Assignment 3

anonymous

1 General information

2 Inference for normal mean and deviation (3 points)

Loading the library and the data.

```
data("windshieldy1")
# The data are now stored in the variable `windshieldy1`.
# The below displays the data:
windshieldy1
```

```
[1] 13.357 14.928 14.896 15.297 14.820 12.067 14.824 13.865 17.447
```

The below data is **only for the tests**, you need to change to the full data `windshieldy1` when reporting your results.

```
windshieldy_test <- c(13.357, 14.928, 14.896, 14.820)
```

2.1 (a)

$$Likelihood : p(y|\mu, \sigma^2) \propto \sigma^{-n} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - \mu)^2\right)$$

$$prior : p(\mu, \sigma^2) \propto \frac{1}{\sigma^2}$$

$$posterior : p(\mu, \sigma^2|y) \propto \sigma^{-n-2} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - \mu)^2\right)$$

2.2 (b)

Under non informative prior the marginal posterior for normalized μ has distribution ~t_{n_1}
The unknown μ lies with 95% confidence between 13.47808 15.74436, and the mean of it's distribution is 14.61122

Keep the below name and format for the functions to work with `markmyassignment`:

```
# Useful functions: mean(), length(), sqrt(), sum()
# and qtnew(), dtnew() (from aaltobda)


mu_point_est <- function(data) {
    # Do computation here, and return as below.
    # This is the correct return value for the test data provided above.
    y_mean = mean(data)
    n = length(data)
    sd = sqrt(sum((data - y_mean)**2)/(n-1))

    mean_estimate= qtnew(0.5, df=n-1, mean=y_mean, scale=(sd/sqrt(n)))
    mean_estimate

}
mu_interval <- function(data, prob = 0.95) {
    # Do computation here, and return as below.
    # This is the correct return value for the test data provided above.
    y_mean = mean(data)
    n = length(data)
    sd = sqrt(sum((data - y_mean)**2)/(n-1))

    low_estimate= qtnew((1-prob)/2, df=n-1, y_mean, (sd/sqrt(n)))

    high_estimate= qtnew(1-(1-prob)/2, df=n-1,  y_mean, (sd/sqrt(n)))
    c(low_estimate, high_estimate)
    # c(13.3, 15.7)

}
# mu_point_est(windshieldy1)
# mu_interval(windshieldy1)
```

You can plot the density as below if you implement `mu_pdf` to compute the PDF of the posterior $p(\mu|y)$ of the average hardness μ.

```r
mu_pdf <- function(data, x){
    # Compute necessary parameters here.
    # These are the correct parameters for `windshieldy_test`
    # with the provided uninformative prior.
    location = mean(data)
    df = length(data)-1
    scale = sqrt(sum((data - location)**2)/df)
    # df = 3
    # location = 14.5
    # scale = 0.3817557
    # Use the computed parameters as below to compute the PDF:

    dtnew(x, df, location, scale)
}

x_interval = mu_interval(windshieldy1, .999)
lower_x = x_interval[1]
upper_x = x_interval[2]
x = seq(lower_x, upper_x, length.out=1000)
plot(
    x, mu_pdf(windshieldy1, x), type="l",
    xlab=TeX(r'(average hardness $\mu$)'),
    ylab=TeX(r'(PDF of the posterior $p(\mu|y)$)')
)
```

