Problem Set 5, Winter 2024

Michael Ghattas

```r
knitr::opts_chunk$set(echo = TRUE)
# Load any packages
library(tidyverse)   # For data manipulation and visualization
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(lmtest)      # For likelihood ratio tests
```

```
## Loading required package: zoo
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

Question 1 - 10 points (A-D)

The relationship between percents, odds, and odds ratios is salient to interpreting logistic regression output. If you're not familiar with the relationship between probability and odds, have a look at the Week 5 live session slides. There is a section on probability and odds that I did not discuss during the live session but included in the slides as a reference.

If the odds of an event equal b, what is the probability p of the event? This question has four parts:

A) Write a function to compute the probability from the odds:

```r
prob.from.odds <- function(b) {
  return(b / (1 + b))
}
```

B) Test your function by inputting three test values - 5, 10, and 20 - and showing what the output of your function is for these values. That is, when the odds are 5, 10, and 20, what are the associated probabilities?

```
# Test your function by running this code chunk, which uses 5, 10, and 20 as inputs for your function.
prob.from.odds(5)
```

```
## [1] 0.8333333
```

```
prob.from.odds(10)
```

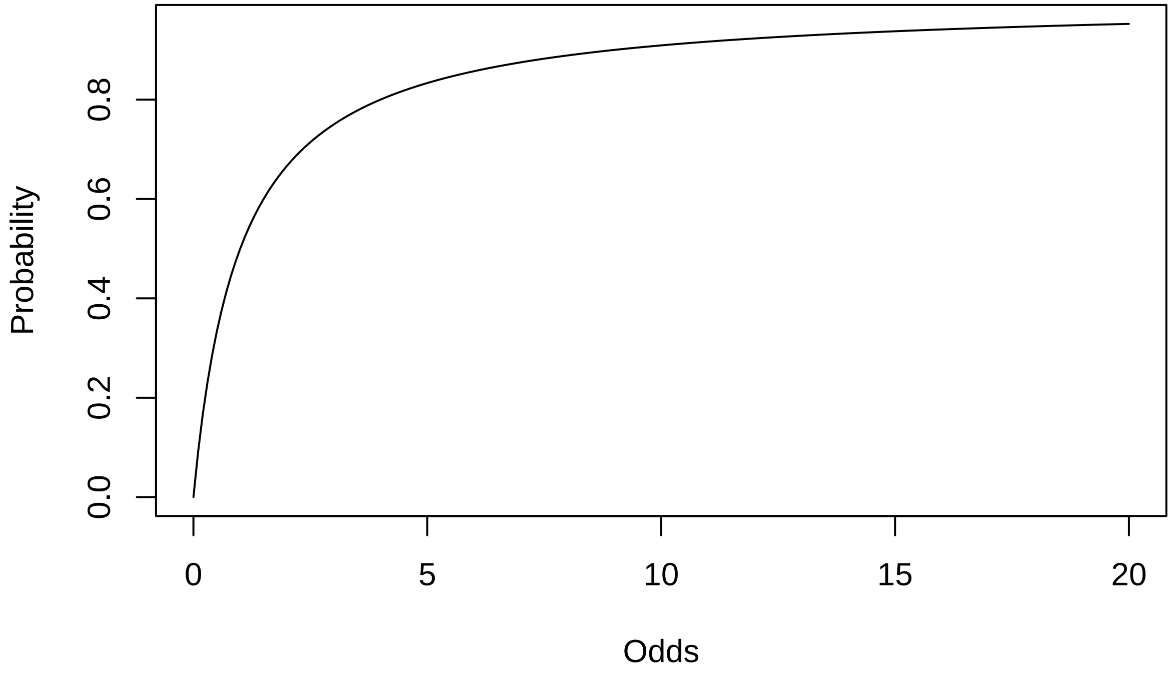
```
## [1] 0.9090909
```

```
prob.from.odds(20)
```

```
## [1] 0.952381
```

C) Create a plot that visually demonstrates how the probability changes within in the range of odds=0 to odds=20. Be sure probability is on the y axis and odds are on the x axis.

