Stat 505 Assignment 7 Solutions

1. In basketball, a player often gets to shoot a one-and-one meaning if the first free throw goes in, he/she gets to shoot another. Assume the two shots are independent, and that different players have different probabilities of making their free throws, p (assumed constant for any individual). You can look at players stats on wnba.com or nba.com to see what personal career averages on FT's are. I see numbers ranging from 0.59 to 0.90.

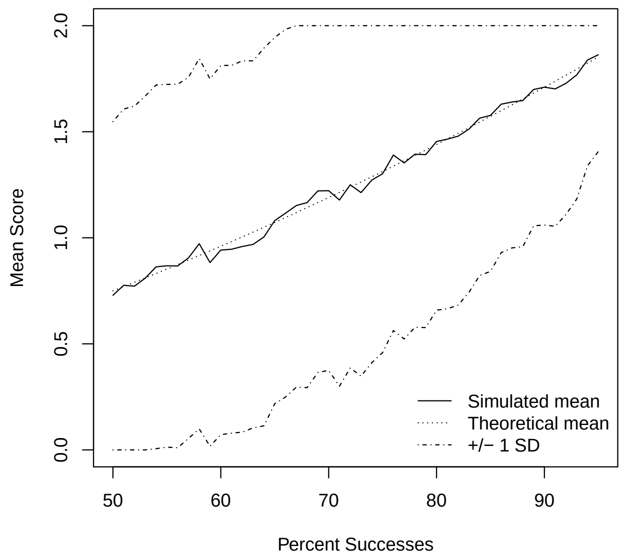
   (a) Build an R function to simulate the process of shooting a one and one. It should take as input the player's probability of making a free throw. Output: 0, 1, or 2 points (successes). Show the function and table the output of 100 runs with $p = .5$.

```
> oneANDone <- function(p, nsim) {
      shot1 <- rbinom(nsim, 1, p)
      shot2 <- ifelse(shot1 == 1, rbinom(nsim, 1, p), 0)
      shot1 + shot2
  }
> table(oneANDone(0.5, 100))
```

```
 0  1  2
43 27 30
```

   (b) For each integer percentage from 50% to 95%, run your simulation 1000 times and store the scores (0, 1, or 2 pts). Plot the average score against the percentage.

```
> mean.p <- sd.p <- rep(NA, 46)
> for (p in 50:95) {
      tt <- oneANDone(p/100, 1000)
      mean.p[p - 49] <- mean(tt)
      sd.p[p - 49] <- sd(tt)
  }
> plot(50:95, mean.p, type = "l", ylim = c(0, 2), xlab = "Percent Successes",
      ylab = "Mean Score")
> curve(x/100 + x^2/10000, add = TRUE, lty = 3)
> lines(50:95, pmax(0, mean.p - sd.p), lty = 4)
> lines(50:95, pmin(2, mean.p + sd.p), lty = 4)
> legend("bottomright", lty = c(1, 3, 4), c("Simulated mean",
      "Theoretical mean", "+/- 1 SD"), bty = "n")
```



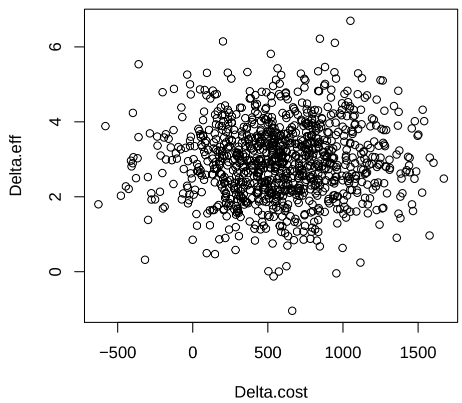
(c) Add “error bands” to encompass plus/minus 1 standard deviation from the mean.

(d) Use conditional probability to find the expected value of the number of points for a 1-and-1 as a function of p, the probability of success. Add this curve to your plot with a different line type. Include a legend.
We get 0 only if the first shot is missed, an event with probability $1-p$. *One occurs when the first is hit and the second missed,* $P(1) = p(1-p)$, *and 2 occurs with probability* p^2. *The expected score is* $0 \times (1-p) + 1 \times p(1-p) + 2 \times p^2 = p(1+p)$.

2. Exercise 8 p 152-3
Cost vs treatment: $\widehat{\Delta}_{\text{cost}} = \overline{y}_A - \overline{y}_B = 600$ with SE 400
Effectiveness: $\widehat{\Delta}_{\text{eff}} = \overline{z}_A - \overline{z}_B = 3.0$ with SE 1
Need to simulate $\Delta_{\text{cost}}/\Delta_{\text{eff}}$.

