Ordinal Regression Model

Group 97

11/15/2021

Goals:

1) build an ordinal regression model (first phase - progress report)
2) improve and evaluate model (compare with other models)
3) generate predicted probabilities for the visualization

```
## -- Attaching packages ---------------------------------------------------------------------------------------

## v ggplot2 3.3.3     v dplyr   1.0.3
## v tibble  3.0.1     v stringr 1.4.0
## v tidyr   1.0.3     v forcats 0.5.0
## v purrr   0.3.4

## -- Conflicts ------------------------------------------------------------------------------------------------
## x dplyr::between()   masks data.table::between()
## x dplyr::filter()    masks stats::filter()
## x dplyr::first()     masks data.table::first()
## x dplyr::lag()       masks stats::lag()
## x dplyr::last()      masks data.table::last()
## x purrr::transpose() masks data.table::transpose()

## Loading required package: foreign

## Loading required package: MASS

## Warning: package 'MASS' was built under R version 4.0.5

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select

## Loading required package: Hmisc

## Warning: package 'Hmisc' was built under R version 4.0.5
```

```
## Loading required package: lattice

## Loading required package: survival

## Loading required package: Formula

##
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:dplyr':
##
##     src, summarize

## The following objects are masked from 'package:base':
##
##     format.pval, units

## Loading required package: reshape2

##
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':
##
##     smiths

## The following objects are masked from 'package:data.table':
##
##     dcast, melt

## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode

## The following object is masked from 'package:purrr':
##
##     some

## Warning: package 'lmtest' was built under R version 4.0.5

## Loading required package: zoo

##
## Attaching package: 'zoo'
```

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

```
## Warning: package 'AER' was built under R version 4.0.5
```

```
## Loading required package: sandwich
```

```
## Warning: package 'sandwich' was built under R version 4.0.5
```

Step 1: Build the Ordinal Regression Model (First Phase)

1a: Loading the data

If you are running this on your own machine, download the cleaned, imputed data set from our github repo at: https://github.gatech.edu/DVA-group97/our-lovely-repo/blob/master/Merging_and_Cleaning_Files/imputed_data.csv

We cleaned the data previously. See folder 1-Cleaning_Data as part of this submission for the cleaning code.

In addition to loading the data, we re-type the categorical variables as factors as opposed to integers. The survey responses were originally numeric codes in the raw data.

```
## Classes 'data.table' and 'data.frame':   318989 obs. of  17 variables:
##  $ WEEK          : Factor w/ 21 levels "13","14","15",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ DATE          : chr  "8/19/2020" "8/19/2020" "8/19/2020" "8/19/2020" ...
##  $ PWEIGHT       : num  1149.7 1028.2 193.5 136.7 51.5 ...
##  $ EEDUC         : Factor w/ 7 levels "1","2","3","4",..: 7 7 6 4 4 3 5 4 6 7 ...
##  $ RHISPANIC     : Factor w/ 2 levels "1","2": 1 1 1 2 1 2 1 1 1 1 ...
##  $ RRACE         : Factor w/ 4 levels "1","2","3","4": 1 1 1 4 1 1 1 1 1 1 ...
##  $ THHLD_NUMKID  : Factor w/ 6 levels "0","1","2","3",..: 4 3 4 3 2 4 2 3 2 2 ...
##  $ THHLD_NUMPER  : Factor w/ 10 levels "1","10","2","3",..: 6 5 6 5 3 6 4 6 4 4 ...
##  $ INCOME        : Factor w/ 8 levels "1","2","3","4",..: 8 4 6 5 2 1 4 3 6 7 ...
##  $ CURFOODSUF    : Factor w/ 4 levels "1","2","3","4": 2 1 1 2 3 2 4 2 1 1 ...
##  $ TSPNDFOOD     : num  400 200 200 200 100 250 50 150 200 140 ...
##  $ TSPNDPRPD     : num  75 250 20 100 15 80 0 0 50 40 ...
##  $ EST_ST        : Factor w/ 51 levels "1","10","11",..: 1 1 11 11 49 5 42 33 42 49 ...
##  $ COMPAVAIL     : Factor w/ 5 levels "1","2","3","4",..: 2 1 1 1 2 2 1 3 1 1 ...
##  $ INTRNTAVAIL   : Factor w/ 5 levels "1","2","3","4",..: 2 1 3 1 1 3 1 2 1 1 ...
##  $ SCHLHRS       : Factor w/ 4 levels "1","2","3","4": 4 2 1 3 3 4 2 1 1 3 ...
##  $ RACE_ETHNICITY: chr  "White" "White" "White" "Hispanic" ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

1b: Building the Ordinal Regression Model - First Attempt

We use the polr function from the MASS library to run ordinal regressions with our data. We use ordinal regression because out outcome variable SCHLHRS is a an ordinal variable representing how much live virtual contact students had with teachers. We chose this as out outcome variable because it had not yet been explored in peer-reviewed research.

For the progress report and demo-ing our visualization for user feedback, we built a model with just RACE_ETHNICITY, INCOME, and INTRNTAVAIL (internet availability) to predict SCHLHRS. We chose

these predictors initially because these were previously identified as relevant to the digital divide according to our literature review. We also wanted to limit the number of factors to three to keep our calculator visualization simpler for the user.

In the output table below, all of the predictors at every level were significant at the 5% significance level, except for one level of RACE_ETHNICITY. Each predictor has multiple levels in the model because they are categorical. In the next code cell, we check to see if keeping RACE_ETHNICITY still improves the model by performing a likelihood ratio test.

