Problem Set 5, Fall 2021

Ben Karabinus

```
knitr::opts_chunk$set(echo = TRUE)

# load necessary libraries
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.
3.1 ──

## ✓ ggplot2 3.3.5     ✓ purrr   0.3.4
## ✓ tibble  3.1.4     ✓ dplyr   1.0.7
## ✓ tidyr   1.1.3     ✓ stringr 1.4.0
## ✓ readr   2.0.1     ✓ forcats 0.5.1

## ── Conflicts ────────────────────────────────────────── tidyverse_conflict
s() ──
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(lmtest) # For lrtest()

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

library(ggplot2)
```

Question 1 - 10 points

The relationship between percents, odds, and odds ratios is salient to interpreting logistic regression output.

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

1) Write a function to compute the probability from the odds.

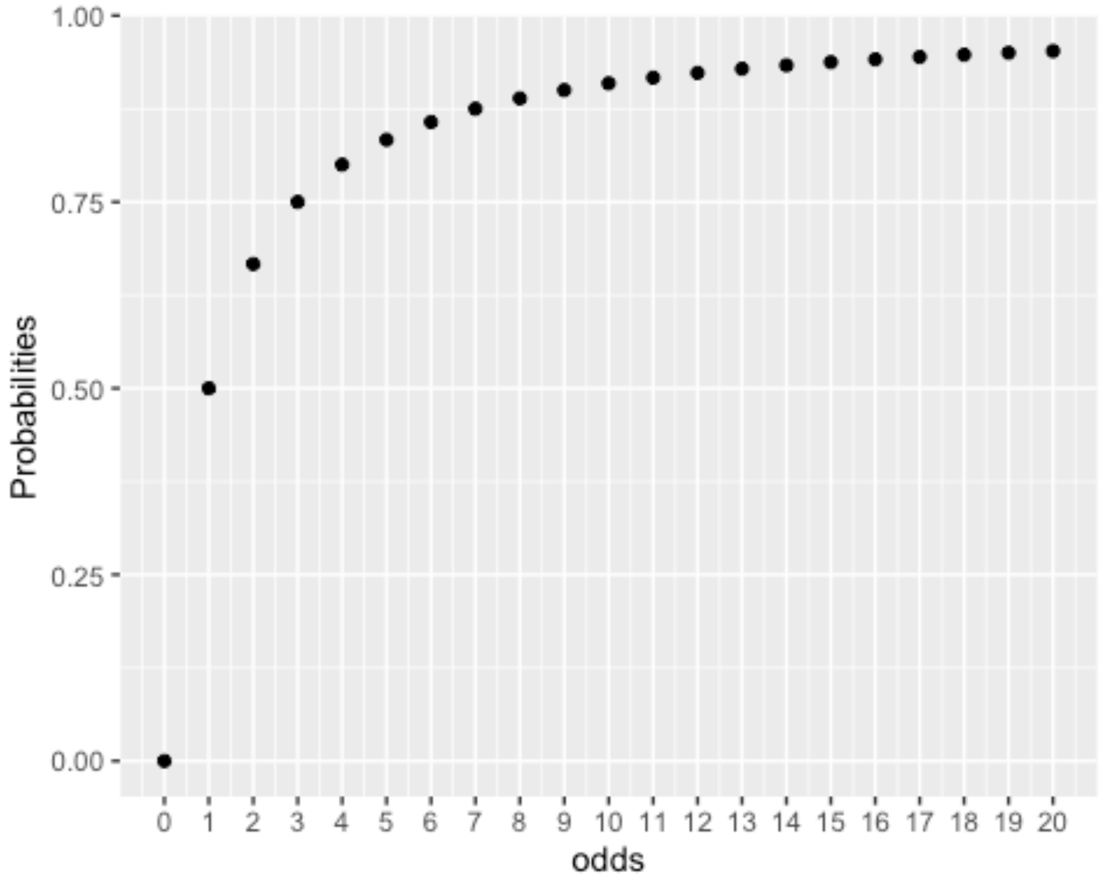
```
# function to compute probability from odds
prob.from.odds <- function(b){
  p <- b/(1+b)
  return(p)
}
```

