

```

---
title: STAT 457 Homework 05
author: Martha Eichlersmith
date: 2019-11-16
output:
  pdf_document:
    fig_caption: yes
header-includes:
  - \usepackage{color}
  - \usepackage{mathtools}
  - \usepackage{amssbsy} #bold in mathmode
  - \usepackage{nicefrac} # for nice fracs
  - \usepackage{booktabs}
  - \usepackage{geometry}
  - \usepackage{caption} #to remove automatic table name and
number - \captionsetup[table]{labelformat=empty}, put code under
---
geometry: "left=1.75cm,right=1.75cm,top=1.5cm,bottom=2cm"

---

\captionsetup[table]{labelformat=empty}
```{r setup, echo=FALSE, results="hide", warning=FALSE,
message=FALSE}
library(ggplot2) #ggplot
library(readr) #import CSV
library(gridExtra) #organize plots
library(grid) #organize plots
library(latex2exp) #latex in ggplot titles
library(matlib) #A = matrix, inv(A) = A^{-1}
library(numDeriv) #calculate numerical first and second order
derivatives
library(gtable) #for tablegrob functions
#library(kableExtra) #for kable functions
library(dplyr) #for piping
library(MCMCpack) #for dirichelt
knitr::opts_chunk$set(echo=FALSE, fig.width = 10, fig.height = 4)
decimal <- function(x, k) trimws(format(round(x, k), nsmall=k))
dec <- 5
#knitr::opts_chunk$set(echo=FALSE) #using knitr for this option
but don't have to load
```

## Problem 1
Consider two urns each containing an unknown mixture of blue and

```

\$\$

Problem 1a

Let π denote the proportion of blue marbles in urn #1 and let ψ denote the corresponding proportion in urn #2. Under the (i) Haldane, (ii) flat and (iii) non-informative priors, compute $p\left(\ln\left[\frac{\pi}{1-\pi}\right] > \ln\left[\frac{\psi}{1-\psi}\right] \mid \text{data}\right)$ using the normal approximation.

\$\$

$$p\left(\ln\left[\frac{\pi}{1-\pi}\right] > \ln\left[\frac{\psi}{1-\psi}\right] \mid \text{data}\right)$$

=

$$p\left(\ln\left[\frac{\pi}{1-\pi}\right] - \ln\left[\frac{\psi}{1-\psi}\right] > 0 \mid \text{data}\right)$$

\$\$

\$\$

$$\begin{aligned} & p(\pi) \propto \text{Beta}(\alpha_0, \beta_0) \\ & p(\psi) \propto \text{Beta}(\alpha_0, \beta_0) \\ & p(\pi \mid Y) \propto \text{Beta}(y_{\pi} + \alpha_0, n_{\pi} - y_{\pi} + \beta_0) \\ & p(\psi \mid Y) \propto \text{Beta}(y_{\psi} + \alpha_0, n_{\psi} - y_{\psi} + \beta_0) \end{aligned}$$

$$p(\pi) \propto \text{Beta}(\alpha_0, \beta_0)$$

$$p(\psi) \propto \text{Beta}(\alpha_0, \beta_0)$$

$$p(\pi \mid Y) \propto \text{Beta}(y_{\pi} + \alpha_0, n_{\pi} - y_{\pi} + \beta_0)$$

$$p(\psi \mid Y) \propto \text{Beta}(y_{\psi} + \alpha_0, n_{\psi} - y_{\psi} + \beta_0)$$

$$\text{Normal Approx Mean} \propto \ln\left(\frac{\alpha \cdot \delta}{\beta \cdot \gamma}\right)$$

$$\text{Normal Approx Variance} \propto \frac{1}{\alpha} + \frac{1}{\beta} + \frac{1}{\gamma} + \frac{1}{\delta}$$

$$\text{Normal Approx} \propto \mathcal{N}\left(\ln\left(\frac{\alpha \cdot \delta}{\beta \cdot \gamma}\right), \sqrt{\frac{1}{\alpha} + \frac{1}{\beta} + \frac{1}{\gamma} + \frac{1}{\delta}}\right)$$

$$\sqrt{\frac{1}{\alpha} + \frac{1}{\beta} + \frac{1}{\gamma} + \frac{1}{\delta}}$$

