Lab 4: Logistic Regression (Part 2)

<u>To submit your work, insert screenshots of your code and outputs (both numeric outputs and graphs) under respective problem prompts. Many steps also require a written answer, and you should insert your written or typed answer below the prompt.</u>

Suppose you are investigating allegations of gender discrimination in the hiring practices of a particular firm. An equal-rights group claims that females are less likely to be hired than males with the same background, experience, and other qualifications. You collected data on 28 former applicants. The variables in the dataset include:

- *HIRE (1 = hired, 0 = not hired)*
- *Years of higher education (EDUC)*
- *Years of work experience (EXP)*
- *GENDER (1 = male, 0 = female).*

1. Download the data file “DISCRIM.csv” from Canvas.

   DISCRIM.csv 11/18/2023 9:00 PM

2. Start R or R Studio. Load the “car” package.

```
rm(list=ls())
setwd("C:/Users/jyang/OneDrive - Arizona State University/10 Classes_OneDrive/2023_STP530_Regression/R")
library(car)
# install.packages("caret")
library(ggplot2)
library(lattice)
library(caret)
# install.packages("OptimalCutpoints")
library(OptimalCutpoints)
```

```
> library(car)
Loading required package: carData
> # install.packages("caret")
> library(ggplot2)
> library(lattice)
> library(caret)
> # install.packages("OptimalCutpoints")
> library(OptimalCutpoints)
```

3. Import the data into R. Name the imported data **hire.data**. View the data and make sure the data have been imported correctly.

```
# Load data
hire.data <- read.csv("DISCRIM.csv")

# Check data
head(hire.data )
str(hire.data )
```

```
> # Load data
> hire.data <- read.csv("DISCRIM.csv")
>
> # Check data
> head(hire.data )
  HIRE EDUC EXP GENDER
1    0    6   2      0
2    0    4   0      1
3    1    6   6      1
4    1    6   3      1
5    0    4   1      0
6    1    8   3      0
> str(hire.data )
'data.frame':	28 obs. of  4 variables:
 $ HIRE  : int  0 0 1 1 0 1 0 0 0 1 ...
 $ EDUC  : int  6 4 6 6 4 8 4 4 6 8 ...
 $ EXP   : int  2 0 6 3 1 3 2 4 1 10 ...
 $ GENDER: int  0 1 1 1 0 0 1 0 0 0 ...
```

4. Fit a logistic regression model.

```
m <- glm(HIRE ~ EDUC + EXP + GENDER, data=hire.data,
               family=binomial)
summary(m)
```

```
37  # Fit logistic regression model
38  m <- glm(HIRE ~ EDUC + EXP + GENDER, data=hire.data , family=binomial)
39  summary(m)
```

```
> # Fit logistic regression model
> m <- glm(HIRE ~ EDUC + EXP + GENDER, data=hire.data , family=binomial)
> summary(m)

Call:
glm(formula = HIRE ~ EDUC + EXP + GENDER, family = binomial,
    data = hire.data)

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -14.2483     6.0805  -2.343   0.0191 *
EDUC          1.1549     0.6023   1.917   0.0552 .
EXP           0.9098     0.4293   2.119   0.0341 *
GENDER        5.6037     2.6028   2.153   0.0313 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 35.165  on 27  degrees of freedom
Residual deviance: 14.735  on 24  degrees of freedom
AIC: 22.735

Number of Fisher Scoring iterations: 7
```

5. Write out the fitted model (original form) with the estimated coefficient values and meaningful variable names.

Logit(HIRE) = log-odd{HIRE}

$= ln\frac{pi}{1-pi}$ = -14.2483 + 1.1549(EDUC) + 0.9098(EXP) + 5.6037(GENDER.male)

