Problem Set 1

Group members: Xuchen Liu, Binqian Chai, He Jiang, Ella Lin

- 1.) The R command cor(x,y) computes the sample correlation between vectors x and y. Using the data pickup.csv,
- a.) Compute the correlation between miles and price in this sample.

```
pickup <- read.csv("./data/pickup.csv")
cor(pickup$miles, pickup$price)
```

```
## [1] -0.6458428
```

- b.) Suppose this data was formed by randomly taking a small sample from all of the Craigslist listings available at a particular time. The sample correlation in part (a) is an estimate of the correlation in this larger population. Form a 95% CI for this population correlation using the bootstrap (you can use either of the methods from class).

```
set.seed(0)
n <- 46
B <- 1000
sample <- data.frame(miles=pickup$miles, price=pickup$price)
bootstrap_cors = rep(0, B)
for (b in 1:B) {
  boot_sample_indices <- sample.int(n, replace=TRUE)
  bootsample_miles <- sample$miles[boot_sample_indices]
  bootsample_price <- sample$price[boot_sample_indices]
  bootstrap_cors[b] = cor(bootsample_miles,bootsample_price)
}
cor(sample$miles, sample$price) + c(-1,1)*quantile(abs(bootstrap_cors-cor(sample$miles, sample$price)),0.95)
```

```
## [1] -0.7940271 -0.4976585
```

- 2.) Recall the basic (no interactions) regression specification for the orange juice data from the lec03-regression.nb.html notes:

$$log(sales_i) = \beta_1 + \log(price_i) \cdot \beta_2 + minutemaid_i \cdot \beta_3 + tropicana_i \cdot \beta_4 + \varepsilon_i$$

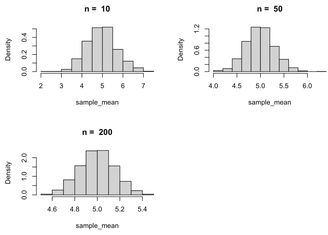
  In the lecture notes, we computed the an estimate of the percent increase in sales of Tropicana relative to the baseline (Dominick's brand) for a given price, using the formula (exp(β̂4) − 1) · 100. In this exercise we will form a 95% CI for this quantity using the bootstrap. For each bootstrap replication, compute the boostrap version of the estimate (exp(β̂4) − 1) · 100. Use these bootstrap replications to form a 95% CI (you can use either of the methods from class). As with the previous exercise, be sure to use the same resampled observations for the whole data frame in each bootstrap replication.

```
set.seed(0)
orange <- read.csv("./data/oj.csv")
sample2 <- data.frame(sales=orange$sales, price=orange$price, brand=orange$brand)
sample2$brand <- factor(sample2$brand)
OLS_model <- glm(log(sales) ~ log(price) + brand, data=sample2)
esti <- (exp(coef(OLS_model)['brandtropicana']) - 1) * 100
n <- 28947
B <- 1000
bootstrap_estis = rep(0, B)
for (b in 1:B) {
  boot_sample_indices2 <- sample.int(n, replace=TRUE)
  boot_sample2 <- data.frame(sales=sample2$sales[boot_sample_indices2], price=sample2$price[boot_sample_indices
2], brand=sample2$brand[boot_sample_indices2])
  boot_sample2$brand <- factor(boot_sample2$brand)
  OLS_model <- glm(log(sales) ~ log(price) + brand, data=boot_sample2)
  bootstrap_estis[b] = (exp(coef(OLS_model)['brandtropicana']) - 1) * 100
}
esti + c(-1,1)*quantile(abs(bootstrap_estis-esti),0.95)
```

```
## [1] 345.8152 377.7673
```

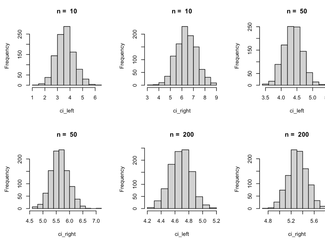
- 3.) We will perform a Monte Carlo (i.e. simulation) analysis of confidence intervals (CIs) for the mean. Recall that the CIs we constructed in class are supposed to contain the true parameter about 95% of the time over repeated samples. Since we already covered the exponential distribution in class, we'll simulate from a different distribution: the Poisson distribution. Note that the Poisson distribution is a discrete distribution: a Poisson random variable takes on possible values 0,1,2,..., rather than taking on any positive real number as with the exponential distribution. The Poisson distribution is indexed by a parameter λ, which is equal to the mean of the distribution. For n = 10, 50, 200 and B = 1000, run the following simulation exercise. Draw a sample of size n from the Poisson distribution with λ = 5 (rpois(n, lambda=5) in R). Do this B times, each time computing the sample mean and standard error, and the endpoints of the 95% CI from the lecture notes (the one that adds and subtracts two times the standard error).
- a.) Report a histogram of the sample mean over the B simulations (3 histograms, one for each sample size n).

```
set.seed(0)
par(mfrow=c(2,2))
n_size <- c(10,50,200)
B <- 1000
sample_mean <- rep(0, B)
sample_se <- rep(0, B)
ci_left <- rep(0, B)
ci_right <- rep(0, B)
for (n in n_size) {
  for (b in 1:B) {
    each <- rpois(n, lambda=5)
    sample_mean[b] <- mean(each)
    sample_se[b] <- (sd(each)/sqrt(n))
    ci_left[b] <- (mean(each)-2*(sd(each)/sqrt(n)))
    ci_right[b] <- (mean(each)+2*(sd(each)/sqrt(n)))
  }
  hist(sample_mean, freq=FALSE, main=paste("n = ", n))
}
```



