
5	1.20	923.16	0.20

- Based on (Guo, Niu, and Szidarovszky 2013) study of 3 components.
  - We added components 4 and 5.
  - **Shapes** greater than 1, indicating wear-outs.
- **Probabilities** are comparable: it is *reasonably well-designed*.
  - **Reliability**: Components 1 and 3 *most* and *least* reliable, respectively.
  - **Simulation Study**: Only show estimates for these two components.

- This study is **centered** around a series system with **five** Weibull components.
- It's **based** on a paper that studies a 3-component series system.
- We **added** components 4 and 5 to make it more complex.
- We show the **shape** and **scale** parameters for each component.
- We see that the **shape** parameters are **greater** than 1, which indicates **wear-out** failures.
- We also show the **probability** of each component being the cause of failure in the last column.
- Since the probabilities are **comparable**, no weak links, it's reasonably **well-designed**.
- Component 1 is the **most** reliable and component 3 is the **least** reliable.
- In our simulation study, we only show the estimates for these two

Synthetic Data and Simulation Values

How is the data generated in our simulation study?

- **Component Lifetimes** (latent $T_{i1}, \cdots, T_{im}$) generated for each system.
  - **Observed Data** is a function of latent components.
- **Right-Censoring** amount controlled with simulation value q.
  - Quantile q is probability system won't be right-censored.
  - Solve for right-censoring time τ in $\Pr\{T_i \leq \tau\} = q$.
  - $S_i = \min(T_i, \tau)$ and $\delta_i = 1_{\{T_i \leq \tau\}}$.
- **Candidate Sets** are generated using the *Bernoulli Masking Model*.
  - Masking level controlled with simulation value p.
  - Failed component (latent K_i) placed in candidate set (observed $\mathcal{C}_i$).
  - Each functioning component included with probability p.

Synthetic Data and Simulation Values
How is the data generated in our simulation study?
Component Lifetimes (latent $T_{i1}, \cdots, T_{im}$) generated for each system.
Observed Data is a function of latent components.
Right-Censoring amount controlled with simulation value q.
Quantile q is probability system won't be right-censored.
Solve for right-censoring time τ in $\Pr\{T_i \leq \tau\} = q$.
$S_i = \min(T_i, \tau)$ and $\delta_i = 1_{\{T_i \leq \tau\}}$.
Candidate Sets are generated using the Bernoulli Masking Model.
Masking level controlled with simulation value p.
Failed component (latent K_i) placed in candidate set (observed $\mathcal{C}_i$).
Each functioning component included with probability p.

- Let's talk about how we **generate** the data for our **simulation** study.
- First, we generate the latent **component lifetimes** for the system just discussed.
- Then, we generate the data we actually **see** based on these lifetimes.
- The **right-censored** lifetimes, the **censoring indicators**, and **candidate sets**.
- In the simulations, we **control** the amount of **right-censoring** with the value q, the probability the system won't be right-censored.
- We use the **Bernoulli Masking Model** to generate the candidate sets.
- We **control** the masking level with the value p, the **Bernoulli** probability.
- – Explain procedure for generating candidates –

Bernoulli Masking Model: Satisfying Masking Conditions

The Bernoulli Masking Model *satisfies* the masking conditions:

- **Condition 1**: The failed component deterministically placed in candidate set.
- **Condition 2** and **3**: Bernoulli probability p is same for all components and fixed by us.
  - Probability of candidate set is constant conditioned on component failure within set.
  - Probability of candidate set, conditioned on a component failure, only depends on the p.

Future Research: Realistically conditions may be violated.

- Explore sensitivity of likelihood model to violations.

Bernoulli Masking Model: Satisfying Masking Conditions

The Bernoulli Masking Model *satisfies* the masking conditions:

- **Condition 1**: The failed component deterministically placed in candidate set.
- **Condition 2** and **3**: Bernoulli probability p is same for all components and fixed by us.
  - Probability of candidate set is constant conditioned on component failure within set.
  - Probability of candidate set, conditioned on a component failure, only depends on the p.