} \backslash

\right)

\end{aligned}

\$\$

```{r pla, warning=FALSE}

n.pi <- 18

n.psi <- 6

y.pi <- 14

y.psi <- 2

```

Repeat (1a) by drawing deviates from the appropriate beta
distributions. Quantify the Monte Carlo error in your value.
```{r plb, warning=FALSE}
set.seed(050101)

n.pi <- 18
n.psi <- 6
y.pi <- 14
y.psi <- 2

func_pval <- function(it, a0, b0){
  a0 <- 0
  b0 <- 0
  alpha <- y.pi + a0
  gamma <- y.psi + a0
  beta <- n.pi - y.pi + b0
  delta <- n.psi - y.psi + b0

  post.pi <- rbeta(it, alpha, beta)
  post.psi <- rbeta(it, gamma, delta)

  inside.log <- (post.pi/(1 - post.pi))/(post.psi/(1 - post.psi))
  #inside.log <- (alpha*delta) / (beta*gamma)
  difflogodds <- log(inside.log)
  se <- sd(difflogodds)/sqrt(it)
  pval <- 1 - pnorm(0, mean(difflogodds), sd(difflogodds))
  results <- c(pval, se)
}

func_logdiff <- function(it.vec, a0, b0){
  a0.vec <- rep(a0, length(it.vec))
  b0.vec <- rep(b0, length(it.vec))
  pvals <- mapply(func_pval, it.vec, a0.vec, b0.vec)
  table <- rbind(it.vec, decimal(pvals, dec))
  rownames(table) <- c("Iterations", "p-value", "Standard Error")
  table
}

it.vec <- c(1e+04, 1e+05, 1e+06)

dt <- cbind(func_logdiff(it.vec, 0, 0), func_logdiff(it.vec, 1,
1), func_logdiff(it.vec, .5, .5))

knitr::kable(dt, booktabs=T, 'latex', caption="Probability that
the log differences are greater than 0") %>%

```

```

\newpage
### Problem 1d
Is delinquency related to birth order?
```{r 1d-counts}
alpha <- c(127, 123, 93, 17)/360 #most delinquent
gamma <- c(345, 209, 158, 65)/777 #least delinquent
parameters <- rbind(alpha, gamma)
total <- c(sum(alpha), sum(gamma))
table.count <- decimal(cbind(parameters, total), dec)
rownames(table.count) <- c("Most Delinquent", "Least Delinquent")
colnames(table.count) <- c("Oldest", "In-between", "Youngest",
"Only Child", "Total")
knitr::kable(table.count, booktabs=TRUE, 'latex', caption="Row
Proportion") %>%
 kableExtra::kable_styling(latex_options="hold_position")
```

```{r 1d-function}
func_diff <- function(matrix){
diff.matrix <- cbind(#most - least
 matrix[,1] - matrix[,5] #diff for Oldest
 ,matrix[,2] - matrix[,6] #diff for in-between
 ,matrix[,3] - matrix[,7] #diff for youngest
 ,matrix[,4] - matrix[,8] #diff for only-child
)
diff.matrix
}

func_values <- function(vec){
 it <- length(vec)
 CI.val <- decimal(quantile(vec, c(0.025, 0.975)), dec)
 CI.95 <- paste("(", paste(CI.val, collapse=" ", ")", ")")
 mean <- decimal(mean(vec), dec)
 pval <- length(vec[vec > 0])/it
 vec <- c(it, mean, CI.95, pval)
 vec
}

func_delinq <- function(it){
set.seed(050104)
prop <- rdirichlet(it, t(parameters))
diff <- cbind(#most - least
 prop[,1] - prop[,5] #diff for Oldest
 ,prop[,2] - prop[,6] #diff for in-between

```