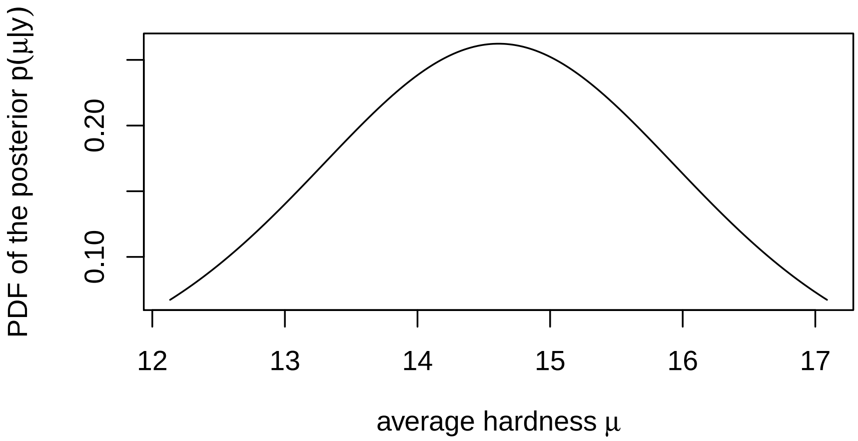


Figure 1: PDF of the posterior $p(\mu|y)$ of the average hardness μ

2.3 (c)

The posterior predictive distribution for a future observation given the uninformative prior and the likelihood following a normal distribution is distributed as $\tilde{y} \sim t_{n-1}$with location $= \bar{y}$ and scale $= (1 + \frac{1}{n}^{1/2}) * s$

Keep the below name and format for the functions to work with `markmyassignment`:

```
# Useful functions: mean(), length(), sqrt(), sum()
# and qtnew(), dtnew() (from aaltobda)

mu_pred_point_est <- function(data) {
    # Do computation here, and return as below.
    # This is the correct return value for the test data provided above.
    y_mean = mean(data)
    n = length(data)
    sd = sqrt(sum((data - y_mean)**2)/(n-1))

    mean_estimate= qtnew(0.5, df=n-1, mean=y_mean, scale=sd*(1+1/n)^(1/2))
    mean_estimate
    # 14.5
```

```
}
mu_pred_interval <- function(data, prob = 0.95) {
    # Do computation here, and return as below.
    # This is the correct return value for the test data provided above.
    y_mean = mean(data)
    n = length(data)
    sd = sqrt(sum((data - y_mean)**2)/(n-1))

    low_estimate= qtnew((1-prob)/2, df=n-1, mean=y_mean,scale=sd*(1+1/n)^(1/2))

    high_estimate= qtnew(1-(1-prob)/2, df=n-1,  mean=y_mean, scale=sd*(1+1/n)^(1/2))
    c(low_estimate, high_estimate)
    # c(11.8, 17.2)

}

mu_pred_point_est(windshieldy_test)
```

```
[1] 14.50025
```

```
mu_pred_interval(windshieldy_test)
```

```
[1] 11.78361 17.21689
```

You can plot the density as below if you implement `mu_pred_pdf` to compute the PDF of the posterior predictive $p(\tilde{y}|y)$ of a new hardness observation $\tilde{y}$.

```
mu_pred_pdf <- function(data, x){
    # Compute necessary parameters here.
    # These are the correct parameters for `windshieldy_test`
    # with the provided uninformative prior.
    df = 4
    location = 14.5
    scale = 1.553903
    # Use the computed parameters as below to compute the PDF:

    dtnew(x, df, location, scale)
}

x_interval = mu_pred_interval(windshieldy1, .999)
lower_x = x_interval[1]
upper_x = x_interval[2]
```

```
x = seq(lower_x, upper_x, length.out=1000)
plot(
    x, mu_pred_pdf(windshieldy1, x), type="l",
    xlab=TeX(r'(new hardness observation $\tilde{y}$)'),
    ylab=TeX(r'(PDF of the posterior predictive $p(\tilde{y}|y)$)')
)
```

