Bios 6301: Assignment 3

Erin Fey

Grade 50/50

Due Tuesday, 11 October, 1:00 PM

50 points total.

$5^{n=day}$ points taken off for each day late.

This assigment includes turning in the first two assignments. All three should include knitr files (named `homework1.rmd`, `homework2.rmd`, `homework3.rmd`) along with valid PDF output files. Inside each file, clearly indicate which parts of your responses go with which problems (you may use the original homework document as a template). Add your name as `author` to the file's metadata section. Raw R code/output or word processor files are not acceptable.

Failure to properly name files or include author name may result in 5 points taken off.

Question 1

10 points

1. Use GitHub to turn in the first three homework assignments. Make sure the teacher (couthcommander) and TA (chipmanj) are collaborators. (5 points)
2. Commit each assignment individually. This means your repository should have at least three commits. (5 points)

Question 2

15 points

Write a simulation to calculate the power for the following study design. The study has two variables, treatment group and outcome. There are two treatment groups (0, 1) and they should be assigned randomly with equal probability. The outcome should be a random normal variable with a mean of 60 and standard deviation of 20. If a patient is in the treatment group, add 5 to the outcome. 5 is the true treatment effect. Create a linear model for the outcome by the treatment group, and extract the p-value (hint: see assigment1). Test if the p-value is less than or equal to the alpha level, which should be set to 0.05.

Repeat this procedure 1000 times. The power is calculated by finding the percentage of times the p-value is less than or equal to the alpha level. Use the `set.seed` command so that the professor can reproduce your results.

1. Find the power when the sample size is 100 patients. (10 points)

```r
set.seed(100)
nperson=100
group <- c(1:nperson)
nsim <- 1000
pvals <- numeric(nsim)
for (i in seq_along(pvals)) {
  treat <- rbinom(nperson, size = 1, prob = 0.5)
  outcome <- rnorm(nperson, mean = 60, sd = 20)
  outcome <- ifelse(treat[group] == 1, outcome[group]+5, outcome[group])
  pvals[i] <- summary(lm(outcome ~ treat))$coefficients[2,4]
```

```
}
mean(pvals < 0.05)
```

```
## [1] 0.236
```

1. Find the power when the sample size is 1000 patients. (5 points)

```
set.seed(1000)
nperson=1000
group <- c(1:nperson)
nsim <- 1000
pvals <- numeric(nsim)
for (i in seq_along(pvals)) {
  treat <- rbinom(nperson, size = 1, prob = 0.5)
  outcome <- rnorm(nperson, mean = 60, sd = 20)
  outcome <- ifelse(treat[group] == 1, outcome[group]+5, outcome[group])
  pvals[i] <- summary(lm(outcome ~ treat))$coefficients[2,4]
}
mean(pvals < 0.05)
```

```
## [1] 0.968
```

Question 3

15 points

Obtain a copy of the football-values lecture. Save the `2016/proj_wr16.csv` file in your working directory. Read in the data set and remove the first two columns.

```
#football <- read.csv("/Users/erinfey/Desktop/proj_wr16.csv")
football <- read.csv("proj_wr16.csv")
football <- football[,-(1:2)]
```

1. Show the correlation matrix of this data set. (3 points)

```
cor(football)
```

```
##           rush_att  rush_yds   rush_tds    rec_att    rec_yds    rec_tds
## rush_att 1.0000000 0.9906030 0.88608205 0.19706851 0.14473723 0.13548999
## rush_yds 0.9906030 1.0000000 0.91252627 0.18745520 0.13765791 0.12772327
## rush_tds 0.8860820 0.9125263 1.00000000 0.06914613 0.03114206 0.03163468
## rec_att  0.1970685 0.1874552 0.06914613 1.00000000 0.99002712 0.96757796
## rec_yds  0.1447372 0.1376579 0.03114206 0.99002712 1.00000000 0.98209522
## rec_tds  0.1354900 0.1277233 0.03163468 0.96757796 0.98209522 1.00000000
## fumbles  0.1844220 0.1881021 0.10845675 0.43577978 0.40349289 0.35852435
## fpts     0.1766540 0.1698501 0.06567865 0.98754942 0.99760259 0.99058639
##            fumbles       fpts
## rush_att 0.1844220 0.17665405
## rush_yds 0.1881021 0.16985010
## rush_tds 0.1084568 0.06567865
## rec_att  0.4357798 0.98754942
## rec_yds  0.4034929 0.99760259
## rec_tds  0.3585244 0.99058639
## fumbles  1.0000000 0.38269698
## fpts     0.3826970 1.00000000
```