Future Research: Realistically conditions may be violated.

- Explore sensitivity of likelihood model to violations.

- It's important to show how our **Bernoulli** masking model used in our simulation study **satisfies** these masking conditions.
- We obviously satisfy **Condition 1** because the failed component is always placed in the candidate set.
- We satisfy **Condition 2** because the Bernoulli probability is the same for all components. As we vary the component failure within the set, the probability of the set doesn't change.
- We satisfy **Condition 3** because, conditioned on a failed component, the probability of the candidate set only depends on the Bernoulli probability, which is fixed by us and doesn't interact with the the system parameters.
- In **real life**, these conditions may be violated. Future research could explore the **sensitivity** of our likelihood model to violations of these conditions.

Performance Metrics

Objective: Evaluate the MLE and BCa confidence intervals' performance across various scenarios.

- Visualize the **simulated** sampling distribution of MLEs and 95% CIs.
- **MLE Evaluation**:
  - **Accuracy**: Bias
  - **Precision**: Dispersion of MLEs
    - 95% quantile range of MLEs.
- **95% CI Evaluation**:
  - **Accuracy**: Coverage probability (CP).
    - *Correctly Specified* CIs: CP near 95% (> 90% acceptable).
  - **Precision**: Width of median CI.

Performance Metrics

Objective: Evaluate the MLE and BCa confidence intervals' performance across various scenarios.

- Visualize the simulated sampling distribution of MLEs and 95% CIs.
- MLE Evaluation:
  - Accuracy: Bias
  - Precision: Dispersion of MLEs
    - 95% quantile range of MLEs.
- 95% CI Evaluation:
  - Accuracy: Coverage probability (CP).
    - Correctly Specified CIs: CP near 95% (> 90% acceptable).
  - Precision: Width of median CI.

- We want to evaluate the accuracy and precision of our MLE and CIs under various conditions.
- For the **MLE**, we're looking at its **bias** and **spread**.
- A **tight spread** indicates **high precision**, but if it's biased, we can't trust it.
- For the **CIs**, when we talk about accuracy, we're looking at **coverage probability**.
- We want our intervals to be **correctly specified**, meaning they cover the true parameter value around 95% of the time.
- Our **goal** is to get close to the nominal 95% level, but we'll consider anything above 90
- As for **precision**, we use the width of these intervals.
- A **narrow width** points to a **higher precision**, but that's meaningless if the CP is too low.

Scenario: Impact of Right-Censoring

Assess the impact of right-censoring on MLE and CIs.

- **Right-Censoring**: Failure observed with probability q: 60% to 100%.
  - Right censoring occurs with probability $1 - q$: 40% to 0%.
- **Bernoulli Masking Probability**: Each component is a candidate with probability p fixed at 21.5%.
  - Estimated from original study (Guo, Niu, and Szidarovszky 2013).
  - Chance of **no** masking: $\Pr\{\text{only failed component in } \mathcal{C}_i\} \approx 62\%$.
- **Sample Size**: n fixed at 90.
  - Small enough to show impact of right-censoring.

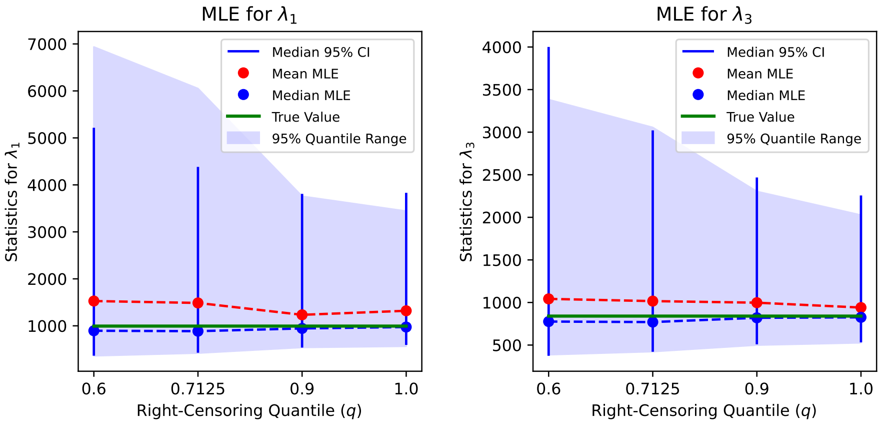
Scenario: Impact of Right-Censoring

Assess the impact of right-censoring on MLE and CIs.