```
##
## Re-fitting to get Hessian

##                               Value  Std. Error     t value       p value
## RACE_ETHNICITYBlack    -0.05249087 0.019987227  -2.6262205  8.633887e-03
## RACE_ETHNICITYHispanic -0.01473432 0.018651709  -0.7899718  4.295442e-01
## RACE_ETHNICITYOther    -0.14224787 0.022705659  -6.2648643  3.731503e-10
## RACE_ETHNICITYWhite    -0.05655135 0.015826845  -3.5731285  3.527415e-04
## INCOME2                 0.06815524 0.017842963   3.8197268  1.335996e-04
## INCOME3                 0.13543107 0.016977449   7.9771156  1.497926e-15
## INCOME4                 0.20015102 0.015448951  12.9556386  2.183116e-38
## INCOME5                 0.27790115 0.015757462  17.6361622  1.299956e-69
## INCOME6                 0.37003933 0.015020988  24.6348192 5.352962e-134
## INCOME7                 0.43895616 0.016908203  25.9611366 1.361270e-148
## INCOME8                 0.57907336 0.016249227  35.6369779 3.749406e-278
## INTRNTAVAIL2           -0.24324446 0.009789207 -24.8482300 2.702327e-136
## INTRNTAVAIL3           -0.47401877 0.018893876 -25.0884877 6.642154e-139
## INTRNTAVAIL4           -0.72920561 0.037489054 -19.4511607  2.849083e-84
## INTRNTAVAIL5           -0.71268212 0.047253556 -15.0820844  2.124543e-51
## 1|2                    -1.48065233 0.020219952 -73.2272911  0.000000e+00
## 2|3                    -1.17468581 0.020132778 -58.3469302  0.000000e+00
## 3|4                    -0.32124815 0.020012061 -16.0527268  5.470684e-58
```

To see if it was worth keeping the RACE_ETHNICTY variable (since not all levels were significant), we performed a likelihood ratio test to see if adding it to the model significantly improved it beyond the effect of just having more variables in general.

The test came out significant (p-value = 3.538e-11), suggesting that keeping RACE_ETHNICITY in the model likely improves the model even if one level is not significant.

