HDS502 - Assignment 2

Matthew Onimus

10Jul2022

R Set Up and Data Read

I will start by loading the in the packages to be used for the initial data analysis. I will mainly be working in the `tidyverse` ecosystem.

```
library(tidyverse) # used for reading data, data cleaning, visualizations
library(haven) # used to read in SAS, Strata, etc. files
library(survey) # used for assignment 4
library(foreign) # used for assignment 4
library(tidymodels) # a group of packages used to build models
library(here) # used for easy file path finding
library(rlang) # fancy data masking functions
library(kableExtra) # used to create tables in the document

sessionInfo()
```

```
## R version 3.6.3 (2020-02-29)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Linux Mint 20.3
##
## Matrix products: default
## BLAS:   /usr/lib/x86_64-linux-gnu/blas/libblas.so.3.9.0
## LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.9.0
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] grid      stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] kableExtra_1.3.4   rlang_1.0.2        here_1.0.1         yardstick_0.0.9
##  [5] workflowsets_0.2.1 workflows_0.2.6    tune_0.2.0         rsample_0.1.1
##  [9] recipes_0.2.0      parsnip_0.2.1      modeldata_0.1.1    infer_1.0.0
## [13] dials_0.1.1        scales_1.1.1       broom_0.8.0        tidymodels_0.2.0
## [17] foreign_0.8-75     survey_4.1-1       survival_3.1-8     Matrix_1.2-18
## [21] haven_2.3.1        forcats_0.5.0      stringr_1.4.0      dplyr_1.0.9
```

```
## [25] purrr_0.3.4        readr_2.1.1        tidyr_1.2.0        tibble_3.1.6
## [29] ggplot2_3.3.6      tidyverse_1.3.0
##
## loaded via a namespace (and not attached):
##  [1] colorspace_2.0-1   ellipsis_0.3.2     class_7.3-15       rprojroot_2.0.2
##  [5] fs_1.5.0           rstudioapi_0.13    listenv_0.8.0      furrr_0.2.2
##  [9] prodlim_2019.11.13 fansi_0.5.0        lubridate_1.7.9.2  xml2_1.3.2
## [13] codetools_0.2-16   splines_3.6.3      knitr_1.33         jsonlite_1.7.2
## [17] pROC_1.17.0.1      dbplyr_2.0.0       compiler_3.6.3     httr_1.4.2
## [21] backports_1.2.1    assertthat_0.2.1   fastmap_1.1.0      cli_3.3.0
## [25] htmltools_0.5.2    tools_3.6.3        gtable_0.3.0       glue_1.6.2
## [29] Rcpp_1.0.8.3       cellranger_1.1.0   DiceDesign_1.9     vctrs_0.4.1
## [33] svglite_2.0.0      iterators_1.0.13   timeDate_3043.102  gower_0.2.2
## [37] xfun_0.30          globals_0.14.0     rvest_0.3.6        lifecycle_1.0.1
## [41] future_1.21.0      MASS_7.3-51.5      ipred_0.9-12       hms_1.1.1
## [45] parallel_3.6.3     yaml_2.2.1         rpart_4.1-15       stringi_1.7.6
## [49] foreach_1.5.1      lhs_1.1.1          hardhat_0.2.0      lava_1.6.9
## [53] systemfonts_1.0.1  pkgconfig_2.0.3    evaluate_0.14      lattice_0.20-40
## [57] tidyselect_1.1.2   parallelly_1.24.0  plyr_1.8.6         magrittr_2.0.1
## [61] R6_2.5.1           generics_0.1.2     DBI_1.1.0          pillar_1.6.4
## [65] withr_2.4.3        nnet_7.3-13        modelr_0.1.8       crayon_1.4.2
## [69] utf8_1.2.2         tzdb_0.2.0         rmarkdown_2.14     readxl_1.3.1
## [73] webshot_0.5.2      reprex_0.3.0       digest_0.6.27      munsell_0.5.0
## [77] GPfit_1.0-8        viridisLite_0.4.0  mitools_2.4
```

Next, we need to read the data in to R. It was provided in .dta format from MEDS.

```
h209 <- read_stata(here('assignment2/h209.dta'))
```