```
# Create your plot. Probability should be on the Y-axis and odds should be on the X-axis.
odds <- seq(0, 20, 0.1)
prob <- prob.from.odds(odds)
plot(odds, prob, type="l", xlab="Odds", ylab="Probability")
```



D) Answer the following question:

Based on what you see in your plot, what happens to a computed probability as the associated odds increase? This can be answered in one sentence.

Your answer here: As the odds increase, the probability increases as it approaches 1.

CONTEXT: Pew Research Center data

The data in "pew_data.RData" comes from the Pew Research Center, an organization that conducts nationally-representative public opinion polls on a variety of political and social topics. Dr. Durso constructed this data set from the 2017 Pew Research Center Science and NewsSurvey, downloaded from https://www.journalism.org/datasets/2018/ on 4/16/2019.

There are 224 variables in this data set, but only a subset will be used in this problem set. For this problem set, the outcome of interest will be the LIFE variable, which was presented to respondents like so:

"In general, would you say life in America today is better, worse or about the same as it was 50 years ago for people like you?"

Possible responses included:

1 = Better today

2 = Worse today

3 = About the same as it was 50 years ago

-1 = Refused

Preamble to Questions 2-6 - Read this before starting on Question 2.

Using the data contained in "pew", you will fit three logistic regression models using the LIFE variable as the outcome.

Model 1: Include income as a continuous predictor** and gender as a categorical predictor.

Model 2: In addition to the predictors in Model 1, include ethnicity and education as categorical predictors.

Model 3: In addition to the predictors in Model 2, include the ideology variable.

** I wrote an aside about this variable. You do *not* have to read it, but if you want to, scroll to the end of the document to find it.

Question 2 - 5 points (A-F)

First, you will need to process the data. The Pew data is stored in an RData file, so the first line loads the RData file into memory. The second line creates a data set called "pew" that contains just the variables we'll use in this problem set. Run the code chunk and continue.

```r
load("pew_data.RData")
pew<-dplyr::select(dat,PPINCIMP,PPGENDER,PPETHM,IDEO,PPEDUCAT,LIFE)
table(pew$LIFE, exclude = NULL)
```

```
##
##   -1    1    2    3
##   18 1596 1900  510
```

Next, have a look at each variable in the data set. The RData format allowed for metadata about variables to be preserved along with the data itself. In the code chunk below, each variable has three lines of code associated with it. The first displays the text of the question, the second displays the set of potential responses, and the third displays the number of respondents that gave each response. Once you've reviewed the output, answer the six questions below.

```
attributes(pew$LIFE)$label # LIFE
```

```
## [1] "In general, would you say life in America today is better, worse or about the same as it was 50
```

```
attributes(pew$LIFE)$labels
```

```
##                               Refused                          Better today
##                                    -1                                     1
##                           Worse today About the same as it was 50 years ago
##                                     2                                     3
```

```
table(pew$LIFE, exclude = NULL)
```

```
##
##   -1    1    2    3
##   18 1596 1900  510
```

```
attributes(pew$PPINCIMP)$label #income
```

```
## [1] "Household Income"
```

```
attributes(pew$PPINCIMP)$labels
```

```
##            Not asked              REFUSED     Less than $5,000
##                   -2                   -1                    1
##     $5,000 to $7,499     $7,500 to $9,999   $10,000 to $12,499
##                    2                    3                    4
##   $12,500 to $14,999   $15,000 to $19,999   $20,000 to $24,999
##                    5                    6                    7
##   $25,000 to $29,999   $30,000 to $34,999   $35,000 to $39,999
##                    8                    9                   10
##   $40,000 to $49,999   $50,000 to $59,999   $60,000 to $74,999
##                   11                   12                   13
##   $75,000 to $84,999   $85,000 to $99,999 $100,000 to $124,999
##                   14                   15                   16
## $125,000 to $149,999 $150,000 to $174,999 $175,000 to $199,999
##                   17                   18                   19
## $200,000 to $249,999     $250,000 or more
##                   20                   21
```

```
table(pew$PPINCIMP, exclude = NULL)
```

```
##
##   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20
##  66  31  40  91  77 104 144 183 179 167 258 321 378 285 319 486 226 253 160 125
##  21
## 131
```