```
##
## Re-fitting to get Hessian

##                    Value  Std. Error     t value
## INCOME2       0.06833818 0.017832944    3.832131
## INCOME3       0.13486530 0.016945406    7.958812
## INCOME4       0.19870693 0.015354255   12.941489
## INCOME5       0.27570705 0.015588620   17.686431
## INCOME6       0.36788007 0.014781711   24.887516
## INCOME7       0.43764246 0.016669478   26.254119
## INCOME8       0.57923551 0.015941156   36.335854
## INTRNTAVAIL2 -0.24353271 0.009784642  -24.889283
## INTRNTAVAIL3 -0.47602425 0.018884225  -25.207508
## INTRNTAVAIL4 -0.73521379 0.037475911  -19.618304
## INTRNTAVAIL5 -0.71881767 0.047233384  -15.218424
```

```
## 1|2          -1.42988642 0.013189420 -108.411620
## 2|3          -1.12390627 0.013052822  -86.104467
## 3|4          -0.27062960 0.012875435  -21.019065

## Likelihood ratio test
##
## Model 1: SCHLHRS ~ RACE_ETHNICITY + INCOME + INTRNTAVAIL
## Model 2: SCHLHRS ~ INCOME + INTRNTAVAIL
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1  18 -326354
## 2  14 -326382 -4 54.823  3.538e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Step 2: Improve and Evaluate the Model

For the first phase of our project (see first part above), we built a model with just 3 predictors to ensure the design of our tool was simple for our users. Like mentioned previously, we chose the 3 based off our literature survey. The 3 predictors were: RACE_ETHNICITY, INCOME, INTRNTAVAIL.

According to our user survey, there was a slight preference for more variables. The next preference was to keep the original 3 variables. Based on this feedback, we decided to keep our original 3 predictors and test out adding one more variable to improve the model. We only considered up to 4 variables to keep the model and visualization simple for users, especially since a significant proportion of respondents preferred to keep only 3 predictors.

In the code cells following, we build 4 more models (each with one predictor added to the original 3) and compare them to the original model and among each other to evaluate them and select the best one.

We evaluate the models based on the following:

- Check assumptions log regression(see report)
- Accuracy (limitations)
- Likelihood ratio test
- AIC

2a: Creating Training, Validation, and Test Data Sets

So we can evaluate the models we create and pick the most suitable one, let's divide the data into training, Validation, and test data sets.

```
## [1] TRUE
```

2b: Building More Models with One More Predictor

Below, we create more models with one more variable added to our original model with 3 variables (RACE_ETHNICITY, INCOME, INTRNTAVAIL)

We considered adding one of the following variables: COMPAVAIL,THHLD_NUMKID, EEDUC, CUR-FOODSUF. These variables were included in the realm of possibility because they seemed relevant based on our lit search and they were also present in the same weeks as our outcome variable SCHLHRS.

After we build all the models below, he have 5 models in total to compare:

- m1: SCHLHRS ~ RACE_ETHNICITY+INCOME+INTRNTAVAIL (original)
- m2: SCHLHRS ~ RACE_ETHNICITY+INCOME+INRNTAVAIL+COMPAVAIL
- m3: SCHLHRS ~ RACE_ETHNICITY+INCOME+INRNTAVAIL+CURFOODSUF
- m4: SCHLHRS ~ RACE_ETHNICITY+INCOME+INRNTAVAIL+EEDUC
- m5: SCHLHRS ~ RACE_ETHNICITY+INCOME+INRNTAVAIL+THHLD_NUMKID

