Homework 7, STA 360

Jerry Xin

Total points: 10 (reproducibility) + 30 (Q1) = 40 points.

General instructions for homeworks: Please follow the uploading file instructions according to the syllabus. You will give the commands to answer each question in its own code block, which will also produce plots that will be automatically embedded in the output file. Each answer must be supported by written statements as well as any code used. Your code must be completely reproducible and must compile. Syllabus: (https://github.com/resteorts/modern-bayes/blob/master/syllabus/syllabus-sta602-spring19.pdf)

Advice: Start early on the homeworks and it is advised that you not wait until the day of. While the professor and the TA's check emails, they will be answered in the order they are received and last minute help will not be given unless we happen to be free.

Commenting code Code should be commented. See the Google style guide for questions regarding commenting or how to write code https://google.github.io/styleguide/Rguide.xml. No late homework's will be accepted.

Please look over the homework before lab this week. TA's will answer questions on the homework this week regarding these two problems below. I recommend that you work through them as much as possible before lab this week.

1. Multivariate Normal, 30 points, 10 points each) Hoff exercise 7.3 (Australian crab data).
2. Imputation, 50 points, 10 points each) Hoff 7.4 (Marriage data) (This is left as an optional exercise that will not be graded.)

7.3 Australian crab data: The files bluecrab.dat and orangecrab.dat contain measurements of body depth (Y1) and rear width (Y2), in millimeters,made on 50 male crabs from each of two species, blue and orange. We will model these data using a bivariate normal distribution.

```
library(plyr)
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:plyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
```

```
##     intersect, setdiff, setequal, union
```

```
library(xtable)
library(reshape)
```

```
##
## Attaching package: 'reshape'
```

```
## The following object is masked from 'package:dplyr':
##
##     rename
```

```
## The following objects are masked from 'package:plyr':
##
##     rename, round_any
```

```
library(tidyverse)
```

```
## Warning in system("timedatectl", intern = TRUE): running command 'timedatectl'
## had status 1
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v tibble  3.1.5     v purrr   0.3.4
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.2     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::arrange()   masks plyr::arrange()
## x purrr::compact()   masks plyr::compact()
## x dplyr::count()     masks plyr::count()
## x tidyr::expand()    masks reshape::expand()
## x dplyr::failwith()  masks plyr::failwith()
## x dplyr::filter()    masks stats::filter()
## x dplyr::id()        masks plyr::id()
## x dplyr::lag()       masks stats::lag()
## x dplyr::mutate()    masks plyr::mutate()
## x reshape::rename()  masks dplyr::rename(), plyr::rename()
## x dplyr::summarise() masks plyr::summarise()
## x dplyr::summarize() masks plyr::summarize()
```

```
library(ggrepel)
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
bluecrab = as.matrix(read.table("data/bluecrab.dat"))
orangecrab = as.matrix(read.table("data/orangecrab.dat"))
```

a) (10 points) For each of the two species, obtain posterior distributions of the population mean theta and covariance matrix sigma as follows: Using the semiconjugate prior distributions for theta and sigma, set μ0 equal to the sample mean of the data, (caret) 0 and S0 equal to the sample covariance matrix and (caret)0 = 4. Obtain 10,000 posterior samples of theta and sigma. Note that this "prior" distribution loosely centers the parameters around empirical estimates based on the observed data (and is very similar to the unit information prior described in the previous exercise). It cannot be considered as our true prior distribution, as it was derived from the observed data. However, it can be roughly considered as the prior distribution of someone with weak but unbiased information.

```
set.seed(123)
crab.mcmc = lapply(list('bluecrab' = bluecrab, 'orangecrab' = orangecrab), function(crab) {
  c = ncol(crab)
  r = nrow(crab)
  ybar = colMeans(crab)

  # Prior parameters

  mu_0 = ybar
  lambda_0 = s_0 = cov(crab)
  nu_0 = 4

  num = 10000
  theta_2 = matrix(nrow = num, ncol = c)
  sigma_2 = array(dim = c(c, c, num))

  # Start with sigma sample
  sigma = s_0

  # Also, inv = solve to make it more readable
  inv = solve

  for (n in 1:num) {
    # Update theta
    lambda_n = inv(inv(lambda_0) + r * inv(sigma))
    mu_n = lambda_n %*% (inv(lambda_0) %*% mu_0 + r * inv(sigma) %*% ybar)
    theta = mvrnorm(n = 1, mu_n, lambda_n)

    # Update sigma
    resid = t(crab) - c(theta)
    s_theta = resid %*% t(resid)
    s_n = s_0 + s_theta
    sigma = inv(rWishart(1, nu_0 + r, inv(s_n))[, , 1])

    theta_2[n, ] = theta
    sigma_2[, , n] = sigma
  }

  list(theta = theta_2, sigma = sigma_2)
})
```