- b.) Report a histogram of the lower endpoint of the 95% CI, and another histogram of the upper endpoint of the 95% CI (6 total histograms).

```
set.seed(0)
par(mfrow=c(2,3))
n_size <- c(10,50,200)
B <- 1000
sample_mean <- rep(0, B)
sample_se <- rep(0, B)
ci_left <- rep(0, B)
ci_right <- rep(0, B)
for (n in n_size) {
  for (b in 1:B) {
    each <- rpois(n, lambda=5)
    sample_mean[b] <- mean(each)
    sample_se[b] <- (sd(each)/sqrt(n))
    ci_left[b] <- (mean(each)-2*(sd(each)/sqrt(n)))
    ci_right[b] <- (mean(each)+2*(sd(each)/sqrt(n)))
  }
  hist(ci_left, main=paste("n = ", n))
  hist(ci_right, main=paste("n = ", n))
}
```



- c.) For each n, report the proportion of the B CIs that contain the population mean. Is it close to 95%?

```
set.seed(0)
n_size <- c(10,50,200)
B <- 1000
sample_mean <- rep(0, B)
sample_se <- rep(0, B)
ci_left <- rep(0, B)
ci_right <- rep(0, B)
for (n in n_size) {
  count <- 0
  for (b in 1:B) {
    each <- rpois(n, lambda=5)
    sample_mean[b] <- mean(each)
    sample_se[b] <- (sd(each)/sqrt(n))
    ci_left[b] <- (mean(each)-2*(sd(each)/sqrt(n)))
    ci_right[b] <- (mean(each)+2*(sd(each)/sqrt(n)))
    if (ci_left[b] < 5 && ci_right[b] > 5){
      count = count + 1
    }
  }
  cat("For n = ", n, ", the proportion of the B CIs that contain the population mean is ", count/B*100, "%.\n", s
ep="")
}
```

```
## For n = 10, the proportion of the B CIs that contain the population mean is 91.7%.
## For n = 50, the proportion of the B CIs that contain the population mean is 95.8%.
## For n = 200, the proportion of the B CIs that contain the population mean is 95.2%.
```

The proportion for each n is close to 95%.

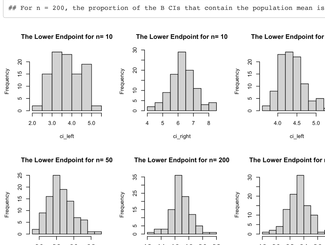
- d.) Bonus question: repeat parts (b) and (c) using a bootstrap CI (you can pick either of the bootstrap CIs we covered in class). This means that, for each simulated sample b = 1,...,B, you will draw B bootstrap samples from the simulated sample. Since this is computationally intensive (B^2 total bootstrap samples), try B = 100.

```
par(mfrow=c(2,3))
n_size <- c(10,50,200)
B <- 100
for (n in n_size) {
  set.seed(0)
  num <- 0
  ci_left <- rep(0, B)
  ci_right <- rep(0, B)
  for (a in 1:B) {
    sample3 <- rpois(n, lambda=5)
    sample3_mean <- mean(sample3)
    boot_mean <- rep(0, B)
    for (b in 1:B) {
      boot_sample_indices3 <- sample.int(n, replace=TRUE)
      boot_sample3 <- sample3[boot_sample_indices3]
      boot_mean[b] <- mean(boot_sample3)
    }
    ci_left[a] <- sample3_mean-2*sd(boot_mean)
    ci_right[a] <- sample3_mean+2*sd(boot_mean)
    if (ci_left[a] < 5 && ci_right[a] > 5){
      num = num + 1
    }
  }
  cat("For n = ", n, ", the proportion of the B CIs that contain the population mean is ", num/B*100, "%.\n", sep
="")
  hist(ci_left, main=paste("The Lower Endpoint for n=", n))
  hist(ci_right, main=paste("The Lower Endpoint for n=", n))
}
```

```
## For n = 10, the proportion of the B CIs that contain the population mean is 92%.
```

```
## For n = 50, the proportion of the B CIs that contain the population mean is 92%.
```

```
## For n = 200, the proportion of the B CIs that contain the population mean is 95%.
```



The proportion for each n is close to 95%.