Problem Set 5, Winter 2021

Adam Ten Hoeve

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

# Load any packages, if any, that you use as part of your answers here
# For example:
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lmtest)
```

```
## Loading required package: zoo

##
## Attaching package: 'zoo'

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

Question 1 - 10 points

The relation of odds and odds ratios to probabilities is important to interpretation of the output of a logistic regression model.

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

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

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.

3) Create a plot that visually demonstrates how the probability changes within in the range $[0.1, 20]$. ; probability should be on the y axis and odds should be on the x axis.

4) Answer a question about the plot, which is shown below the lask code chunk for this question.

```
# 1) Write a function

prob.from.odds<-function(b){
  # Your function code here
  return(b / (b+1))
}
```

```
# 1) Test your function. Use 5, 10, and 20 as inputs for this function.
prob.from.odds(5)
```

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

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

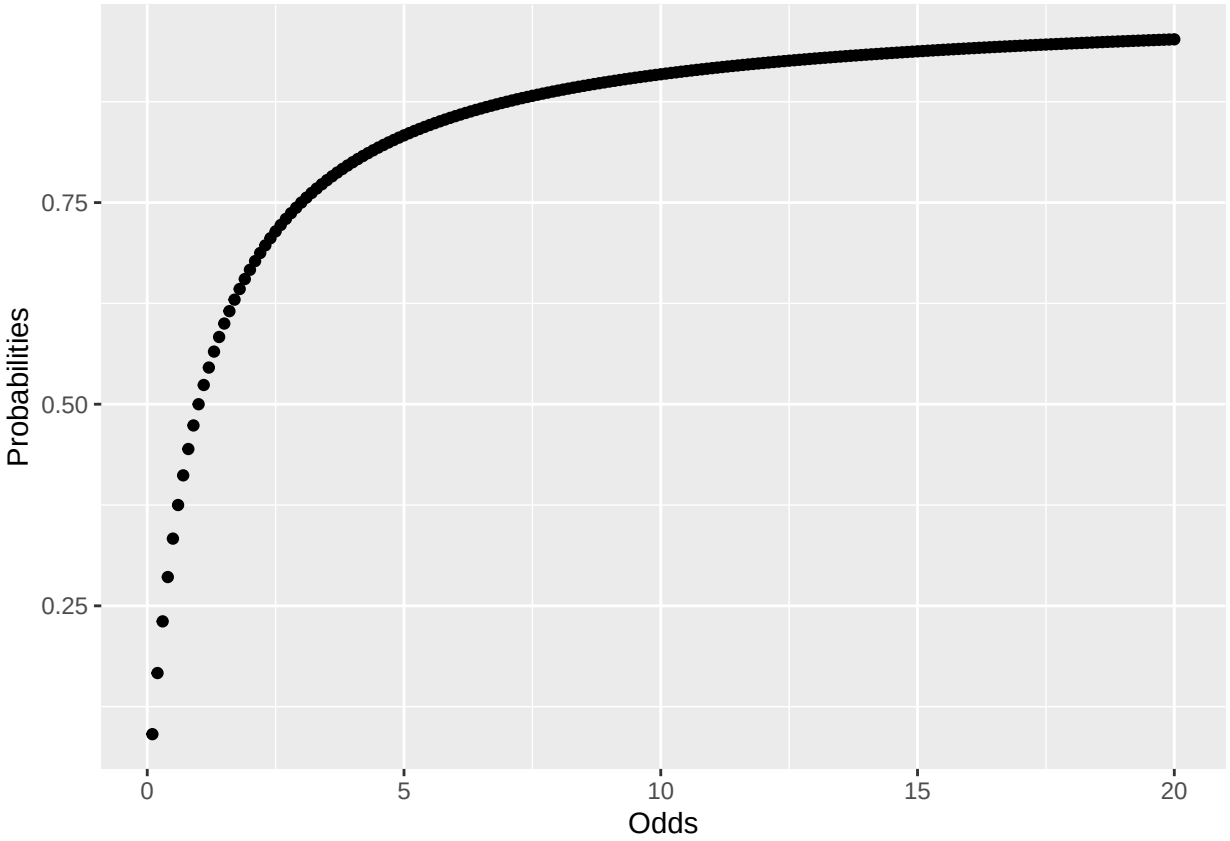
```
## [1] 0.9090909
```

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

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

```
# 1) Create your plot. Probability should be on the Y-axis and odds should be on the X-axis.
odds = seq(0.1, 20, 0.1)
probs = prob.from.odds(odds)
odds.df = data.frame(odds, probs)

g = ggplot(odds.df, aes(x=odds, y=probs)) +
      geom_point() +
      labs(x="Odds", y="Probabilities")
g
```



4) 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 of an event increases, so does the probability of that event.