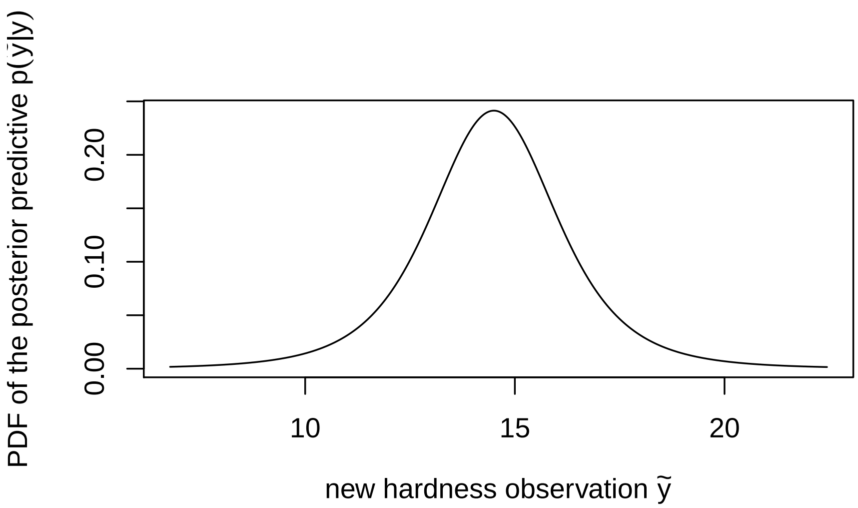


Figure 2: PDF of the posterior predictive $p(\tilde{y}|y)$ of a new hardness observation $\tilde{y}$

3 Inference for the difference between proportions (3 points)

3.1 (a)

Using independent distributions.
P0

$$\text{likelihood:}p(y|p_0) \propto p_0^{39} * (1-p_0)^{635}$$
$$\text{prior:}p(p_0) \propto p_0^{\alpha-1} * (1-p_0)^{\beta-1}$$
$$\text{posterior:}p(p_0|y) \propto Beta(\alpha+39, \beta+674-39)$$

P1

$$\text{likelihood:}p(y|p_1) \propto p_1^{22} * (1-p_1)^{658}$$

$$\text{prior:} p(p_1) \propto p_1^{\alpha-1} * (1-p_1)^{\beta-1}$$
$$\text{posterior:} p(p_1|y) \propto Beta(\alpha + 22, \beta + 680 - 22)$$

3.2 (b)

Assuming uninformative uniform prior distributions for p0 and p1 (control and treatment) we find that that the mean odds ratio is 0.57 and that the 0.95 credible interval is [0.32, 0.92]. The probability is 0.95 that the true treatment effect is in the interval [0.32, 0.92]. This means that with high probability the patients in treatment group are less likely to die.

The below data is **only for the tests**:

```
set.seed(4711)
no_samples = 1000
p0 = rbeta(no_samples, 5, 95)
p1 = rbeta(no_samples, 10, 90)
```

Keep the below name and format for the functions to work with `markmyassignment`:

```
# Actual data
set.seed(4711)
no_samples = 1000
p0_data = rbeta(no_samples, 40, 636)
p1_data = rbeta(no_samples, 23, 659 )
```

```
# Useful function: mean(), quantile()

posterior_odds_ratio_point_est <- function(p0, p1) {
    odds_ratio = (p1 / (1-p1)) / (p0 / (1-p0))
    mean_odds_ratio = mean(odds_ratio)
    mean_odds_ratio
    # Do computation here, and return as below.
    # This is the correct return value for the test data provided above.
    # 2.650172

}
posterior_odds_ratio_interval <- function(p0, p1, prob = 0.95) {
    odds_ratio = (p1 / (1-p1)) / (p0 / (1-p0))

    left_tail = (1-prob)/2
    right_tail =  1 - left_tail
```

```r
    c(quantile(odds_ratio, probs=c(left_tail, right_tail)))
    # Do computation here, and return as below.
    # This is the correct return value for the test data provided above.
    # c(0.6796942,7.3015964)

}

posterior_odds_ratio_point_est(p0_data, p1_data) # 0.57
```

```
[1] 0.5710218
```

```r
posterior_odds_ratio_interval(p0_data, p1_data) # 0.3221829 0.9220926
```

```
     2.5%     97.5%
0.3221829 0.9220926
```