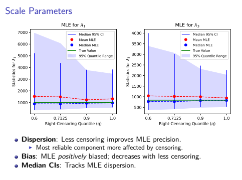
- **Right-Censoring**: Failure observed with probability q: 60% to 100%.
  - Right censoring occurs with probability $1 - q$: 40% to 0%.
- **Bernoulli Masking Probability**: Each component is a candidate with probability p fixed at 21.5%.
  - Estimated from original study (Guo, Niu, and Szidarovszky 2013).
  - Chance of **no** masking: $\Pr\{\text{only failed component in } \mathcal{C}_i\} \approx 62\%$.
- **Sample Size**: n fixed at 90.
  - Small enough to show impact of right-censoring.

- We assess the impact of **right-censoring** on the MLE and confidence intervals.
- We **vary** the probability of observing a failure from 60% to 100%.
- We fix the masking probability at 21.5%, which is the probability that each component is a candidate.
- This masking probability is **based** on estimates from the original study.
- We fix the **sample size** at 90, which was small enough to show the impact of right-censoring on the MLE, but large enough so that the convergence rate was reasonable.

Scale Parameters

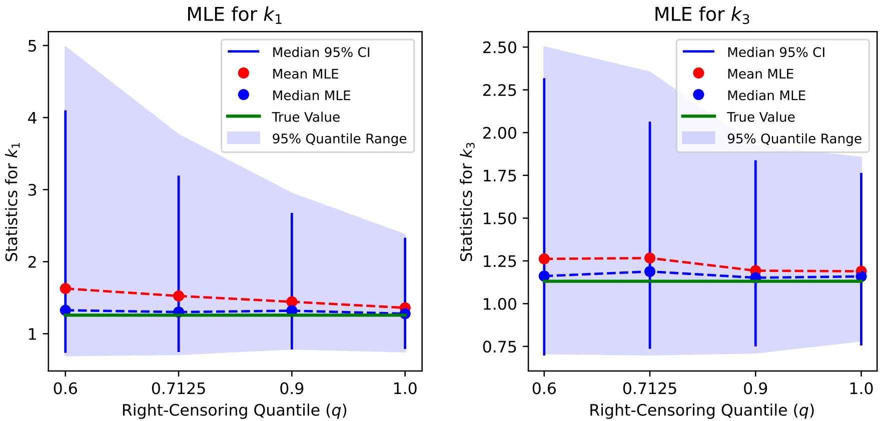


- **Dispersion**: Less censoring improves MLE precision.
  - Most reliable component more affected by censoring.
- **Bias**: MLE *positively* biased; decreases with less censoring.
- **Median CIs**: Tracks MLE dispersion.

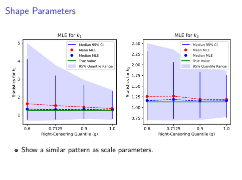


- Here, we show two graphs for the **scale parameters** of the two components, the most reliable on the left and the least reliable on the right.
- In **light solid blue**, we show the dispersion of the MLE. We see that it improves with less censoring.
- We see that the **more reliable** component has more dispersion than the other component.
- This is **due** to more reliable components being more likely to be censored.
- In the **dashed red line**, we show the mean of the MLEs. In green, we show the true values.
- The MLEs are **positively** biased, but that bias decreases as censoring level is reduced.
- In the **dark blue** vertical lines, we show the medians of the

