Exponential Distribution

Abhijith Asok, Chris Hilger, Liam Loscalzo, Katherine Wang

Point Estimation: Methods

We compared the following point estimators:

- Maximum Likelihood Estimator
- Unbiased correction for the MLE
- Second Method of Moment Estimator

##add criteria for comparing estimators!

Maximum Likelihood Estimator

```
##code here
```

Unbiased correction for the MLE

```
##code here
```

Second Method of Moment Estimator

```
##code here
```

Confidence Intervals: Methods

We compared the following confidence intervals:

- Wald-based Confidence Interval
- Gamma-based Confidence Interval
- Score-based Confidence Interval
- Bootstrap Confidence Interval

##add criteria for comparing CIs!!

Wald Confidence Interval

```
wald_ci <- function(N, rate, alpha = 0.05){
  x <- rexp(N, rate = rate)
  x_bar <- mean(x)
  se <- sd(x)/sqrt(N)
  ci <- x_bar + c(-1, 1)*qnorm(1 - (alpha / 2))
  return(ci)
}
```

Gamma Confidence Interval

If $X_1, X_2, ..., X_n \overset{\text{iid}}{\sim} \text{Exponential}(\lambda)$, then $\sum_i^n x_i \sim \text{Gamma}(n, \lambda) \implies \lambda\bar{x} \sim \text{Gamma}(n, n)$. Let g_y be the yth percentile of this distribution. Then we can say:

$$1 - \alpha = P(g_{\alpha/2} \leq \lambda\bar{x} \leq g_{1-\alpha/2}) = P(g_{\alpha/2}/\bar{x} \leq \lambda\bar{x} \leq g_{1-\alpha/2}/\bar{x})$$

And therefore, a $(1 - \alpha)\%$ confidence interval for λ is

$$(g_{\alpha/2}/\bar{x},\ g_{1-\alpha/2}/\bar{x})$$

```
gamma_ci <- function(N, rate, alpha = 0.05){
  # Inspiration: https://math.stackexchange.com/questions/128
  x <- rexp(N, rate = rate)
  x_bar <- mean(x)
  ci_rate <- qgamma(c(alpha / 2, 1 - (alpha / 2)), N, N) / x_b
  ci_mean <- 1 / ci_rate
  return(c(ci_mean[2], ci_mean[1]))
}
```

Score Confidence Interval

```
score_ci <- function(N, rate, alpha = 0.05) {
  x <- rexp(N, rate = rate)
  xbar <- mean(x)
  ci_lambda <- (1/xbar)*(1 + c(-1, 1) * qnorm(1 - alpha / 2) /
  ci_mean <- 1 / ci_lambda
  return(c(ci_mean[2], ci_mean[1]))
}
```

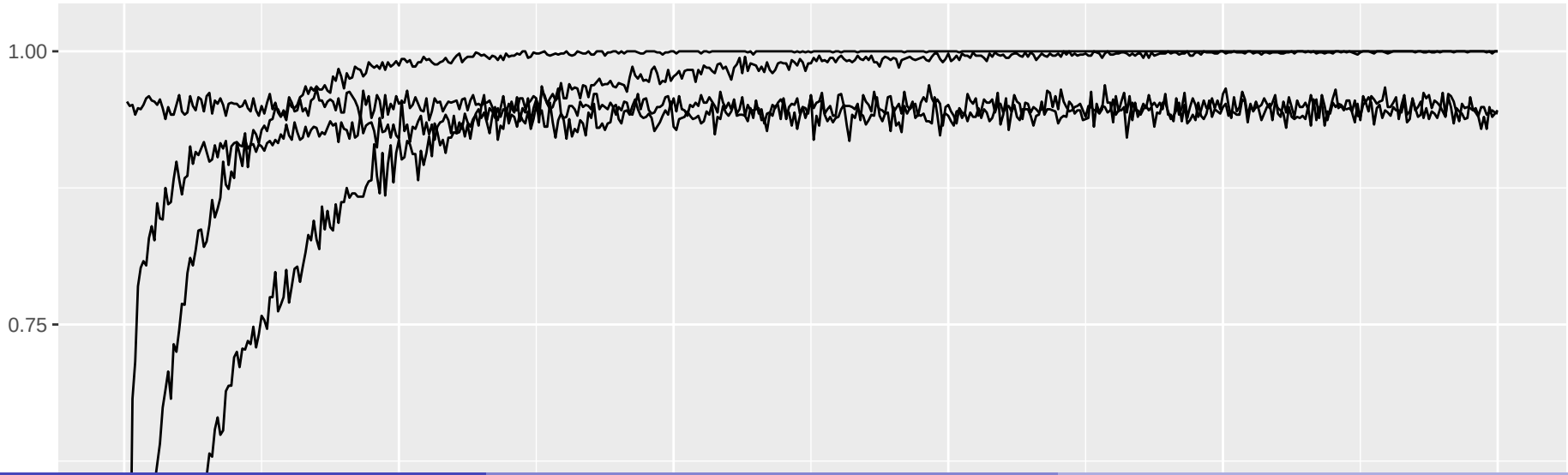
Bootstrap Confidence Interval

```r
bootstrap_ci <- function(N, rate, alpha = 0.05){
  # Function to calculate bootstrap CI
  x <- rexp(N, rate = rate)
  x_bar <- mean(x)
  # Number of bootstrap samples
  nb <- 1000
  # Take boostrap samples
  bootstrap_samples <- sample(x, N * nb, replace = TRUE) %>%
    matrix(nrow = N, ncol = nb)
  # Get means of columns
  means <- colMeans(bootstrap_samples)
  # Get deltas (x* - x)
  deltas <- means - x_bar
  deltas <- sort(deltas)
  # Calculate CIs
  ci <- x_bar - quantile(deltas, probs = c(alpha / 2, 1 - (alp
  return(c(ci[2], ci[1]))
}
```

Other

```
## Parsed with column specification:
## cols(
##   n = col_integer(),
##   bs = col_double(),
##   gamma = col_double(),
##   wald = col_double(),
##   score = col_double()
## )

## Warning: Ignoring unknown parameters: se
```



Summary of Findings and Recommendations

```
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00
```