Testing Figure 1

2022-11-18

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.4
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.2      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Load data

```
PM = read.csv('data/external/AHRI_DATASET_PM_MANUSCRIPT_DATA.csv')
```

Clean data

```
# only necessary columns for figure 1's multivariate logistical regression
PM_cleaned = PM %>%
  select(
    CASEID_7139,
    SEX,
    AGE,
    ETHNICITY,
    REGION,
    CCI_SCORE,
    GAD7_GE10,
    PHQ9_GE10,
    INSURANCE,
    PM1_GEN_HEALTH,
    PM1_DIAG_CONDITION,
    PM1_UNDIAG_CONCERN
    ) %>%
  mutate(
    PM_12M = PM1_GEN_HEALTH + PM1_UNDIAG_CONCERN + PM1_DIAG_CONDITION
    )

# calculate boolean for PM_12M
PM_cleaned$PM_12M = PM_cleaned$PM_12M %>%
  recode(`-297` = 0, `0` = 0, `1` = 1, `2` = 1, `3` = 1)
```

```
## refactor all booleans so they make sense
```

Multivariate logistical regression

```
## modified helper from https://rdrr.io/github/eringrand/RUncommon/src/R/logistic.regression.or.ci.R
logistic.regression.or.ci <- function(regress.out, level = 0.95) {
  usual.output <- summary(regress.out)
  z.quantile <- stats::qnorm(1 - (1 - level) / 2)
  number.vars <- length(regress.out$coefficients)
  OR <- exp(regress.out$coefficients[-1])
  temp.store.result <- matrix(rep(NA, number.vars * 2), nrow = number.vars)
  for (i in 1:number.vars) {
    temp.store.result[i, ] <- summary(regress.out)$coefficients[i] +
      c(-1, 1) * z.quantile * summary(regress.out)$coefficients[i + number.vars]
  }
  intercept.ci <- temp.store.result[1, ]
  slopes.ci <- temp.store.result[-1, ]
  OR.ci <- exp(slopes.ci)

  output <- list(
    regression.table = usual.output, intercept.ci = intercept.ci,
    slopes.ci = slopes.ci, OR = OR, OR.ci = OR.ci
  )
  return(output)
}
```

```
full_model = glm(PM_12M ~ SEX + AGE + ETHNICITY + REGION + CCI_SCORE + GAD7_GE10 + PHQ9_GE10 + INSURANCE
full_model_results = logistic.regression.or.ci(full_model)
```

```
full_model_results
```

