Assignment Five

Michael Kerr

2024-01-30

Introduction

The basic two-parameter Weibull distribution is of the form:

$$f(x;k,\lambda)=\frac{k}{\lambda}\left(\frac{x}{\lambda}\right)^{k-1}\exp\left\{-\left(\frac{x}{\lambda}\right)^{k}\right\}1_{\mathbb{R}_+}(x)$$

where k, λ are labeled the 'scale' and 'shape' parameters, respectively. This distribution function forms the basis for many adjusted distributions and has been referenced over 1000 times (Lai, Murthy, and Xie (2011)). The Weibull distribution, named after Waloddi Weibull for his extensive research on it, is used most among survival analysts for studying lifetimes. Waloddi developed this distribution for breaking strength of materials and found that it was particularly useful for describing the distribution of the survival of mechanical components. It is now commonly used for survival purposes: to analyse the behaviour of time-to-failure, or death in the case of medicine.

Makalic and Schmidt (2023) introduce new bias adjusted maximum likelihood estimates for purposes of survival analysis. They use methods of traditional maximum likelihood estimation (MLE), as described in this paper, and then compare it to bias adjusted MLEs for complete data and type 1 data. The methods of bias adjustment involve simulation of small and finite sample sizes while also keeping $\lambda = 1$. They perform simulation on simulated censored data and eventually real-life data to evaluate their hypotheses. This report is a demonstration of the basis of this paper, and of bootstrap and simulation in general. This report does not go into the survival parts of the Weibull distribution, and assumes that λ is chosen such that $\mathbb{E}[f(x;k,\lambda)] = \lambda\Gamma\left(1+1/k\right) = 1$. Therefore $\lambda = \frac{1}{\Gamma(1+1/k)}$.

The first two sections go over the idea around inverse-transform sampling and maximum-likelihood estimation. After which is the methodology, algorithm, results discussion then specifications.

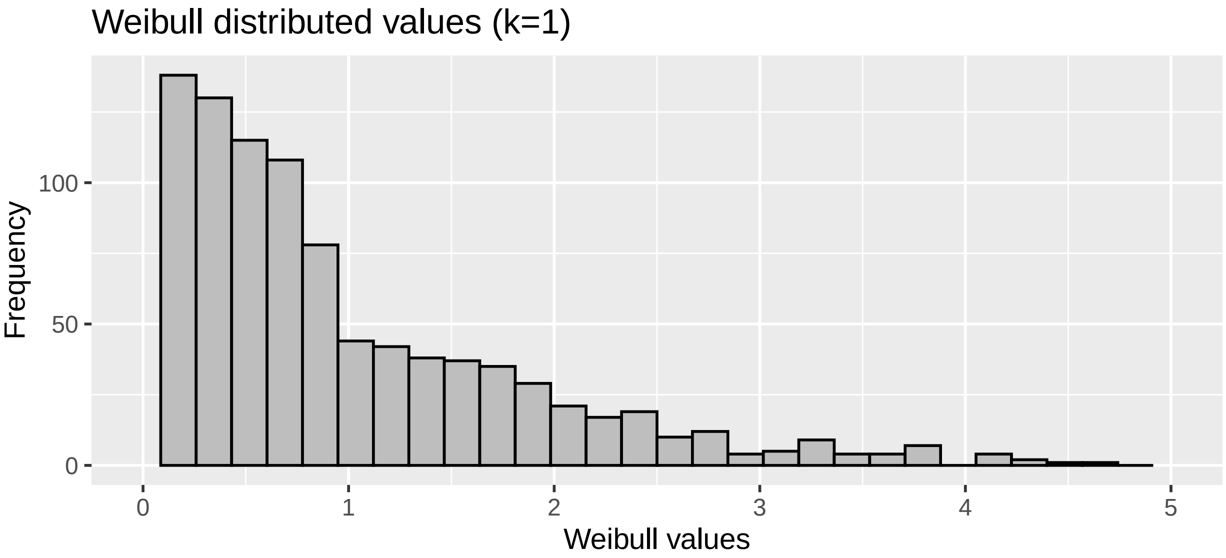
Generating Weibull distributed data

Using inverse transform sampling, it is possible to generate values from any distribution, D, given a known CDF. If Y is a continuous random variable, then $F_Y^{-1}(u) \sim D$ where $u \sim U(0,1)$.