6. **Confidence interval of model coefficient.** Compute the 95% confidence interval of the slope coefficient of EDUC. The point estimate and the standard error are given in the model summary output. For the distribution multiplier, use the z-distribution (standard normal) instead of t-distribution. <u>Report and interpret the confidence interval</u>.

```
# Confidence interval for model coefficients

summary(m)$coefficients
qnorm(p=.975) # z-distribution

# 95% CI of b_EDU, using z-distribution
1.1549 - 1.96 * 0.6023 # LL
summary(m)$coefficients[2, 1] - qnorm(.975) * summary(m)$coefficients[2, 2]

1.1549 + 1.96 * 0.6023 # UL
summary(m)$coefficients[2, 1] + qnorm(.975) * summary(m)$coefficients[2, 2]
```

```
> #-------------------------------------------------------
> # Confidence interval for model coefficients
>
> summary(m)$coefficients
               Estimate Std. Error   z value   Pr(>|z|)
(Intercept) -14.2482575  6.0805351 -2.343257 0.01911620
EDUC          1.1548804  0.6022944  1.917468 0.05517846
EXP           0.9098486  0.4292934  2.119410 0.03405584
GENDER        5.6036794  2.6027819  2.152958 0.03132200
> qnorm(p=.975) # z-distribution
[1] 1.959964
>
> # 95% CI of b_EDU, using z-distribution
> 1.1549 - 1.96 * 0.6023 # LL
[1] -0.025608
> summary(m)$coefficients[2, 1] - qnorm(.975) * summary(m)$coefficients[2, 2]
[1] -0.0255949
>
> 1.1549 + 1.96 * 0.6023 # UL
[1] 2.335408
> summary(m)$coefficients[2, 1] + qnorm(.975) * summary(m)$coefficients[2, 2]
[1] 2.335356
```

- With 95% confidence, the b1 (EDUC) falls between -0.025608 and 2.335356.
- With 95% confidence, we estimate that the log-odds of success (being hired) changes by somewhere between -0.025608 and 2.335356 for each 1-year increase in higher education while holding all other predictors (EXP and DENDER) constant.
- With 95% confidence, we estimate that the odds of success (being hired) changes by a factor of somewhere within (e^ -0.025608, e^2.335356) = (0.975, 10.333) for each 1-year increase in higher education while holding all other predictors (EXP and DENDER) constant. This implies that the odds of being hired could decrease by ~ 2.5% or increase by ~ 933.3%, per additional year of education.