```
attributes(pew$PPGENDER)$label #gender
```

```
## [1] "Gender"
```

```
attributes(pew$PPGENDER)$labels
```

```
## Not asked   REFUSED      Male    Female
##        -2        -1         1         2
```

```
table(pew$PPGENDER, exclude = NULL)
```

```
##
##    1    2
## 1993 2031
```

```
attributes(pew$PPETHM)$label #ethnicity
```

```
## [1] "Race / Ethnicity"
```

```
attributes(pew$PPETHM)$labels
```

```
##              Not asked                REFUSED    White, Non-Hispanic
##                     -2                     -1                      1
##    Black, Non-Hispanic    Other, Non-Hispanic               Hispanic
##                      2                      3                      4
## 2+ Races, Non-Hispanic
##                      5
```

```
table(pew$PPETHM, exclude = NULL)
```

```
##
##    1    2    3    4    5
## 2862  392  166  447  157
```

```
attributes(pew$IDEO)$label #ideology
```

```
## [1] "In general, would you describe your political views as..."
```

```r
attributes(pew$IDEO)$labels
```

```
##           Refused Very conservative      Conservative          Moderate 
##                -1                 1                 2                 3 
##           Liberal      Very liberal 
##                 4                 5 
```

```r
table(pew$IDEO, exclude = NULL)
```

```
## 
##   -1    1    2    3    4    5 
##  116  314 1095 1624  616  259 
```

```r
attributes(pew$PPEDUCAT)$label #education
```

```
## [1] "Education (Categorical)"
```

```r
attributes(pew$PPEDUCAT)$labels
```

```
##                   Not asked                     REFUSED 
##                          -2                          -1 
##       Less than high school                 High school 
##                           1                           2 
##                Some college Bachelor’s degree or higher 
##                           3                           4 
```

```r
table(pew$PPEDUCAT, exclude = NULL)
```

```
## 
##    1    2    3    4 
##  303 1130 1147 1444 
```

A) How many people’s response was “Refused”, “Not asked”, or “NA” for the LIFE variable?

Your answer here: Refused (-1): 18 respondents.

B) How many people’s response was “Refused”, “Not asked”, or “NA” for the PPINCIMP variable?

Your answer here: Refused (-1): 66 respondents, Not asked (-2): 31 respondents, Total = 66 + 31 = 97 respondents.

C) How many people’s response was “Refused”, “Not asked”, or “NA” for the PPGENDER variable?

Your answer here: Refused (-1): 0 respondents, Not asked (-2): 0 respondents, Total 0 respondents.

D) How many people’s response was “Refused”, “Not asked”, or “NA” for the PPETHM variable?

Your answer here: Refused (-1): 392 respondents, Not asked (-2): 2862 respondents, Total = 392 + 2862 = 3254 respondents.

E) How many people's response was "Refused", "Not asked", or "NA" for the IDEO variable?

Your answer here: Refused (-1): 116 respondents.

F) How many people's response was "Refused", "Not asked", or "NA" for the PPEDUCAT variable?

Your answer here: Refused (-1): 303 respondents, Not asked (-2): 1130 respondents, Total = 303 + 1130 = 1433 respondents.

Question 3 - 5 points (A-C)

Be sure to have completed Question 2 before beginning this question.

You'll conduct what's called a "complete cases" analysis, where an analysis is conducted only on cases that have information for all variables used in the analysis. There are some situations were this is appropriate and others where other ways of handling missing data should be used (for more information, see http://galton.uchicago.edu/~eichler/stat24600/Admin/MissingDataReview.pdf). For the purposes of this problem set, we'll assume that this is a situation where complete cases analysis is appropriate.

Use the code chunk below to drop all rows that have one or more instances of "Refused", "Not asked", or "NA" in the six variables in the pew data set. You'll do this by first making a copy of the pew data set, then dropping cases from the copy; this will make it easier to check your work. Once you've done this, answer the question below.