2. Generate a data set with 30 rows that has a similar correlation structure. Repeat the procedure 10,000 times and return the mean correlation matrix. (10 points)

```
library(MASS)
times <- 10000
football2 <- 0
for (i in seq(times)) {
  corr2 <- mvrnorm(n = 30, mu = colMeans(football), Sigma = var(football))
  football2 <- football2 + cor(corr2)/times
}
football2
```

```
##          rush_att  rush_yds   rush_tds    rec_att   rec_yds    rec_tds
## rush_att 1.0000000 0.9902341 0.88241167 0.19362908 0.1418744 0.13289492
## rush_yds 0.9902341 1.0000000 0.90964176 0.18401614 0.1348112 0.12530075
## rush_tds 0.8824117 0.9096418 1.00000000 0.06921105 0.0315649 0.03213981
## rec_att  0.1936291 0.1840161 0.06921105 1.00000000 0.9896449 0.96653342
## rec_yds  0.1418744 0.1348112 0.03156490 0.98964486 1.0000000 0.98149861
## rec_tds  0.1328949 0.1253007 0.03213981 0.96653342 0.9814986 1.00000000
## fumbles  0.1827721 0.1861464 0.10810836 0.42843163 0.3965239 0.35290690
## fpts     0.1732943 0.1665474 0.06553721 0.98711766 0.9975202 0.99026166
##            fumbles       fpts
## rush_att 0.1827721 0.17329432
## rush_yds 0.1861464 0.16654738
## rush_tds 0.1081084 0.06553721
## rec_att  0.4284316 0.98711766
## rec_yds  0.3965239 0.99752024
## rec_tds  0.3529069 0.99026166
## fumbles  1.0000000 0.37620335
## fpts     0.3762033 1.00000000
```