To generate Weibull distributed random variables, we use this method. The Weibull distribution (with λ s.t. $\mathbb{E}(X) = 1$) has CDF $F(x;k) = 1 - \exp(-\left(x\Gamma(1+1/k)\right)^{k})$. This is then inverted to get:

$$x=\frac{\log\left(\frac{1}{1-u}\right)^{\frac{1}{k}}}{\Gamma\left(1+\frac{1}{k}\right)}$$

So generating values of $u \sim U(0,1)$ and plugging them into the above will generate Weibull random variables. The following shows Weibull distributed values with $k = 1$.



Maximum likelihood estimator

The maximum likelihood estimator is a way of estimating the most optimal value of a parameter using the likelihood function.

Makalic and Schmidt (2023) show that parameter estimation can be accomplished using the technique of maximum likelihood estimation for the Weibull distribution. This involves maximizing the likelihood function (often the log-likelihood). The Weibull likelihood and log-likelihood function is:

$$L\left(k, \lambda | x_i\right) = \prod_{i=1}^{n} \frac{k}{\lambda} \left(\frac{x_i}{\lambda}\right)^k \exp\left(-\left(\frac{x_i}{\lambda}\right)^k\right)$$

$$\ell(k, \lambda | x_i) = -n \log\left(\frac{\lambda^k}{k}\right) + (k-1)\left(\sum_{i=1}^{n} \log x_i\right) - \sum_{i=1}^{n} \left(\frac{x_i}{\lambda}\right)^k$$

In maximizing the log-likelihood, it is found that the maximum likelihood estimate of λ and k are:

$$\widehat{\lambda} = \left(\frac{1}{n} \sum_{i=1}^{n} y_i^k\right)^{\frac{1}{k}}$$

$$0 = \frac{n}{k} + \sum_{i=1}^{n} \log x_i - \frac{n \sum_{i=1}^{n} x_i^k \log x_i}{\sum_{i=1}^{n} x_i^k}$$

where k must be solved numerically in the second equation.

Methodology

Each loop tested parameters n and k, where n is a sample of $(10, 100, 500)$ Weibull distributed variables and $k \in (0.5, 1, 2, 4)$. n samples were taken for each maximum likelihood estimate of k which was repeated $\frac{30,000}{n}$ times so $3000, 300, 60$ repetitions per n. These maximum likelihood estimates were compared to bootstrap maximum likelihood estimates. These bootstraps were sampled with replacement 100 times per repetition.

k cannot be determined analytically so numerical approximation is used. The function uniroot optimizes the objective function, as described in the maximum likelihood estimator section, using the bisection method (Burden and Faires (1985)). To get the Weibull data for maximum likelihood estimation inverse transform sampling was used with seed=38 using the Mersenne-Twister algorithm.

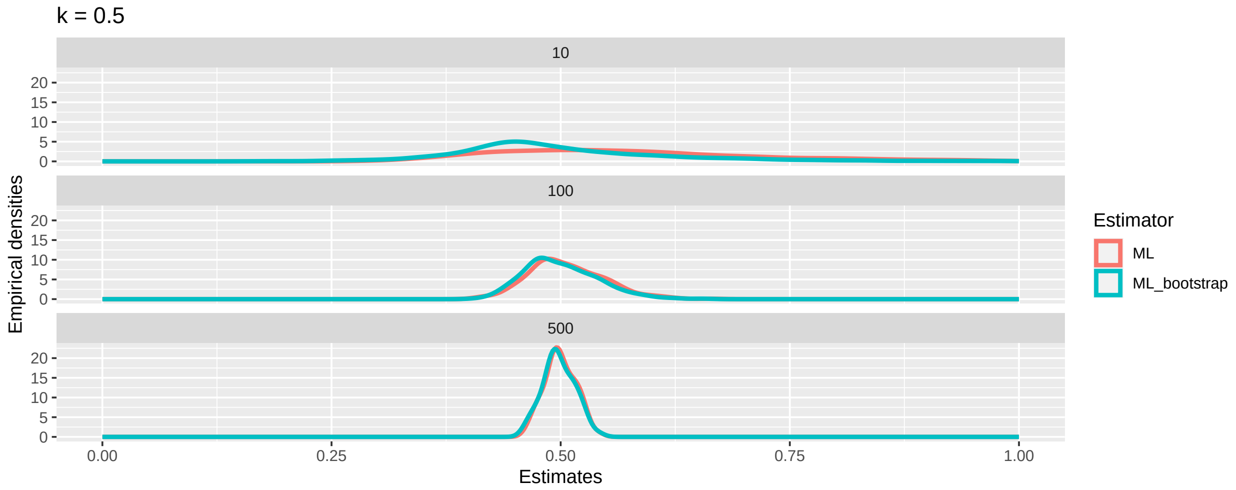
The bias and mean-squared error are calculated by using the following:

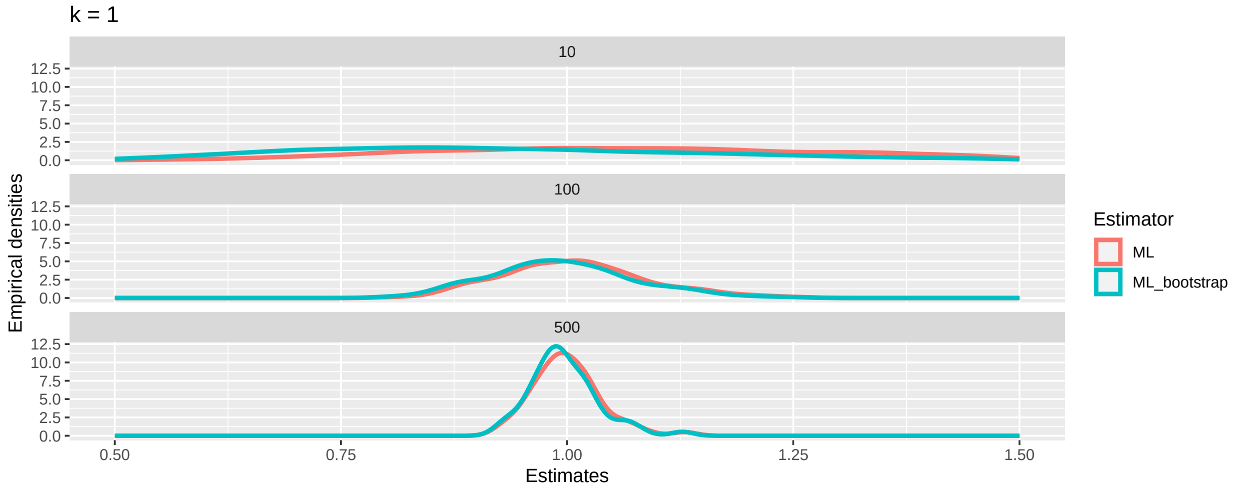
$$\begin{aligned}\text{Bias}(\widehat{\theta}(X)) &= \mathbb{E}(\widehat{\theta}(X)) - \theta \\ \text{MSE}(\widehat{\theta}(X)) &= \text{Bias}(\widehat{\theta}(X))^2 + \text{Var}(\widehat{\theta}(X))\end{aligned}$$