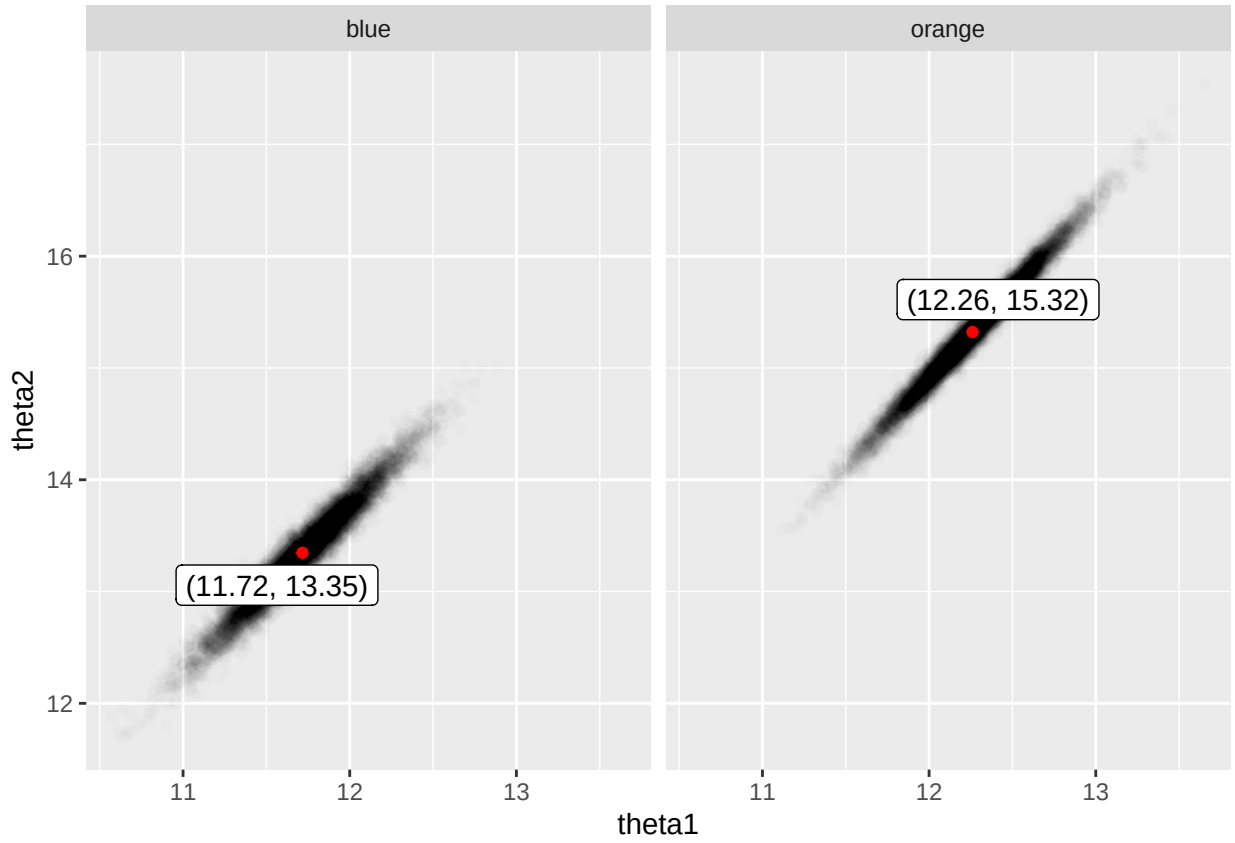
b) (10 points) Plot values of theta = (theta1, theta2)' for each group and compare. Describe any size differences between the two groups.

```
bluecrab.df = data.frame(crab.mcmc$bluecrab$theta, species = 'blue')
orangecrab.df = data.frame(crab.mcmc$orangecrab$theta, species = 'orange')

colnames(bluecrab.df) = colnames(orangecrab.df) = c('theta1', 'theta2', 'species')
crab.df = rbind(bluecrab.df, orangecrab.df)
bluecrab.means = as.data.frame(t(as.matrix(colMeans(bluecrab.df[, c('theta1', 'theta2')]))))
orangecrab.means = as.data.frame(t(as.matrix(colMeans(orangecrab.df[, c('theta1', 'theta2')]))))
bluecrab.means$species = 'blue'
orangecrab.means$species = 'orange'

crab.means = rbind(bluecrab.means, orangecrab.means)


ggplot(crab.df, aes(x = theta1, y = theta2)) +
  geom_point(alpha = 0.01) +
  geom_point(data = crab.means, color = 'red') +
  geom_label_repel(data = crab.means, aes(label = paste0("(", round(theta1, 2), ", ", round(theta2, 2),
  facet_wrap(~ species)
```