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.

The variable "LIFE" contains the responses of participants to the following question:

"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?"

1 = Better today

2 = Worse today

3 = About the same as it was 50 years ago

-1 = Refused

Preamble to Questions 2-5 - Read this before starting on Question 2.

Using the data contained in "pew", you will fit two 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.

Question 2 - 5 points

Before beginning your analysis, you will need to do some processing of your data:

1) Recode the LIFE variable such that "Worse today" equals 1 and the other responses are equal to zero.
2) Set the four predictors you will use in the two models - income, gender, ethnicity, and education - to the variable types indicated in the preamble.

Run the provided code in the chunk below first.

```
load("pew_data.RData")
pew<-dplyr::select(dat,PPINCIMP,PPGENDER,PPETHM,IDEO,PPEDUCAT,LIFE)
pew<-filter(pew,LIFE>0) # filters out cases with responses less than zero.

# This code displays counts of different responses for six variables in the pew data.

table(pew$PPINCIMP)
```

```
##
##   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20
##  66  30  37  90  77 104 144 183 179 164 257 321 377 284 317 486 225 251 160 123
##  21
## 131
```

```
table(pew$PPGENDER)
```

```
##
##    1    2
## 1988 2018
```

```
table(pew$PPETHM)
```

```
##
##    1    2    3    4    5
## 2849  391  166  443  157
```

```
table(pew$IDEO)
```

```
##
##   -1    1    2    3    4    5
##  112  314 1091 1616  614  259
```

```
table(pew$PPEDUCAT)
```

```
##
##    1    2    3    4
##  301 1124 1142 1439
```

```r
table(pew$LIFE)
```

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

```r
# This code displays the questions and labels associated with different numeric values in six variables

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

```
##            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
```

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

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

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

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

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

```
attributes(pew$PPEDUCAT)$labels #education
```

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

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

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

Next, write the processing code needed to recode the outcome variable and to set the predictor variables to the correct variable types. Once you've done this, display your data set using the str() function.

```
# Your processing code here
pew$LIFE = ifelse(pew$LIFE==2, 1, 0)
pew$LIFE = as.factor(pew$LIFE)
pew$PPINCIMP = as.numeric(pew$PPINCIMP)
pew$PPGENDER = as.factor(pew$PPGENDER)
pew$PPETHM = as.factor(pew$PPETHM)
pew$PPEDUCAT = as.factor(pew$PPEDUCAT)

# Don't forget to display your final data set with the str() function!
str(pew)
```

```
## tibble [4,006 x 6] (S3: tbl_df/tbl/data.frame)
##  $ PPINCIMP: num [1:4006] 16 19 12 12 21 18 19 16 7 10 ...
##  $ PPGENDER: Factor w/ 2 levels "1","2": 1 2 1 1 1 1 2 2 2 2 ...
##  $ PPETHM  : Factor w/ 5 levels "1","2","3","4",..: 1 2 4 4 1 5 1 5 1 1 ...
##  $ IDEO    : 'labelled' num [1:4006] 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: Factor w/ 4 levels "1","2","3","4": 4 4 2 1 3 3 4 4 2 4 ...
##  $ LIFE    : Factor w/ 2 levels "0","1": 2 2 1 2 2 1 2 1 2 2 ...
```

Question 3 - 10 points

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

Now that your data have been processed, please fit two logistic regression models as described in the preamble and display the results using the summary() function. As a reminder, here are the two models:

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.

