# BDA - Assignment 3

Anonymous

8/3/2020

## Q1 - Inference for normal mean and deviation

Data is describing windshields tested for hardness and some basic statstics about the data can be applied:

```
head(windshieldy1)
```

```
## [1] 13.357 14.928 14.896 15.297 14.820 12.067
```

```
summary(windshieldy1)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   12.07   13.87   14.82   14.61   14.93   17.45
```

```
n = length(windshieldy1) # sample size
var = var(windshieldy1) # sample variance
mu = mean(windshieldy1) # sample mean
sigma = sd(windshieldy1) # standard deviation
n; var; mu; sigma # sample size, variance, mean and sd
```

```
## [1] 9
```

```
## [1] 2.173153
```

```
## [1] 14.61122
```

```
## [1] 1.474162
```

Assumptions:
The observations follow a normal distribution with an unknown standard deviation $\sigma$, and the model for the observations is:

$$p(y) = \mathcal{N}(\mu, \sigma)$$

A noninformative prior distribution, assuming prior independence of location and scale parameters, is uniform on $(\mu, \log \sigma)$ or, equivalently $p(\mu, \sigma) \propto (\sigma^2)^{-1}$

Under this conventional improper prior density, the joint posterior distribution is proportional to the likelihood function multiplied by the factor $1/\sigma^2$:

$$p(\mu, \sigma^2|y) = \sigma^{-n-2} exp(\frac{1}{2\sigma^2}[(n-1)s^2 + n(\bar{y} - \mu)^2])$$

where $s^2 = \frac{1}{n-1} \sum_{i=1}^{n} (y_i - \bar{y})^2$

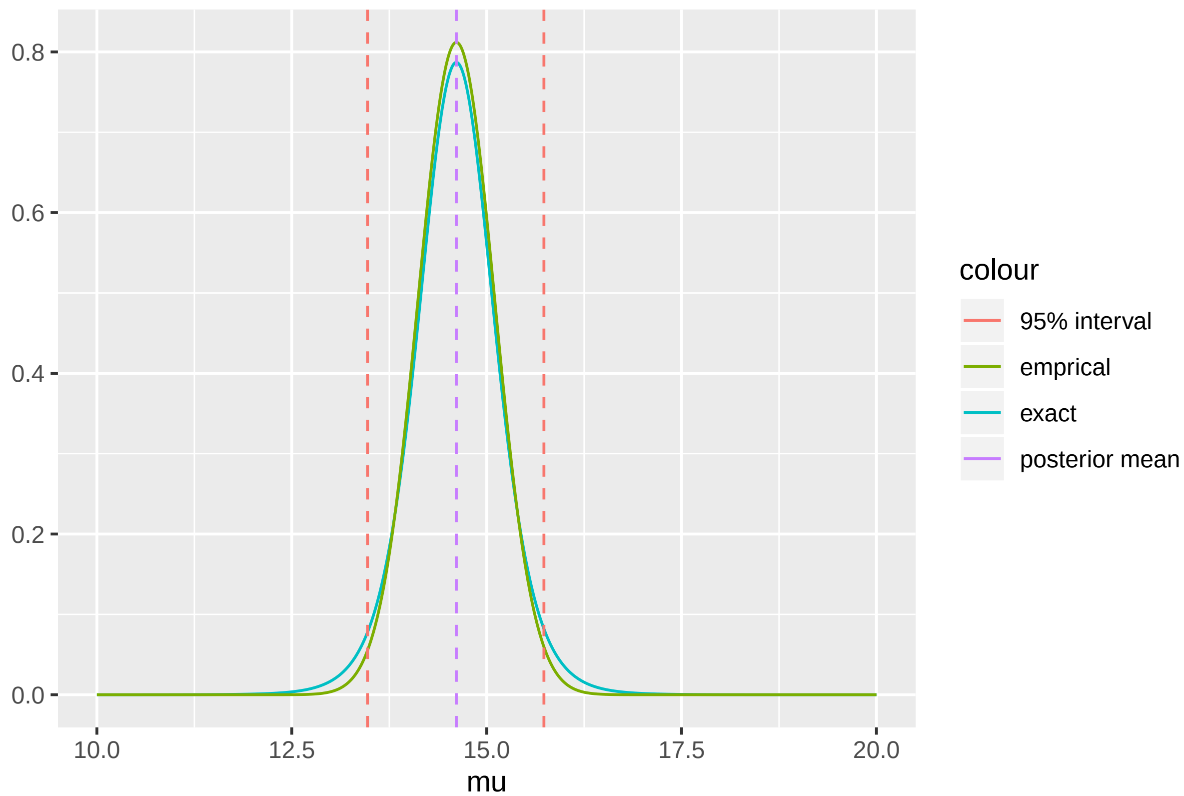
### a)

