````
---
title: STAT 457 Homework 03
author: Martha Eichlersmith
date: 2019-10-25
output:
  pdf_document:
    fig_caption: yes
header-includes:
  - \usepackage{color}
  - \usepackage{mathtools}
  - \usepackage{amsbsy} #bold in mathmode
  - \usepackage{nicefrac} # for nice fracs
---
```{r, echo=FALSE, results="hide", warning=FALSE, message=FALSE}
library(ggplot2)
library(readr)
library(gridExtra)
library(grid)
library(png)
library(downloader)
library(grDevices)
library(latex2exp)
library(knitr)
library(leaps)
library(directlabels)
library(diffusr)
library(MASS)
library(invgamma)
library(condMVNorm)
library(asht) #bfTest: Behrens-Fisher Test
library(mvnfast) #multi-variate t
library(matlib) #A = matrix, inv(A) = A^{-1}
decimal <- function(x, k) trimws(format(round(x, k), nsmall=k))
dec <- 5
```

## Problem 1
Consider an *iid* sample of size $n$ from the $\mathcal{N}(\mu,
\sigma^2)$ distribution, where $\sigma^2$ is **known**. Derive
the posterior distribution of $\mu$ under the prior $\mathcal{N}
(\mu_0, \sigma^2_0)$.
$$
\begin{aligned}
Y \mid \mu, \sigma^2 & \sim \mathcal{N}(\mu, \sigma^2) = \frac{1}
{\sqrt{2 \pi \sigma^2 }} \exp \left\{ - \frac{ (x - \mu )}{2
````

```
\right)
\right\}
\\[0.5ex]
& = \exp \left\{ - \frac{1}{2} \frac{1}{\sigma^2 \sigma_0^2}
\left(
\mu^2 (\sigma^2 + n \sigma_0^2) - 2 \mu (\mu_0 \sigma^2 +
\sigma_0^2 n \bar{x}) + (\mu_0^2 \sigma^2 + \sigma_0^2 \sum_{i=1}
^n x_i^2)
\right)\right\}
\\[0.5ex]
& \propto \exp \left\{ - \frac{1}{2} \left[
\mu^2 \left( \frac{1}{\sigma_0^2} + \frac{n}{\sigma^2} \right) -
2 \mu \left( \frac{\mu_0}{\sigma_0^2} + \frac{n \bar{x}}
{\sigma^2} \right)
+ k
\right] \right\} \quad \text{ where } k \text{ is some
normalizing constant}
\\[0.5ex]
& = \exp \left\{ - \frac{1}{2} \left( \frac{1}{\sigma_0^2} +
\frac{n}{\sigma^2} \right) \left[
\mu^2 \frac{ \frac{1}{\sigma_0^2} + \frac{n}{\sigma^2} }{
\frac{1}{\sigma_0^2} + \frac{n}{\sigma^2} }
- 2 \mu \frac{ \frac{\mu_0}{\sigma_0^2} + \frac{n\bar{x}}
{\sigma^2} }{ \frac{1}{\sigma_0^2} + \frac{n}{\sigma^2} } +k
\right] \right\}
\\[0.5ex]
& \propto \exp \left\{  - \frac{1}{2} \left( \frac{1}{\sigma_0^2}
+ \frac{n}{\sigma^2} \right) \left[
\mu - \frac{ \frac{\mu_0}{\sigma_0^2} + \frac{n\bar{x}}{\sigma^2}
}{ \frac{1}{\sigma_0^2} + \frac{n}{\sigma^2} }
\right]^2 \right\}
\\[0.5ex]
& \sim \mathcal{N}(\hat{\mu}, \hat{\sigma}_\mu^2 )
\\[0.5ex]
\hat{\mu} & = \left. \left( \frac{\mu_0}{\sigma_0^2} +
\frac{n\bar{x}}{\sigma^2} \right) \right/ \left(  \frac{1}
{\sigma_0^2} + \frac{n}{\sigma^2} \right)
\\[0.5ex]
\hat{\sigma}_\mu^2 & = \left( \frac{1}{\sigma_0^2} + \frac{n}
{\sigma^2} \right)^{-1}
\end{aligned}
$$
\newpage

## Problem 2
```

````
\\[0.5ex]
\implies p(t \mid \pmb{Y}) & =
\left( \tfrac{1}{2} \right) s^{-(n + 1)} \left( \tfrac{1}{2}(n -
1) \right)^{- \frac{(n + 1)}{2}} \ \Gamma \left( \tfrac{n+1}{2}
\right)
\left\{ 1 + \tfrac{1}{n - 1} t^2 \right\}^{ - \tfrac{(n + 1)}{2}}
\left( \tfrac{s}{\sqrt{n}} \right)
\\[0.5ex]
& \propto \frac{ \Gamma \left( \frac{n + 1}{2} \right)}{ \Gamma
\left( \frac{n}{2} \right)} \cdot \frac{1}{\sqrt{n \pi}} \left( 1
+ \tfrac{1}{n - 1} t^2 \right)^{ - \frac{(n+1)}{2}}
\\[0.5ex]
& \sim t \text{ distribuiton with }n \text{ degrees of freedom}
\\[0.5ex]
\implies p(\mu \mid \pmb{Y} )  & = \frac{ \Gamma \left( \frac{n +
1}{2} \right)}{ \Gamma \left( \frac{n}{2} \right)} \frac{ s /
\sqrt{n}}{ \sqrt{n \pi} \left[ 1 + \mu^2 /n \right]^{ (n + 1)/2}
} + \bar{x}
\end{aligned}
$$

**t.test:**
```{r, echo=FALSE}
ttest2a <- t.test(plants)
ttest2a <- c(mean(plants), ttest2a$conf.int[1:2], (1-
(ttest2a$p.value)/2))