Shape Parameters

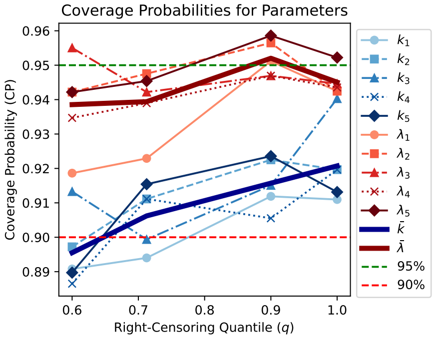


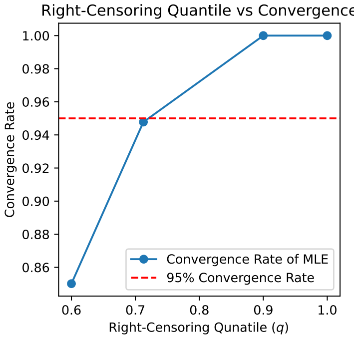
- Show a similar pattern as scale parameters.



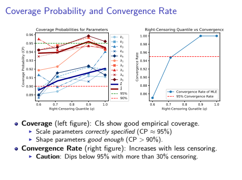
- We see similar results for the **shape parameters**.
- So, let's move on to evaluating the **accuracy** of the **confidence intervals**, where we do see some notable differences.

Coverage Probability and Convergence Rate





- **Coverage** (left figure): CIs show good empirical coverage.
  - Scale parameters *correctly specified* (CP $\approx$ 95%)
  - Shape parameters *good enough* (CP $>$ 90%).
- **Convergence Rate** (right figure): Increases with less censoring.
  - **Caution**: Dips below 95% with more than 30% censoring.



- On the **left** figure, we show the impact of **right-censoring** on the **coverage probability**.
- In the **bold** red line, we show the **mean** coverage for the scale parameters.
- It shows that the **coverage** is **correctly specified** across all censoring levels.
- In the **bold** blue line, we show the **mean** coverage for the shape parameters. They are **acceptable**, with coverage above 90%.
- In the **right** figure, we show the **convergence rate** for the MLE.
- At more than 30% censoring, the convergence rate dips below 95%.
- Combined with moderate failure masking and small samples, we suggest **caution** in interpreting the results.

Key Takeaways: Right-Censoring

Right-censoring has a notable impact on the MLE:

- **MLE Precision**:
  - Improves notably with reduced right-censoring levels.
  - More reliable components benefit more from reduced right-censoring.
- **Bias**:
  - MLEs show positive bias, but decreases with reduced right-censoring.
- **Convergence Rates**:
  - MLE convergence rate improves with reduced right-censoring.
  - Dips: $< 95\%$ at $> 30\%$ right-censoring.

BCa confidence intervals show good empirical coverage.

- CIs offer reliable *empirical coverage*.
- Scale parameters *correctly specified* across all right-censoring levels.

Key Takeaways: Right-Censoring

Right-censoring has a notable impact on the MLE:

- **MLE Precision**:
  - Improves notably with reduced right-censoring levels.
  - More reliable components benefit more from reduced right-censoring.
- **Bias**:
  - MLEs show positive bias, but decreases with reduced right-censoring.
- **Convergence Rates**:
  - MLE convergence rate improves with reduced right-censoring.
  - Dips: $< 95\%$ at $> 30\%$ right-censoring.

BCa confidence intervals show good empirical coverage.

- CIs offer reliable *empirical coverage*.
- Scale parameters *correctly specified* across all right-censoring levels.

Scenario: Impact of Failure Masking

Assessing the impact of the failure masking level on MLE and CIs.

- **Bernoulli Masking Probability**: Vary Bernoulli probability p from 10% to 70%.
- **Right-Censoring**: q fixed at 82.5%.
  - Right-censoring occurs with probability $1 - q$: 17.5%.
  - Censoring less prevalent than masking.
- **Sample Size**: n fixed at 90.
  - Small enough to show impact of masking.