In the cell below, we build the 5 models and print the summary tables for each.

```
##
## Re-fitting to get Hessian

## Call:
## polr(formula = SCHLHRS ~ RACE_ETHNICITY + INCOME + INTRNTAVAIL,
##     data = train)
##
## Coefficients:
##                            Value Std. Error  t value
## RACE_ETHNICITYBlack    -0.07249    0.02391  -3.0317
## RACE_ETHNICITYHispanic -0.01620    0.02232  -0.7256
## RACE_ETHNICITYOther    -0.15198    0.02713  -5.6025
## RACE_ETHNICITYWhite    -0.06048    0.01893  -3.1949
## INCOME2                 0.05423    0.02135   2.5406
## INCOME3                 0.12282    0.02029   6.0517
## INCOME4                 0.19683    0.01849  10.6466
## INCOME5                 0.28085    0.01889  14.8702
## INCOME6                 0.35870    0.01796  19.9778
## INCOME7                 0.42269    0.02023  20.8992
## INCOME8                 0.56768    0.01942  29.2328
## INTRNTAVAIL2           -0.24802    0.01173 -21.1415
## INTRNTAVAIL3           -0.48055    0.02259 -21.2709
## INTRNTAVAIL4           -0.73492    0.04486 -16.3822
## INTRNTAVAIL5           -0.74184    0.05609 -13.2253
##
## Intercepts:
##     Value    Std. Error t value
## 1|2  -1.5009   0.0242   -62.0753
## 2|3  -1.1932   0.0241   -49.5677
## 3|4  -0.3401   0.0239   -14.2149
##
## Residual Deviance: 456224.46
## AIC: 456260.46

##
## Re-fitting to get Hessian

## Call:
## polr(formula = SCHLHRS ~ RACE_ETHNICITY + INCOME + INTRNTAVAIL +
##     COMPAVAIL, data = train)
##
## Coefficients:
##                            Value Std. Error  t value
## RACE_ETHNICITYBlack    -0.07992    0.02393  -3.3396
## RACE_ETHNICITYHispanic -0.01254    0.02234  -0.5611
```

```
## RACE_ETHNICITYOther    -0.15064    0.02716  -5.5474
## RACE_ETHNICITYWhite    -0.05933    0.01894  -3.1321
## INCOME2                 0.04820    0.02138   2.2542
## INCOME3                 0.11276    0.02033   5.5471
## INCOME4                 0.18150    0.01852   9.7984
## INCOME5                 0.26032    0.01893  13.7513
## INCOME6                 0.33027    0.01801  18.3398
## INCOME7                 0.38516    0.02029  18.9824
## INCOME8                 0.52773    0.01949  27.0735
## INTRNTAVAIL2           -0.10596    0.01295  -8.1816
## INTRNTAVAIL3           -0.22347    0.02480  -9.0096
## INTRNTAVAIL4           -0.37530    0.04867  -7.7107
## INTRNTAVAIL5           -0.27627    0.06626  -4.1692
## COMPAVAIL2             -0.31846    0.01382 -23.0451
## COMPAVAIL3             -0.45600    0.02350 -19.4073
## COMPAVAIL4             -0.51663    0.04459 -11.5865
## COMPAVAIL5             -0.66941    0.06005 -11.1468
##
## Intercepts:
##     Value    Std. Error t value
## 1|2  -1.5580   0.0243   -64.1376
## 2|3  -1.2491   0.0242   -51.6575
## 3|4  -0.3931   0.0240   -16.3632
##
## Residual Deviance: 455359.74
## AIC: 455403.74

##
## Re-fitting to get Hessian

## Call:
## polr(formula = SCHLHRS ~ RACE_ETHNICITY + INCOME + INTRNTAVAIL +
##     CURFOODSUF, data = train)
##
## Coefficients:
##                            Value Std. Error  t value
## RACE_ETHNICITYBlack    -0.058142    0.02395  -2.4279
## RACE_ETHNICITYHispanic -0.005708    0.02235  -0.2555
## RACE_ETHNICITYOther    -0.137394    0.02716  -5.0589
## RACE_ETHNICITYWhite    -0.065140    0.01894  -3.4389
## INCOME2                 0.040135    0.02139   1.8762
## INCOME3                 0.093956    0.02041   4.6032
## INCOME4                 0.143868    0.01878   7.6591
## INCOME5                 0.206221    0.01938  10.6421
## INCOME6                 0.267378    0.01866  14.3272
## INCOME7                 0.321489    0.02097  15.3280
## INCOME8                 0.457964    0.02031  22.5432
## INTRNTAVAIL2           -0.207106    0.01195 -17.3350
## INTRNTAVAIL3           -0.408377    0.02299 -17.7657
## INTRNTAVAIL4           -0.661725    0.04514 -14.6580
## INTRNTAVAIL5           -0.682327    0.05629 -12.1217
## CURFOODSUF2            -0.155019    0.01061 -14.6109
## CURFOODSUF3            -0.248443    0.01798 -13.8150
## CURFOODSUF4            -0.324628    0.03311  -9.8040
```

