**Lab 4: Logistic Regression (Part 2)**

**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.**

*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.
2. Start R or R Studio. Load the “car” package.
3. Import the data into R. Name the imported data **hire.data**. View the data and make sure the data have been imported correctly.
4. Fit a logistic regression model.

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

1. Write out the fitted model (original form) with the estimated coefficient values and meaningful variable names.
2. **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. Report and interpret the confidence interval.
3. **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. Report and interpret the confidence interval in the context of the problem.

# 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)

1. **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.
   1. **Step 1:** State the assumptions.
   2. **Step 2:** State full model and the reduced model, then state the null and alternative hypotheses.
   3. **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.)
   4. **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)
   5. **Step 5:** Make a conclusion.
2. **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.  
   1. **Step 1:** State the assumptions.
   2. **Step 2:** State the null and alternative hypotheses.
   3. **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.
   4. **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)
   5. **Step 5:** Make a conclusion.
3. **Pseudo R-square.**   
   1. Use the pertinent objects from Problem 14 above to compute McFadden’s pseudo   
      R-squared. Report and interpret the results.
   2. 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])

1. **Diagnostics.**   
   1. **Influential points.** Run the code below to generate the influence measure plot. Describe your impression of the plot.   
        
      car::influencePlot(m)
   2. **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))
   3. **DFbetas.** Run the code below to see the DFbetas. Which observation has the strongest influence on the slope coefficient of GENDER?  
        
      dfbetas­­­­(m)