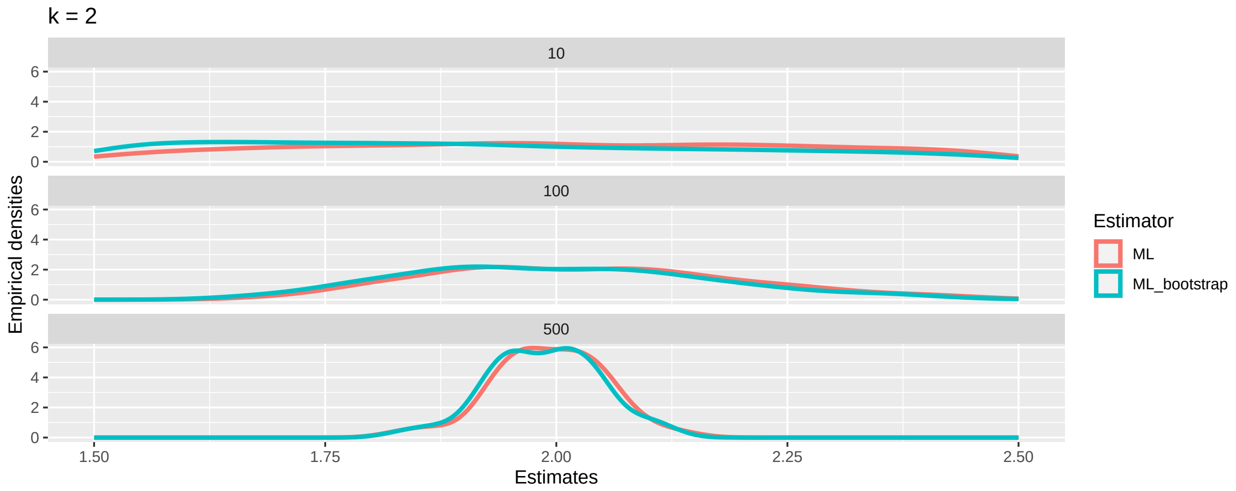
Simulation algorithm

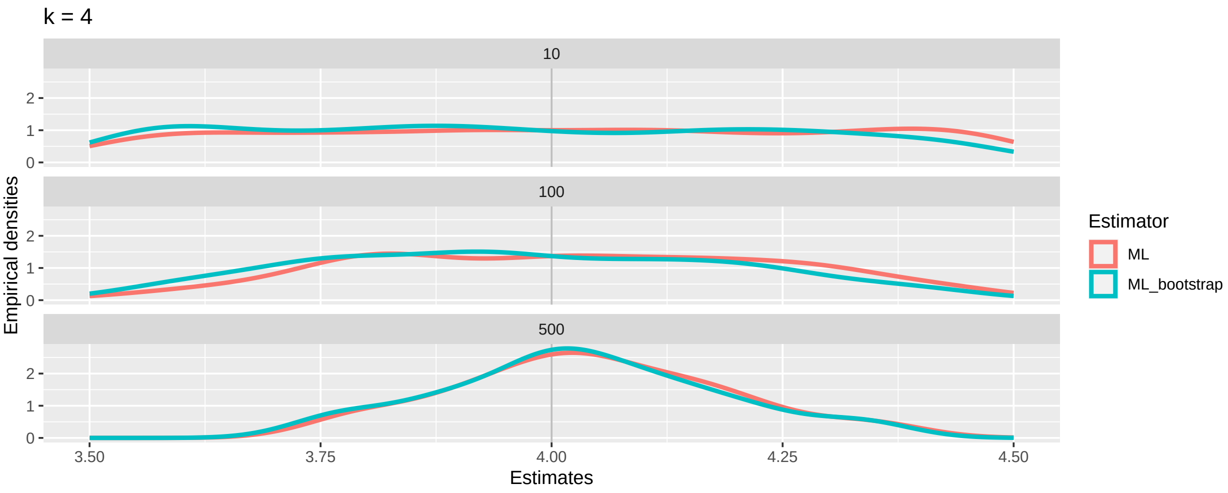
```
for k in (0.5, 1, 2, 4)
  for n in (10, 100, 500)
    for rep in 30000/n
      Generate X ~ Weibull(k)
      Generate maximum likelihood of k from MLE
      Do 100 times:
        Generate B ~ Weibull(k) with replacement
        Generate maximum likelihood of k from MLE bootstrap
      Calculate bootstrap using mean of B (2*k_MLE - mean(B))
```