```
# Code to drop all observations with one or more "Refused", "Not asked", or "NA" in any of the six vari
pew.complete <- pew[complete.cases(pew), ]
nrow(pew.complete)
```

```
## [1] 4024
```

A) How many rows remain in your data set once you've dropped all cases with at least one "Refused", "Not asked", or NA?

Your answer here: 4024 rows remain in the pew.complete data set after dropping incomplete cases.

Now, use the table() function to display the counts of the responses for all six variables to verify that none of these responses remain and answer the question below:

```
# Code to display counts of the responses of the six variables here (remember, use pew.complete as the
table(pew.complete$LIFE)
```

```
##
##   -1    1    2    3
##   18 1596 1900  510
```

B) Looking at the LIFE variable in the pew.complete data set, how many people said that life was "Worse today"?

Your answer here: 1900 people said that life was "Worse today" (coded as 2 in the data)

C) Again looking at the LIFE variable in the pew.complete data set, how many people said that life was either "Better today" or "About the same"?

Your answer here: Better today (coded as 1): 1596 people, About the same (coded as 3): 510 people. So, 2106 people in the data set said that life was either "Better today" or "About the same."

Question 4 - 10 points

Be sure to complete Question 3 before starting this one.

Now that you've dropped the incomplete cases, we can move on to analysis. Use the pew.complete data set. First, you will set up your outcome variable. Re-code the LIFE variable such that "Worse today" is equal to one and "Better today"/"About the same" are equal to 0. Be sure to display the frequencies of the recoded variable.

```
# Code to re-code outcome
pew.complete$worse <- ifelse(pew.complete$LIFE == 2, 1, 0)
# Don't forget to display a table showing the frequencies of the re-coded outcome
table(pew.complete$worse, exclude = NULL)
```

```
##
##    0    1
## 2124 1900
```

Next, check that all six variables are of the appropriate type. Income should be numeric- or integer-type variables, and gender, ethnicity, ideology, education category, and the re-coded life variable should be factor-type variables. Check that you've done this correctly by using the str() function.

```
# Code to set variables to their appropriate types
pew.complete$income <- as.numeric(pew.complete$PPINCIMP)
pew.complete$gender <- factor(pew.complete$PPGENDER)
pew.complete$eth <- factor(pew.complete$PPETHM)
pew.complete$ideo <- factor(pew.complete$IDEO)
pew.complete$edu <- factor(pew.complete$PPEDUCAT)
pew.complete$worse <- factor(pew.complete$worse)
# Display the variable types using the str() function
str(pew.complete)
```