Looking at the plots of theta 1 and theta 2, we can see that on average, orange crabs seem to have a larger theta 1 value than blue crabs, and on average, orange crabs seem to have a larger theta 2 value than blue crabs.

```
mean(orangecrab.df$theta1 > bluecrab.df$theta1)
```

```
## [1] 0.8996
```

```
mean(orangecrab.df$theta2 > bluecrab.df$theta2)
```

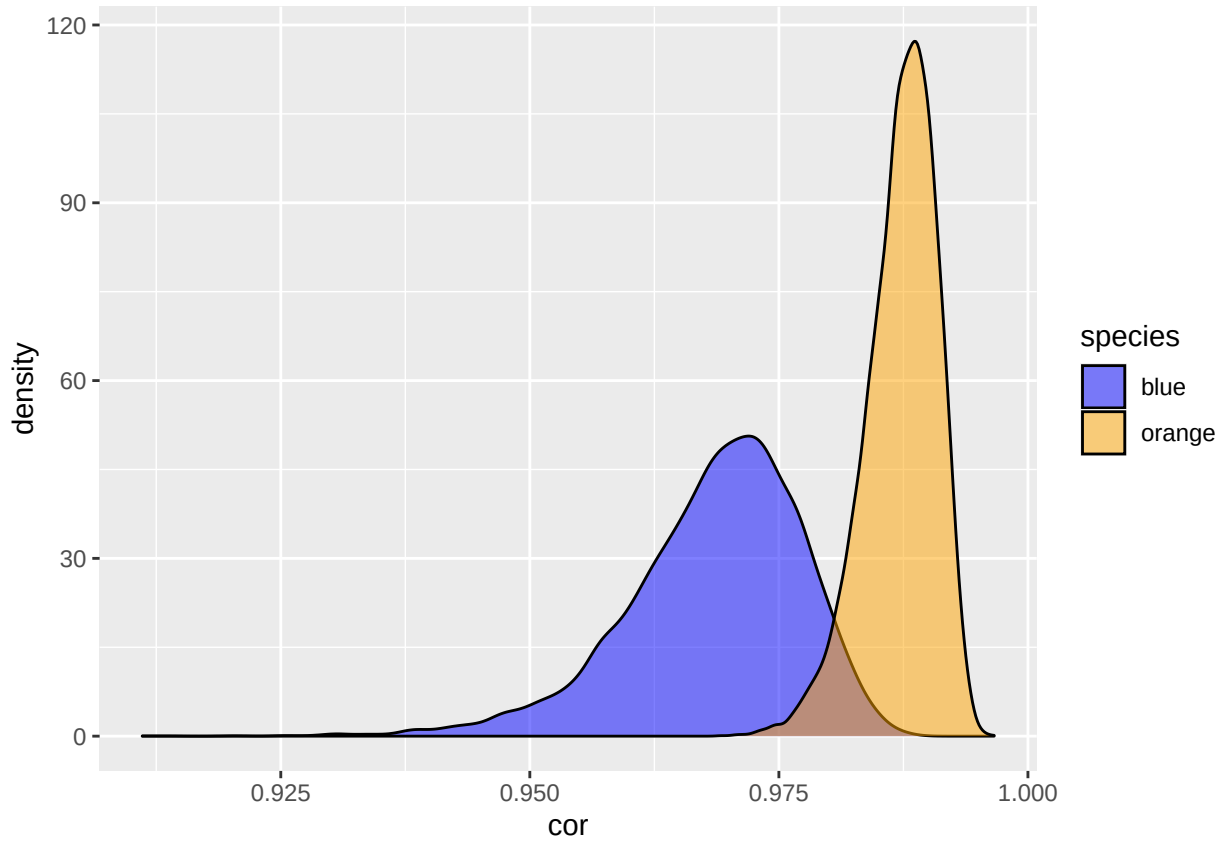
```
## [1] 0.9983
```

Looking at the plots of theta 1 and theta 2, we can see that on average, orange crabs seem to have a larger theta 1 value than blue crabs, and on average, orange crabs seem to have a larger theta 2 value than blue crabs. This is further shown by the looking at the probability that theta 1 is greater for orange crabs than blue crabs, which is 0.8967 and the probability that theta 2 is greater for orange crabs than blue crabs, which is 0.9975. This shows that 89.67 percent of the time theta 1 is greater for orange crabs compared to blue crabs, which is not statistically significant at an alpha level of 0.05. This also shows that 99.75% of the time theta 2 is greater for orange crabs compared to blue crabs, which is significant at an alpha level of 0.05.

c) (10 points) From each covariance matrix obtained from the Gibbs sampler, obtain the corresponding correlation coefficient. From these values, plot posterior densities of the correlations p_blue and p_orange for the two