Reproducing Example 4

I will start by reproducing example 4 (https://github.com/HHS-AHRQ/MEPS/blob/master/R/workshop_exercises/exercise_4.R) before creating my own model based on the recommended outcome variables.

```
# Keep only needed variables ----------------------------------------------------

h209Sub <- h209 %>%
  select(DUPERSID, VARPSU, VARSTR,
         ADFLST42, AGELAST, SEX, RACETHX, INSCOV18, matches("SAQ"))

# Create variables ----------------------------------------------------------
#  - Convert ADFLST42 from 1/2 to 0/1 (for logistic regression)
#  - Create 'subpop' to exclude people with Missing 'ADFLST42'


h209x <- h209Sub %>%
  mutate(

    # Convert outcome from 1/2 to 0/1:
    flu_shot = case_when(
      ADFLST42 == 1 ~ 1,
      ADFLST42 == 2 ~ 0,
      # note, case when is more sensitive now and will not
      # coerce variables for you, you need to explicitly state `as.numeric`
      # for this to run
```

```
      TRUE ~ as.numeric(ADFLST42)),

    # Create subpop to exclude Missings
    subpop = (ADFLST42 >= 0))

saq_dsgn = svydesign(
  id = ~VARPSU,
  strata = ~VARSTR,
  weights = ~SAQWT18F,
  data = h209x,
  nest = TRUE)

flu_dsgn = subset(saq_dsgn, subpop)


# QC sub-design
saq_dsgn$variables %>% count(flu_shot)
```

```
##   flu_shot     n
## 1      -15   399
## 2       -1 10891
## 3        0 10939
## 4        1  8232
```

```
flu_dsgn$variables %>% count(flu_shot)
```

```
##   flu_shot     n
## 1        0 10939
## 2        1  8232
```

```
# Calculate survey estimates ----------------------------------------------------
#  - Percentage of people with a flu shot
#  - Logistic regression: to identify demographic factors associated with
#    receiving a flu shot

# Percentage of people with a flu shot
svymean(~flu_shot, design = flu_dsgn)
```

```
##             mean     SE
## flu_shot 0.42095 0.0064
```

```
# Logistic regression
# - specify 'family = quasibinomial' to get rid of warning messages

svyglm(
  flu_shot ~ AGELAST + as.factor(SEX) + as.factor(RACETHX) + as.factor(INSCOV18),
  design = flu_dsgn, family = quasibinomial) %>%
  summary
```

```
##
## Call:
## svyglm(formula = flu_shot ~ AGELAST + as.factor(SEX) + as.factor(RACETHX) +
##     as.factor(INSCOV18), design = flu_dsgn, family = quasibinomial)
##
## Survey design:
## subset(saq_dsgn, subpop)
```

```
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)           -2.141692   0.081346 -26.328  < 2e-16 ***
## AGELAST                0.031347   0.001207  25.961  < 2e-16 ***
## as.factor(SEX)2        0.272896   0.033345   8.184 2.03e-13 ***
## as.factor(RACETHX)2    0.379073   0.061079   6.206 6.52e-09 ***
## as.factor(RACETHX)3   -0.066147   0.079983  -0.827 0.409719
## as.factor(RACETHX)4    0.425476   0.098933   4.301 3.29e-05 ***
## as.factor(RACETHX)5    0.203291   0.109916   1.850 0.066620 .
## as.factor(INSCOV18)2 -0.159193   0.044643  -3.566 0.000506 ***
## as.factor(INSCOV18)3 -1.366126   0.086787 -15.741  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for quasibinomial family taken to be 1.006476)
##
## Number of Fisher Scoring iterations: 4
```

Creating New Model

I will be creating a similar model but for the doctor checking BP. I will clean up the variables a little bit first to convert them into factors with the corresponding labels. I will also remove any blood pressure results that are not a yes or no; these are the data that is represented by -15 and -1.

```
h209Clean <- h209 %>%
  select(DUPERSID, VARPSU, VARSTR, drCheckBP = ADBPCK42,
         ADFLST42, AGELAST, SEX, RACETHX, INSCOV18, matches("SAQ")) %>%
  filter(drCheckBP >= 0) %>%
  mutate(
    SEX = factor(if_else(SEX == 1, 'male', 'female'), levels = c('male', 'female')),
    RACETHX = factor(case_when(
      RACETHX == 1 ~ 'HISPANIC',
      RACETHX == 2 ~ 'NON-HISPANIC WHITE',
      RACETHX == 3 ~ 'NON-HISPANIC BLACK',
      RACETHX == 4 ~ 'NON-HISPANIC ASIAN',
      RACETHX == 5 ~ 'NON-HISPANIC OTHER/MULTIPLE'
    )),
    INSCOV18 = factor(case_when(
      INSCOV18 == 1 ~ "private",
      INSCOV18 == 2 ~ 'public',
      INSCOV18 == 3 ~ 'uninsured'
    )),
    drCheckBP = factor(if_else(drCheckBP == 1, 'yes', 'no'))
  )

## check the count of the outcome variable

h209Clean %>% count(drCheckBP) # note this match the expected from MEDS (https://meps.ahrq.gov/mepsweb/
```

```
## # A tibble: 2 x 2
##   drCheckBP     n
##   <fct>     <int>
## 1 no         3162
## 2 yes       15998
```

```r
# I will use the same survey design as the example but I will update the
# model to reflect my cleaned up variables

bpDsgn = svydesign(
  id = ~VARPSU,
  strata = ~VARSTR,
  weights = ~SAQWT18F,
  data = h209Clean,
  nest = TRUE)


# Percentage of people who had their BP checked
svymean(~drCheckBP, design = bpDsgn)
```

```
##                  mean     SE
## drCheckBPno  0.16176 0.0047
## drCheckBPyes 0.83824 0.0047
```

```r
# Logistic regression
# - specify 'family = quasibinomial' to get rid of warning messages

svyglm(
  drCheckBP ~ AGELAST + SEX + RACETHX + INSCOV18,
  design = bpDsgn, family = quasibinomial) %>%
  tidy() %>%
  kbl() %>%
  kable_styling(latex_options = "HOLD_position")
```

term	estimate	std.error	statistic	p.value
(Intercept)	-0.5720322	0.0932592	-6.1337896	0.0000000
AGELAST	0.0387793	0.0015981	24.2664073	0.0000000
SEXfemale	0.6209931	0.0535417	11.5983049	0.0000000
RACETHXNON-HISPANIC ASIAN	0.0512999	0.1190063	0.4310688	0.6671215
RACETHXNON-HISPANIC BLACK	0.4807177	0.0983089	4.8898718	0.0000029
RACETHXNON-HISPANIC OTHER/MULTIPLE	0.5104138	0.1560306	3.2712420	0.0013663
RACETHXNON-HISPANIC WHITE	0.7449486	0.0723415	10.2976618	0.0000000
INSCOV18public	-0.2390403	0.0667901	-3.5789779	0.0004832
INSCOV18uninsured	-1.6528537	0.0763599	-21.6455699	0.0000000

Based upon the model results, there are a few interesting observations. The first that being a female increases your odds of having your blood pressure checked relative to male. The second is, relative to Hispanic, every other race has higher odds of having their blood pressure checked. Finally, if you are hoping to have your blood pressure checked, having insurance is your best bet.