7. **Confidence interval of the predicted probability.** Use the following code to compute the 95% confidence interval of the predicted probability of being hired for the individual described above. <u>Report and interpret the confidence interval in the context of the problem</u>.

```
# Compute the lower and upper limits on the logit scale first
my.pred <- predict(m, newdata=data.frame(EDUC=6, EXP=3,
                   GENDER=1), level=.95, type="link", se.fit=T)
```

```
z.crit <- qnorm(p=.975)

LL.logit <- my.pred$fit - z.crit * my.pred$se.fit
UL.logit <- my.pred$fit + z.crit * my.pred$se.fit

# Then convert the two limits to the probability scale
LL.pi <- exp(LL.logit) / (1 + exp(LL.logit))
UL.pi <- exp(UL.logit) / (1 + exp(UL.logit))
c(LL.pi, UL.pi)
```

```
#----------------- HW11 ---------------------------------
# To compute the confidence interval for pi, you need to start from the logit
# scale (i.e., the "link" type).

my.pred <- predict(m, newdata=data.frame(EDUC=6, EXP=3, GENDER=1),level=.95,
                   type="link", se.fit=T)
my.pred # 1.030039

z.crit <- qnorm(p=.975)
z.crit # 1.959964

LL.logit <- my.pred$fit - z.crit * my.pred$se.fit
LL.logit # -1.00459
1.01425 - 1.959964 * 1.030039 # -1.004589, manual, LL.logit

UL.logit <- my.pred$fit + z.crit * my.pred$se.fit
UL.logit # 3.03309
1.01425 + 1.959964 * 1.030039 # 3.033089, manual, UL.logit

# Then convert the lower and upper limits to the probability scale
LL.pi <- exp(LL.logit) / (1 + exp(LL.logit))
LL.pi # 0.2680399
exp(-1) / (1 + exp(-1)) # 0.2689414, manual calculation

UL.pi <- exp(UL.logit) / (1 + exp(UL.logit))
UL.pi # 0.9540469
exp(3) / (1 + exp(3)) # 0.9525741, manual calculation

c(LL.pi, UL.pi)
```

```
> # To compute the confidence interval for pi, you need to start from the logit
> # scale (i.e., the "link" type).
>
> my.pred <- predict(m, newdata=data.frame(EDUC=6, EXP=3, GENDER=1),level=.95,
+                   type="link", se.fit=T)
> my.pred # 1.030039
$fit
      1
1.01425

$se.fit
[1] 1.030039

$residual.scale
[1] 1

>
> z.crit <- qnorm(p=.975)
> z.crit # 1.959964
[1] 1.959964
>
> LL.logit <- my.pred$fit - z.crit * my.pred$se.fit
> LL.logit # -1.00459
       1
-1.00459
> 1.01425 - 1.959964 * 1.030039 # -1.004589, manual, LL.logit
[1] -1.004589
```

```
> UL.logit <- my.pred$fit + z.crit * my.pred$se.fit
> UL.logit # 3.03309
       1
3.03309
> 1.01425 + 1.959964 * 1.030039 # 3.033089, manual, UL.logit
[1] 3.033089
>
> # Then convert the lower and upper limits to the probability scale
> LL.pi <- exp(LL.logit) / (1 + exp(LL.logit))
> LL.pi # 0.2680399
        1
0.2680399
> exp(-1) / (1 + exp(-1)) # 0.2689414, manual calculation
[1] 0.2689414
>
> UL.pi <- exp(UL.logit) / (1 + exp(UL.logit))
> UL.pi # 0.9540469
        1
0.9540469
> exp(3) / (1 + exp(3)) # 0.9525741, manual calculation
[1] 0.9525741
>
> c(LL.pi, UL.pi)
        1         1
0.2680399 0.9540469
```

- The lower or upper limit of the 95% confidence interval is ~ 0.268 (26.8%) or ~ 0.954 (95.4%), for a male with 6 years of education (EDUC=6) and 3 years of experience (EXP=3).
- We are 95% sure that the probability of this individual being hired for a male with 6 years of education (EDUC=6) and 3 years of experience (EXP=3) falls between 26.8% and 95.4%.

8. **Likelihood-ratio test of comparing two nested models.** The likelihood-ratio test of comparing two nested models is analogous to the general linear test approach F-test. Follow prompts below to complete the 5 steps of the test comparing the *full model*, which we have examined so far, against the *reduced model* where the term **GENDER** is dropped from the model.

   a. **Step 1:** State the assumptions.
   Large sample, independent observations.

   b. **Step 2:** State full model and the reduced model, then state the null and alternative hypotheses.
   Full Model: $ln\frac{pi}{1-pi}$ = β0 + β(EDUC) + β(EXP) + β(GENDER)
   Reduced Model: $ln\frac{pi}{1-pi}$ = β0 + β(EDUC) + β(EXP)
   H0: β(GENDER) = 0, drop β(GENDER)
   H1: β(GENDER) ≠ 0, don't drop β(GENDER)