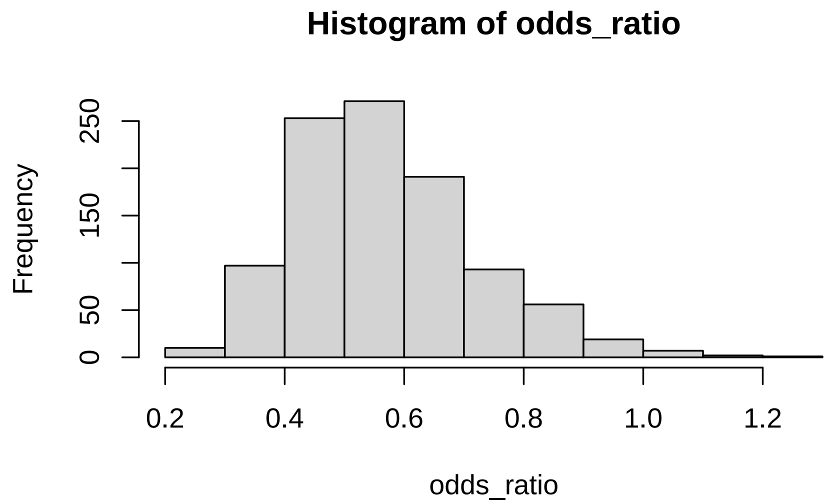
```r
library(dplyr)
```

```
Warning: package 'dplyr' was built under R version 4.0.5
```

```r
library(tidyr)
```

```
Warning: package 'tidyr' was built under R version 4.0.5
```

```r
odds_ratio = (p1_data / (1-p1_data)) / (p0_data / (1-p0_data))
hist(odds_ratio)
```




```
# data = data.frame(group = c(replicate(no_samples, "control"), replicate(no_samples, "tr
# data = data %>% mutate(parameter = c(p0_data, p1_data))
#
# ggplot(data= data) +
#   geom_histogram(aes(parameter, color=group), fill="white", alpha=0.2, position='identi
```

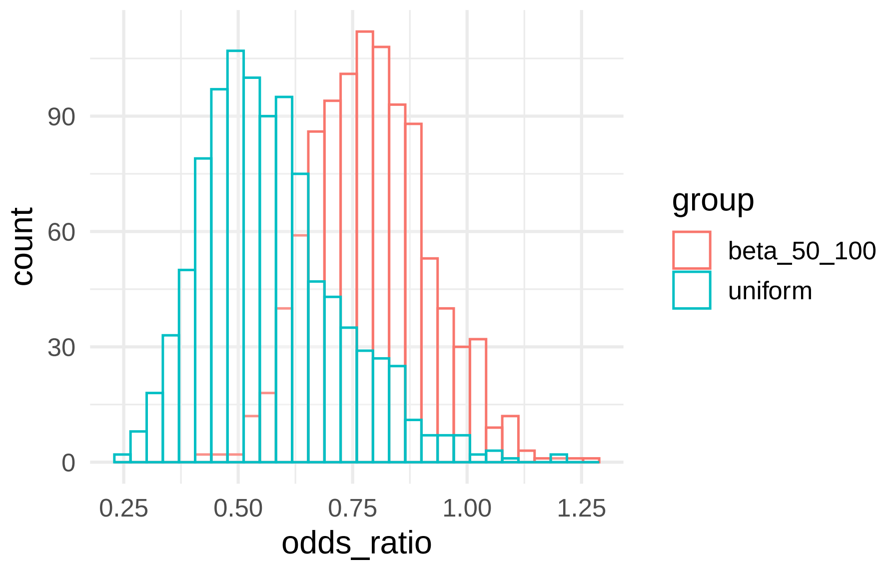
3.3 (c)