results2a <- data.frame(t(decimal(ttest2a, dec)))
colnames(results2a) <- c("mu.hat", "CI.lower", "CI.upper", "P(mu
> 0 | Y)")
kable(results2a)
```


\newpage
### Problem 2b
Sort data and throw out the $i^\text{th}$ value.  Simulate from
the posterior and sort data again.  Plot the mean of the
simulated $i^\text{th}$ data and the actual $i^\text{th}$ data.

```{r, echo=FALSE}
func_simvals.of.ith <- function(ith, rep, n, xbar, sd){
  bigmatrix <- matrix(rnorm(n*rep, mean=xbar, sd=sd), nrow=n,
ncol=rep)
  samp.of.ith <- apply(bigmatrix, 2, sort)[ith,]
````
````

```
                     -6.167448,
                      1.215080,
                      8.208091,
                     14.255372,
                     20.838637,
                     27.000632,
                     33.465391,
                     39.391671,
                     46.125491,
                     54.086084,
                     64.936747,
                     79.251345)
#plant.sim.100  <- func_simvals(100000, plants) #run before hand,
takes about 1-2 min
plant.sim.100  <- c(-24.787618,
                   -16.258298,
                   -14.894389,
                    -6.044791,
                     1.228475,
                     8.145858,
                    14.315700,
                    20.713908,
                    26.924897,
                    33.500010,
                    39.491774,
                    46.371982,
                    54.329831,
                    64.973360,
                    79.487609)
#plant.sim.1000 <- func_simvals(1000000, plants) #run before
hand, takes about 13-14 min
plant.sim.1000 <- c(-24.772450,
                   -16.307124,
                   -14.904322,
                    -6.017075,
                     1.238906,
                     8.153171,
                    14.283596,
                    20.721454,
                    26.883395,
                    33.469015,
                    39.494382,
                    46.331012,
                    54.301564,
                    64.978522,
```

```
  geom_text(aes(x=Actual, y=Simulated, label=ith), hjust=0,
vjust=0, size=4.5)+
  ggtitle(paste("Actual vs Expected (via Simulation)"))+
  xlab("Actual Plant Height Differences") +
  ylab("Simulated (without ith value) Plant Height Differences")+
  facet_wrap(~Iterations, nrow=1, scales="free")

plants.df <- data.frame("plants" = plants, "ith" = c(1:15))
plot2b.02 <- ggplot(plants.df, aes(sample=plants)) +
  ggtitle("Normal QQ Plot") +
  xlab("Theoretical") + ylab("Actual Plant Height Differences") +
  geom_qq_line(size=1, color="red", linetype="dashed") +
  geom_qq(size=4, color="grey")

Glist <- list(plot2b.01, plot2b.02)
grid.arrange(grobs=Glist, ncol=2, widths = c(3,1))
```