   c. **Step 3:** Compute the test-statistic. The test-statistic can be calculated by "*Residual deviance of the reduced model – Residual deviance of the full model*". Find the two quantities from the model summary outputs of *each* model. (You have already seen the summary of the full model. You need to fit the reduced model now and see its summary.)

```
#----------------------------------------------------------
# Likelihood-Ratio Test for GENDER
m.F <- glm(HIRE~EDUC+EXP+GENDER, data=hire.data , family=binomial)
summary(m.F)

m.R <- glm(HIRE~EDUC+EXP, data=hire.data , family=binomial)
summary(m.R)
```

```
> m.F <- glm(HIRE~EDUC+EXP+GENDER, data=hire.data , family=binomial)
> summary(m.F)

Call:
glm(formula = HIRE ~ EDUC + EXP + GENDER, family = binomial,
    data = hire.data)

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -14.2483     6.0805  -2.343   0.0191 *
EDUC          1.1549     0.6023   1.917   0.0552 .
EXP           0.9098     0.4293   2.119   0.0341 *
GENDER        5.6037     2.6028   2.153   0.0313 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 35.165  on 27  degrees of freedom
Residual deviance: 14.735  on 24  degrees of freedom
AIC: 22.735

Number of Fisher Scoring iterations: 7
```

```
> m.R <- glm(HIRE~EDUC+EXP, data=hire.data , family=binomial)
> summary(m.R)

Call:
glm(formula = HIRE ~ EDUC + EXP, family = binomial, data = hire.data)

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -5.4419     2.3361  -2.330   0.0198 *
EDUC          0.5404     0.3287   1.644   0.1002
EXP           0.3717     0.1670   2.226   0.0260 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 35.165  on 27  degrees of freedom
Residual deviance: 26.056  on 25  degrees of freedom
AIC: 32.056

Number of Fisher Scoring iterations: 5
```

d. **Step 4:** Find the p-value. The p-value is found by locating the test-statistic value above on a reference distribution that is a Chi-square distribution with a degree of freedom of "*reduced model residual deviance df – full model residual deviance df*" (both quantities found from the model summary outputs), and finding the **right-tail probability**. Tweak the code below to get that p-value.

```
pchisq(q = ?, df = ?, lower.tail=F)
```

e.

```
# Test-statistic
26.056 - 14.735

# DF
25 - 24

# P-value
pchisq(q=11.321, df=25-24, lower.tail=F)
```

```
> # Test-statistic
> 26.056 - 14.735
[1] 11.321
>
> # DF
> 25 - 24
[1] 1
>
> # P-value
> pchisq(q=11.321, df=25-24, lower.tail=F)
[1] 0.0007663543
```

f. **Step 5:** Make a conclusion.
With the p-value = 0.00076 < 0.05 = α, we reject H0 and take H1. We conclude that the full model fit the data significantly better than the reduced model. Cender contributes significantly to the model on top of EDU and EXP. There is enough evidence for gender discrimination after accounting for education and work experience.

9. **Likelihood-ratio test of global model utility.** The likelihood-ratio test of *globel model utility* is a likelihood-ratio test comparing the fitted model with the *null model* where all slope coefficients are set to 0. This is analogous to the F-test of global model utility in the ordinary multiple regression models. Follow prompts below to complete the 5 steps of this test.

a. **Step 1:** State the assumptions.
Large sample, independent observations.

Step 2: State the null and alternative hypotheses.
Full Model: $ln\frac{pi}{1-pi}$ = β0 + β(EDUC) + β(EXP) + β(GENDER)
Null Model: $ln\frac{pi}{1-pi}$ = β0
H0: β(EDUC) = β(EXP) = β(GENDER) = 0, drop all
H1: Not all β is zero. At least one of β(EDUC), β(EXP), or β(GENDER) is NOT zero.

b. **Step 3:** Compute the test-statistic. The test-statistic can be calculated by "*Null deviance – Residual deviance*". Find the two quantities from the model summary output.

```
#----------------------------------------------------------
# Likelihood-Ratio Test for Null vs Full
m.F <- glm(HIRE~EDUC+EXP+GENDER, data=hire.data , family=binomial)
summary(m.F)

35.165 - 14.735 # Test-statistic, 20.43
27 - 24 # DF

# P-value
pchisq(q=20.43, df=27-24, lower.tail=F)
```

```
> m.F <- glm(HIRE~EDUC+EXP+GENDER, data=hire.data , family=binomial)
> summary(m.F)

Call:
glm(formula = HIRE ~ EDUC + EXP + GENDER, family = binomial,
    data = hire.data)

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -14.2483     6.0805  -2.343   0.0191 *
EDUC          1.1549     0.6023   1.917   0.0552 .
EXP           0.9098     0.4293   2.119   0.0341 *
GENDER        5.6037     2.6028   2.153   0.0313 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 35.165  on 27  degrees of freedom
Residual deviance: 14.735  on 24  degrees of freedom
AIC: 22.735

Number of Fisher Scoring iterations: 7
```