3. Generate a data set with 30 rows that has the exact correlation structure as the original data set. (2 points)

```
football3 <- mvrnorm(n = 30, mu = colMeans(football), Sigma = var(football), empirical = TRUE)
football3
```

```
##         rush_att    rush_yds     rush_tds    rec_att    rec_yds    rec_tds
##  [1,]  0.0902384  -2.5279000 -0.107582111  33.910693  490.79343  2.9160060
##  [2,]  0.7144450   5.2670900  0.001398947   1.148729  -68.16764 -1.5477117
##  [3,] -1.5895402 -12.9731850 -0.132810712 -18.267059 -259.85482 -1.7211184
##  [4,]  1.1596093   7.2007924  0.050096499  53.420898  730.83436  4.4521102
##  [5,]  0.5428394   2.5272232  0.067319360  23.453523  291.91091  1.4480400
##  [6,]  1.3916603   8.9840435  0.002428525  93.723402 1259.76526  8.7506686
##  [7,] -0.4325914  -4.4767024 -0.025170590  50.138344  680.24962  4.1762852
##  [8,] -2.0470856 -11.3418416 -0.118264798  25.376518  355.17863  2.0892143
##  [9,] -0.0299001   0.5452748 -0.002783108  34.371500  448.62104  3.2707365
## [10,]  1.6258796   7.3130072  0.043292099  30.338740  452.59934  2.4451678
## [11,] -0.5650389  -2.7539599 -0.043817353 -17.658570 -262.87905 -1.7145282
## [12,] -2.8308791 -17.7324867 -0.135922327  19.591514  206.13758  1.7699021
## [13,] -4.9820483 -29.5885354 -0.216185560   3.463429  123.84535 -0.2405928
## [14,] -0.5047502  -7.6483948 -0.054746489  10.019537  114.07255  0.8220655
## [15,]  3.1780039  18.4299745 -0.009231437  66.252195  831.57444  5.1327121
## [16,]  4.6095103  27.9581474  0.204770763 -19.713518 -267.85247 -2.1445425
## [17,]  1.0183495   8.2870049  0.064424007  25.127385  272.60994  1.9790730
## [18,]  0.4007560   2.1264452  0.010025785  41.006988  535.03141  3.3499405
```

```
## [19,]  3.3453523  16.9169392  0.111894805  48.935775  679.66640  4.9488463
## [20,] -0.7806804  -3.4029450  0.045711896  17.783214  195.65005  0.5637421
## [21,]  3.6028887  21.6036590  0.122117413 -11.534493 -133.76197 -0.8869267
## [22,]  0.2911036   1.5598782 -0.022372747  28.646089  366.45322  1.3649941
## [23,]  6.8309906  42.4378326  0.257080356  18.595965  136.92789  0.4613699
## [24,]  1.7185189   7.9047065 -0.013366703  61.972725  774.61986  5.3237353
## [25,]  2.8093186  15.5322028  0.072873403  70.858678  879.41881  5.7519236
## [26,]  1.5685811  10.6742855  0.081289239  52.877867  662.64948  4.4387853
## [27,] -0.9770863  -5.4688067 -0.034577501  39.109833  627.98777  4.9465050
## [28,] -0.5490212   0.6665364  0.035892366   9.492728  234.93380  2.3393889
## [29,] -0.7216516  -5.6088100  0.003264171  22.701449  262.47073  1.9099786
## [30,]  1.4825981  11.6749442  0.088630816  48.547282  696.02024  3.8758341
##           fumbles       fpts
##  [1,] -0.53126648  66.682964
##  [2,] -0.10188757 -15.512749
##  [3,] -0.09552139 -38.129392
##  [4,]  0.71226053  99.338770
##  [5,]  0.06618022  38.616461
##  [6,]  0.85010875 177.611939
##  [7,]  0.86745994  90.448330
##  [8,]  0.36730013  45.880148
##  [9,]  0.45702326  63.679003
## [10,]  0.55650967  59.544051
## [11,]  0.37936140 -38.284476
## [12,]  0.73568828  26.955039
## [13,]  0.16866436   6.470755
## [14,]  0.24591560  14.814932
## [15,]  0.61969895 114.682787
## [16,]  0.63178066 -36.755357
## [17,]  0.03767150  40.543898
## [18,] -0.01170495  73.808054
## [19,] -0.38532503 100.730544
## [20,]  0.15182409  22.719136
## [21,]  0.19929955 -16.078879
## [22,]  0.68206000  43.613208
## [23,]  0.34564556  21.521587
## [24,]  0.74600703 108.821120
## [25,]  1.24639543 122.143258
## [26,]  0.35710857  93.561654
## [27,]  0.26744305  91.039124
## [28,] -0.33702599  38.375292
## [29,] -0.15184107  37.531051
## [30,]  0.72563511  93.245033
```

Question 4

10 points

Use LaTeXto create the following expressions.

1. Hint: `\Rightarrow` (4 points)

$$
\begin{aligned}
P(B) &= \sum_j P(B|A_j)P(A_j), \\
&\Rightarrow p(A_i|B) = \frac{P(B|A_i)P(A_i)}{\sum_j P(B|A_j)P(A_j)}
\end{aligned}
$$

2. Hint: `\zeta` (3 points)

$$
\hat{f}(\zeta) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i x \zeta} dx \tag{1}
$$

3. Hint: `\partial` (3 points)

$$
\mathbf{J} = \frac{d\mathbf{f}}{d\mathbf{x}} = [\frac{\partial \mathbf{f}}{\partial x_1} \cdots \frac{\partial \mathbf{f}}{\partial x_n}] = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix} \tag{2}
$$