```
## tibble [4,024 x 12] (S3: tbl_df/tbl/data.frame)
##  $ PPINCIMP: 'labelled' num [1:4024] 16 19 12 12 21 18 19 16 7 10 ...
##   ..- attr(*, "label")= chr "Household Income"
##   ..- attr(*, "format.spss")= chr "F2.0"
##   ..- attr(*, "labels")= Named num [1:23] -2 -1 1 2 3 4 5 6 7 8 ...
##   .. ..- attr(*, "names")= chr [1:23] "Not asked" "REFUSED" "Less than $5,000" "$5,000 to $7,499" ..
##  $ PPGENDER: 'labelled' num [1:4024] 1 2 1 1 1 1 2 2 2 2 ...
##   ..- attr(*, "label")= chr "Gender"
##   ..- attr(*, "format.spss")= chr "F2.0"
##   ..- attr(*, "labels")= Named num [1:4] -2 -1 1 2
##   .. ..- attr(*, "names")= chr [1:4] "Not asked" "REFUSED" "Male" "Female"
##  $ PPETHM  : 'labelled' num [1:4024] 1 2 4 4 1 5 1 5 1 1 ...
##   ..- attr(*, "label")= chr "Race / Ethnicity"
##   ..- attr(*, "format.spss")= chr "F2.0"
##   ..- attr(*, "labels")= Named num [1:7] -2 -1 1 2 3 4 5
##   .. ..- attr(*, "names")= chr [1:7] "Not asked" "REFUSED" "White, Non-Hispanic" "Black, Non-Hispani
##  $ IDEO    : 'labelled' num [1:4024] 1 3 2 3 2 3 2 3 3 2 ...
##   ..- attr(*, "label")= chr "In general, would you describe your political views as..."
##   ..- attr(*, "format.spss")= chr "F4.0"
##   ..- attr(*, "labels")= Named num [1:6] -1 1 2 3 4 5
##   .. ..- attr(*, "names")= chr [1:6] "Refused" "Very conservative" "Conservative" "Moderate" ...
##  $ PPEDUCAT: 'labelled' num [1:4024] 4 4 2 1 3 3 4 4 2 4 ...
##   ..- attr(*, "label")= chr "Education (Categorical)"
##   ..- attr(*, "format.spss")= chr "F2.0"
##   ..- attr(*, "labels")= Named num [1:6] -2 -1 1 2 3 4
##   .. ..- attr(*, "names")= chr [1:6] "Not asked" "REFUSED" "Less than high school" "High school" ...
##  $ LIFE    : 'labelled' num [1:4024] 2 2 1 2 2 1 2 1 2 2 ...
##   ..- attr(*, "label")= chr "In general, would you say life in America today is better, worse or abou
##   ..- attr(*, "format.spss")= chr "F4.0"
##   ..- attr(*, "labels")= Named num [1:4] -1 1 2 3
##   .. ..- attr(*, "names")= chr [1:4] "Refused" "Better today" "Worse today" "About the same as it wa
##  $ worse   : Factor w/ 2 levels "0","1": 2 2 1 2 2 1 2 1 2 2 ...
##  $ income  : num [1:4024] 16 19 12 12 21 18 19 16 7 10 ...
##  $ gender  : Factor w/ 2 levels "1","2": 1 2 1 1 1 1 2 2 2 2 ...
##  $ eth     : Factor w/ 5 levels "1","2","3","4",..: 1 2 4 4 1 5 1 5 1 1 ...
##  $ ideo    : Factor w/ 6 levels "-1","1","2","3",..: 2 4 3 4 3 4 3 4 4 3 ...
##  $ edu     : Factor w/ 4 levels "1","2","3","4": 4 4 2 1 3 3 4 4 2 4 ...
```

Finally, you will fit three logistic regression models using the re-coded LIFE variable and display the results:

Model 1: Include income as a continuous predictor and gender as a categorical predictor.

```r
Model.1 <- glm(worse ~ income + gender, data = pew.complete, family = binomial)
summary(Model.1)
```

```
##
## Call:
## glm(formula = worse ~ income + gender, family = binomial, data = pew.complete)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.494938   0.103562   4.779 1.76e-06 ***
## income      -0.055251   0.006987  -7.907 2.63e-15 ***
## gender2      0.215937   0.064003   3.374 0.000741 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5566.0  on 4023  degrees of freedom
## Residual deviance: 5485.4  on 4021  degrees of freedom
## AIC: 5491.4
##
## Number of Fisher Scoring iterations: 4
```

Model 2: In addition to the predictors in Model 1, include ethnicity and education as categorical predictors.

```
Model.2 <- glm(worse ~ income + gender + eth + edu, data = pew.complete, family = binomial)
summary(Model.2)
```

```
##
## Call:
## glm(formula = worse ~ income + gender + eth + edu, family = binomial,
##     data = pew.complete)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.517898   0.150129   3.450 0.000561 ***
## income      -0.045866   0.007829  -5.858 4.68e-09 ***
## gender2      0.205465   0.064632   3.179 0.001478 **
## eth2        -0.406624   0.112215  -3.624 0.000291 ***
## eth3        -0.053722   0.165110  -0.325 0.744901
## eth4        -0.331653   0.107695  -3.080 0.002073 **
## eth5         0.010342   0.167551   0.062 0.950781
## edu2         0.040497   0.134925   0.300 0.764069
## edu3         0.215399   0.136867   1.574 0.115538
## edu4        -0.382420   0.140662  -2.719 0.006554 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5566.0  on 4023  degrees of freedom
## Residual deviance: 5413.1  on 4014  degrees of freedom
## AIC: 5433.1
##
## Number of Fisher Scoring iterations: 4
```

Model 3: In addition to the predictors in Model 2, include the ideology variable.