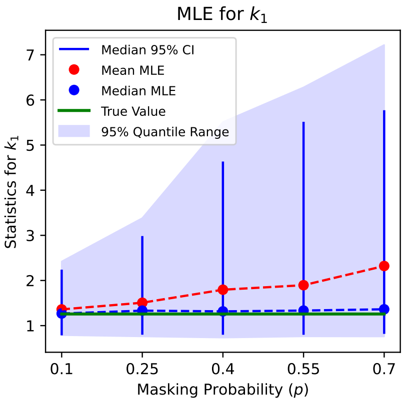
Scenario: Impact of Failure Masking

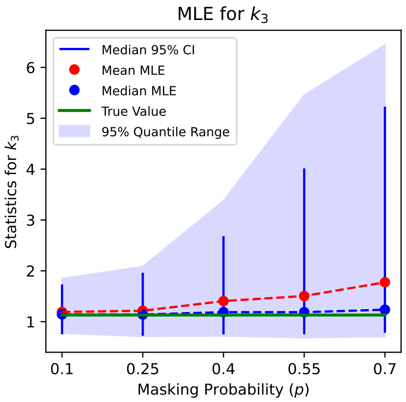
Assessing the impact of the failure masking level on MLE and CIs.

- **Bernoulli Masking Probability**: Vary Bernoulli probability p from 10% to 70%.
- **Right-Censoring**: q fixed at 82.5%.
  - Right-censoring occurs with probability $1 - q$: 17.5%.
  - Censoring less prevalent than masking.
- **Sample Size**: n fixed at 90.
  - Small enough to show impact of masking.

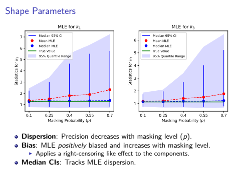
- Here, we assess the impact of **masking levels** on the MLE and confidence intervals.
- We **vary** the Bernoulli masking probability from 10% to 70%.
- We fix the right-censoring probability at 17.5%.
- The **chances** of **censoring** are less than masking.
- We fix the **sample size** at 90, which was small enough to show the impact of masking on the MLE, but large enough so that the convergence rate was reasonable.

Shape Parameters



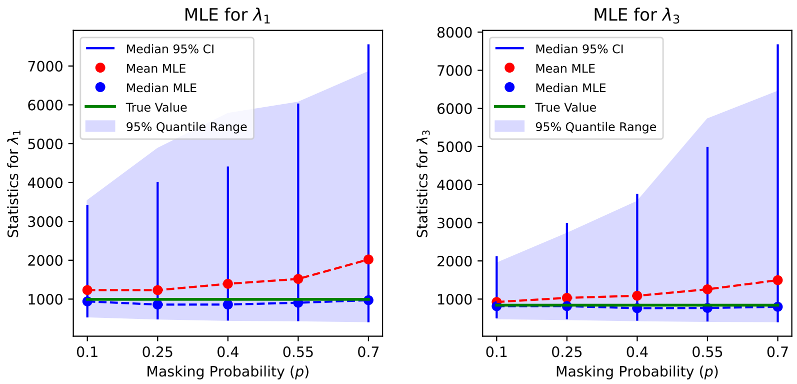


- **Dispersion**: Precision decreases with masking level (p).
- **Bias**: MLE *positively* biased and increases with masking level.
  - Applies a right-censoring like effect to the components.
- **Median CIs**: Tracks MLE dispersion.

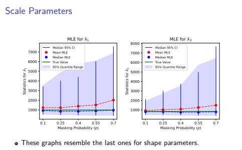


- Here, we show the impact of **masking** on the MLE and confidence intervals, this time for the **shape parameters**.
- In **light solid blue**, we show the dispersion of the MLE. We see that as increases with masking level.
- Unlike for the scale parameter, the **more reliable** component on the left has only slightly more dispersion than the other component.
- In the **dashed red line**, we show the bias. The MLE is **positively** biased, and increases with masking level.
- In the **dark blue** vertical lines, we show the median confidence intervals.
- Again, we see they they **track** the MLE's dispersion.