```
##
## Intercepts:
##     Value    Std. Error t value
## 1|2  -1.6244   0.0251   -64.7129
## 2|3  -1.3165   0.0250   -52.6699
## 3|4  -0.4621   0.0248   -18.6097
##
## Residual Deviance: 455875.02
## AIC: 455917.02

##
## Re-fitting to get Hessian

## Call:
## polr(formula = SCHLHRS ~ RACE_ETHNICITY + INCOME + INTRNTAVAIL +
##     EEDUC, data = train)
##
## Coefficients:
##                           Value Std. Error  t value
## RACE_ETHNICITYBlack    -0.05124    0.02397  -2.1377
## RACE_ETHNICITYHispanic  0.03955    0.02250   1.7573
## RACE_ETHNICITYOther    -0.11441    0.02722  -4.2023
## RACE_ETHNICITYWhite    -0.03635    0.01900  -1.9133
## INCOME2                 0.03365    0.02142   1.5706
## INCOME3                 0.07750    0.02047   3.7860
## INCOME4                 0.12432    0.01885   6.5944
## INCOME5                 0.18080    0.01947   9.2850
## INCOME6                 0.23362    0.01885  12.3954
## INCOME7                 0.27667    0.02129  12.9953
## INCOME8                 0.40122    0.02088  19.2194
## INTRNTAVAIL2           -0.24561    0.01174 -20.9166
## INTRNTAVAIL3           -0.46460    0.02263 -20.5320
## INTRNTAVAIL4           -0.71451    0.04492 -15.9051
## INTRNTAVAIL5           -0.71579    0.05619 -12.7381
## EEDUC2                 -0.02469    0.05960  -0.4143
## EEDUC3                  0.04260    0.05211   0.8174
## EEDUC4                  0.16212    0.05153   3.1459
## EEDUC5                  0.20519    0.05233   3.9209
## EEDUC6                  0.30908    0.05173   5.9751
## EEDUC7                  0.35791    0.05200   6.8831
##
## Intercepts:
##     Value    Std. Error t value
## 1|2  -1.3272   0.0551   -24.0851
## 2|3  -1.0190   0.0551   -18.5065
## 3|4  -0.1644   0.0550    -2.9883
##
## Residual Deviance: 455727.19
## AIC: 455775.19

##
## Re-fitting to get Hessian

## Call:
```

```
## polr(formula = SCHLHRS ~ RACE_ETHNICITY + INCOME + INTRNTAVAIL +
##     THHLD_NUMKID, data = train)
##
## Coefficients:
##                            Value Std. Error t value
## RACE_ETHNICITYBlack    -0.07279    0.02394  -3.041
## RACE_ETHNICITYHispanic -0.02582    0.02236  -1.155
## RACE_ETHNICITYOther    -0.16062    0.02717  -5.912
## RACE_ETHNICITYWhite    -0.06692    0.01896  -3.530
## INCOME2                 0.05187    0.02136   2.428
## INCOME3                 0.11872    0.02031   5.846
## INCOME4                 0.19025    0.01850  10.283
## INCOME5                 0.27008    0.01891  14.283
## INCOME6                 0.34547    0.01799  19.207
## INCOME7                 0.40771    0.02027  20.116
## INCOME8                 0.54726    0.01948  28.098
## INTRNTAVAIL2           -0.25747    0.01176 -21.895
## INTRNTAVAIL3           -0.49551    0.02264 -21.887
## INTRNTAVAIL4           -0.74921    0.04492 -16.680
## INTRNTAVAIL5           -0.74324    0.05616 -13.235
## THHLD_NUMKID1           0.21815    0.05308   4.110
## THHLD_NUMKID2           0.37588    0.05310   7.079
## THHLD_NUMKID3           0.38745    0.05376   7.207
## THHLD_NUMKID4           0.45619    0.05609   8.134
## THHLD_NUMKID5           0.31607    0.05948   5.314
##
## Intercepts:
##     Value    Std. Error t value
## 1|2  -1.2027   0.0575   -20.9320
## 2|3  -0.8947   0.0574   -15.5820
## 3|4  -0.0402   0.0574    -0.7008
##
## Residual Deviance: 455818.62
## AIC: 455864.62
```

