

Homework 3

Senan Hogan-H.

22 February 2018

Question 2.

d.

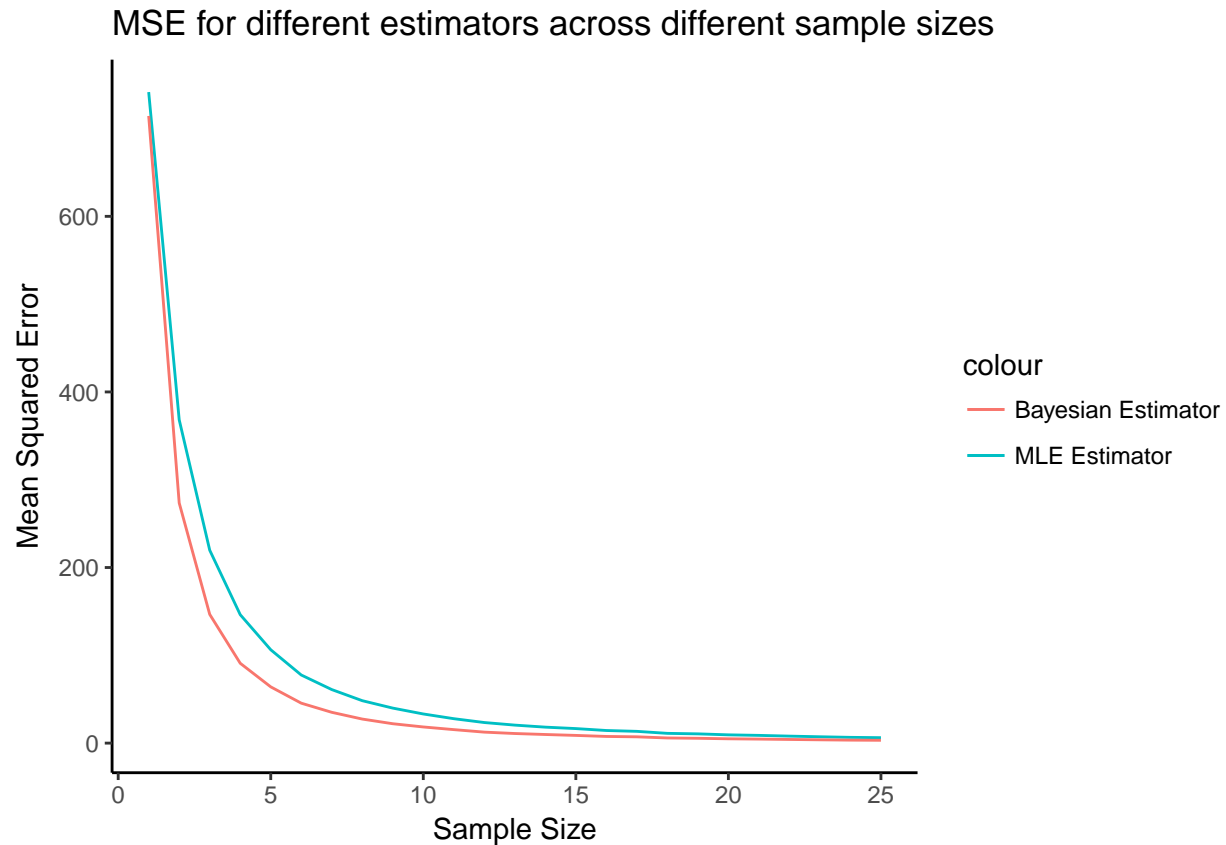
Define $\theta = 47$, with reasonable prior $\text{Pareto}(x_0, \alpha)$ where $x_0 = 4$, $\alpha = 1$.

```
theta <- 47
x_0 <- 4
alpha <- 1

MSE_bayes <- MSE_mle <- n_value <- c()

sample_size <- seq(1,25)
for (n in sample_size){
  estimate_mle <- estimate_bayes <- c()
  for (i in c(0:10000)){
    # Generate X_1 , ... , X_n from uniform(0, theta)
    x <- runif(n, 0, theta)
    estimate_mle <- c(estimate_mle, max(x))
    estimate_bayes <- c(estimate_bayes,
                      (alpha + n)*max(c(x_0, x))/(alpha+n-1))
  }
  MSE_mle <- c(MSE_mle, mean((estimate_mle - theta)^2))
  MSE_bayes <- c(MSE_bayes, mean((estimate_bayes - theta)^2))
  n_value <- c(n_value, n)
}

data.frame(MSE_bayes, MSE_mle, n_value) %>%
  ggplot(aes(x=n_value)) +
  geom_line(aes(y=MSE_mle, colour='MLE Estimator')) +
  geom_line(aes(y=MSE_bayes, colour='Bayesian Estimator')) +
  labs(x= 'Sample Size', y='Mean Squared Error',
       title = 'MSE for different estimators across different sample sizes') +
  theme_classic()
```