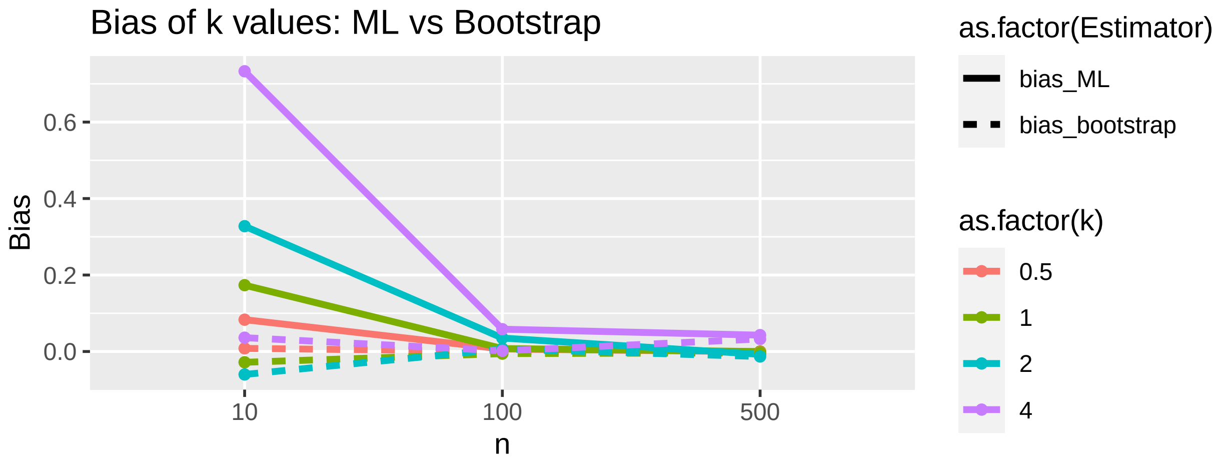
Results

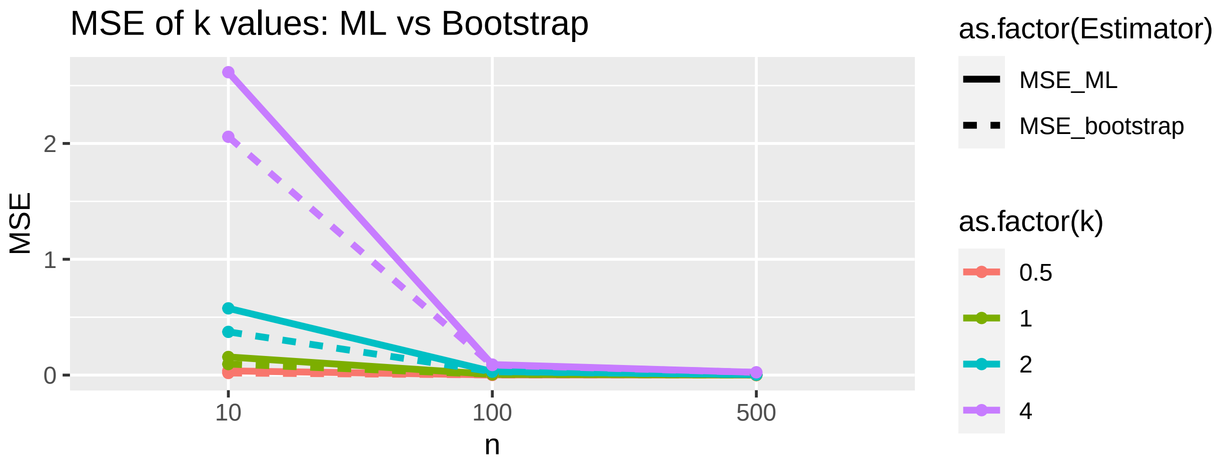

k = 0.5
10
100
500
Empirical densities
0
5
10
15
20
0.00
0.25
0.50
0.75
1.00
Estimates
Estimator
ML
ML_bootstrap

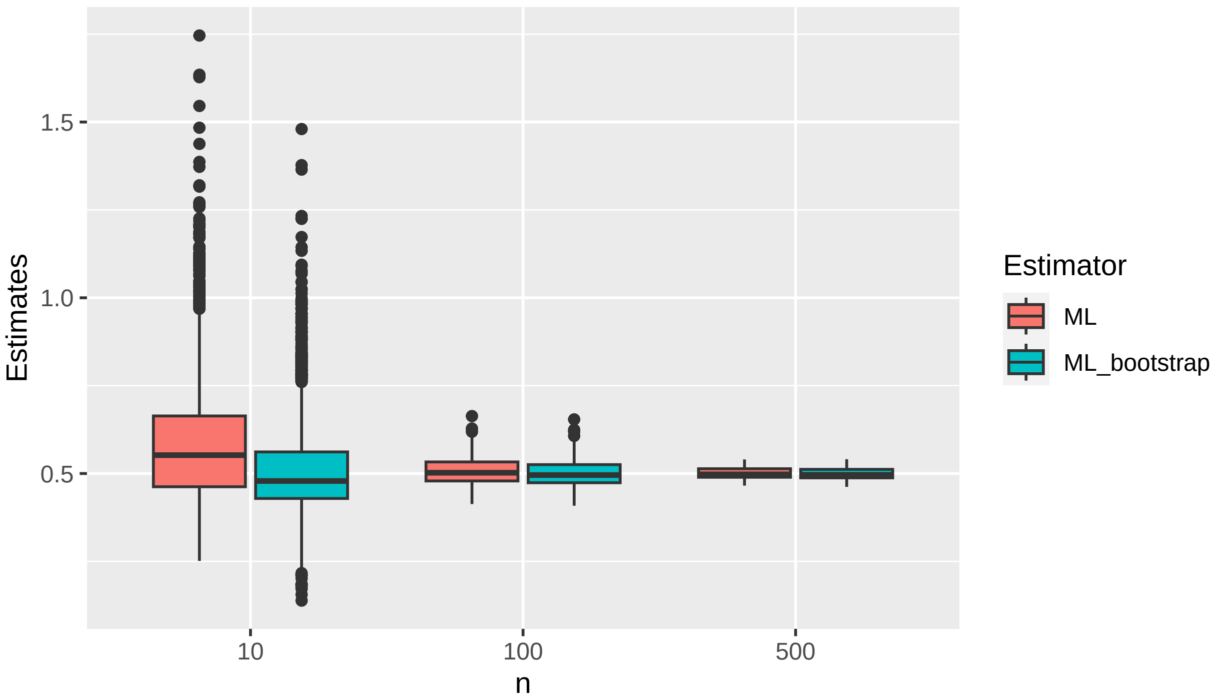

k = 1
10
100
500
Empirical densities
0.0
2.5
5.0
7.5
10.0
12.5
0.50
0.75
1.00
1.25
1.50
Estimates
Estimator
ML
ML_bootstrap


k = 2
10
100
500
Empirical densities
0
2
4
6
1.50
1.75
2.00
2.25
2.50
Estimates
Estimator
ML
ML_bootstrap