Comparing the uniform prior with a beta distribution prior with parameters 50/100 (roughly 50% of deaths in each group) we see that the posterior odds ratio are shifted towards the 1, meaning that data shows a probable effect of treatment but observations are not enough to totally dominate the prior and have an effect as strong as with the uniform prior.

```
# Uniform prior
set.seed(4711)
no_samples = 1000
p0_data = rbeta(no_samples, 40, 636)
p1_data = rbeta(no_samples, 23, 659 )
odds_ratio_uniform = (p1_data / (1-p1_data)) / (p0_data / (1-p0_data))

# 50/100
p0_data_2 = rbeta(no_samples, 89, 685)
```

```
p1_data_2 = rbeta(no_samples, 72, 708 )
odds_ratio_50_100 = (p1_data_2 / (1-p1_data_2)) / (p0_data_2 / (1-p0_data_2))




dataplot = data.frame(group = c(replicate(no_samples, "uniform"), replicate(no_samples, "
dataplot = dataplot %>% mutate(odds_ratio = c(odds_ratio_uniform,odds_ratio_50_100)
                       )
ggplot(data=dataplot) +
  geom_histogram(aes(x=odds_ratio, colour=group),  fill="white", alpha=0.2, position='ide
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
# data = data %>% pivot_longer(!x, names_to = "odds_ratio", values_to = "density")
```

4 Inference for the difference between normal means (3 points)

Loading the library and the data.

```r
data("windshieldy2")
# The new data are now stored in the variable `windshieldy2`.
# The below displays the first few rows of the new data:
head(windshieldy2)
```

```
[1] 15.980 14.206 16.011 17.250 15.993 15.722
```

4.1 (a)

windshieldy1

$$Likelihood : p(y|\mu_1, \sigma_1^2) \propto {\sigma_1}^{-n_1} \exp{(-\frac{1}{2\sigma_1^2}\sum_{i=1}^{n}(y_{1i} - \mu_1)^2)}$$

$$prior : p(\mu_1, \sigma_1^2) \propto \frac{1}{\sigma_1^2}$$

$$posterior : p(\mu_1, \sigma_1^2|y_1) \propto \sigma_1^{-n_1-2} \exp{(-\frac{1}{2\sigma_1^2}\sum_{i=1}^{n}(y_{1i} - \mu_1)^2)}$$

Same for windshieldy2 but with it's corresponding subscript.

4.2 (b)

Write your answers and code here!

```r
mu_diff_point_est <- function(data1, data2) {
    # Do computation here, and return as below.
    # This is the correct return value for the test data provided above.
    y1_mean = mean(data1)
    n_1 = length(data1)
    sd_1 = sqrt(sum((data1 - y1_mean)**2)/(n_1-1))
    posterior1 = rtnew(1000, df=n_1-1, mean=y1_mean, scale=(sd_1/sqrt(n_1)))

    y2_mean = mean(data2)
    n_2 = length(data2)
    sd_2 = sqrt(sum((data2 - y2_mean)**2)/(n_2-1))
    posterior2 = rtnew(1000, df=n_2-1, mean=y2_mean, scale=(sd_2/sqrt(n_2)))

    mu_diff = posterior1 - posterior2
    mean(mu_diff)
}
```

```r
mu_diff_interval <- function(data1, data2, prob=0.95) {
    # Do computation here, and return as below.
    # This is the correct return value for the test data provided above.
    y1_mean = mean(data1)
    n_1 = length(data1)
    sd_1 = sqrt(sum((data1 - y1_mean)**2)/(n_1-1))
    posterior1 = rtnew(1000, df=n_1-1, mean=y1_mean, scale=(sd_1/sqrt(n_1)))

    y2_mean = mean(data2)
    n_2 = length(data2)
    sd_2 = sqrt(sum((data2 - y2_mean)**2)/(n_2-1))
    posterior2 = rtnew(1000, df=n_2-1, mean=y2_mean, scale=(sd_2/sqrt(n_2)))

    mu_diff = posterior1 - posterior2

    low = (1-prob)/2
    high = 1 -low
    c(quantile(mu_diff, c(low, high)))
}

plot_difference <- function(data1, data2){
    y1_mean = mean(data1)
    n_1 = length(data1)
    sd_1 = sqrt(sum((data1 - y1_mean)**2)/(n_1-1))
    posterior1 = rtnew(1000, df=n_1-1, mean=y1_mean, scale=(sd_1/sqrt(n_1)))

    y2_mean = mean(data2)
    n_2 = length(data2)
    sd_2 = sqrt(sum((data2 - y2_mean)**2)/(n_2-1))
    posterior2 = rtnew(1000, df=n_2-1, mean=y2_mean, scale=(sd_2/sqrt(n_2)))

    mu_diff = posterior1 - posterior2
    hist(mu_diff)
}

# Useful functions: mean(), length(), sqrt(), sum(),
# rtnew() (from aaltobda), quantile() and hist().
```