The unknown $\mu$ describes the average hardness, and we wish to investigate this using a Bayesian point estimate and a 95% posterior interval, including plot of the density:

```r
num_samples <- 100000
x <- seq(10, 20, 0.01)
exact_posterior_mu <- dtnew(x, df=n-1, mean=mu, scale=sigma/sqrt(n))
emprical_posterior_mu <- dnorm(x, mu, sigma/sqrt(n))
data <- windshieldy1

mu_point_est <- function(data){
  n <- length(data)
  mu <- mean(data)
  sigma <- sd(data)
  rtg <- rt(num_samples, df=n-1)
  rr <- (rtg * sigma/sqrt(n) ) + mu
  mu_post <- mean(rr)
  return(mu_post)
}
mu_interval <- function(data, prob){
  n <- length(data)
  mu <- mean(data)
  sigma <- sd(data)
  rtg <- rt(num_samples, df=n-1)
  rr <- (rtg * sigma/sqrt(n) ) + mu
  q <- quantile(rr, c((1-prob)/2, prob+(1-prob)/2), names = FALSE)
  return(q)
}

# plot of the density:
ggplot() +
  geom_line(aes(x, exact_posterior_mu, color='exact')) +
  geom_line(aes(x, emprical_posterior_mu, color='emprical')) +
  geom_vline(aes(xintercept = mu_point_est(data), color = 'posterior mean'),
             linetype = 'dashed', show.legend = F) +
  geom_vline(aes(xintercept = c(mu_interval(data, prob = 0.95)), color = '95% interval'),
             linetype = 'dashed', show.legend = F) +
  labs(title = '', x = 'mu', y = '')
```




```r
mu_point_est(data); mu_interval(data, prob = 0.95) # mean of mu and posterior interval
```

```
## [1] 14.61251
```

```
## [1] 13.47136 15.74238
```

We find:
Posterior mean expected value = 14.61%
95% Posterior Mean interval = 13.49 - 15.75

**b)**

We wish to investigate the hardness of the next windshield coming from the production - before it's hardness are measured, using Bayesian point, intervals and plot of the density.
To draw from the posterior predictive distribution, we first draw $(\mu, \sigma^2)$ from the joint posterior distribution and then simulate $\tilde{y} \propto \mathcal{N}(\mu, \sigma^2)$. Posterior predictive distribution based on integrating $(\mu, \sigma^2)$:

$$p(\tilde{y}|\sigma^2, y) = \int p(\tilde{y}|\mu, \sigma^2, y)p(\mu|\sigma^2, y)d\mu = \mathcal{N}(\tilde{y}|\bar{y}, (1 + \frac{1}{n})\sigma^2)$$

Analytical form of posterior predictive distribution:

$$p(\tilde{y}|\sigma^2, y) = t_{n-1}(\bar{y}, (1 + \frac{1}{n})s^2)$$

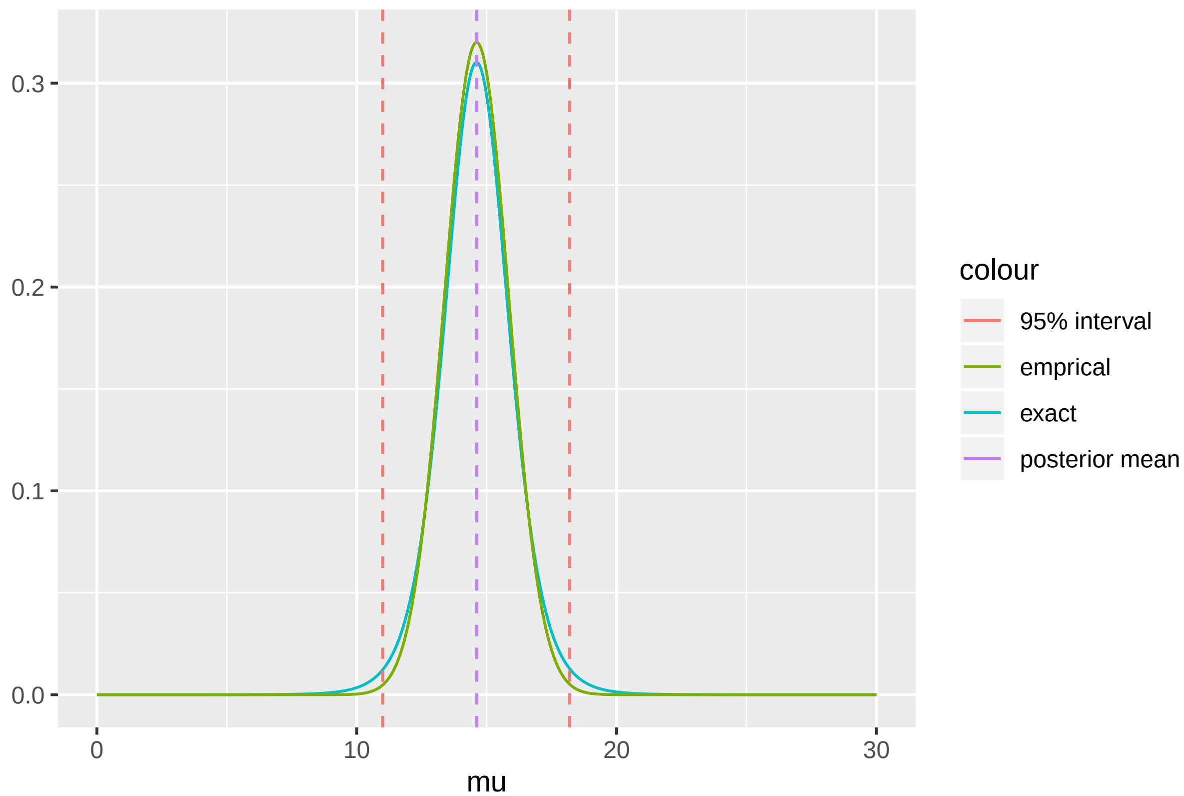
```r
x <- seq(0, 30, 0.01)
exact_posterior_pred <- dtnew(x, df=n-1, mean=mu, scale=sqrt(sigma*sqrt(1+1/n)))
emprical_posterior_pred <- dnorm(x, mu, sqrt(sigma*sqrt(1+1/n)))
mu_pred_point_est <- function(data){
  n <- length(data)
  mu <- mean(data)
  sigma <- sd(data)
  rtg <- rt(num_samples, df=n-1)
  rr <- (rtg * sigma*sqrt(1+(1/n)) ) + mu
  mu_post <- mean(rr)
  return(mu_post)
}
mu_pred_interval <- function(data, prob = 0.95){
  n <- length(data)
  mu <- mean(data)
  sigma <- sd(data)
  rtg <- rt(num_samples, df=n-1)
  rr <- (rtg * sigma*sqrt(1+(1/n)) ) + mu
  q <- quantile(rr, c((1-prob)/2, prob+(1-prob)/2), names = FALSE)
  return(q)
}
mu_pred_point_est(data) # mean posterior predictor
```

```
## [1] 14.61323
```

```r
mu_pred_interval(data, prob = 0.95) # 95 % interval of posterior predictive
```

```
## [1] 11.03065 18.21109
```

```r
ggplot() +
  geom_line(aes(x, exact_posterior_pred, color='exact')) +
  geom_line(aes(x, emprical_posterior_pred, color='emprical')) +
  geom_vline(aes(xintercept = mu_pred_point_est(data), color = 'posterior mean'),
             linetype = 'dashed', show.legend = F) +
  geom_vline(aes(xintercept = c(mu_pred_interval(data, prob = 0.95)), color = '95% interval'),
             linetype = 'dashed', show.legend = F) +
labs(title = '', x = 'mu', y = '')
```




### Q 2 - Inference for the difference between proportions

The observational model:

$$p(y_0, y_1) \propto p_0^{y_0}(1-p_0)^{n_0-y_0} p_1^{y_1}(1-p_1)^{n_1-y_1}$$

We use independent Beta distribution as priors:

$$p(p_i) = Beta(\alpha_i, \beta_i)$$

So posterior distributions are independent:

$$p(p_i|y_i) = Beta(y_i + \alpha_i, n_i - y_i + \beta_i)$$

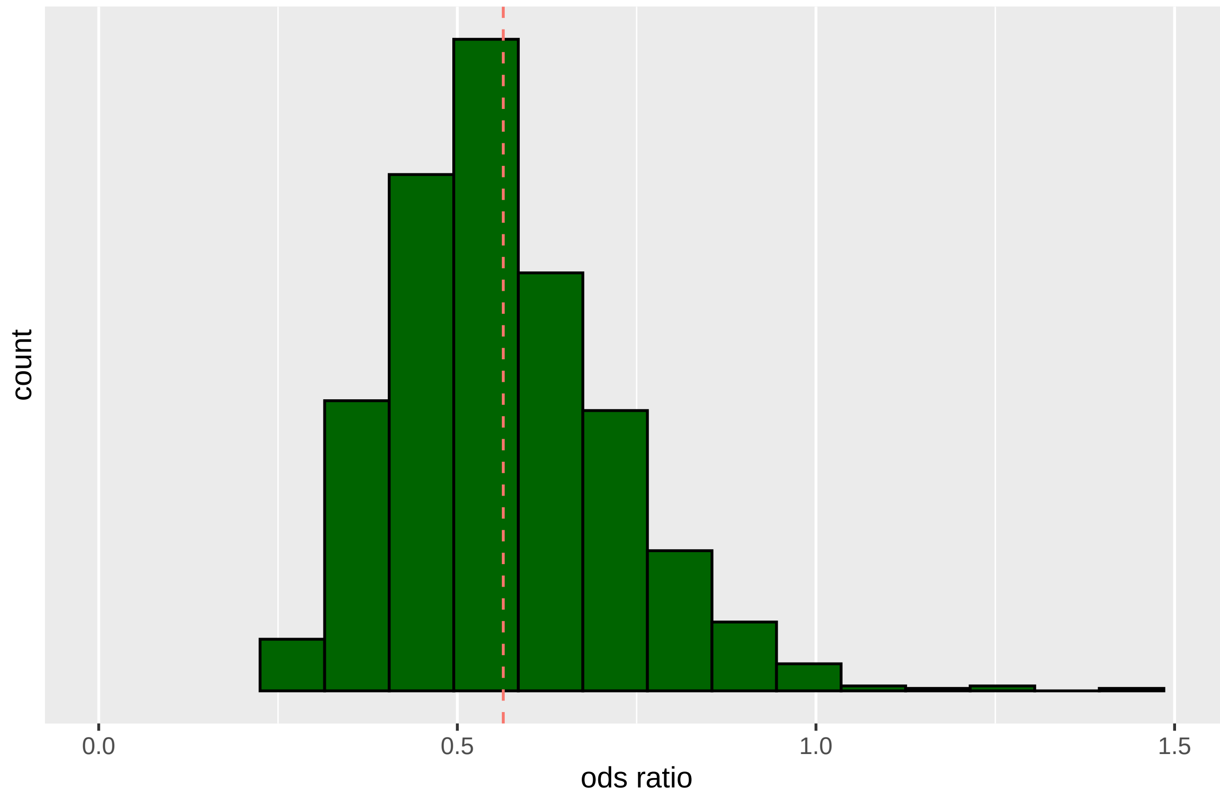
where $i \in 0, 1, n_0 = 674, n_1 = 680, y_0 = 39, y_1 = 22$

$$p(p_0, p_1|y_0, y_1) \propto p(p_0|y_0)p(p_1|y_1) \propto Beta(\alpha_0, \beta_0)Beta(\alpha_1, \beta_1) \propto Beta(\alpha_0 + \alpha_1, \beta_0 + \beta_1)$$

## a) Summarizing the posterior distribution for the odds ratio and computing the point and interval estimatas, including a plot:

We have odds ratio as $\psi = \frac{p_1/(1-p_1)}{p_0/(1-p_0)}$. For computing the posterior of odds ratio we use sampling from $p(p_i|yi_)$ and then simulate $\psi$ based on $\psi = \frac{p_1/(1-p_1)}{p_0/(1-p_0)}$.

```r
# given from the text
n0 <- 674
y0 <- 39
n1 <- 680
y1 <- 22
a0 <- 1
b0 <- 1
a1 <- 1
b1 <- 1
post_alpha0 <- a0 + y0
post_beta0 <- b0 + n0 - y0
post_dist0 <- rbeta(1000, post_alpha0, post_beta0)
post_alpha1 <- a1 + y1
post_beta1 <- b1 + n1 - y1
post_dist1 <- rbeta(1000, post_alpha1, post_beta1)
posterior_odds_ratio_point_est <- function(p0, p1){
  psi <- (p1/(1-p1))/(p0/(1-p0))
  return(mean(psi))
}
posterior_odds_ratio_interval <- function(p0, p1, prob = 0.9){
  psi <- (p1/(1-p1))/(p0/(1-p0))
  q <- c(quantile(psi, (1-prob)/2), quantile(psi, prob+(1-prob)/2))
  return(q)
}
odds_ratio <- (post_dist1/(1-post_dist1))/(post_dist0/(1-post_dist0))
ggplot() +
  geom_histogram(aes(odds_ratio), binwidth = 0.09, fill = 'darkgreen', color = 'black') +
  coord_cartesian(xlim = c(0, 1.5)) +
  scale_y_continuous(breaks = NULL) +
  labs(title = '', x = 'ods ratio')+
  geom_vline(aes(xintercept = mean(odds_ratio), color = 'q'),
             linetype = 'dashed', show.legend = F)
```




```
posterior_odds_ratio_point_est(post_dist0, post_dist1)
```

```
## [1] 0.5640348
```

```
posterior_odds_ratio_interval(post_dist0, post_dist1, prob = 0.95)
```

```
##      2.5%     97.5%
## 0.3237415 0.9115434
```