G^2 = Deviance(R) - Deviance(F) = 35.165 - 14.735 = 20.43

c. **Step 4:** Find the p-value. The p-value is found by locating the test-statistic value above on a reference distribution that is a Chi-square distribution with a degree of freedom of "*Null deviance df – Residual deviance df*" (both quantities found from the model summary output), and finding the **right-tail probability**. Tweak the code below to get that p-value.

```
pchisq(q = ?, df = ?, lower.tail=F)
```

```
# P-value
pchisq(q=20.43, df=27-24, lower.tail=F)
```

```
> 35.165 - 14.735 # Test-statistic, 20.43
[1] 20.43
> 27 - 24 # DF
[1] 3
>
> # P-value
> pchisq(q=20.43, df=27-24, lower.tail=F)
[1] 0.0001382406
```

d. **Step 5:** Make a conclusion.

With the p-value = 0.000138 < 0.05 = α, we reject H0 and take H1. We conclude that the full model fit the data significantly better than the null model. At least one of the slope coefficients in the model is significantly different from zero, suggesting that at least one of EDU, EXP, or GENDER is useful for predicting the binary outcome.

10. **Pseudo R-square.**

a. Use the pertinent objects from Problem 14 above to compute McFadden's pseudo R-squared. Report and interpret the results.

```
#-------------------------------------------------------
# Pseudo R-squares

# McFadden's Pseudo R-square

1 - m$deviance / m$null.deviance  # The deviances given in R outputs are -2 times of the log-likelihood
logLik(m)  # This gives the log-likelihood of the model
```

```
> 1 - m$deviance / m$null.deviance  # The deviances given in R outputs are -2 times of the log-likelihood
[1] 0.5809766
> logLik(m)  # This gives the log-likelihood of the model
'log Lik.' -7.367422 (df=4)
```

Pseudo R-square (ρ^2) values are not directly comparable to R-squared values from linear regression and generally are lower. A value of 0.2 to 0.4 is often considered as a good fit in logistic regression contexts. Here, the McFadden's Pseudo R-square value is 0.5809766 which is considered quite high (R^2 > 0.9, Fig.5.5). This suggests that the model is a relatively good fit to the data.