The Bayesian estimator requires a smaller sample size (until $n = 25$) for a smaller MSE. In other words, produces a better estimate in smaller sample sizes.

e.

It is possible to select a prior (call it a stupid prior) such that no sample size can produce a better estimate than the MLE for a given distribution. A stupid prior, like one centred around a wrong θ estimate that assigns zero probability to the true value of θ , is such an example.

Question 3.

a.

Aim to estimate p in a binomial distribution. Start with prior distribution of $Beta(\alpha, \beta)$, where $\alpha = \beta = 1$.

```
# Graph for different n values.

# definitions for prior
p <- 0.5
alpha <- beta <- 1

#vectors for final storage
coverage_rate <- n_value <- c()

for (n in seq(1, 100, length.out = 100)){
  coverage <- c()
  for (i in c(1:1000)){
```

```

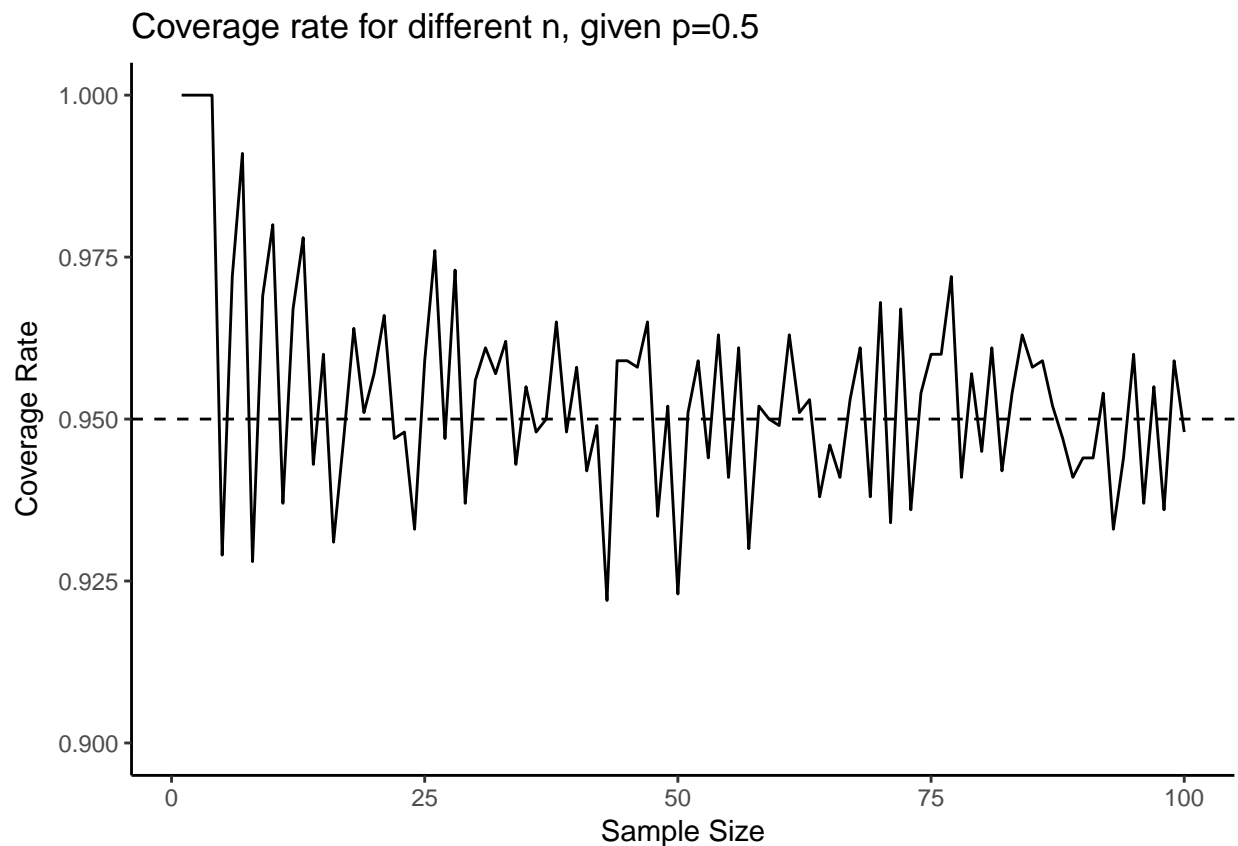
# draw from the binomial distribution
x <- sum(rbinom(n, 1, p))

# form credible interval from posterior distribution.
CI <- qbeta(c(0.025, 0.975), alpha + x, beta + n - x)

# document coverage
ifelse(CI[1] < p & p < CI[2],
       coverage <- c(coverage, 1),
       coverage <- c(coverage, 0))
}
coverage_rate <- c(coverage_rate, mean(coverage))
n_value <- c(n_value, n)
}

data.frame(coverage_rate, n_value) %>%
  ggplot(aes(x=n_value, y=coverage_rate)) +
  geom_hline(yintercept=0.95, linetype='dashed') +
  geom_line() +
  ylim(0.9,1) +
  labs(x= 'Sample Size', y='Coverage Rate',
       title = 'Coverage rate for different n, given p=0.5') +
  theme_classic()