```
Model.3 <- glm(worse ~ income + gender + eth + edu + ideo, data = pew.complete, family = binomial)
summary(Model.3)
```

```
##
## Call:
## glm(formula = worse ~ income + gender + eth + edu + ideo, family = binomial,
##     data = pew.complete)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.78154    0.23649   3.305 0.000951 ***
## income      -0.04654    0.00787  -5.913 3.36e-09 ***
## gender2      0.23047    0.06517   3.537 0.000405 ***
## eth2        -0.36129    0.11320  -3.192 0.001414 **
## eth3        -0.01606    0.16607  -0.097 0.922946
## eth4        -0.31303    0.10845  -2.886 0.003898 **
## eth5         0.01153    0.16809   0.069 0.945295
## edu2         0.03896    0.13542   0.288 0.773562
## edu3         0.23537    0.13755   1.711 0.087068 .
## edu4        -0.33396    0.14154  -2.359 0.018302 *
## ideo1        0.02588    0.22496   0.115 0.908420
## ideo2       -0.24225    0.20271  -1.195 0.232064
## ideo3       -0.29936    0.19885  -1.505 0.132216
## ideo4       -0.62808    0.21080  -2.980 0.002887 **
## ideo5       -0.28476    0.23147  -1.230 0.218598
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5566  on 4023  degrees of freedom
## Residual deviance: 5387  on 4009  degrees of freedom
## AIC: 5417
##
## Number of Fisher Scoring iterations: 4
```

Question 5 - 10 points (A-C)

Now that you've fit the three models, you will now conduct two nested model tests to determine the best of the three models. Once you've done so, answer the three questions below. Be sure to conduct likelihood ratio tests, not F-change tests.

Nested model test 1: Model 1 vs Model 2

```
# Code to conduct a nested model test between Model 1 and Model 2 here
lrtest(Model.1, Model.2)
```

```
## Likelihood ratio test
##
## Model 1: worse ~ income + gender
## Model 2: worse ~ income + gender + eth + edu
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   3 -2742.7
## 2  10 -2706.5  7 72.286  5.094e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Nested model test 2: Model 2 vs Model 3

```
# Code to conduct a nested model test between Model 2 and Model 3 here
lrtest(Model.2, Model.3)
```

```
## Likelihood ratio test
##
## Model 1: worse ~ income + gender + eth + edu
## Model 2: worse ~ income + gender + eth + edu + ideo
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1  10 -2706.5
## 2  15 -2693.5  5 26.112  8.488e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

A) Based on the results of the nested model test between Model 1 and Model 2, which would you choose?

Your answer here: The likelihood ratio test between Model 1 and Model 2 shows a significant improvement in fit when moving to Model 2 (Chisq = 72.286, p < 0.001). Therefore, Model 2 is preferred over Model 1.

B) Based on the results of the nested model test between Model 2 and Model 3, which would you choose?

Your answer here: The likelihood ratio test between Model 2 and Model 3 shows a significant improvement in fit when moving to Model 3 (Chisq = 26.112, p < 0.001). Therefore, Model 3 is preferred over Model 2.

C) Based on the results of the two nested model tests, which of the three models - Model 1, Model 2, or Model 3 - would you choose?

Your answer here: Since both nested model tests show significant improvements in fit when moving to the next model, Model 3 is the best choice out of the three models based on the likelihood ratio tests.

Question 6 - 10 points (A-H)

For the model you chose in Question 5, construct a confusion matrix comparing the actual 0/1 values for the re-coded LIFE variable and the predicted 0/1 values. For this question, do so manually (i.e., using the table() function) and not by using a package to do it for you (i.e., do not use confusionMatrix() from the caret package). Construct your confusion matrix such that the rows and columns are labeled; that is, it should be clear what the rows and columns represent without reading your code. Once you've done that, answer the four questions below.

First, display the counts of the *actual* rating of worse the predicted values and display a table of the predicted outcome and the actual outcome.

```r
# Code to display actual binary outcome counts - be sure this displays in your knitted document
table(pew.complete$worse)
```

```
##
##    0    1
## 2124 1900
```

Next, compute the *binarized predictions* based on the model you chose.

