Week_3

2022-10-30

Q1

Simulate (s,S) inventory system using the calling sequence below.

```r
set.seed(12345)
Y <- MySim(123, 2000, 55555)

### calculate 95% confidence intervals for point estimates
# mean
mean_95 <- c(mean(Y) - qnorm(0.975)*sd(Y)/length(Y),mean(Y) + qnorm(0.975)*sd(Y)/length(Y))
# 80% quantile
l_hat <- floor(length(Y)*0.8 - qnorm(0.975)*sqrt(length(Y)*0.8*(1-0.8)))
u_hat <- ceiling(length(Y)*0.8 + qnorm(0.975)*sqrt(length(Y)*0.8*(1-0.8)))
# prob Y>=210
s_2 <- (length(Y)/(length(Y)-1))*sum(Y>=210)/length(Y)*(1-sum(Y>=210)/length(Y))
prob_95 <- c( sum(Y>=210)/length(Y)- qnorm(0.975)*sqrt(s_2)/length(Y),
              sum(Y>=210)/length(Y)+ qnorm(0.975)*sqrt(s_2)/length(Y))
```

Using 2000 replications, the estimates are:

- mean: 193.4812
- standard deviation: 7.1717139
- 80th percentile of average cost: 199.6
- probability that cost is greater than or equal to $210: 0.0105
- 95% confidence intervals on your point estimates:
  - mean: (193.4741718, 193.4882282)
  - 80th percentile: $[Y_{(1564)}, Y_{(1636)}]$
  - probability that cost is greater than or equal to $210: (0.0104001, 0.0105999)

Q2: MSER statistic in the tandem queue system

(a)

Generate queue 1 and 2 waiting times with c(1,1) servers and probbaility 0.3 of skipping second queue.

Estimate deletion points.

```
mser_q1 <- MSER(Waits$Waits1)
mser_q2 <- MSER(Waits$Waits2)

dEstimated_q1 <- mser_q1$d
dEstimated_q2 <- mser_q2$d
```

The estimate of optimal number of waiting times to delete is 155 for queue 1 and 165 for queue 2.

```
df_q1 <- mser_q1 %>% as_tibble() %>% select(MSER) %>% rowid_to_column()
df_q2 <- mser_q2 %>% as_tibble() %>% select(MSER) %>% rowid_to_column()
```

```
# {plot(1:length(mser_q1$MSER),mser_q1$MSER,type="l")
# abline(v=dEstimated_q1,col="firebrick")
# text(x=320,y=0.1,label="deletion point estimate")}

# use ggplot instead to make nicer
ggplot(df_q1) + geom_line(aes(x=rowid,y=MSER)) +
  geom_vline(xintercept = dEstimated_q1,col="firebrick") +
  annotate("text", x=dEstimated_q1-30, y=0.07, label="deletion point", angle=90, size=3, color="firebric
  theme_minimal()
```