groups. Evaluate differences between the two species by comparing these posterior distributions. In particular, obtain an approximation to Pr(p_blue < p_orange|yblue, yorange). What do the results suggest about differences between the two populations?

```
bluecrab.cor = apply(crab.mcmc$bluecrab$sigma, MARGIN = 3, FUN = function(covmat) {
  covmat[1, 2] / (sqrt(covmat[1, 1] * covmat[2, 2]))
})
orangecrab.cor = apply(crab.mcmc$orangecrab$sigma, MARGIN = 3, FUN = function(covmat) {
  covmat[1, 2] / (sqrt(covmat[1, 1] * covmat[2, 2]))
})
cor.df = data.frame(species = c(rep('blue', length(bluecrab.cor)), rep('orange', length(orangecrab.cor)),
                    cor = c(bluecrab.cor, orangecrab.cor))

ggplot(cor.df, aes(x = cor, fill = species)) +
  geom_density(alpha = 0.5) +
  scale_fill_manual(values = c('blue', 'orange'))
```




```
mean(orangecrab.cor > bluecrab.cor)
```

```
## [1] 0.9915
```

The Plot of the posterior densities of correlation coefficients for orange and blue crabs shows that on average, the correlation coefficient is higher for orange crabs than blue crabs, for both measurements of theta 1 and theta 2. Through our calculation of Pr(p_blue < p_orange|yblue, yorange) = 0.9894, we can see that the probability that the correlation coefficient of orange crabs is greater than the correlation coefficient of blue crabs is 98.94%, which is significance at an alpha level of 0.05.