Assignment 3

Ya-Chen Lin

2015 M09 23

Question 02:

Write a simulation to calculate the power for the following study design.

Find the power when the sample size is 100 patients.

```
set.seed(1)
#Create a function with n=100 and return the pvalue
p<- function(){
treatment <- matrix(rbinom(n=100, size=1, prob=1/2))
outcome <- matrix(rnorm(100, mean=60, sd=20))
study <- cbind(treatment, outcome)
colnames(study) <- c('treatment' ,'outcome')
study <- data.frame(study)
for(i in 1:100){
  if(study[i,1] == 1){
    study[i,2] <- study[i,2] + 5
  }
}
mod.study <- lm(outcome ~ treatment, data=study)
pvalue <- summary(mod.study)$coefficients['treatment',4]
return(pvalue)
}
#Simulate this function for 1000 times
loops <- 1000
pb <- c()
ps <- 0
for(i in seq(loops)){
  pb[i] <- p()
}
power100 <- sum(pb[] <= 0.05) *100 / loops
power100
```

```
## [1] 28.1
```

Find the power when the sample size is 1000 patients.

```
set.seed(2)
#Create a function with n=1000 and return pvalue
p<- function(){
treatment <- matrix(rbinom(n=1000, size=1, prob=1/2))
outcome <- matrix(rnorm(1000, mean=60, sd=20))
study <- cbind(treatment, outcome)
colnames(study) <- c('treatment' ,'outcome')
study <- data.frame(study)
for(i in 1:1000){
```

```
  if(study[i,1] == 1){
    study[i,2] <- study[i,2] + 5
  }
}
mod.study <- lm(outcome ~ treatment, data=study)
pvalue <- summary(mod.study)$coefficients['treatment',4]
return(pvalue)
}
#Simulate this function for 1000 times
loops <- 1000
pb <- c()
ps <- 0
for(i in seq(loops)){
  pb[i] <- p()
}
power1000 <- sum(pb[] <= 0.05) *100 / loops
power1000
```

```
## [1] 97.5
```

Question 03:

Obtain a copy of the football-values lecture.

```
rb <- read.csv('https://raw.githubusercontent.com/couthcommander/football-values/master/2015/proj_rb15.
rb <- rb[, 3:10]
```

1. Show the correlation matrix of this data set.

```
cor(rb)
```

```
##           rush_att  rush_yds  rush_tds   rec_att   rec_yds   rec_tds
## rush_att 1.0000000 0.9975511 0.9723599 0.7694384 0.7402687 0.5969159
## rush_yds 0.9975511 1.0000000 0.9774974 0.7645768 0.7345496 0.6020994
## rush_tds 0.9723599 0.9774974 1.0000000 0.7263519 0.6984860 0.5908348
## rec_att  0.7694384 0.7645768 0.7263519 1.0000000 0.9944243 0.8384359
## rec_yds  0.7402687 0.7345496 0.6984860 0.9944243 1.0000000 0.8518924
## rec_tds  0.5969159 0.6020994 0.5908348 0.8384359 0.8518924 1.0000000
## fumbles  0.8589364 0.8583243 0.8526904 0.7459076 0.7224865 0.6055598
## fpts     0.9824135 0.9843044 0.9689472 0.8556928 0.8340195 0.7133908
##            fumbles      fpts
## rush_att 0.8589364 0.9824135
## rush_yds 0.8583243 0.9843044
## rush_tds 0.8526904 0.9689472
## rec_att  0.7459076 0.8556928
## rec_yds  0.7224865 0.8340195
## rec_tds  0.6055598 0.7133908
## fumbles  1.0000000 0.8635550
## fpts     0.8635550 1.0000000
```

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.

```
set.seed(3)
rb <- read.csv('https://raw.githubusercontent.com/couthcommander/football-values/master/2015/proj_rb15.
rb <- rb[, 3:10]
means.rb <- colMeans(rb)
vcov.rb <- var(rb)
library(MASS)
rb.sim.c <- 0
loops <- 10000
for ( i in seq(loops)){
rb.sim <- mvrnorm (30, mu=means.rb, Sigma = vcov.rb)
rb.sim <- data.frame(rb.sim)
rb.sim.c <- rb.sim.c + (cor(rb.sim) / loops)
}
rb.sim.c
```

```
##           rush_att  rush_yds  rush_tds   rec_att   rec_yds   rec_tds
## rush_att 1.0000000 0.9974738 0.9714690 0.7635276 0.7338655 0.5896312
## rush_yds 0.9974738 1.0000000 0.9767571 0.7585406 0.7280285 0.5946157
## rush_tds 0.9714690 0.9767571 1.0000000 0.7199586 0.6916975 0.5830992
## rec_att  0.7635276 0.7585406 0.7199586 1.0000000 0.9942040 0.8337476
## rec_yds  0.7338655 0.7280285 0.6916975 0.9942040 1.0000000 0.8475462
## rec_tds  0.5896312 0.5946157 0.5830992 0.8337476 0.8475462 1.0000000
## fumbles  0.8548191 0.8542738 0.8483354 0.7388825 0.7150244 0.5972068
## fpts     0.9818372 0.9837567 0.9679101 0.8512292 0.8289990 0.7064388
##            fumbles      fpts
## rush_att 0.8548191 0.9818372
## rush_yds 0.8542738 0.9837567
## rush_tds 0.8483354 0.9679101
## rec_att  0.7388825 0.8512292
## rec_yds  0.7150244 0.8289990
## rec_tds  0.5972068 0.7064388
## fumbles  1.0000000 0.8592837
## fpts     0.8592837 1.0000000
```

