Monday Submission - cleaning real test data and running final model

2022-10-16

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

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

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

Please note: All of our work for training and testing the various models is available on our shared Github. We did not include all of the models we tried here, just the coefficients to predict from our final model we selected.

Section 1. Read and clean data

Clean data according to how we did in the training data.

```
dta <- read_csv("project_test_data_noformat_nomissing.csv")
```

```
## Rows: 4135 Columns: 62

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (62): id, FAMSIZE, ELDCH, SEX, AGE, MARST, BIRTHYR, MARRNO, MARRINYR, YR...

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# Clean data according to how we did in the training data.
# 1. Family size: continuous variable, we can keep this variable since

# 2. ELDCH: age of eldest own child in household, less related to this adult's hearing situation, so we

# 3.sex_f: sex of the individuals. Keep the original value. Just keep in mind that
# 1= male; 2= female
dta <- dta %>% dplyr::select(-ELDCH)
dta <- dta %>%
```

```r
  mutate(SEX = if_else(SEX == 1, "male", "female"))

# 4.marst_f: Marital status, just collapse married into one group
dta <- dta %>%
  mutate(MARST = case_when(
    MARST %in% c(1,2,3) ~ "Married",
    MARST == 4 ~ "Divorced",
    MARST == 5 ~ "Widowed",
    MARST == 6 ~ "Never married"
  ))


# 5. marrno_f: time married. This one carries similar information as married.
# I think times married is less related to hearing loss if the marriage status has already been conside
dta <- dta %>% dplyr::select(-MARRNO)



# 6. MARRINYR: married within the last year
# this one is confusing and is less useful than marriage status, so we decided to remove this variable

dta <- dta %>% dplyr::select(-MARRINYR)

#7. DIVINYR: divorced in the last year, change the coding method to make NA explicitly
# NA: missing, 0: No, 1:Yes

dta$DIVINYR[dta$DIVINYR==0] <- NA
dta$DIVINYR <- dta$DIVINYR-1

#8. WIDINYR_f: widowed in the last year. Same idea as divorced
dta$WIDINYR[dta$WIDINYR==0] <- NA
dta$WIDINYR <- dta$WIDINYR-1

#9. FERTYR_f: didn't find the definition.

#10. Race: I would create white, black, Asian, or others
dta <- dta %>%
  mutate(RACE = case_when(
    RACE == 1 ~ "white",
    RACE == 2 ~ "Black",
    RACE %in% c(4,5,6) ~"Asian",
    RACE %in% c(3, 7, 8, 9) ~ "Otherse"
  ))

# 11. HISPAN: collapse to year vs. no
# Assumption made: if the person doesn't report his/her hispanic, we treat it as no
dta <- dta %>%
  mutate(HISPAN = if_else(HISPAN ==0 , 0 , 1))


# 12. SPEAKENG_f: speak english or not, it covers similar information as race and education, so we deci

dta <- dta %>% dplyr::select(-SPEAKENG)
```

```r
# 13. RACNUM_f: race groups, less useful information given we have race category so we decided to remov
dta <- dta %>% dplyr::select(-RACNUM)

# 14. HCOVANY: health insurance coverage
# 0 :No, 1:Yes
dta$HCOVANY <- dta$HCOVANY -1



# 15 HCOVPRIV_f: delete
# 16 HINSEMP_f: delete
# 17 HINSPUR_f: delete
# 18 HCOVPUB_f: delete
# 19 HINSCAID_f: delete
# 20 HINSCARE_f: delete
# 21 HINSVA_f: delete
# 22 HINSIHS_f: delete
dta <- dta %>% dplyr::select(-c(HCOVPRIV, HINSEMP, HINSPUR, HCOVPUB,
                          HINSCAID, HINSCARE, HINSVA, HINSIHS))


# 23 EDUC: I would like to convert it into
# 1: less than high school
# 2: high school
# 3. college or greater
dta <- dta %>%
  mutate(EDUC = case_when(
    EDUC <= 3 ~ "Less than high school",
    EDUC %in% c(4,5,6) ~ "High school",
    EDUC >= 7 ~ "College or greater"
  ))


# 24.SCHLTYPE_f: school type , less important, so we decided to remove this variable based on subject-m
# 25. EMPSTAT: employment status, recoding it to make NA explicit
# 26. LABFORCE: labor force status
# WE just create a new variable WORK_STATUS
dta$WORK_STATUS <- NA
dta$WORK_STATUS <- ifelse(dta$EMPSTAT==1, "Employed", dta$WORK_STATUS)