The points that stray from the line and might be considered as outliers are the 1st, 2nd, and 3rd ordered values.
The normal QQ plot also suggests that the 1st and 2nd values are outliers.

Looking at simulated distribution after taking out $\pmb{i^{\text{th}}}$ value

```
```{r, echo=FALSE, fig.height=2, fig.width=3, message=FALSE,
results='hide'}
func_simvals.of.ith <- function(ith, rep, n, xbar, sd){
  bigmatrix <- matrix(rnorm(n*rep, mean=xbar, sd=sd), nrow=n,
ncol=rep)
  samp.of.ith <- apply(bigmatrix, 2, sort)[ith,]
}
data <- plants
func_sim.ith.vals <- function(rep, ith){
#rep <- 10
#ith <- 1
  ordered <- sort(data)
  n <- length(data)
  order.i <- c()
  for (i in 1:n){
    order.i <- cbind(order.i, ordered[-i])}
  xbar.vec <- apply(order.i, 2, mean)
  sd.vec <- apply(order.i, 2, sd)
  ith.vec <- c(1:n)
  n.vec <- rep(n, n)
```
```

````
return(plot) }
```

```{r, echo=FALSE, fig.height=4, fig.width=10, message=FALSE,
warning=FALSE}
set.seed(030202) #Homeowrk 03 | Problem 02 | Part 02
plot.1st <- func_simdensityplots(plants,  1)
plot.2nd <- func_simdensityplots(plants,  2)
plot.3rd <- func_simdensityplots(plants,  3)
Glist <- list(plot.1st, plot.2nd, plot.3rd)
grid.arrange(grobs=Glist, ncol=3)
```
The first and second actual values (-67, -48) are in the bottom
2% for the 1st and 2nd order statistics, respectively so they may
be considered as outliers.  The third actual value (6) would not
be considered an outlier since it is in the top 7.12% of values
expected for the third ordered value.



### Problem 2c (Extra Credit)
Unfortunately, did not have time to attempt extra credit.
```{r, echo=FALSE }
#
# Suppose that the data follow the $t$ distribution with mean $
\mu$, variance $\sigma^2$ (unknown), on 4 degrees of freedom.
What is your best guess for $\mu$?  Obtain a 97% credible
interval for $\mu$ and interpret this interval.
# **OUTSTANDING**
```



\newpage
## Problem 3
The following data, taken from Wallace (1980), represent the
hours of post-operative pain relief for subjects receiving one of
two drugs:
```{r}
drug1 <- c(2, 4, 4, 5, 6, 8, 13)
drug2 <- c(0, 0, 0, 1, 1, 2, 2, 2 ,3, 3 ,3, 4, 8)
```

```{r, echo=FALSE}
xbar.1 <- mean(drug1)
````

````
Assume that $\sigma_1^2 \neq \sigma_2^2$.  Via simulation, repeat
3a.
```{r, echo=FALSE}
set.seed(030302) #Homeowrk 03 | Problem 03 | Part 02

sim <- function(rep, data1, data2){
xbar.1 <- mean(data1)
xbar.2 <- mean(data2)
sd.1 <- sd(data1)
sd.2 <- sd(data2)
n.1 <- length(data1)
n.2 <- length(data2)
df.1 <- n.1 - 1
df.2 <- n.2 - 1
mu.1.star <- xbar.1 - rt(rep, df.1)*sd.1/sqrt(n.1)
mu.2.star <- xbar.2 - rt(rep, df.2)*sd.2/sqrt(n.2)
diff.vec <- mu.1.star - mu.2.star
sim_CI <-quantile(diff.vec, probs=c(.025, 0.975))[1:2]
sim_CIlength <- (as.vector(sim_CI[2]) - as.vector(sim_CI[1]))
sim_p <- length(diff.vec[diff.vec > 0 ])/length(diff.vec)
result <- c(sim_CI, sim_CIlength, sim_p)
return(result)
}