2) 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? Be sure that your outputted probabilities display in your knitted document.

```
# five
five <- prob.from.odds(5)
# ten
ten <- prob.from.odds(10)
# fifteen
fifteen <- prob.from.odds(15)
# print probabilities
five

## [1] 0.8333333

ten

## [1] 0.9090909

fifteen

## [1] 0.9375
```

3) 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 dataframe of integers 0-20
prob.df <-data.frame(odds = seq.int(0,20,1))
# apply function to calculate probs
prob.df$probs <- apply(prob.df, 1, prob.from.odds)
# Create prob_plt
prob_plt <- ggplot(prob.df, aes(x=odds, y=probs))+
            geom_point()+
            ylab("Probabilities")+
            scale_x_continuous(breaks = seq.int(0, 20, 1))
prob_plt
```



4) 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:

The computed probability increases rapidly at first and then begins to level off as the associated odds increase and the computed probability approaches 1.0.

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

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 and subset the data
load("pew_data.RData")
pew<-dplyr::select(dat,PPINCIMP,PPGENDER,PPETHM,IDEO,PPEDUCAT,LIFE)
```

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.

```r
# view questions and attributes
attributes(pew$LIFE)$label # LIFE

## [1] "In general, would you say life in America today is better, worse or a
bout the same as it was 50 years ago for people like you?"

attributes(pew$LIFE)$labels
```

```
##                               Refused                          Better toda
y
##                                    -1                                     
1
##                           Worse today About the same as it was 50 years ag
o
##                                     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  1
9  20 
##   66  31  40  91  77 104 144 183 179 167 258 321 378 285 319 486 226 253 16
0 125 
##   21 
##  131
```

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

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

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

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

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

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

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

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

```r
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
```

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

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

```r
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

table(pew$PPEDUCAT, exclude = NULL)

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

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

Your answer here:

18

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

Your answer here:

0

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

Your answer here:

0

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

Your answer here:

0

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

Your answer here:

116

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

Your answer here:

0

Question 3 - 5 points

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.

```
# create pew.complete
pew.complete <- pew
# drop any NA's  (I don't believe there were any but caution)
pew.complete %>% drop_na()

## # A tibble: 4,024 × 6
##    PPINCIMP   PPGENDER   PPETHM     IDEO       PPEDUCAT   LIFE
##    <labelled> <labelled> <labelled> <labelled> <labelled> <labelled>
##  1 16         1          1          1          4          2
##  2 19         2          2          3          4          2
##  3 12         1          4          2          2          1
##  4 12         1          4          3          1          2
##  5 21         1          1          2          3          2
##  6 18         1          5          3          3          1
##  7 19         2          1          2          4          2
##  8 16         2          5          3          4          1
##  9  7         2          1          3          2          2
## 10 10         2          1          2          4          2
## # … with 4,014 more rows

# remove rows containing -1 or -2
#pew.complete <- filter(pew.complete, LIFE != -1, IDEO != -1 )
pew.complete <- filter_all(pew.complete, all_vars(. != -1))
pew.complete <- filter_all(pew.complete, all_vars(. != -2))
# print num rows in each dataframe
nrow(pew)

## [1] 4024

nrow(pew.complete)
```

```
## [1] 3894
```

1) 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:

3894

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:

```
# LIFE
table(pew.complete$LIFE, exclude = NULL)

## 
##    1    2    3 
## 1570 1832  492 

# PPINCIMP
table(pew.complete$PPINCIMP, exclude = NULL)

## 
##   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  1
9  20 
##  61  29  34  88  73 104 131 177 169 159 250 312 368 274 306 478 222 248 16
0 120 
##  21 
## 131 

# PPGENDER
table(pew.complete$PPGENDER, exclude = NULL)

## 
##    1    2 
## 1946 1948 

#PPETHM
table(pew.complete$PPETHM, exclude = NULL)

## 
##    1    2    3    4    5 
## 2780  378  164  423  149 

#IDEO
table(pew.complete$IDEO, exclude = NULL)

## 
##    1    2    3    4    5 
##  314 1091 1616  614  259 

#PPEDUCAT
table(pew.complete$PPEDUCAT, exclude = NULL)
```

```
##
##    1    2    3    4
##  281 1074 1114 1425
```

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

Your answer here:

1832

3) 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:

2062

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.