dta$WORK_STATUS <- ifelse(dta$EMPSTAT ==2, "Unemployed", dta$WORK_STATUS)
dta$WORK_STATUS <- ifelse(dta$EMPSTAT == 3|dta$LABFORCE==1, "Not in labor force", dta$WORK_STATUS)
dta <- dta %>% dplyr::select(-EMPSTAT,-LABFORCE)

# 27. CLASSWKR:self-employed vs. work for wages.
dta$CLASSWKR[dta$CLASSWKR == 0 ] <- NA
dta$CLASSWKR <- ifelse(dta$CLASSWKR == 1, "Self-employed","Works for wages")

# 28. WKSWORK2: Weeks worked last year, intervalled. Just convert it to a categorical variable
dta$WKSWORK2[dta$WKSWORK2==0] <- NA
dta <- dta %>%
  mutate(WKSWORK2 = case_when(
    WKSWORK2 == 1 ~ "1-13 weeks",
    WKSWORK2 == 2 ~ "14-26 weeks",
```

```
    WKSWORK2 == 3 ~ "27-39 weeks",
    WKSWORK2 == 4 ~ "40-47 weeks",
    WKSWORK2 == 5 ~ "48-49 weeks",
    WKSWORK2 == 6 ~ "50-52 weeks"
  ))

# 29. value WRKLSTWK_f: worked last week ? this one is not of much use and can be sensible to the
# survey time, sso we decided to remove this variable based on subject-matter knowledge
dta <- dta %>% dplyr::select(-WRKLSTWK)

# 30. value ABSENT: absent last week. Same reason, just delete it
dta <- dta %>% dplyr::select(-ABSENT)

# 31. Looking : looing for work, less important than work status, delete it
dta <- dta %>% dplyr::select(-LOOKING)

# 32. AVAILBLE:
# 1. avaliable for work or has a work
# 2. Not avaliable for work because of illness
# 3. Not avaliable for work because of other reasons

dta$AVAILBLE[dta$AVAILBLE %in% c(0, 5)] <- NA
dta$AVAILBLE <- ifelse(dta$AVAILBLE %in% c(1,4), "Avaliable for work or has a job", dta$AVAILBLE)
dta$AVAILBLE <- ifelse(dta$AVAILBLE == 2, "Not avaliable because of illness", dta$AVAILBLE)
dta$AVAILBLE <- ifelse(dta$AVAILBLE == 3, "Not avaliable because of oteher reasons", dta$AVAILBLE)

# 34. DIFFHEAR: that's our outcome, but that is not in the testing dataset so we will leave out this co
# 0:NO, 1:yes

#dta$DIFFHEAR[dta$DIFFHEAR==0] <- NA
#dta$DIFFHEAR <- dta$DIFFHEAR - 1

# 35. VETSTAT: just keep this information for veteran
dta$VETSTAT[dta$VETSTAT %in% c(0, 9)] <- NA
dta$VETSTAT <- ifelse(dta$VETSTAT == 1, "Not a veteran", "Veteran")

# 36. VETSTATD_f
# 37. VET01LTR_f
# 38. VET90X01_f
# 39. VET75X90_f
# 40. VETVIETN_f
# 41. VET55X64_f
# 42. VETKOREA_f
# 43. VET47X50_f
# 44. VETWWII_f
# 45. VETOTHER_f
# delete

dta <- dta %>% dplyr::select(-c(VETSTATD, VET01LTR, VET90X01, VET75X90, VETVIETN,
                          VET55X64, VETKOREA, VET47X50, VETWWII, VETOTHER))

# 46. TRANWOR: not much useful because a lot of 0 (NAs)
dta <- dta %>% dplyr::select(-TRANWORK)
```

```r
# 46. Carpool: convert it
dta$CARPOOL[dta$CARPOOL==0] <- NA
dta$CARPOOL <- if_else(dta$CARPOOL==1, "Drives alone", "Carpools")

# 47. RIDERS: we already have the carpool variable, which makes this variable less useful so we decided
dta <- dta %>% dplyr::select(-RIDERS)

# 48. GCHHOUSE: Own grandchildren living in household
dta$GCHOUSE[dta$GCHOUSE==0] <- NA
dta$GCHOUSE <- dta$GCHOUSE-1

# 49. GCMONTHS: less useful compared to GXRESPON so we decided to remove this variable based on subject
dta <- dta %>% dplyr::select(-GCMONTHS)

# 50. GCRESPON
dta$GCRESPON[dta$GCRESPON == 0] <- NA
dta$GCRESPON <- dta$GCRESPON - 1

#51. AGE: keep the original continuous variable

# 52. UHRSWORK: keep the original continuous variable
# if work hour is 0, we treat it as 0 instead of missing

# 53. WORKEDYR_f: worked last year
dta$WORKEDYR[dta$WORKEDYR==0] <- NA
dta <- dta %>% mutate(WORKEDYR = case_when(
  WORKEDYR == 1 ~ "No",
  WORKEDYR ==2 ~ "No, but worked 1-5 yrs age",
  WORKEDYR == 3 ~"Yes"
))

# 54. YRMARR: year of married, not much useful but keep it here

# 55. TRANTIME: time of transportaton, keep original value
dta$TRANTIME[dta$TRANTIME==0]<- NA

## 56. Arrive and departs
dta$ARRIVES[dta$ARRIVES == 0] <- NA
dta$ARRIVES[dta$ARRIVES <= 1200] <- "AM"
dta$ARRIVES[dta$ARRIVES > 1200] <- "PM"

dta$DEPARTS[dta$DEPARTS == 0] <- NA
dta$DEPARTS[dta$DEPARTS <= 1200] <- "AM"
dta$DEPARTS[dta$DEPARTS > 1200] <- "PM"
## 57. VETDISAB VA service-connected disability rating

dta$VETDISAB[dta$VETDISAB == 0] <- NA

dta$VETDISAB <-if_else(dta$VETDISAB== 1, 0, 1)
```