```

table <- cbind(it.1[,i], it.2[,i] , it.3[,i])
rownames(table) <- c("Iterations", "Mean", "95% CI", "P(diff>0)")
title <- paste("Difference in Proportion for", name, ": Most
Delinquent - Least Delinquent")
knitr::kable(table, booktabs=T, 'latex', caption=title) %>%
 kableExtra::kable_styling(latex_options="hold_position") #hold
table in place
}

```

```

func_table(1, "Oldest")
func_table(2, "In-Between")
func_table(3, "Youngest")
func_table(4, "Only")
...

```

```

```{r 1d-chisqtest}
func_pval.chisq <- function(pvals){
  x <- -2*sum(log(as.numeric(pvals)))
  combined.pval <- pnorm(x, 2*length(pvals))
}

```

```

pvals.vec <- cbind(it.1[4,], pval.2 <- it.2[4,], pval.3 <- it.
3[4,])

```

```

combined <- apply(pvals.vec, 2, func_pval.chisq)
combined.pvals <- rbind(it.vec, decimal(combined, dec))
rownames(combined.pvals) <- c("Iterations", "Combined p-value")

```

```

title <- paste("Combined p-values:", "$X = -2\\sum_{i=1}^4(p_i) \\sim \\chi^2_{df=8}$")
knitr::kable(combined.pvals, booktabs=T, 'latex', caption=title)
%>%
  kableExtra::kable_styling(latex_options="hold_position" )
...

```

There is evidence that birth order has an effect on delinquency rates.

```

\newpage
## Problem 2

```

$$2\sin^{-1}\sqrt{\hat{p}} \text{ \& \stackrel{\mathcal{D}}{\rightarrow} \mathcal{N} \left(2\sin^{-1}\sqrt{p}, 1 \right)}$$

Problem 3
 Let x_1, \dots, x_n be an iid sample from $\mathcal{N}(\theta, 1)$ and let y_1, \dots, y_n be an independent iid sample from $\mathcal{N}(\phi, 1)$. Derive the distribution of $\overline{x} / \overline{y}$ (where $\overline{y} \neq 0$) via the delta method.

$$\begin{aligned} \overline{x} &\sim \mathcal{N}(\theta, \frac{1}{n}) \\ \overline{y} &\sim \mathcal{N}(\phi, \frac{1}{n}) \\ \text{Let } h(x, y) &= x / y \\ \text{Let } h(B) &= \overline{x} / \overline{y} \\ \text{Let } h(\beta) &= \theta / \phi \\ \sqrt{n} \left(h(B) - h(\beta) \right) &\stackrel{\mathcal{D}}{\rightarrow} \mathcal{N} \left(0, \nabla h(\beta)^T \Sigma \nabla h(\beta) \right) \\ \Sigma &= \begin{bmatrix} 1/n & 0 \\ 0 & 1/n \end{bmatrix} \\ \nabla h(\beta)^T &= \begin{bmatrix} \frac{\partial h}{\partial x} & \frac{\partial h}{\partial y} \end{bmatrix}_{\theta, \phi} \\ &= \begin{bmatrix} \frac{1}{y} & -\frac{x}{y^2} \end{bmatrix}_{\theta, \phi} \\ &= \begin{bmatrix} \frac{1}{\phi} & -\frac{\theta}{\phi^2} \end{bmatrix} \\ \nabla h(\beta)^T \Sigma \nabla h(\beta) &= \begin{bmatrix} \frac{1}{\phi} & -\frac{\theta}{\phi^2} \end{bmatrix} \begin{bmatrix} 1/n & 0 \\ 0 & 1/n \end{bmatrix} \begin{bmatrix} \frac{1}{\phi} & -\frac{\theta}{\phi^2} \end{bmatrix}^T \end{aligned}$$

```

\\[1ex]
\frac{ \overline{x}}{\overline{y}} & \stackrel{\mathcal{D}}{\rightarrow}
\mathcal{N} \left(
\frac{\theta}{\phi}, \frac{1}{\phi^2} - \frac{\sigma^2}{\phi^4}
\right)
\end{aligned}
$$