```
# re-code the outcome variable
pew.complete$worse <- as.factor(ifelse(pew.complete$LIFE == 2, 1, 0))

# display the frequencies of the recoded outcome
table(pew.complete$worse, exclude = NULL)
```

```
##
##    0    1
## 2062 1832
```

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.

```
# set variables to the appropriate type
pew.complete$income <- as.numeric(pew.complete$PPINCIMP)
pew.complete$gender <- as.factor(pew.complete$PPGENDER)
pew.complete$eth <- as.factor(pew.complete$PPETHM)
pew.complete$ideo <- as.factor(pew.complete$IDEO)
pew.complete$edu <- as.factor(pew.complete$PPEDUCAT)
pew.complete$worse <- as.factor(pew.complete$worse)
```

```r
# print data structure
str(pew.complete)
```

```
## tibble [3,894 × 12] (S3: tbl_df/tbl/data.frame)
##  $ PPINCIMP: 'labelled' num [1:3894] 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 [1:3894] 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 [1:3894] 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-His
panic" "Black, Non-Hispanic" ...
##  $ IDEO    : 'labelled' num [1:3894] 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" "Conser
vative" "Moderate" ...
##  $ PPEDUCAT: 'labelled' num [1:3894] 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 [1:3894] 2 2 1 2 2 1 2 1 2 2 ...
##   ..- attr(*, "label")= chr "In general, would you say life in America tod
ay is better, worse or about the same as it was 50 years ago for people like
you?"
##   ..- 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 was 50 years ago"
##  $ worse   : Factor w/ 2 levels "0","1": 2 2 1 2 2 1 2 1 2 2 ...
##  $ income  : num [1:3894] 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/ 5 levels "1","2","3","4",..: 1 3 2 3 2 3 2 3 3 2 ..
.
##  $ 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.

```
# create glm (binomial)
Model.1 <- glm(worse ~ income+gender,
               data=pew.complete, family="binomial")
# print model summary
summary(Model.1)
```

```
## 
## Call:
## glm(formula = worse ~ income + gender, family = "binomial", data = pew.com
plete)
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -1.4655  -1.1023  -0.9518   1.2064   1.4712  
## 
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)    
## (Intercept)  0.500606   0.105903   4.727 2.28e-06 ***
## income      -0.055665   0.007135  -7.802 6.09e-15 ***
## gender2      0.210773   0.065059   3.240   0.0012 ** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 5384.6  on 3893  degrees of freedom
## Residual deviance: 5307.0  on 3891  degrees of freedom
## AIC: 5313
## 
## Number of Fisher Scoring iterations: 4
```

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

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

```
## 
## Call:
## glm(formula = worse ~ income + gender + eth + edu, family = "binomial",
##     data = pew.complete)
## 
## Deviance Residuals: 
```

```
##     Min       1Q   Median       3Q      Max  
## -1.5786  -1.1040  -0.8716   1.1667   1.6828  
## 
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)    
## (Intercept)  0.51597    0.15458   3.338 0.000844 ***
## income      -0.04750    0.00797  -5.959 2.53e-09 ***
## gender2      0.20024    0.06570   3.048 0.002306 ** 
## eth2        -0.45265    0.11458  -3.950 7.80e-05 ***
## eth3        -0.08136    0.16634  -0.489 0.624749    
## eth4        -0.31838    0.11016  -2.890 0.003852 ** 
## eth5        -0.03420    0.17183  -0.199 0.842257    
## edu2         0.06893    0.13943   0.494 0.621044    
## edu3         0.23814    0.14104   1.688 0.091316 .  
## edu4        -0.33809    0.14444  -2.341 0.019242 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 5384.6  on 3893  degrees of freedom
## Residual deviance: 5238.0  on 3884  degrees of freedom
## AIC: 5258
## 
## Number of Fisher Scoring iterations: 4
```