Scale Parameters

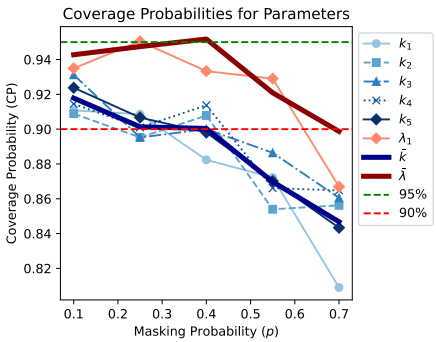


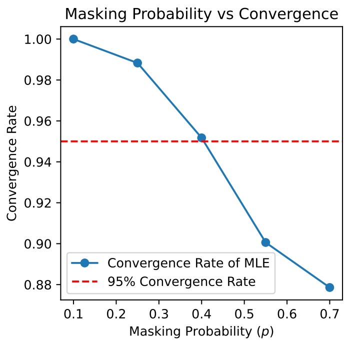
- These graphs resemble the last ones for shape parameters.



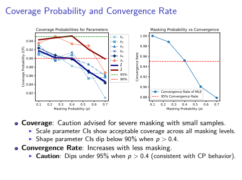
- We see similar results for the **scale parameters**.
- So, let's move on to evaluating the **accuracy** of the **confidence intervals**, where we do continue to see some differences.

Coverage Probability and Convergence Rate





- **Coverage**: Caution advised for severe masking with small samples.
  - Scale parameter CIs show acceptable coverage across all masking levels.
  - Shape parameter CIs dip below 90% when $p > 0.4$.
- **Convergence Rate**: Increases with less masking.
  - **Caution**: Dips under 95% when $p > 0.4$ (consistent with CP behavior).



Key Takeaways: Masking

The masking level of component failures profoundly affects the MLE:

- **MLE Precision**:
  - Decreases with more masking.
- **MLE Bias**:
  - Positive bias is amplified with increased masking.
  - Masking exhibits a right-censoring-like effect.
- **Convergence Rate**:
  - Commendable for Bernoulli masking levels $p \leq 0.4$.
    - *Extreme* masking: some masking occurs 90% of the time at $p = 0.4$.

The BCa confidence intervals show good coverage:

- **Scale** parameters maintain good coverage across all masking levels.
- **Shape** parameter coverage dip below 90% when $p > 0.4$.
  - Caution advised for severe masking with small samples.

Key Takeaways: Masking

The masking level of component failures profoundly affects the MLE:

- **MLE Precision**:
  - Decreases with more masking.
- **MLE Bias**:
  - Positive bias is amplified with increased masking.
  - Masking exhibits a right-censoring-like effect.
- **Convergence Rate**:
  - Commendable for Bernoulli masking levels $p \leq 0.4$.
    - *Extreme* masking: some masking occurs 90% of the time at $p = 0.4$.

The BCa confidence intervals show good coverage:

- **Scale** parameters maintain good coverage across all masking levels.
- **Shape** parameter coverage dip below 90% when $p > 0.4$.
  - Caution advised for severe masking with small samples.

Scenario: Impact of Sample Size

Assess the mitigating affects of sample size.

- **Sample Size**: We vary the same size n from 50 to 500..
- **Right-Censoring**: q fixed at 82.5%
  - 17.5% chance of right-censoring.
- **Bernoulli Masking Probability**: p fixed at 21.5%
  - Some masking occurs 62% of the time.