```



```

# Graph for different p values.

```

```

# definitions for prior
n <- 100
alpha <- beta <- 1

#vectors for final storage
coverage_rate <- p_value <- c()

for (p in seq(0, 1, length.out = 1000)){
  coverage <- c()
  for (i in c(1:1000)){
    # draw from the binomial distribution
    x <- sum(rbinom(n, 1, p))

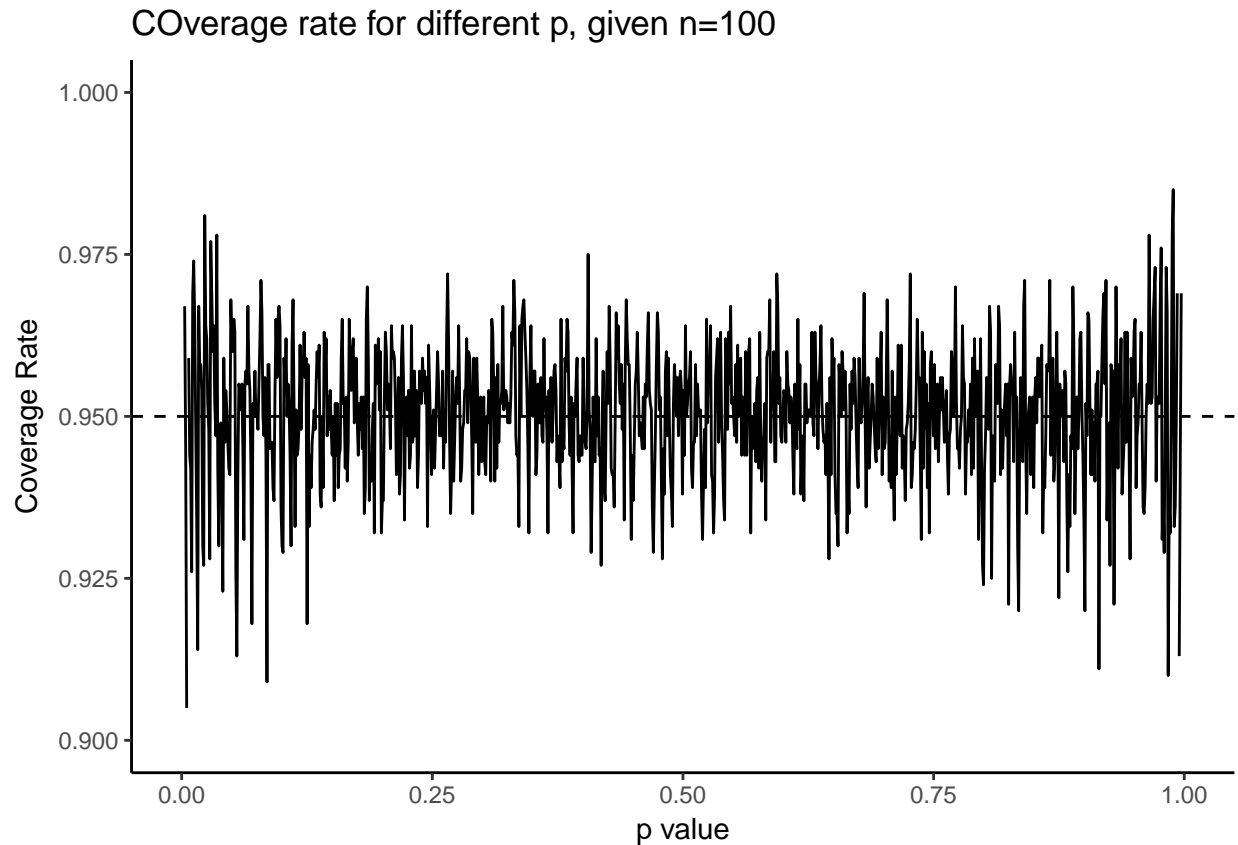
    # form credible interval from posterior distribution.
    CI <- qbeta(c(0.025, 0.975), alpha + x, beta + n - x)

    # document coverage
    ifelse(CI[1] < p & p < CI[2],
           coverage <- c(coverage, 1),
           coverage <- c(coverage, 0))
  }
  coverage_rate <- c(coverage_rate, mean(coverage))
  p_value <- c(p_value, p)
}

data.frame(coverage_rate, p_value) %>%
  ggplot(aes(x=p_value, y=coverage_rate)) +
  geom_hline(yintercept=0.95, linetype='dashed') +
  geom_line() +
  ylim(0.9,1) +
  labs(x= 'p value', y='Coverage Rate',
       title = 'Coverage rate for different p, given n=100') +
  theme_classic()

```

```
## Warning: Removed 4 rows containing missing values (geom_path).
```



The Bayesian credible interval suffers from the same problem to a much lesser extent than the frequentist confidence interval, even from a completely non-informative prior. The main takeaway is that the credible interval spends much less time achieving a coverage lower than stated than does the frequentist approach.

b.

```
# check average coverage
mean(coverage_rate)

## [1] 0.948

# Check difference from stated coverage
abs(mean(coverage_rate)-0.95)

## [1] 0.002036

# proportion below stated coverage
below_coverage <- ifelse(
  coverage_rate < 0.95,
  1,
  0)
mean(below_coverage)

## [1] 0.44
```

The credible interval suffers from the same problem, oscillating around 0.95 (not below).

This is slightly different to the sense in which the confidence interval works, instead referring to coverage probability over a lifetime of experiments of many values of p . This is demonstrated by the graph of many p values for the experiments.