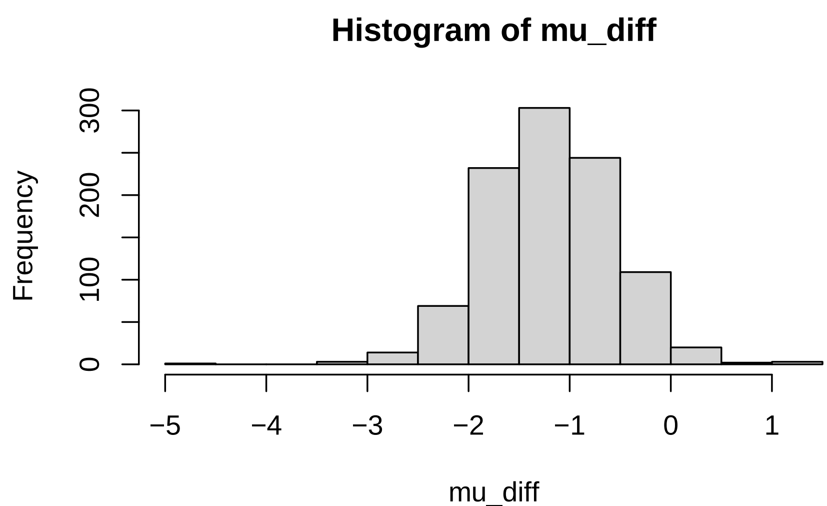
```r
mu_diff_point_est(windshieldy1, windshieldy2)
```

```
[1] -1.21092
```

```r
mu_diff_interval(windshieldy1, windshieldy2)
```

```
      2.5%      97.5% 
-2.4306047  0.1050471 
```

```
plot_difference(windshieldy1, windshieldy2)
```



Assuming uninformative uniform prior distributions for μ and σ^2 of both groups we find that that the mean difference in mu is -1.23 and that the 0.95 credible interval is -2.345138407 -0.005582468. The probability is 0.95 that the true mean difference is in the interval -2.345138407 -0.005582468. This means that with high probability the hardness is the first production line is lower than in the second one. ## (c)

Since the hardness is a continuous variable I think it's not possible to determine if the means are exactly the same, we could get a density for the mean difference = 0 but that's it.

What we can do is to calculate the probability of a small interval containing 0. The probability the mu difference is between -0.05 and 0.05 is around 0.7%

```
mu_zero_interval <- function(data1, data2, sides=0.05) {
    # Do computation here, and return as below.
    # This is the correct return value for the test data provided above.
    y1_mean = mean(data1)
    n_1 = length(data1)
    sd_1 = sqrt(sum((data1 - y1_mean)**2)/(n_1-1))
```

```
    posterior1 = rtnew(1000, df=n_1-1, mean=y1_mean, scale=(sd_1/sqrt(n_1)))

    y2_mean = mean(data2)
    n_2 = length(data2)
    sd_2 = sqrt(sum((data2 - y2_mean)**2)/(n_2-1))
    posterior2 = rtnew(1000, df=n_2-1, mean=y2_mean, scale=(sd_2/sqrt(n_2)))

    mu_diff = posterior1 - posterior2

    low = 0 - sides
    high = 0  + sides
    inrange = sum(mu_diff > low & mu_diff < high)
    n  = length(mu_diff)
    inrange/n
}
```

```
mu_zero_interval(windshieldy1, windshieldy2)
```

```
[1] 0.007
```