P8131 HW8

Brian Jo Hsuan Lee

4/13/2022

Load packages

```
library(tidyverse)
library(readxl)
library(gee)
library(lme4)
library(nlme)
```

Problem 1

Import data

```
data =
  read_excel("HW8-HEALTH.xlsx") %>%
  janitor::clean_names() %>%
  mutate(
    id = factor(id),
    time = factor(time),
    txt = factor(txt, levels = c("Control", "Intervention")),
    health = factor(health, levels = c("Poor", "Good")),
    agegroup = factor(agegroup)
  )
```

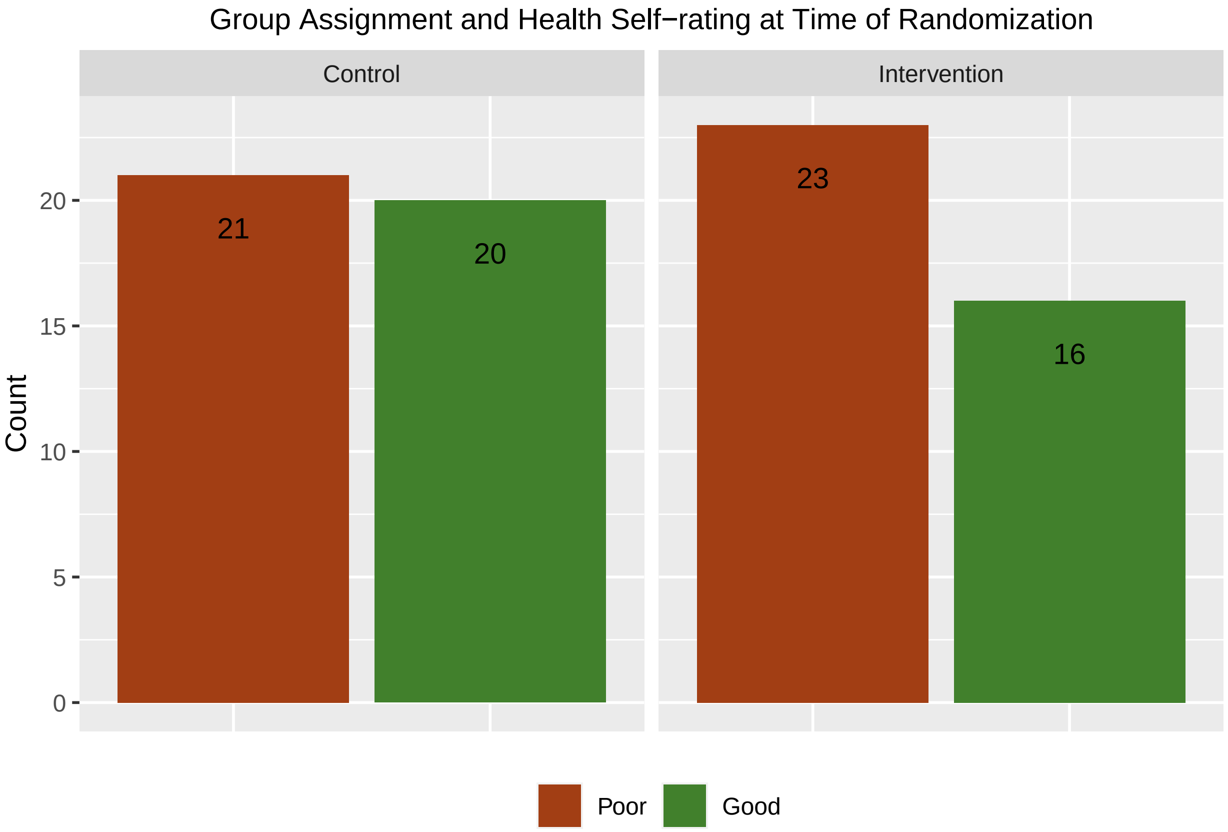
a) **Interpret and discuss the bivariate, cross-sectional relationship between group assignment and health self-rating.**

Samples that were given the control treatment (no educational intervention) had a more even-split health responses, where as lower proportion of samples in the intervention treatment reported good health. By count, there are more samples who reported poor health in the intervention group than the control group, even when the total sample count in the control group (41) exceeds that of the intervention group (39).

If the baseline health status for the 2 groups are indeed statistically unequivalent, the discrepancy could impact study conclusions when the rooted differences between the groups are ignored.

```
# filter in only the first visits
data_bl =
  data %>%
  filter(
    time == "1"
  )

# plot the response counts for both the control and the intervention group
data_bl %>%
  group_by(txt, health) %>%
  summarize(count = n()) %>%
  ggplot(aes(x = health, y = count, fill = health)) +
  geom_col() +
  scale_fill_manual(values = c("#A23E14", "#41802C")) +
  facet_grid(cols = vars(txt)) +
  geom_text(aes(label = count), vjust = 3) +
  labs(
    title = "Group Assignment and Health Self-rating at Time of Randomization",
    y = "Count"
  ) +
  theme(
    axis.title.x = element_blank(),
    axis.text.x = element_blank(),
    axis.ticks.x = element_blank(),
    legend.title = element_blank(),
    legend.position = "bottom",
    plot.title = element_text(size = 11, hjust = 0.5)
  )
```