We exclude several variable that are not included in our trainning dataset

```
dta <- dta %>% dplyr::select(-AVAILBLE, -CARPOOL,-TRANTIME,-GCRESPON,-WKSWORK2, -VETDISAB, -DIVINYR, -W
write.csv(dta, "clean_data_newtest.csv")
```

Use the trained LASSO to predict probabilities

- This is the model we derived from LASSO with CV (See separate file in the Github for doing this based on the training dataset.

```
logit <-  1.363527e+01 + 5.332333e-02*dta$AGE  -9.844508e-03*dta$BIRTHYR  +
      -2.885644e-01*(dta$SEX=="female") +
      1.700971e-05*(dta$SEX =="male") + 6.480554e-02* (dta$RACE == "Otherse") +
      7.366961e-03*(dta$EDUC == "Less than high school")+
      2.106638e-03*(dta$WORKEDYR == "No") +
      -2.180949e-018*(dta$VETSTAT == "Not a veteran") + 7.611356e-05*(dta$VETSTAT == "Veteran")+
      -2.785161e-02 *(dta$WORK_STATUS == "Employed")
dta$ProbHL <- exp(logit)/(1+exp(logit))
```

- We set the probability cutoff at 0.2

```
dta$HearLossYN <- ifelse(dta$ProbHL<0.2,0,1)
```

- Save the data as a csv file

```
dta$ID <- dta$id
dta_save <- dta %>% dplyr::select(c(ProbHL,HearLossYN,ID))
write.csv(dta_save, "group3_data.csv")
```