Point estimate = 0.5719
The 95% interval = 0.3232 to 0.9281

**b)**

Discussion of the sensitivity:
The posterior is not sensitive to the prior, since the posterior is not close to the prior

```
A0 <- c(1, 2, 0.5, 5)
B0 <- c(1, 10, 10 ,100)
A1 <- c(1, 2, 0.4, 4)
B1 <- c(1, 10, 10, 100)
post_mean = c()
post_int = matrix(rep(0,2*length(A0)), ncol=2)
```

```r
for(i in 1:length(A0)){
  a0 <- A0[i]
  b0 <- B0[i]
  a1 <- A1[i]
  b1 <- B1[i]

  post_alpha0 <- a0 + y0
  post_beta0 <- b0 + n0 - y0
  prior_dist0 <- rbeta(1000, a0, b0)
  post_dist0 <- rbeta(1000, post_alpha0, post_beta0)

  post_alpha1 <- a1 + y1
  post_beta1 <- b1 + n1 - y1
  prior_dist1 <- rbeta(1000, a1, b1)
  post_dist1 <- rbeta(1000, post_alpha1, post_beta1)

  prior_dist <- rbeta(1000, a0+a1, b0+b1)
  post_mean[i] <- posterior_odds_ratio_point_est(post_dist0, post_dist1)
  post_int[i, ] <- posterior_odds_ratio_interval(post_dist0, post_dist1, prob = 0.95)


}
post_mean
```

```
## [1] 0.5722603 0.5787384 0.5587729 0.5931242
```

```r
post_int
```

```
##           [,1]      [,2]
## [1,] 0.3202242 0.9277851
## [2,] 0.3297420 0.9295484
## [3,] 0.3092586 0.9071357
## [4,] 0.3399453 0.9548902
```

| Parameters of the prior distribution | | Summaries of the posterior distribution | |
|---|---|---|---|
| $\frac{\alpha_0+\alpha_1}{\alpha_0+\beta_0+\alpha_1+\beta_1}$ | $\alpha_0+\beta_0+\alpha_1+\beta_1$ | mean of $\psi$ | 95% posterior interval for $\pi$ |
| 0.5 | 4 | 0.5706 | [0.3137, 0.9642] |
| 0.1667 | 24 | 0.5849 | [0.3311, 0.9221] |
| 0.0431 | 20.9 | 0.5664 | [0.3155, 0.9026] |
| 0.0431 | 209 | 0.5956 | [0.3369, 0.9474] |

## Q3 - Inference for the difference between normal means

### a)

Ivestigating $\mu_d$ qith Bayesian point estimate, 95% posterior interval and a histogram:

Uninformative joint prior: $p(\mu, \sigma_2) \propto \frac{1}{\sigma_2^2}$ likelihood: $p(y_2|\mu, \sigma_2) = \mathcal{N}(\mu, \sigma_2)$ Marginal posterior for $\mu$: $p(\mu|y_2) = t_{n-1}(y_2, \frac{s^2}{n})$

$\mu_d$ will be calculated by sampling from $\mu_1$ and $\mu_2$ and then calculating $\mu_1 - \mu_2$.

```r
data("windshieldy1")
data("windshieldy2")
post_mean <- function(data){
  n <- length(data)
  mu <- mean(data)
  sigma <- sd(data)
  rtg <- rt(num_samples, df=n-1)
  rr <- (rtg * sigma/sqrt(n) ) + mu
  return(rr)
}

data1 <- windshieldy1
data2 <- windshieldy2
n2 <- length(data2)
mu_difference <- post_mean(data1) - post_mean(data2)
posterior_mean <- mean(mu_difference)
cat("mean \n")
```

```
## mean
```

```r
posterior_mean
```

```
## [1] -1.208907
```