2b: Comparing AIC Scores

We pulled the AIC scores for each model above and print them in the table below. We can use AIC to evaluate model fit.

Lower scores are better. m2 (RACE_ETHNICITY, INCOME, INTRNTAVAIL, and COMPAVAIL) had the lowest AIC score of 455403.7.

```
##   models      aic
## 1     m1 456260.5
## 2     m2 455403.7
## 3     m3 455917.0
## 4     m4 455775.2
## 5     m5 455864.6
```

2c: Getting Accuracy From Predictions

In the code cell below, we generate predictions and calculate accuracy based on the validation data set.

Model m3 appear to have the highest accuracy (0.6196786). However, all 5 models have a very similar level of accuracy, differing by less than .001

```
##   models  accuracy
## 1     m1 0.6194905
## 2     m2 0.6195950
## 3     m3 0.6196786
## 4     m4 0.6195532
## 5     m5 0.6196159
```

It's possible that all of the models have a similar accuracy level because they mostly predict the value "4" for SCHLHRS, since this is the most common outcome for that variable across the full data set. The models assign values based on the highest probability. "4" = 4 or more days of virtual contact.

62.3% of the full data set has the value of "4" for SCHLHRS. This matches up with the models' accuracy levels of the models, which were all slightly below 62%.

In general, most of the models predicted "4" for most of the validation set. So it makes sense if they predict about 62% correctly if they are labeling almost all of them as "4" as the data in general is ~62% the value "4".

See this article for limitations of using accuracy as a means of evaluating models: https://medium.com/@limavallantin/why-you-should-not-trust-only-in-accuracy-to-measure-machine-learning-performance-a72cf00b4516

```
## [1] 0.6230183
```

```
##     1     2     3     4
##     0     0     0 47849
```

```
##     1     2     3     4
##    76     0     0 47773
```

```
##     1     2     3     4
##    48     0     0 47801
```

```
##     1     2     3     4
##    29     0     0 47820
```

```
##     1     2     3     4
##    18     0     0 47831
```

2d: Likelihood Ratio Tests (Wilks Test)

https://www.statology.org/likelihood-ratio-test-in-r/

The likelihood ratio test compares the goodness of fit of two nested models. Meaning, one "fuller" model with more variables vs one "nested" model that has the same variables but less of them.

The null hypothesis is that both models are equally good, suggesting that you should use the model with less variables. The alternative hypothesis is that full model fits the data better, suggesting that you should use that one instead.

For this test, we can only compare m2-5 (full models) with m1 (nested models). This is because model m1 has 3 variables nested relative to m2-5 which have the same 4 as m1 plus an additional.

In total, we perform 4 tests, comparing m2-m5 each with m1. All 4 came out significant with p values of (2.2e-16), suggesting that adding a fourth variable (any of these 4) provides a better fit than just using the 3 from m1.

```
## Likelihood ratio test
##
## Model 1: SCHLHRS ~ RACE_ETHNICITY + INCOME + INTRNTAVAIL
## Model 2: SCHLHRS ~ RACE_ETHNICITY + INCOME + INTRNTAVAIL + COMPAVAIL
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1  18 -228112
## 2  22 -227680  4 864.72  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Likelihood ratio test
##
## Model 1: SCHLHRS ~ RACE_ETHNICITY + INCOME + INTRNTAVAIL
## Model 2: SCHLHRS ~ RACE_ETHNICITY + INCOME + INTRNTAVAIL + CURFOODSUF
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1  18 -228112
## 2  21 -227938  3 349.44  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Likelihood ratio test
##
## Model 1: SCHLHRS ~ RACE_ETHNICITY + INCOME + INTRNTAVAIL
## Model 2: SCHLHRS ~ RACE_ETHNICITY + INCOME + INTRNTAVAIL + EEDUC
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1  18 -228112
## 2  24 -227864  6 497.27  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Likelihood ratio test
##
## Model 1: SCHLHRS ~ RACE_ETHNICITY + INCOME + INTRNTAVAIL
## Model 2: SCHLHRS ~ RACE_ETHNICITY + INCOME + INTRNTAVAIL + THHLD_NUMKID
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1  18 -228112
## 2  23 -227909  5 405.84  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

2e: Selecting and Testing the Final Model

We decided to select model m2 (SSCHLHRS ~ RACE_ETHNICITY+INCOME+INRNTAVAIL+COMPAVAIL) because it had the lowest AIC value and had the highest accuracy (along with m3). We also knew a model with 4 variables rather than 3 was preferable because of the likelihood ratio tests.

Looking at the summary table, all the variables are significant at every level except for RACE_ETHNICITY. We looked into that variable earlier when we build the first model (m1), and it did improve the model as opposed to leaving out.