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

```
# create glm (binomial)
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 = "binomi
al", 
##     data = pew.complete)
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -1.6475  -1.1055  -0.8357   1.1643   1.7785  
## 
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)    
## (Intercept)  0.817519   0.189737   4.309 1.64e-05 ***
## income      -0.048568   0.008005  -6.067 1.30e-09 ***
## gender2      0.229493   0.066200   3.467 0.000527 ***
## eth2        -0.404730   0.115551  -3.503 0.000461 ***
## eth3        -0.046298   0.167266  -0.277 0.781942    
## eth4        -0.295706   0.110867  -2.667 0.007648 ** 
## eth5        -0.026249   0.172180  -0.152 0.878833    
## edu2         0.061066   0.139792   0.437 0.662235    
## edu3         0.246557   0.141439   1.743 0.081297 .  
## edu4        -0.303651   0.144954  -2.095 0.036187 *  
## ideo2       -0.258686   0.131067  -1.974 0.048417 *  
## ideo3       -0.313375   0.126882  -2.470 0.013518 *  
## ideo4       -0.646659   0.144677  -4.470 7.83e-06 ***
## ideo5       -0.308190   0.172905  -1.782 0.074680 .  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 5384.6  on 3893  degrees of freedom
## Residual deviance: 5214.7  on 3880  degrees of freedom
## AIC: 5242.7
## 
## Number of Fisher Scoring iterations: 4
```

Question 5 - 10 points

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

Nested model test 1: Model 1 vs Model 2

```
# alternate method
#anova(Model.1, Model.2, test = "LRT")
# conduct lrttest
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 -2653.5                         
## 2  10 -2619.0  7 68.971  2.385e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Nested model test 2: Model 2 vs Model 3

```
# alternate method
#anova(Model.2, Model.3, test = "LRT")
# conduct lrtest
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 -2619.0
## 2  14 -2607.3  4 23.322  0.0001092 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

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

Your answer here:

I would choose Model.2 the addition of ethnicity and education contribute to a significant increase in predictive power of the model.

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

Your answer here:

I would choose Model.3. The addition of the ideology predictor variable contributed statistically significant predictive power to the model.

3) 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:

I would choose “Model.3.” Model.3 had significantly more predictive power than Model.1 and Model.2.

Question 6 - 10 points

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. 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, compute the predicted values and display a table of the predicted outcome and the actual outcome.

```r
# create prediction
prediction <- predict(Model.3,pew.complete, type="response")
# binarize the outcome
prediction <- as.factor(ifelse(prediction > 0.5, 1, 0))
```

```
# create tables to show actuals and predicted
actuals <- table(pew.complete$worse)
predicted <- table(prediction)
actuals

##
##    0    1
## 2062 1832

predicted

## prediction
##    0    1
## 2255 1639
```

Next, create your confusion matrix using the table() function

```
# create the confusion matrix using table function
confusion.matrix <- table(pew.complete$worse,prediction)
# print the confusion matrix
confusion.matrix

##    prediction
##        0    1
##   0 1345  717
##   1  910  922
```

1) How many true positives did your model produce?

Your answer here:

922

2) How many true negatives did your model produce?

Your answer here:

1345

3) How many false positives did your model produce?

Your answer here:

717

4) How many false negatives did your model produce?

Your answer here:

910

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

```r
# compute the accuracy of the model
# true positives and true negatives are on the diagonals
accuracy <- sum(diag(confusion.matrix))/sum(confusion.matrix)
# compute precision
# (true positive/(true positive + false positive))
precision <- confusion.matrix[2,2]/sum(confusion.matrix[,2])
# Code to compute recall
# (true positive/(true positive + false negative)
recall <- confusion.matrix[2,2]/sum(confusion.matrix[2,]) # Again,
# compute F1 score
F1 <- 2*((precision*recall)/(precision+recall))
```

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

Your answer here:

0.582177709296353

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

Your answer here:

o.562538133007932

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

Your answer here:

0.503275109170306

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

Your answer here:

0.531259003169116

** Strictly speaking, this variable isn't continuous; rather, it's a 21-category variable. Having 20 dummy codes to represent one variable in a model is unusual in practice. Although not ideal from a statistician's perspective, 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, 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.