b. Run the code below to compute the Tjur's pseudo R-squared. Report and interpret the results. (Note: effectively, only the first and last lines of the code are needed to calculate the Tjur's pseudo R-squared. The other code in between are included to help you understand the process.

```
sel <- (lab11data$HIRE == 1)
lab11data[sel,]
lab11data[!sel,]

predict(m, type="response")[sel]
mean(predict(m, type="response")[sel])

predict(m, type="response")[!sel]
mean(predict(m, type="response")[!sel])

mean(predict(m, type="response")[sel]) - mean(predict(m,
     type="response")[!sel])
```

```r
# Tjur'a Pseudo R-square
sel <- hire.data $HIRE == 1
hire.data [sel,]
hire.data [!sel,]

predict(m, type="response")[sel]
mean(predict(m, type="response")[sel]) #0.7428251

predict(m, type="response")[!sel]
mean(predict(m, type="response")[!sel]) #0.1218197

(mean(predict(m, type="response")[sel]) - mean(predict(m, type="response")[!sel]))
0.7428251 - 0.1218197 #0.6210054
```

```
> # Tjur'a Pseudo R-square
> sel <- hire.data $HIRE == 1
> hire.data [sel,]
   HIRE EDUC EXP GENDER
3     1    6   6      1
4     1    6   3      1
6     1    8   3      0
10    1    8  10      0
15    1    4   5      1
18    1    6   1      1
23    1    8   5      1
27    1    4  10      1
28    1    6  12      0
```

```
   HIRE EDUC EXP GENDER
1     0    6   2      0
2     0    4   0      1
5     0    4   1      0
7     0    4   2      1
8     0    4   4      0
9     0    6   1      0
11    0    4   2      1
12    0    8   5      0
13    0    4   2      0
14    0    6   7      0
16    0    6   4      0
17    0    8   0      1
19    0    4   7      0
20    0    4   1      1
21    0    4   5      0
22    0    6   0      1
24    0    4   9      0
25    0    8   1      0
26    0    6   1      1
```

```
> predict(m, type="response")[sel]
         3          4          6         10         15         18         23         27         28 
0.97688294 0.73385108 0.09282087 0.98352456 0.62812621 0.30886153 0.99419785 0.99377805 0.97338251 
> mean(predict(m, type="response")[sel]) #0.7428251
[1] 0.7428251
> 
> predict(m, type="response")[!sel]
           1            2            5            7            8            9           11           12           13           14           16           17           19 
0.0040730658 0.0175489472 0.0001634423 0.0992702542 0.0024990525 0.0016437556 0.0992702542 0.3869926554 0.0004058834 0.2788778392 0.0246124941 0.6443889709 0.0369764902 
          20           21           22           24           25           26 
0.0424842835 0.0061845776 0.1524780906 0.1915301359 0.0163126738 0.3088615307 
> mean(predict(m, type="response")[!sel]) #0.1218197
[1] 0.1218197
```

```
> (mean(predict(m, type="response")[sel]) - mean(predict(m, type="response")[!sel]))
[1] 0.6210054
> 0.7428251 - 0.1218197 #0.6210054
[1] 0.6210054
```

Tjur's Pseudo R-square is approximately 0.6210, suggesting a strong discriminatory power of the model. The higher Tjur's R-square indicates that the model's predictions are, on average, more accurate in terms of reflecting the actual outcomes.

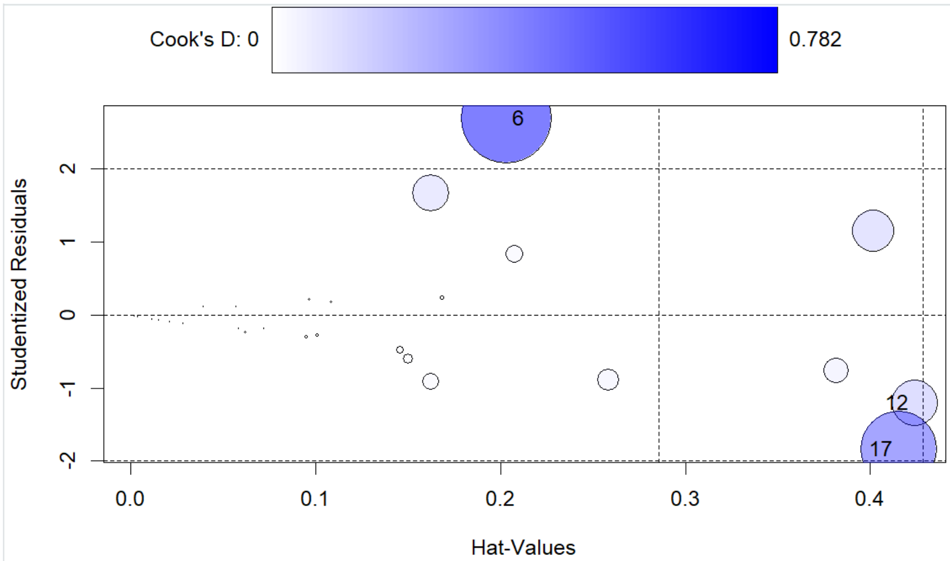
12. **Diagnostics.**

a. **Influential points.** Run the code below to generate the influence measure plot. Describe your impression of the plot.

```
car::influencePlot(m)
```

```
# Diagnostics
# Influence plot
par(mfrow=c(1,1))
car::influencePlot(m)
```

```
> car::influencePlot(m)
     StudRes       Hat     CookD
6   2.691732 0.2031239 0.7815681
12 -1.201389 0.4239325 0.2016170
17 -1.831995 0.4155453 0.5510990
```