```r
# Code to create and display predicted binary outcome counts - be sure the count displays in your knitt
# Compute predicted probabilities for the chosen model (Model 3 from Question 5)
predicted_prob <- predict(Model.3, type = "response")
# Convert probabilities into binary predictions (1 if prob >= 0.5, otherwise 0)
predicted_binary <- ifelse(predicted_prob >= 0.5, 1, 0)
# Display predicted binary outcome counts
table(predicted_binary)
```

```
## predicted_binary
##    0    1
## 2293 1731
```

Next, create your confusion matrix using the table() function. Be sure to label the table axes.

```
# Create the confusion matrix comparing actual vs predicted values
confusion.matrix <- table(Actual = pew.complete$worse, Predicted = predicted_binary)
# Display confusion matrix with labels
confusion.matrix
```

```
##       Predicted
## Actual    0    1
##      0 1373  751
##      1  920  980
```

A) How many true positives did your model produce?

Your answer here: True positives = 980.

B) How many true negatives did your model produce?

Your answer here: True negatives = 1373.

C) How many false positives did your model produce?

Your answer here: False positives = 751.

D) How many false negatives did your model produce?

Your answer here: False negatives = 920.

Now that you've constructed your confusion matrix, use it to compute the four indices of model fit that we dicussed.

```
# Code to compute accuracy
accuracy <- (confusion.matrix[1,1] + confusion.matrix[2,2]) / sum(confusion.matrix)
accuracy
```

```
## [1] 0.5847416
```

```
# Code to compute precision
precision <- confusion.matrix[2,2] / (confusion.matrix[2,2] + confusion.matrix[1,2])
precision
```

```
## [1] 0.5661467
```

```
# Code to compute recall
recall <- confusion.matrix[2,2] / (confusion.matrix[2,2] + confusion.matrix[2,1])
recall
```

```
## [1] 0.5157895
```

```r
# Code to compute F1 score
F1score <- 2 * (precision * recall) / (precision + recall)
F1score
```

```
## [1] 0.5397962
```

E) What is the *accuracy* of this model?

Your answer here: 0.5847 (approximately 58.47% of the predictions were correct).

F) What is the *precision* of this model?

Your answer here: 0.5661 (approximately 56.61% of the predicted positive cases were true positives).

G) What is the *recall* of this model?

Your answer here: 0.5158 (approximately 51.58% of the actual positive cases were correctly identified).

H) What is the *F1 score* of this model?

Your answer here: 0.5398 (this value represents the harmonic mean of precision and recall, balancing the two metrics).

** Strictly speaking, this variable isn't continuous; rather, it's a 21-category variable. Having 20 dummy vectors to represent one variable in a model is unusual in practice. Survey researchers routinely ask about income in this way because many people, especially those who aren't salaried or who have multiple jobs or who belong in multi-income households, can have trouble giving an exact answer to a question about household income. The income categories are used to help the respondent approximate their income in a controlled fashion.

If you have the sample size to support it, one could indeed include 20 dummy codes in a model. If you wanted to test for the effect of income as a whole, you could easily do so using a nested model test. If you were more interested in the effect on the outcome as one goes up categories, though, the coefficients leave something to be desired.

So, what do you do when you don't want 20 dummy codes representing a single variable in a model for whatever reason? Generally, there are two options. The first is to re-categorize into fewer categories, which is a good option if you have new categories that make substantive sense. For example, if information about household size were available in this data set, I would consider dividing the lower bound of each category by the household size to determine to create a smaller set of categories that correspond to different cutoffs based on federal poverty limits (e.g., 100% FPL or less, 100.1%-200% FPL, >200.1% FPL). The downside is that arbitrary re-categorizations may be difficult to justify and difficult to make sense of in the context of the research question. The second option is to treat the variable as "roughly continuous" and include it in the model as a continuous predictor. The downside of this option is that the standard interpretation of the estimated coefficient doesn't hold, so a finding of significance for this variable would have to have a restrained interpretation.

You'll do the latter in this problem set, but it's not the only or even the best choice across situations. As usual, knowledge about the data and the research question will help you make justifiable choices.

END.