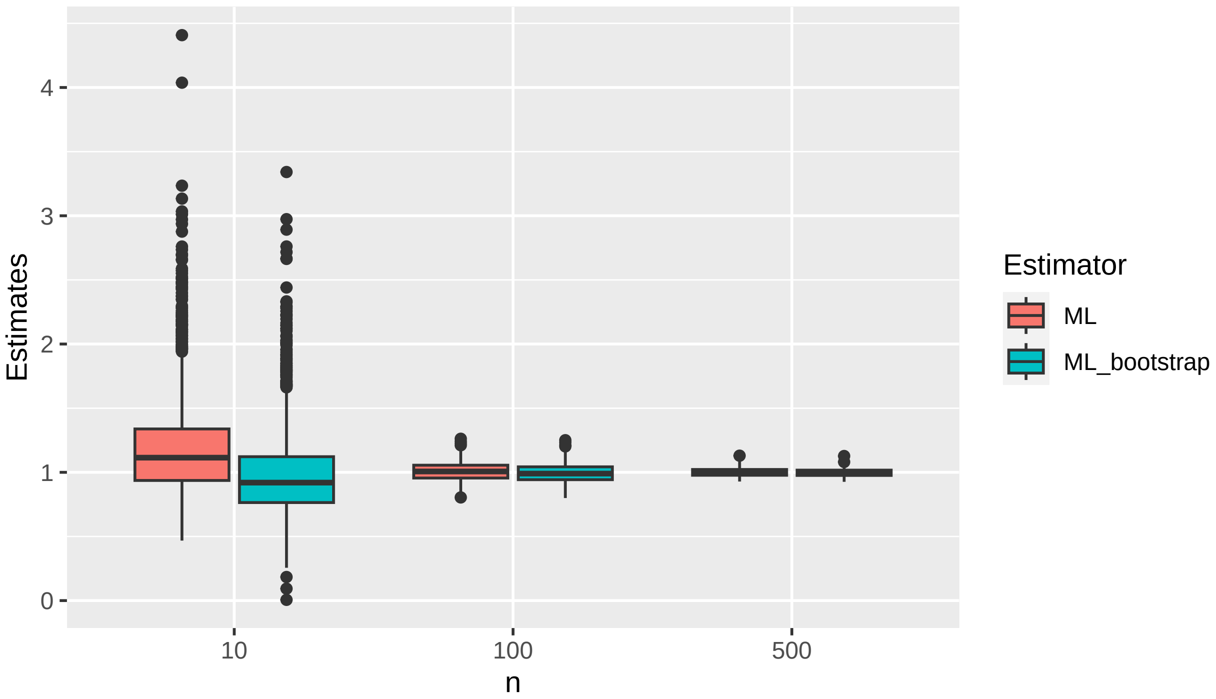

k = 4
10
100
500
Empirical densities
0
1
2
3.50
3.75
4.00
4.25
4.50
Estimates
Estimator
ML
ML_bootstrap


Bias of k values: ML vs Bootstrap
Bias
0.0
0.2
0.4
0.6
10
100
500
n
as.factor(Estimator)
bias_ML
bias_bootstrap
as.factor(k)
0.5
1
2
4