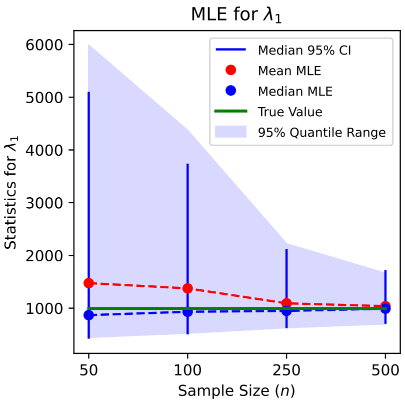
Scenario: Impact of Sample Size

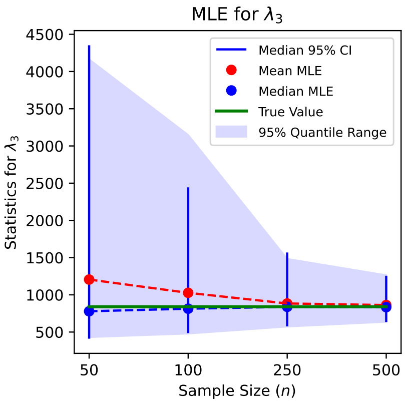
Assess the mitigating affects of sample size.

- **Sample Size**: We vary the same size n from 50 to 500..
- **Right-Censoring**: q fixed at 82.5%
  - 17.5% chance of right-censoring.
- **Bernoulli Masking Probability**: p fixed at 21.5%
  - Some masking occurs 62% of the time.

- We assess the impact of the **sample size** on the MLE and confidence intervals.
- We want to see how will it mitigate the challenges from right-censoring and masking.
- We **vary** the sample sie from size 50 to size 500.
- We fix the masking probability at 21.5% and the right-censoring probability at 17.5%.

Scale Parameters

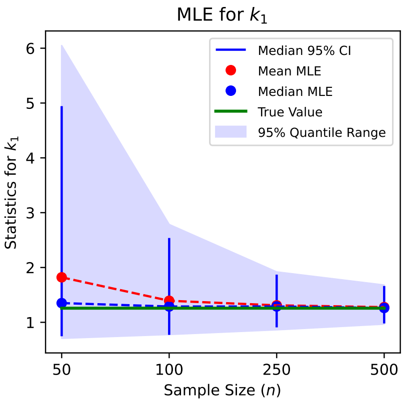


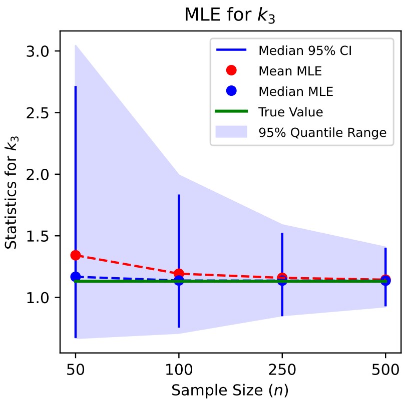


- **Dispersion**: Increasing sample size improves MLE precision.
  - Extremely precise for $n \geq 250$.
- **Bias**: Large *positive* bias initially, but diminishes to zero.
  - Large samples counteract right-censoring and masking effects.
- **Median CIs**: Track MLE dispersion. Very tight for $n \geq 250$.

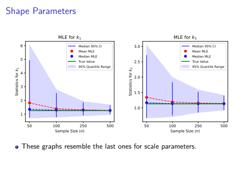
Scale Parameters
- **Dispersion**: Increasing sample size improves MLE precision.
  - Extremely precise for $n \geq 250$.
- **Bias**: Large *positive* bias initially, but diminishes to zero.
  - Large samples counteract right-censoring and masking effects.
- **Median CIs**: Track MLE dispersion. Very tight for $n \geq 250$.

Shape Parameters



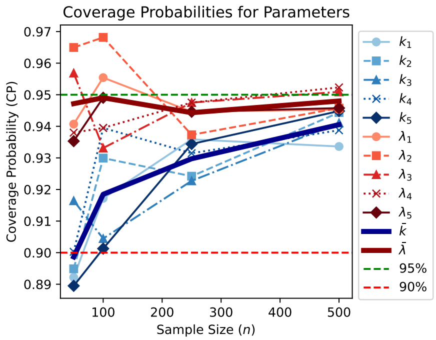


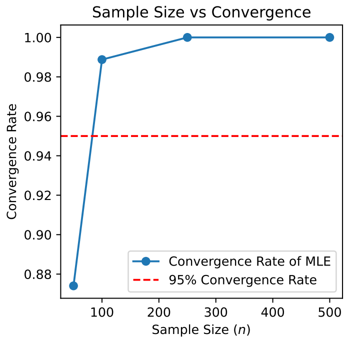
- These graphs resemble the last ones for scale parameters.