```

```

\newpage
## Problem 4
197 animals are distributed into four categories:  $Y = (y_1, y_2, y_3, y_4)$  according to the genetic linkage model  $\left( \frac{2}{2 + \theta}, \frac{1 - \theta}{2}, \frac{1 - \theta}{2}, \frac{\theta}{2} \right)$ . In HW#4 you derived the likelihood for the data  $Y = (125, 18, 20, 34)$  and you derived the likelihood for the data  $Y = (14, 0, 1, 15)$ . In that homework, you also used Newton-Raphson algorithm to obtain the MLE ( $\hat{\theta}$ ) of  $\theta$  and the standard error of  $\hat{\theta}$ .

```

```

$$
\begin{aligned}
L(\theta \mid \mathbf{Y}) &= \\
&\frac{(y_1 + y_2 + y_3 + y_4)!}{y_1! y_2! y_3! y_4!} \\
&\left( \frac{2 + \theta}{2} \right)^{y_1} \\
&\left( \frac{1 - \theta}{2} \right)^{y_2} \\
&\left( \frac{1 - \theta}{2} \right)^{y_3} \\
&\left( \frac{\theta}{2} \right)^{y_4} \\
&\propto (2 + \theta)^{y_1} \cdot (1 - \theta)^{y_2 + y_3} \cdot \theta^{y_4}
\end{aligned}
\\[1ex]
\ell(\theta \mid \mathbf{Y}) &= \log L(\theta \mid \mathbf{Y}) = y_1 \log(2 + \theta) + (y_2 + y_3) \log(1 - \theta) + y_4 \log(\theta)
\\[.5ex]
\frac{\partial \ell}{\partial \theta} &= \frac{y_1}{2 + \theta} - \frac{y_2 + y_3}{1 - \theta} + \frac{y_4}{\theta}
\\[.5ex]
\frac{\partial^2 \ell}{\partial \theta^2} &= -\frac{y_1}{(2 + \theta)^2} - \frac{y_2 + y_3}{(1 - \theta)^2} - \frac{y_4}{\theta^2}
\\[.5ex]
\theta^{(i+1)} &= \theta^{(i)} - \\
&\frac{
\frac{y_1}{(2 + \theta^{(i)})^2} + \frac{y_2 + y_3}{(1 - \theta^{(i)})^2} + \frac{y_4}{\theta^{(i)2}}
}{
\frac{y_1}{2 + \theta^{(i)}} - \frac{y_2 + y_3}{1 - \theta^{(i)}} + \frac{y_4}{\theta^{(i)}}
}

```

```

name <- paste("Normal Likelihood and Normal Approximation for
Y=", print.yval, ")")

table <- data.frame("Statistic"=c("MLE", "Standard Error")
, "Value"=decimal(c(theta.mle, se), dec))
tg <- tableGrob(table)

x <- seq(0, 1, 0.001)
df <- data.frame("X"=x)
ggplot(data=df, aes(x=X))+
  norm.like+normal.approx+
  ggtitle(paste(name))+
  theme(axis.title.x = element_blank())+
  scale_colour_manual("", values = c(colors[1], colors[2])) +
  #theme(legend.position = "bottom")+
  annotation_custom(tg, xmin=.0, ymin=.25, xmax=0.5, ymax=1)

}
...