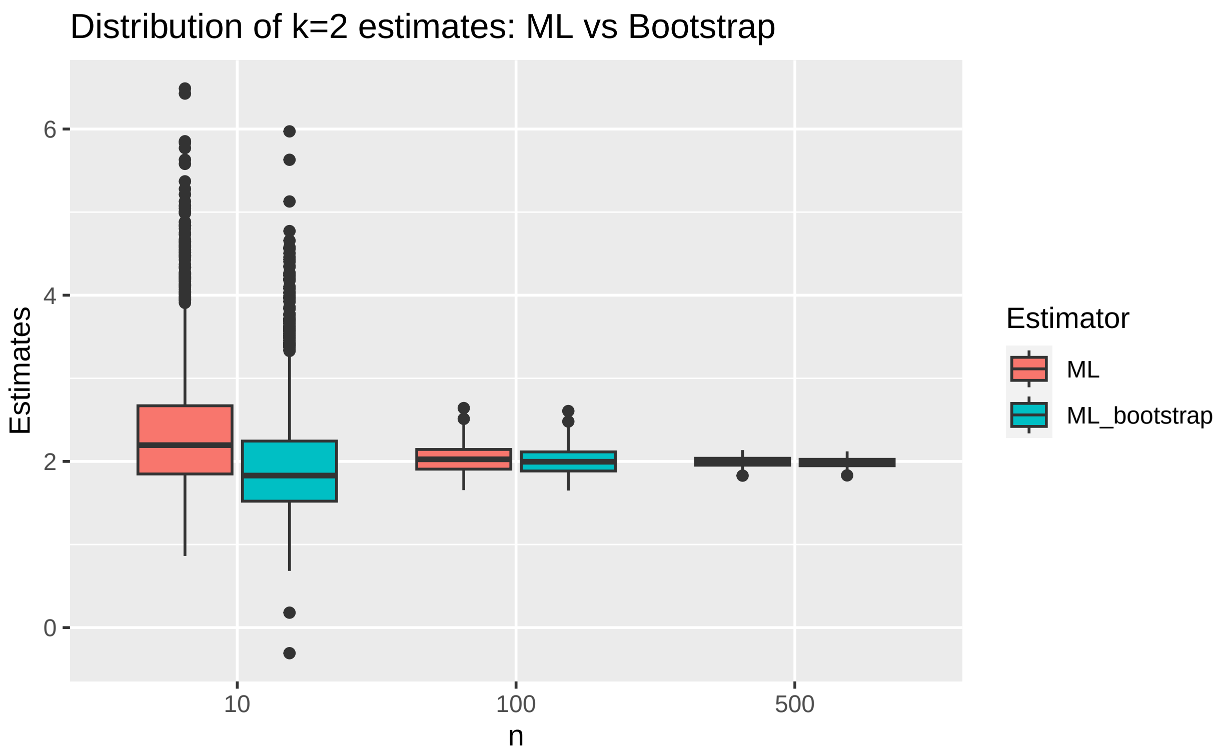

MSE of k values: ML vs Bootstrap
MSE
0
1
2
10
100
500
n
as.factor(Estimator)
MSE_ML
MSE_bootstrap
as.factor(k)
0.5
1
2
4



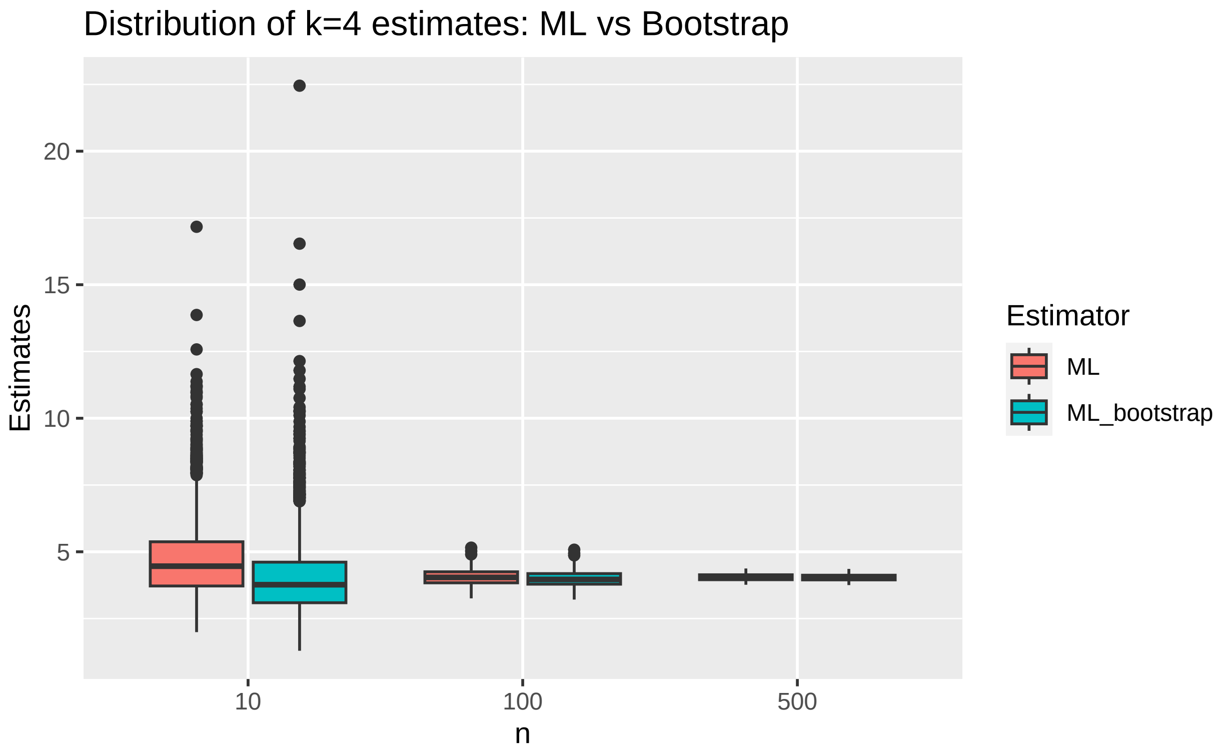
Distribution of k=0.5 estimates: ML vs Bootstrap