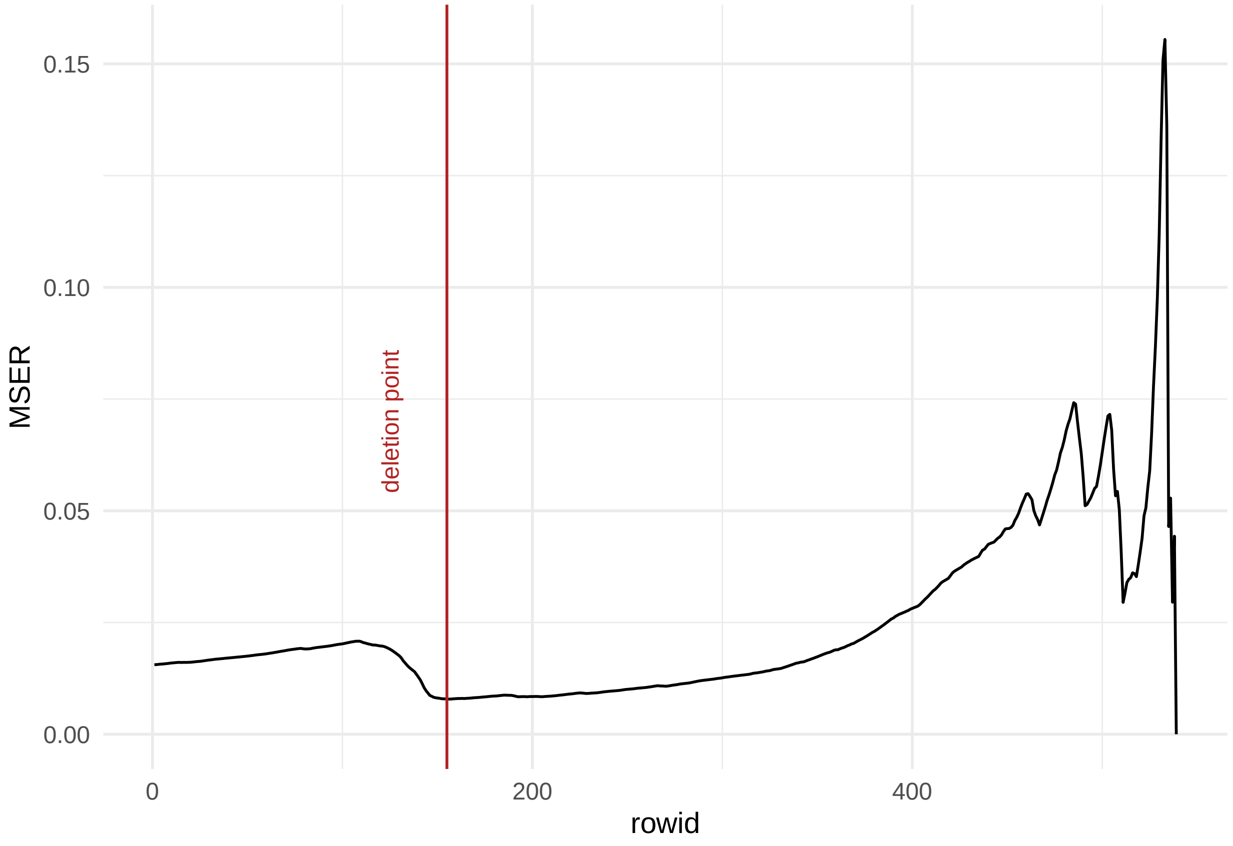

Figure 1: Plot of MSER statistic for queue 1.

```
# use ggplot instead to make nicer
ggplot(df_q2) + geom_line(aes(x=rowid,y=MSER)) + geom_vline(xintercept = dEstimated_q2,col="firebrick")
  annotate("text", x=dEstimated_q2-30, y=0.2, label="deletion point", angle=90, size=3, color="firebric
  theme_minimal()
```

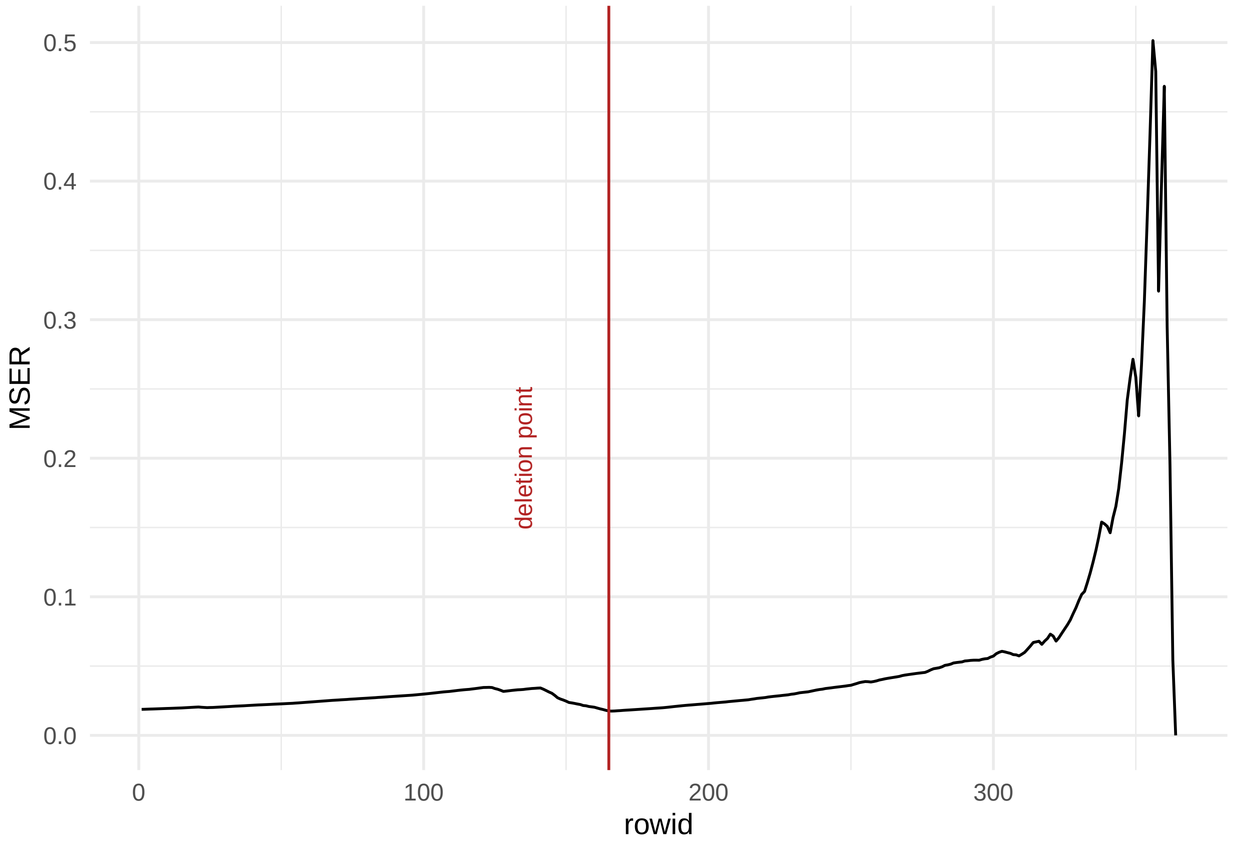

Figure 2: Plot of MSER statistic for queue 2.

(b)

```
# run Tandem with given arguments
Waits <- Tandem(Starting.seed = 12345,c=c(1,1),Tend=5000)
mser_q1 <- MSER(Waits$Waits1)
mser_q2 <- MSER(Waits$Waits2)

dEstimated_q1 <- mser_q1$d
dEstimated_q2 <- mser_q2$d
```

The estimate of optimal number of waiting times to delete is 155 for queue 1 and 165 for queue 2. As there is more information (larger sample), the estimate can be more confident. Since we used the same starting seed, getting same deletion point for sequences of different length is likely (as it so happened in this case of deletion point in local minimum).