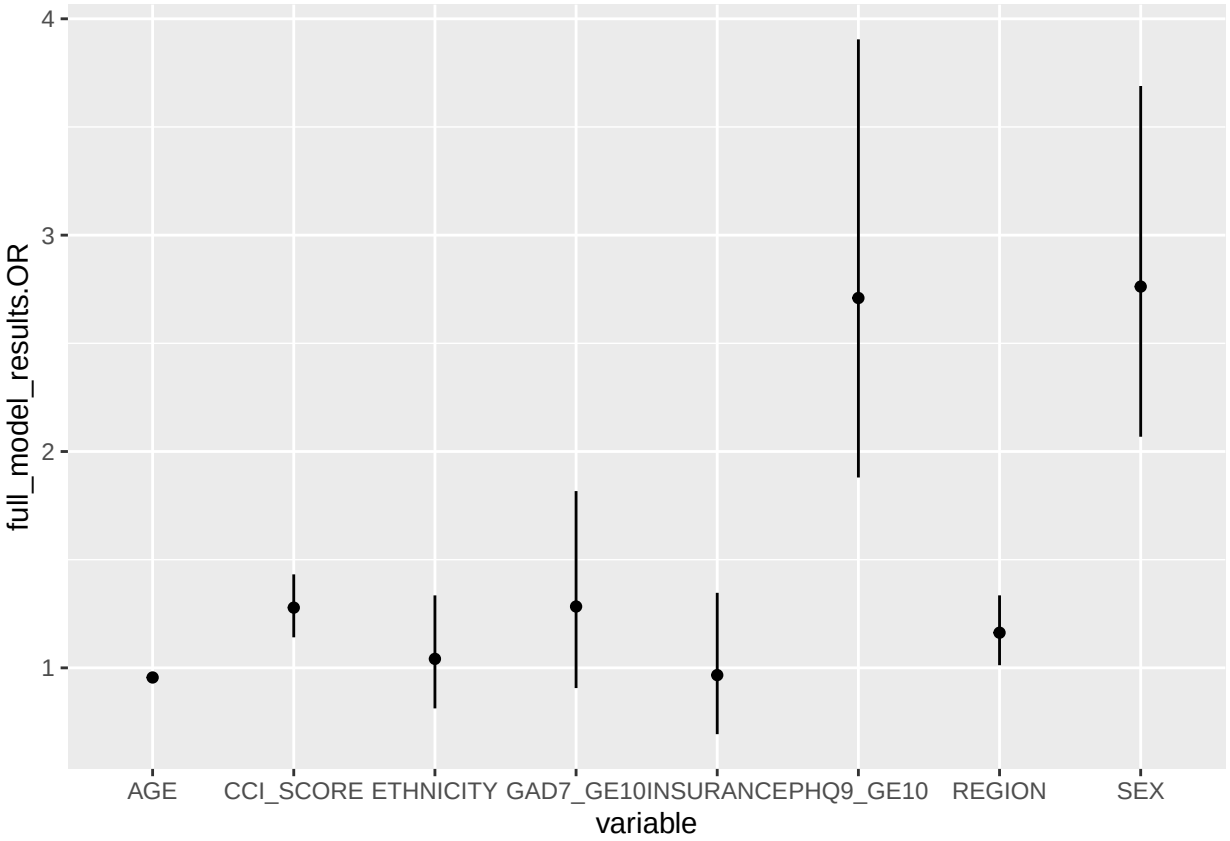
```
## $regression.table
##
## Call:
## glm(formula = PM_12M ~ SEX + AGE + ETHNICITY + REGION + CCI_SCORE +
##     GAD7_GE10 + PHQ9_GE10 + INSURANCE, family = binomial, data = PM_cleaned)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0129  -0.2814  -0.1857  -0.1162   3.1621
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.332470   0.429483  -7.759 8.54e-15 ***
## SEX          1.016064   0.147642   6.882 5.90e-12 ***
## AGE         -0.045376   0.006022  -7.535 4.88e-14 ***
## ETHNICITY    0.040816   0.126622   0.322    0.747
## REGION       0.150605   0.070644   2.132    0.033 *
## CCI_SCORE    0.245608   0.057964   4.237 2.26e-05 ***
## GAD7_GE10    0.249608   0.177366   1.407    0.159
```

```
## PHQ9_GE10    0.996753   0.186480   5.345 9.04e-08 ***
## INSURANCE   -0.034020   0.169303  -0.201    0.841
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2019.1  on 7138  degrees of freedom
## Residual deviance: 1776.7  on 7130  degrees of freedom
## AIC: 1794.7
##
## Number of Fisher Scoring iterations: 7
##
##
## $intercept.ci
## [1] -4.174241 -2.490700
##
## $slopes.ci
##             [,1]        [,2]
## [1,]  0.72669181  1.30543639
## [2,] -0.05717914 -0.03357357
## [3,] -0.20735744  0.28898981
## [4,]  0.01214576  0.28906330
## [5,]  0.13200098  0.35921490
## [6,] -0.09802188  0.59723875
## [7,]  0.63125783  1.36224732
## [8,] -0.36584873  0.29780839
##
## $OR
##       SEX       AGE ETHNICITY    REGION CCI_SCORE GAD7_GE10 PHQ9_GE10 INSURANCE
## 2.7623012 0.9556378 1.0416606 1.1625368 1.2783983 1.2835227 2.7094687 0.9665520
##
## $OR.ci
##           [,1]      [,2]
## [1,] 2.0682272 3.6892987
## [2,] 0.9444249 0.9669838
## [3,] 0.8127291 1.3350781
## [4,] 1.0122198 1.3351762
## [5,] 1.1411094 1.4322046
## [6,] 0.9066291 1.8170944
## [7,] 1.8799738 3.9049591
## [8,] 0.6936077 1.3469037
```

```
df = data.frame(full_model_results$OR)
df = cbind(variable = rownames(df), df)
rownames(df) = 1:nrow(df)

df$or.cimin = full_model_results$OR.ci[,1]
df$or.cimax = full_model_results$OR.ci[,2]
```

```
## try plotting
ggplot(data = df, aes(y = full_model_results.OR, x = variable, ymin = or.cimin, ymax = or.cimax)) +
  geom_linerange() +
  geom_point()
```


4
3
2
1
full_model_results.OR
AGE
CCI_SCORE
ETHNICITY
GAD7_GE10
INSURANCE
PHQ9_GE10
REGION
SEX
variable