Distribution of k=1 estimates: ML vs Bootstrap



Distribution of k=2 estimates: ML vs Bootstrap



Distribution of k=4 estimates: ML vs Bootstrap

k	Estimator	Kurtosis	Skewness
0.5	ML	7.160554	1.536936
0.5	ML_bootstrap	8.527191	1.617505
1.0	ML	10.296470	1.833741

k	Estimator	Kurtosis	Skewness
1.0	ML_bootstrap	8.585926	1.511066
2.0	ML	6.684594	1.494116
2.0	ML_bootstrap	6.412836	1.251265
4.0	ML	8.025075	1.630958
4.0	ML_bootstrap	19.851505	2.506946

Discussion

The performance of each estimator improves significantly as the sample size increases, but of particular note is the much worsened performance of estimator precision for small samples. These estimates get noticably worsened when k increases. The reasons for this, as described in Abbasi et al. (2006), are possibly due to the necessity of numerical optimization when determining k. They say that the usual Newton-Raphson technique of gradient descent is not a good method as it can get stuck add other optimal points, and that as the sample size increases the more complicated the likelihood function to maximize. Some obvious signs of this are the slight peaks that appear near the true value at $k = 0.5$ and $k = 1$.

Bias and variance

The formula for the mean-squared error can be shown to be of the form:

$$MSE(X) = \text{Bias}(X)^2 + \text{Var}(X)$$

Notice that Bias plays a far larger weight than variance meaning that, if the bootstrap estimator is far less biased, it must have a large variance if it has a higher MSE than the ML estimator. Both estimators reduce bias and MSE as the number of samples increase: showing signs of optimization.

Estimator distributions

The kurtosis and skewness between the two estimators vary significantly for all values of k. The bootstrap maximum likelihood estimate has far larger tails and is much less like a peak. These values give significant evidence of large variation between values for not only the bootstrap but the maximum likelihood too. The maximum likelihood is a much more consistent estimator but also far less biased.

Conclusion

Neither estimator is a very good one. Multiple papers (Abbasi et al. (2006), Gibbons and Vance (1981)) agree that simulating two-parameter Weibull distributions using regular bias reduction techniques is not enough for parameter estimation.

Computer hardware and software

- **GPU:** NVIDIA GeForce RTX 3060
- **Processor:** AMD Ryzen 5 5600G with Radeon Graphics
- **OS:** Windows 11 22H2
- **Simulation time:** 15m 19s
- **Software:** R, ggplot2, runif
- **Seed:** Mersenne-Twister; 38

References

Abbasi, Babak, Abdol Hamid Eshragh Jahromi, Jamal Arkat, and Mehdi Hosseinkouchack. 2006. "Estimating the Parameters of Weibull Distribution Using Simulated Annealing Algorithm." *Applied Mathematics and Computation* 183 (1): 85–93. https://doi.org/https://doi.org/10.1016/j.amc.2006.05.063.

Burden, Richard, and Douglas Faires. 1985. "The Bisection Algorithm." *Numerical Analysis*.

Gibbons, Diane I., and Lonnie C. Vance. 1981. "A Simulation Study of Estimators for the 2-Parameter Weibull Distribution." *IEEE Transactions on Reliability* R-30 (1): 61–66. https://doi.org/10.1109/TR.1981.5220965.

Lai, C. D., D. N. P. Murthy, and M. Xie. 2011. "Weibull Distributions." *WIREs Computational Statistics* 3 (3): 282–87. https://doi.org/https://doi.org/10.1002/wics.157.

Makalic, Enes, and Daniel F. Schmidt. 2023. "Maximum Likelihood Estimation of the Weibull Distribution with Reduced Bias." *Stat Comput*. Springer.