sim.10    <- sim(10000   , drug1, drug2)
sim.100   <- sim(100000  , drug1, drug2)
sim.1000  <- sim(1000000 , drug1, drug2)
sim.10000 <- sim(10000000, drug1, drug2)

results3b <- rbind(
   c("n=1e+04", t(decimal(sim.10   , dec)))
  ,c("n=1e+05", t(decimal(sim.100  , dec)))
  ,c("n=1e+06", t(decimal(sim.1000 , dec)))
  ,c("n=1e+07", t(decimal(sim.10000, dec)))
)
colnames(results3b) <- c("Iterations", "CI.lower", "CI.upper",
"CI.length", "P(mu.1-mu.2 > 0 | Y)")
kable(results3b)
```


### Problem 3c
````

~~~~
t.patil<-qt(0.975,b)
se.patil <- a*sqrt(var(drug1)/n.1+var(drug2)/n.2)
upper <- (xbar.1 - xbar.2) + se.patil*t.patil
lower <- (xbar.1 - xbar.2) - se.patil*t.patil
patil <- c(lower, upper, upper - lower, pt((xbar.1 - xbar.2)/
se.patil,b))

#patil.info <- c(f1, f2, b, a, t.patil, se.patil)

#patil.info <- decimal(patil.info, dec)

patil.info <- data.frame(
   "f1" = f1
  ,"f2" = f2
  , "a" = a
  , "b" = b
  , "qt(0.975,b)" = t.patil
  , "se.patil" = se.patil
)

patil.info[1,] <-  decimal(patil.info, dec)
kable(patil.info)

results3c2 <- data.frame(t(decimal(patil, dec)))
colnames(results3c2) <- c("CI.lower", "CI.upper", "CI.length",
"P(mu.1-mu.2 > 0 | Y)")
kable(results3c2)
```




### Problem 3d
Repeat 3a using the Welch $t$ approximation.
```{r, echo=FALSE}
t.Welch <- t.test(drug1, drug2, var.equal=FALSE)
t.Welch <- c(t.Welch$conf.int[1:2], t.Welch$conf.int[2] -
t.Welch$conf.int[1], (1- (t.Welch$p.value)/2))

results3d <- data.frame(t(decimal(t.Welch, dec)))
colnames(results3d) <- c("CI.lower", "CI.upper", "CI.length",
"P(mu.1-mu.2 > 0 | Y)")
kable(results3d)
```
~~~~

````
```{r, echo=FALSE}
func_new.data <- function(data){
X <- c()
Y <- c()
Test <- c()
I <- nrow(data)
for (i in 1:I){
x <- c(data[i, 1:2])
y <- c(rep(I+1-i, 2))
test <- rep(rownames(data)[i], 2)
X <- c(X, x)
Y <- c(Y, y)
Test <- c(Test, test)
}
newdata <- data.frame("X"= X, "Y" = Y, "Test"=Test)
return(newdata)
}

graphdata <- func_new.data(data3)

test.names <- data.frame(
  "number" = c(1:8),
  "names" = rownames(data3))

test.names.order <- as.vector(test.names[order(
test.names$number),  ]$names)

graphdata$Test <- factor(graphdata$Test, levels =
test.names.order)


ggplot(graphdata, aes(x=X, y=Y, color=Test, linetype=Test)) +
geom_line(size=1.5) +
  geom_point(aes(x=X, y=Y, color=Test), size=2.5)+
  xlab("Confidence Interval")+
   theme(
        plot.title=element_text(hjust = 0.5) #hjust=.5 centers
the title
        ,axis.text.y = element_blank()
        #,axis.text.x = element_text(size=8)
        ,axis.title.y=element_blank()
   )
```
````