```r
# Your code for Models 1 and 2 here
glm.1 = glm(LIFE~PPINCIMP+PPGENDER, data=pew, family="binomial")
glm.2 = glm(LIFE~PPINCIMP+PPGENDER+PPETHM+PPEDUCAT, data=pew, family="binomial")

# Don't forget to use the summary() function to display the results of both models!
summary(glm.1)
```

```
##
## Call:
## glm(formula = LIFE ~ PPINCIMP + PPGENDER, family = "binomial",
##     data = pew)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4769  -1.1089  -0.9309   1.1990   1.4711
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.513671   0.104066   4.936 7.97e-07 ***
## PPINCIMP    -0.056283   0.007023  -8.015 1.11e-15 ***
## PPGENDER2    0.223664   0.064154   3.486  0.00049 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5542.9  on 4005  degrees of freedom
## Residual deviance: 5459.4  on 4003  degrees of freedom
## AIC: 5465.4
##
## Number of Fisher Scoring iterations: 4
```

```r
summary(glm.2)
```

```
##
## Call:
## glm(formula = LIFE ~ PPINCIMP + PPGENDER + PPETHM + PPEDUCAT,
##     family = "binomial", data = pew)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5849  -1.1124  -0.8672   1.1591   1.6923
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.538362   0.150663   3.573 0.000353 ***
## PPINCIMP    -0.046946   0.007863  -5.971 2.36e-09 ***
## PPGENDER2    0.213332   0.064795   3.292 0.000993 ***
## PPETHM2     -0.417236   0.112437  -3.711 0.000207 ***
## PPETHM3     -0.062723   0.165224  -0.380 0.704224
## PPETHM4     -0.327827   0.108043  -3.034 0.002411 **
## PPETHM5     -0.001744   0.167669  -0.010 0.991699
```

```
## PPEDUCAT2    0.042780   0.135331   0.316 0.751917
## PPEDUCAT3    0.215947   0.137256   1.573 0.115646
## PPEDUCAT4   -0.383604   0.140973  -2.721 0.006506 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5542.9  on 4005  degrees of freedom
## Residual deviance: 5386.7  on 3996  degrees of freedom
## AIC: 5406.7
##
## Number of Fisher Scoring iterations: 4
```

Question 4 - 10 points

Conduct a likelihood ratio test between these two models. Make sure that your output for this analysis makes sense for what you're doing (hint: check the degrees of freedom).

```
# Your code for the likelihood ratio test here
lrtest(glm.1, glm.2)
```

```
## Likelihood ratio test
##
## Model 1: LIFE ~ PPINCIMP + PPGENDER
## Model 2: LIFE ~ PPINCIMP + PPGENDER + PPETHM + PPEDUCAT
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   3 -2729.7
## 2  10 -2693.3  7 72.728  4.145e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Next, state the null hypothesis of the likelihood ratio test and interpret the obtained result of the test.

Your answer here: The null hypothesis is that the simple and complex models are not significantly different (i.e. have statistically similar goodness of fits).

Based on the results of the likelihood ratio test, which model would you choose?

Your answer here: From the test, we get a p-value of $4.145e-13$. This is very small, therefor we reject the null hypothesis and assert that there is a significant difference between the models, so we should use the more complex model (model 2).

Question 5 - 15 points

Please display confusion matrices for both of the models above and compute the accuracy and precision of both models. Which model is best if choosing based on accuracy? Which model is best if choosing based on precision?

```r
# Your code here
pred.1 = ifelse(predict(glm.1, type="response")<=0.5, 0, 1)
pred.2 = ifelse(predict(glm.2, type="response")<=0.5, 0, 1)

conf.1 = table(pred.1, pew$LIFE)
conf.2 = table(pred.2, pew$LIFE)

conf.1
```

```
##
## pred.1    0    1
##      0 1432 1079
##      1  674  821
```

```r
conf.2
```

```
##
## pred.2    0    1
##      0 1335  910
##      1  771  990
```

```r
# Calculate accuracy = (TP + TN) / (TP + TN + FP + FN)
acc.1 = (conf.1[1,1]+conf.1[2,2]) / sum(conf.1)
acc.2 = (conf.2[1,1]+conf.2[2,2]) / sum(conf.2)
print("Accuracy:")
```

```
## [1] "Accuracy:"
```

```r
acc.1
```

```
## [1] 0.5624064
```

```r
acc.2
```

```
## [1] 0.5803794
```

```r
# Calculate precision = TP / (TP + FP)
prec.1 = (conf.1[2,2]) / (conf.1[2,2] + conf.1[2,1])
prec.2 = (conf.2[2,2]) / (conf.2[2,2] + conf.2[2,1])
print("Precision:")
```

```
## [1] "Precision:"
```

```r
prec.1
```

```
## [1] 0.5491639
```

```
prec.2
```

```
## [1] 0.5621806
```

Best model based on accuracy: Model 2

Best model based on precision: Model 2