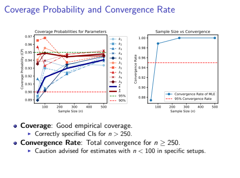
- Again, we see similar results for the **shape parameters**.

Coverage Probability and Convergence Rate





- **Coverage**: Good empirical coverage.
  - Correctly specified CIs for $n > 250$.
- **Convergence Rate**: Total convergence for $n \geq 250$.
  - Caution advised for estimates with $n < 100$ in specific setups.



Key Takeaways: Sample Size

Sample size has a notable impact on the MLE:

- **Precision**: Very precise for large samples ($n > 200$).
- **Bias**: Diminishes to near zero for large samples.
- **Coverage**: Correctly specified CIs for large samples.
- **Convergence**: Total convergence for large samples.

Summary

Larger samples lead to more accurate, unbiased, and reliable estimations.

- Mitigates the effects of right-censoring and masking.

Key Takeaways: Sample Size

Sample size has a notable impact on the MLE:

- **Precision**: Very precise for large samples ($n > 200$).
- **Bias**: Diminishes to near zero for large samples.
- **Coverage**: Correctly specified CIs for large samples.
- **Convergence**: Total convergence for large samples.

Summary

Larger samples lead to more accurate, unbiased, and reliable estimations.

- Mitigates the effects of right-censoring and masking.

Section 4

Overall Conclusion

Section 4
Overall Conclusion

Overall Conclusion

Key Findings:

- Employed maximum likelihood techniques for component reliability estimation in series systems with masked failure data.
- Methods were robust despite masking and right-censoring challenges.

MLE Performance:

- Right-censoring and masking introduce positive bias for our setup.
  - More reliable components are more affected.
- Shape parameters harder to estimate than scale parameters.
- Large samples can mitigate the affects of masking and right-censoring.

BCa Confidence Interval Performance:

- Width of CIs tracked MLE dispersion.
- Good empirical coverage in most scenarios.

Overall Conclusion

Key Findings:

- Employed maximum likelihood techniques for component reliability estimation in series systems with masked failure data.
- Methods were robust despite masking and right-censoring challenges.

MLE Performance:

- Right-censoring and masking introduce positive bias for our setup.
  - More reliable components are more affected.
- Shape parameters harder to estimate than scale parameters.
- Large samples can mitigate the affects of masking and right-censoring.

BCa Confidence Interval Performance:

- Width of CIs tracked MLE dispersion.
- Good empirical coverage in most scenarios.

Future Work and Discussion

Directions to enhance learning from masked data:

- **Relax Masking Conditions**: Assess sensitivity to violations and and explore alternative likelihood models.
- **System Design Deviations**: Assess estimator sensitivity to deviations.
- **Homogenous Shape Parameter**: Analyze trade-offs with the full model.
- **Bootstrap Techniques**: Semi-parametric approaches and prediction intervals.
- **Regularization**: Data augmentation and penalized likelihood methods.
- **Additional Likelihood Contributions**: Predictors, etc.

Future Work and Discussion

Directions to enhance learning from masked data:

- **Relax Masking Conditions**: Assess sensitivity to violations and and explore alternative likelihood models.
- **System Design Deviations**: Assess estimator sensitivity to deviations.
- **Homogenous Shape Parameter**: Analyze trade-offs with the full model.
- **Bootstrap Techniques**: Semi-parametric approaches and prediction intervals.
- **Regularization**: Data augmentation and penalized likelihood methods.
- **Additional Likelihood Contributions**: Predictors, etc.