```
##
## Re-fitting to get Hessian

##
## z test of coefficients:
##
##                          Estimate Std. Error  z value  Pr(>|z|)
## RACE_ETHNICITYBlack    -0.079920   0.023931  -3.3396  0.000839 ***
## RACE_ETHNICITYHispanic -0.012538   0.022344  -0.5611  0.574715
## RACE_ETHNICITYOther    -0.150642   0.027155  -5.5474 2.899e-08 ***
## RACE_ETHNICITYWhite    -0.059334   0.018944  -3.1321  0.001736 **
## INCOME2                 0.048202   0.021384   2.2542  0.024186 *
## INCOME3                 0.112763   0.020328   5.5471 2.904e-08 ***
## INCOME4                 0.181497   0.018523   9.7984 < 2.2e-16 ***
## INCOME5                 0.260318   0.018930  13.7513 < 2.2e-16 ***
## INCOME6                 0.330268   0.018008  18.3398 < 2.2e-16 ***
## INCOME7                 0.385161   0.020290  18.9824 < 2.2e-16 ***
## INCOME8                 0.527731   0.019492  27.0735 < 2.2e-16 ***
## INTRNTAVAIL2           -0.105955   0.012950  -8.1816 2.802e-16 ***
## INTRNTAVAIL3           -0.223474   0.024804  -9.0096 < 2.2e-16 ***
## INTRNTAVAIL4           -0.375303   0.048673  -7.7107 1.251e-14 ***
## INTRNTAVAIL5           -0.276266   0.066263  -4.1692 3.056e-05 ***
## COMPAVAIL2             -0.318458   0.013819 -23.0451 < 2.2e-16 ***
## COMPAVAIL3             -0.456002   0.023496 -19.4073 < 2.2e-16 ***
## COMPAVAIL4             -0.516626   0.044589 -11.5865 < 2.2e-16 ***
## COMPAVAIL5             -0.669406   0.060054 -11.1468 < 2.2e-16 ***
## 1|2                    -1.557970   0.024291 -64.1376 < 2.2e-16 ***
## 2|3                    -1.249144   0.024181 -51.6575 < 2.2e-16 ***
## 3|4                    -0.393138   0.024026 -16.3632 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

To evaluate the final model, we used the test data to get accuracy, which was 62.1%.

Although this number may seem low, we previously explained the limitations of using accuracy above. The accuracy like mentioned before, is more a reflection of the amount of respondents who selected #4, which is the most likely response.

```
## [1] 0.6209455
```

Step 3: Generate Predicted Probabilities with Final Model

3a: Predicted probabilities for all value combos

Since we chose m2 as our final model, we generated predicted probabilities for all the possible value combinations with the 4 predictors (1000 total = 5 categories for RACE_ETHNICITY * 8 categories for INCOME * 5 categories for INTRNTAVAIL * 5 categories for COMPAVAIL). First we made an data frame with all the combinations as input to generate predictions. Then, we made the predictions and cleaned the data to create a csv where each row is has a predicted probability for each of the combos for each level of SCHLHRS (4000 rows total = 1000 combos * 4 levels of SCHLHRS)

3b: Predicted values for comparison purposes

Above, we generated predicted probabilities for the different combinations of the variable values.

Below, we generate predicted values for comparison purposes.

The difference here is that the input data we use for prediction is the full sample. We need to do this separately with the full sample because the full sample is not evenly distributed across the 1000 combinations of predictor values (that we used above).

With the predictions, we calculated the quantiles for each level of SCHLHRS (1 to 4) that we can use for the comparison for our tool.

```
## Using quantile as id variables
```