(a) 1000 draws of each.

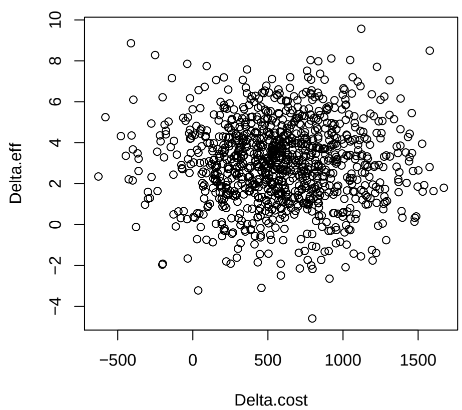
```
> sigma.cost <- 400 * sqrt(50/rchisq(1000, 50))
> sigma.eff <- 1 * sqrt(100/rchisq(1000, 100))
> Delta.cost <- rnorm(1000, 600, sigma.cost)
> Delta.eff <- rnorm(1000, 3, sigma.eff)
> plot(Delta.cost, Delta.eff)
> ratio1 <- Delta.cost/Delta.eff
```



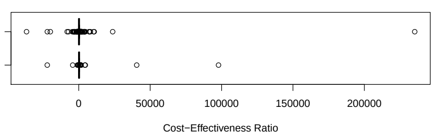
(b) A 50% credible (or confidence) interval for $\Delta_{\text{cost}}/\Delta_{\text{eff}}$ is (111.2, 309.2)
And a 95% credible (or confidence) interval is (-77.9, 723.9)

(c) Now let SE for effectiveness = 2

```
> sigma.eff ← 2 * sqrt(100/rchisq(1000, 100))
> Delta.eff ← rnorm(1000, 3, sigma.eff)
> plot(Delta.cost, Delta.eff)
> ratio2 ← Delta.cost/Delta.eff
```



A 50% credible (or confidence) interval for $\Delta_{\text{cost}}/\Delta_{\text{eff}}$ is (73.4, 297.4)
A 95% credible (or confidence) interval is (-1022, 1771.1) The 50% intervals are similar to those above, but the 95% intervals show the effect of having more Δ_{eff} near zero which throw the ratios out to extremes. I'll do side-by-side boxplots for comparison.



R Code

```
### code chunk number 1: options
###################################################
options(continue=" ",width=76)

### code chunk number 2: oneANDone
###################################################
oneANDone <- function(p,nsim) {
  shot1 <- rbinom(nsim,1,p)
  shot2 <- ifelse(shot1==1, rbinom(nsim,1,p),0)
  shot1+shot2}
table(oneANDone(.5,100))

### code chunk number 3: plot1n1
###################################################
mean.p <- sd.p <- rep(NA,46)
 for(p in 50:95){
   tt <- oneANDone(p/100,1000)
   mean.p[p-49] <- mean(tt)
   sd.p[p-49] <- sd(tt)
 }
 plot(50:95,mean.p,type="l",ylim=c(0,2), xlab="Percent Successes",ylab="Mean Score")
 curve(x/100+x^2/10000,add=TRUE,lty=3)
 lines(50:95,pmax(0,mean.p - sd.p),lty=4)
 lines(50:95,pmin(2,mean.p + sd.p),lty=4)
 legend("bottomright", lty=c(1,3,4), c("Simulated mean","Theoretical mean","+/- 1 SD"), bty="n")

### code chunk number 4: draw1000
###################################################
sigma.cost <- 400*sqrt(50/rchisq(1000,50))
sigma.eff <- 1*sqrt(100/rchisq(1000,100))
Delta.cost <- rnorm(1000,600,sigma.cost)
Delta.eff <- rnorm(1000, 3,sigma.eff)
plot(Delta.cost, Delta.eff)
ratio1 <- Delta.cost/Delta.eff

### code chunk number 5: draw2
###################################################
sigma.eff <- 2*sqrt(100/rchisq(1000,100))
Delta.eff <- rnorm(1000, 3,sigma.eff)
plot(Delta.cost, Delta.eff)
ratio2 <- Delta.cost/Delta.eff

### code chunk number 6: boxplot
###################################################
 boxplot(ratio1,ratio2, horizontal=TRUE,xlab="Cost-Effectiveness Ratio")
```