b. **Cook's d.** Find the *largest* value of Cook's d from the influence plot above. Complete the code below to convert that value to a percentile score on the reference distribution *F(p, n – p)*. Report the percentile score and assess whether it exceeds 50th percentile and is considered overly influential.

```
pf(q = ?, df1 = p, df2 = (n - p))
```

```
# Diagnostics
# Influence plot
par(mfrow=c(1,1))
car::influencePlot(m)

# Largest Cook's D = 0.7815681
p <- 4
n <- nrow(hire.data )
pf(q = .782, df1 = p, df2 = n - p) #0.4520179
0.4520179 *100 # percentile score

# DFBETA

dfbetas(m)

data.frame(hire.data , round(dfbetas(m), 2))
```

```
> # Largest Cook's D = 0.7815681
> p <- 4
> n <- nrow(hire.data )
> pf(q = .782, df1 = p, df2 = n - p) #0.4520179
[1] 0.4520179
> 0.4520179 *100 # percentile score
[1] 45.20179
```

This percentile score with the largest Cook's D value of 0.782 is about 45.2%, which is below the 50th percentile. Hence, the data point with the Cook's D of 0.7815681 is not considered overly influential as it does not exceed the 50th percentile.

c. **DFbetas.** Run the code below to see the DFbetas. Which observation has the strongest influence on the slope coefficient of GENDER?

```
dfbetas(m)
```

```
> # DFBETA
> dfbetas(m)
     (Intercept)          EDUC           EXP        GENDER
1  -0.014762642  0.0105723414  0.0147492413  0.0160108535
2  -0.053419678  0.0504607009  0.0481919930  0.0350299609
3  -0.076346519  0.0605645806  0.0852187192  0.0839297296
4  -0.332365006  0.2852243687  0.3404833421  0.4414867329
5  -0.001009594  0.0008457608  0.0009319448  0.0009576888
6   0.679876340  0.0584955250 -1.0202362324 -1.3473394626
7  -0.179430514  0.1971263632  0.1227777744  0.0699225111
8  -0.010683845  0.0093419237  0.0086625445  0.0099936900
9  -0.006984585  0.0050480283  0.0071580858  0.0074587495
10 -0.066725970  0.0651551597  0.0719799305  0.0479042440
11 -0.179430514  0.1971263632  0.1227777744  0.0699225111
12  0.231241365 -0.6716223326 -0.0724276004  0.3642845882
13 -0.002246936  0.0019032923  0.0020098824  0.0021238718
14 -0.104995891  0.0259017324 -0.0878855654  0.2386493091
15 -0.221838932 -0.1431031002  0.6990131430  0.7546822918
16 -0.058709245  0.0406002189  0.0530055069  0.0673303368
17  1.007243500 -1.4731376500 -0.1706681243 -0.8020667146
18  0.171985299 -0.0615128604 -0.2979583198  0.0753601648
19 -0.092280577  0.0899263810  0.0465433189  0.0829985571
20 -0.103573170  0.1034764731  0.0854583029  0.0581744110
21 -0.022647033  0.0202818129  0.0168949309  0.0210117495
22 -0.158653106  0.1022575490  0.2104754536  0.0911178376
23 -0.026003798  0.0242058099  0.0239649414  0.0242565052
24 -0.287494878  0.3646072878 -0.1135391968  0.2282154079
25 -0.039209007  0.0159091282  0.0511813731  0.0547658952
26 -0.089497037  0.0320098215  0.1550503841 -0.0392156276
27 -0.023964569  0.0146570773  0.0329084134  0.0295003668
28 -0.094292988  0.0763361508  0.1286556732  0.0799699076
```

Observation 6 has the largest absolute **dfbeta value for GENDER with -1.347**. This suggests that observation 6 has the strongest influence on the slope coefficient of GENDER among all the observations in the dataset.