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

```
rbn <- mvrnorm(30, mu=means.rb, Sigma = vcov.rb, empirical = TRUE)
cor(rb)
```

```
##           rush_att  rush_yds  rush_tds   rec_att   rec_yds   rec_tds
## rush_att 1.0000000 0.9975511 0.9723599 0.7694384 0.7402687 0.5969159
## rush_yds 0.9975511 1.0000000 0.9774974 0.7645768 0.7345496 0.6020994
## rush_tds 0.9723599 0.9774974 1.0000000 0.7263519 0.6984860 0.5908348
## rec_att  0.7694384 0.7645768 0.7263519 1.0000000 0.9944243 0.8384359
## rec_yds  0.7402687 0.7345496 0.6984860 0.9944243 1.0000000 0.8518924
## rec_tds  0.5969159 0.6020994 0.5908348 0.8384359 0.8518924 1.0000000
## fumbles  0.8589364 0.8583243 0.8526904 0.7459076 0.7224865 0.6055598
## fpts     0.9824135 0.9843044 0.9689472 0.8556928 0.8340195 0.7133908
##            fumbles      fpts
## rush_att 0.8589364 0.9824135
```

```
## rush_yds 0.8583243 0.9843044
## rush_tds 0.8526904 0.9689472
## rec_att  0.7459076 0.8556928
## rec_yds  0.7224865 0.8340195
## rec_tds  0.6055598 0.7133908
## fumbles  1.0000000 0.8635550
## fpts     0.8635550 1.0000000
```

```
cor(rbn)
```

```
##           rush_att  rush_yds  rush_tds   rec_att   rec_yds   rec_tds
## rush_att 1.0000000 0.9975511 0.9723599 0.7694384 0.7402687 0.5969159
## rush_yds 0.9975511 1.0000000 0.9774974 0.7645768 0.7345496 0.6020994
## rush_tds 0.9723599 0.9774974 1.0000000 0.7263519 0.6984860 0.5908348
## rec_att  0.7694384 0.7645768 0.7263519 1.0000000 0.9944243 0.8384359
## rec_yds  0.7402687 0.7345496 0.6984860 0.9944243 1.0000000 0.8518924
## rec_tds  0.5969159 0.6020994 0.5908348 0.8384359 0.8518924 1.0000000
## fumbles  0.8589364 0.8583243 0.8526904 0.7459076 0.7224865 0.6055598
## fpts     0.9824135 0.9843044 0.9689472 0.8556928 0.8340195 0.7133908
##            fumbles      fpts
## rush_att 0.8589364 0.9824135
## rush_yds 0.8583243 0.9843044
## rush_tds 0.8526904 0.9689472
## rec_att  0.7459076 0.8556928
## rec_yds  0.7224865 0.8340195
## rec_tds  0.6055598 0.7133908
## fumbles  1.0000000 0.8635550
## fpts     0.8635550 1.0000000
```

With empirical = TRUE, we can get data with exact same correlation

Question 04:

1.

```
#\begin{align}
#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_i)}
#\end{align}
```

$$P(B) = \sum_{j} P(B|A_j)P(A_j), \tag{1}$$

$$\Rightarrow P(A_i|B)\frac{P(B|A_i)P(A_i)}{\sum_{j} P(B|A_j)P(A_i)} \tag{2}$$

2.

```
#$\hat{f}(\zeta)=\displaystyle\int_{-\infty}^{\infty} f(x) e^{-2{\pi}ix{\zeta}} dx$
```

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

3.

```
#$\mathbf{J}=\frac{d\mathbf{f}}{d\mathbf{x}}=\left[\frac{\partial \mathbf{f}}{\partial x_1}\cdots\frac{
```

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