```{r p4-netwonraphson}
func_newton.raphson <- function(f, start, it, tol){
 x0 <- start
 k <- c()
 for (i in 1:it) {
 x1 <- x0 - f(x0) #calcualte next value x1
 k[i] <- x1 #store x1
 root.approx <- tail(k, n=1)
 it.completed <- length(k)
 # Once the difference between x0 and x1 becomes sufficiently
small, output the results.
 if (abs(x1 - x0) < tol & !is.na(abs(x1- x0)))
 { root.approx
 break}
 else if(it.completed == it){print(paste("Start at", start,
": diverges"))}
 else{ x0 <- x1}
 }
 return(root.approx)
}
tol <- 1e-5
it <- 1000

func_se <- function(theta.mle){
 secondderiv <- -(yval[1] / (2 + theta.mle)^2) - ((yval[2] +

```



```

theta.mle <- func_newton.raphson(func_a, 0.6, it, tol)
se <- func_se(theta.mle)

func_plots(yval, theta.mle, se)
```

### Problem 4b
Repeat (4a) for $Y = (14, 0, 1, 5)$
```{r p4b, warning=FALSE, fig.height=2.5}
yval <- c(14, 0, 1, 5)
func_b<-function(x){
 (
 yval[1]/(2+x)-(yval[2]+yval[3])/(1-x)+yval[4]/x
) / (
 -yval[1]/(2+x)^2-(yval[2]+yval[3])/(1-x)^2-yval[4]/x^2
) #iterations of newton raphson = theta^{i + 1} = theta^{i} -
(1st deriv of log like) / (2nd deriv of log like)
}

theta.mle <- func_newton.raphson(func_b, 0.9, it, tol)
se <- func_se(theta.mle)

func_plots(yval, theta.mle, se)
```

\newpage
## Problem 5
Use Laplace's method (second order) to compute the posterior mean
(under a flat prior) for the genetic linkage model for both data
sets.

```{r p5-LaplacePostMean}
func_LaplacePostMean <- function(yval, start.hat, start.star){
 n <- sum(yval)

 func_nh <- function(x){ yval[1]*log(2+x) + (yval[2] +
yval[3])*log(1 - x)+ yval[4]*log(x) }
 #-nh(theta)

 func_1st.nh <- function(x){ yval[1]/(2+x)-(yval[2]+yval[3])/(1-x)
+yval[4]/x}
 #d(-nh(theta))/d theta- first derivative

```

```

func_theta.star <-function(x){ (func_1st.nh.star(x)) / (
func_2nd.nh.star(x)) }
 #function for NR to find mle theta star for -nh*(theta)
 #iterations of newton raphson = theta^{i + 1} = theta^{i} -
(1st deriv of log like) / (2nd deriv of log like)

func_sig.star <- function(x){1/sqrt(func_2nd.nh.star(x)/-n) }
 #sigma.star = h''(theta.star) ^{-1/2}

theta.hat <- func_newton.raphson(func_theta.hat, start.hat, it,
tol)
theta.star <- func_newton.raphson(func_theta.star, start.star,
it, tol)
sig.hat <- func_sig.hat(theta.hat)
sig.star <- func_sig.star(theta.star)

post.mean <- (sig.star/sig.hat) *
(exp(func_nh.star(theta.star)))/(exp(func_nh(theta.hat)))

name <- c("theta.hat", "theta.star", "sigma.hat", "sigma.star",
"Posterior Mean")
values <- c(theta.hat, theta.star, sig.hat, sig.star, post.mean)

table <- data.frame("Statistic"=name, "Value"=decimal(values,
dec))

print.yval <- paste(yval, collapse=", ")
name <- paste("Laplace's Method (Second Order) of Posterior Mean
for Y=(", print.yval, ")")

knitr::kable(table, booktabs=T, 'latex', caption=name) %>%
 kableExtra::kable_styling(latex_options="hold_position")
}
...

$$
\begin{aligned}
&L(\theta \mid Y) \propto (2 + \theta)^{y_1} (1 - \theta)^{y_2 + y_3} (\theta)^{y_4} \\
&\ell(\theta \mid Y) \propto y_1 \ln(2 + \theta) + (y_2 + y_3) \ln(1 - \theta) + y_4 \ln(\theta)
\end{aligned}

```

```
```{r p5a}
yval <- c(125, 18, 20, 34)
func_LaplacePostMean(yval, 0.6, 0.6)
```

```{r p5b}
yval <- c(14, 0, 1, 5)
func_LaplacePostMean(yval, 0.9, 0.91)
```
```