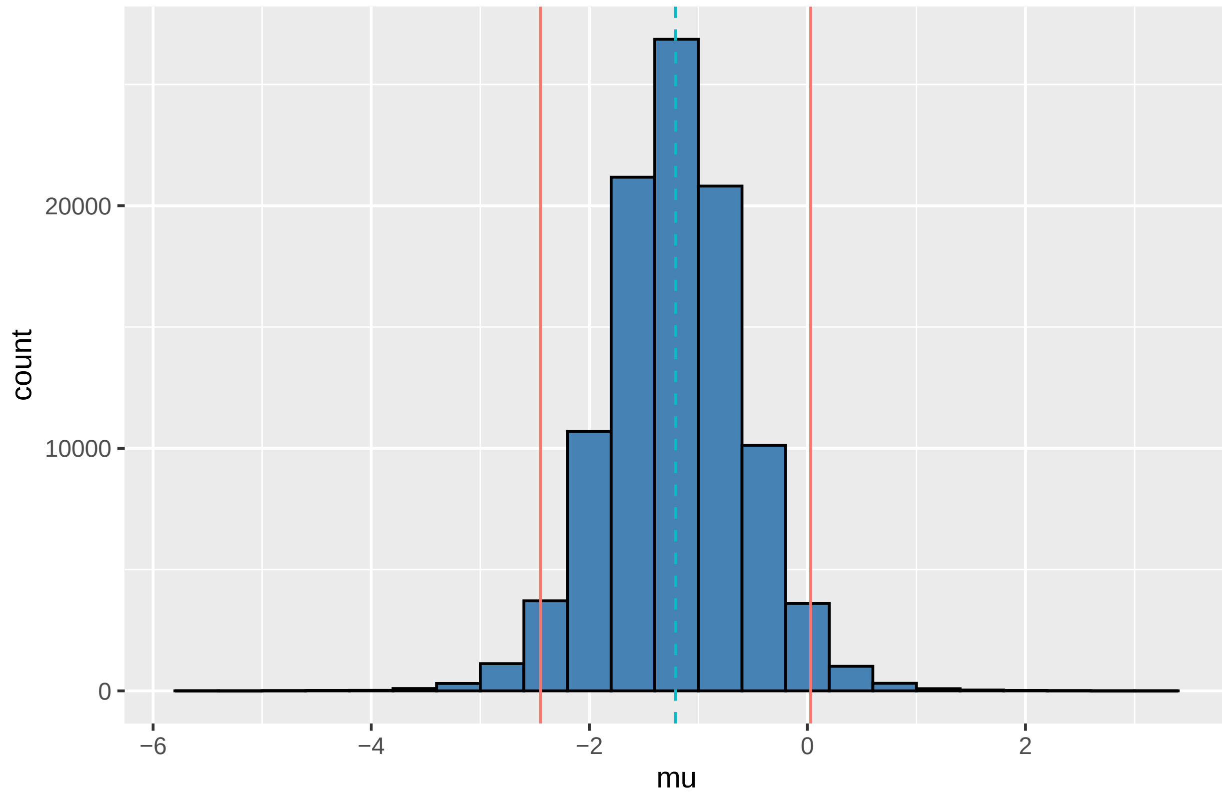
```r
prob <- 0.95
posterior_interval<- quantile(mu_difference, c((1-prob)/2, prob+(1-prob)/2), names = FALSE)
cat("\n interval estimates (95%) \n")
```

```
##
##  interval estimates (95%)
```

```r
posterior_interval
```

```
## [1] -2.446833  0.029522
```

```r
labs <- c('posterior mean')
ggplot() +
  geom_histogram(aes(mu_difference), binwidth = 0.4, fill = 'steelblue', color = 'black') +
  labs(title = '', x = 'mu')+
  geom_vline(aes(xintercept = posterior_mean, color = 'q'),
             linetype = 'dashed', show.legend = F) +
  geom_vline(aes(xintercept = c(posterior_interval), color = '95% interval'),
             linetype = 'solid', show.legend = F)
```




```
p_mu2 = sum(mu_difference<0)/num_samples
p_mu2
```

```
## [1] 0.97269
```

=97% probability that $\mu_2$ is bigger than $\mu_1$.

**b)**

Missing