Problem Set 1

Cianna Devitt (17321885)

Due: February 12, 2023

Instructions

- Please show your work! You may lose points by simply writing in the answer. If the problem requires you to execute commands in `R`, please include the code you used to get your answers. Please also include the `.R` file that contains your code. If you are not sure if work needs to be shown for a particular problem, please ask.
- Your homework should be submitted electronically on GitHub in `.pdf` form.
- This problem set is due before 23:59 on Sunday February 12, 2023. No late assignments will be accepted.

Question 1

The Kolmogorov-Smirnov test uses cumulative distribution statistics test the similarity of the empirical distribution of some observed data and a specified PDF, and serves as a goodness of fit test. The test statistic is created by:

$$D = \max_{i=1:n} \left\{ \frac{i}{n} - F_{(i)}, F_{(i)} - \frac{i-1}{n} \right\}$$

where F is the theoretical cumulative distribution of the distribution being tested and $F_{(i)}$ is the ith ordered value. Intuitively, the statistic takes the largest absolute difference between the two distribution functions across all x values. Large values indicate dissimilarity and the rejection of the hypothesis that the empirical distribution matches the queried theoretical distribution. The p-value is calculated from the Kolmogorov- Smirnoff CDF:

$$p(D \leq x)\frac{\sqrt{2\pi}}{x} \sum_{k=1}^{\infty} e^{-(2k-1)^2\pi^2/(8x^2)}$$

which generally requires approximation methods (see Marsaglia, Tsang, and Wang 2003). This so-called non-parametric test (this label comes from the fact that the distribution of the test statistic does not depend on the distribution of the data being tested) performs poorly in small samples, but works well in a simulation environment. Write an `R` function that implements this test where the reference distribution is normal. Using `R` generate 1,000 Cauchy random variables (`rcauchy(1000, location = 0, scale = 1)`) and perform the test (remember, use the same seed, something like `set.seed(123)`, whenever you're generating your own data).

As a hint, you can create the empirical distribution and theoretical CDF using this code:

```
# create empirical distribution of observed data
ECDF <- ecdf(data)
empiricalCDF <- ECDF(data)
# generate test statistic
D <- max(abs(empiricalCDF - pnorm(data)))
```

```
# Creating data

set.seed(123)
var <- rcauchy(1000, location = 0, scale = 1)

# New Function

KS_TEST <- function(data){
  ECDF <- ecdf(data)
  empiricalCDF <- ECDF(data)
  D <- max(abs(empiricalCDF - pnorm(data)))
  print('One sample Kolmogorov-Smirnoff Test')
  print(paste('Test Statistic: ', D))
}

KS_TEST(var)


> KS_TEST(var)
[1] One sample Kolmogorov-Smirnoff Test
     Test Statistic:  0.13472806160635

```

Question 2

Estimate an OLS regression in `R` that uses the Newton-Raphson algorithm (specifically `BFGS`, which is a quasi-Newton method), and show that you get the equivalent results to using `lm`. Use the code below to create your data.

```
set.seed(123)
data <- data.frame(x = runif(200, 1, 10))
data$y <- 0 + 2.75*data$x +rnorm(200, 0, 1.5)

# Creating a linear liklihood function
linear.lik <- function(theta, y, X){
  n <- nrow(X)
```

```r
    k <- ncol(X)
    beta <- theta[1:k]
    sigma2 <- theta[k +1]^2
    e <- y - X%*%beta
    logl <- -.5*n*log(2*pi) -.5*n*log(sigma2)- ( (t(e) %*%
    e)/ (2*sigma2))
    return(-logl)
  }

%   # Using optim() function and BFGS method, running a Maximum Liklihood
Estimation
  linear.MLE <- optim(fn=linear.lik, par = c(1,1,1), hessian = TRUE, y=
data$y, X=cbind(1, data$x), method = "BFGS")
  linear.MLE$par

  > linear.MLE$par
  [1]  0.1398324  2.7265559 -1.4390716
```

Maximum Liklihood Estimates : β_0 : 0.1398324 β_1 : 2.726559 σ : -1.4390716

```r
summary(lm(y~x, data))

Call:
lm(formula = y ~ x, data = data)

Residuals:
Min        1Q   Median        3Q       Max
-3.1906  -0.9374  -0.1665  0.8931  4.8032

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.13919    0.25276   0.551    0.582
x            2.72670    0.04159  65.564   <2e-16 ***
---

Residual standard error: 1.447 on 198 degrees of freedom
Multiple R-squared:  0.956,  Adjusted R-squared:  0.9557
F-statistic:  4299 on 1 and 198 DF,  p-value: < 2.2e-16
```

Regression with lm() β_0 : 0.139, β_1 : 2.726, σ : 1.447