After a more robust statistical evaluation, the coefficient significance of a logistic model tells a different story. Samples assigned to the intervention did not have a significant decrease in log odds of reporting good health, and this points to that randomization is preserved at the baseline.

```
glm_fit = glm(health ~ txt, family = binomial(link = "logit"), data = data_bl)
summary(glm_fit)
```

```
##
## Call:
```

```
## glm(formula = health ~ txt, family = binomial(link = "logit"), 
##     data = data_bl)
## 
## Deviance Residuals: 
##    Min      1Q  Median      3Q     Max  
## -1.157  -1.157  -1.028   1.198   1.335  
## 
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -0.04879    0.31244  -0.156    0.876
## txtIntervention -0.31412    0.45122  -0.696    0.486
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 110.10  on 79  degrees of freedom
## Residual deviance: 109.62  on 78  degrees of freedom
## AIC: 113.62
## 
## Number of Fisher Scoring iterations: 4
```

The benefit of having longitudinal data is it could be used to control for time-invariant differences within a subject. Having multiple observations per individual allows us to base estimates on the variation within individuals. However, the correlation among the observations from an individual must be taken into account somehow, and there are 2 ways of address such structure.

b) **Interpret health status over time using a GEE model**

Fist, the non-parametric GEE model averages over all individuals to make a population inference by assuming some within-subject covariance structure. For example, according to our summary estimates, compared to the population that reported "poor" health as its baseline response, the "good" health population has a 1.82 increase in log odds of reporting another good health response by the second visit, while adjusting for treatment and age group.

```r
# Create a new column showing baseline health rating, and a new column representing good health as 1, p

data_gee =
  data %>%
  pivot_wider(
    names_from = time,
    values_from = health
  ) %>%
  pivot_longer(
    `2`:`4`,
    names_to = "time",
    values_to = "health"
  ) %>%
  rename("baseline" = `1`) %>%
  mutate(
    time = factor(time),
    baseline = factor(baseline, levels = c("Poor", "Good")),
    health = factor(health, levels = c("Poor", "Good")),
    nhealth = as.numeric(health == "Good")
  )
```

```r
gee_fit = gee(nhealth ~ baseline + txt + time + agegroup,
              data = data_gee,
              family = "binomial",
              id = id,
              corstr = "unstructured",
              scale.fix = FALSE)
```

```
## Beginning Cgee S-function, @(#) geeformula.q 4.13 98/01/27
```

```
## running glm to get initial regression estimate
```

```
##     (Intercept)    baselineGood txtIntervention           time3           time4
##      -1.5199450       1.7192117       2.0042708       0.2575654       0.2366989
##   agegroup25-34     agegroup35+
##       1.1968673       1.3958656
```

```
summary(gee_fit)
```

```
##
##  GEE:  GENERALIZED LINEAR MODELS FOR DEPENDENT DATA
##  gee S-function, version 4.13 modified 98/01/27 (1998)
##
## Model:
##  Link:                      Logit
##  Variance to Mean Relation: Binomial
##  Correlation Structure:     Unstructured
##
## Call:
## gee(formula = nhealth ~ baseline + txt + time + agegroup, id = id,
##     data = data_gee, family = "binomial", corstr = "unstructured",
##     scale.fix = FALSE)
##
## Summary of Residuals:
##         Min          1Q      Median          3Q         Max
## -0.97980130 -0.20060701  0.09442344  0.18344971  0.83995062
##
##
## Coefficients:
##                    Estimate Naive S.E.    Naive z Robust S.E.  Robust z
## (Intercept)     -1.6578607  0.6014505 -2.7564377   0.4533989 -3.656517
## baselineGood     1.8164161  0.5978966  3.0380103   0.5113296  3.552339
## txtIntervention  2.1022271  0.5954429  3.5305269   0.5362768  3.920041
## time3            0.2753559  0.4747047  0.5800572   0.3368572  0.817426
## time4            0.2863563  0.4083916  0.7011809   0.4161352  0.688133
## agegroup25-34    1.3345925  0.5860828  2.2771400   0.5043829  2.645991
## agegroup35+      1.4112905  0.9740226  1.4489299   0.7855584  1.796544
##
## Estimated Scale Parameter:  1.486693
## Number of Iterations:  4
##
## Working Correlation
##           [,1]      [,2]      [,3]
## [1,] 1.0000000 0.1794182 0.5602284
## [2,] 0.1794182 1.0000000 0.2104116
## [3,] 0.5602284 0.2104116 1.0000000
```

c) **Generalized Linear Mixed Model**

Second, GLMMs are an extension of generalized linear models (GLMs) to include both fixed and random effects on a subject level, and therefore their interpretations are similar. Reading from our summary, compared to an individual that reported “poor” health as its baseline response, a “good” health individual has a 2.81 increase in log odds of reporting another good health response by the second month’s visit, while adjusting for treatment and age group.

```
glmm_fit = glmer(nhealth ~ baseline + txt + time + agegroup + (1 | id),
                 family = 'binomial', data = data_gee)
summary(glmm_fit)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula: nhealth ~ baseline + txt + time + agegroup + (1 | id)
##    Data: data_gee
##
##      AIC      BIC   logLik deviance df.resid
##    186.5    212.9    -85.3    170.5      191
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.4477 -0.2302  0.1443  0.2763  1.9348
##
## Random effects:
##  Groups Name        Variance Std.Dev.
##  id     (Intercept) 5.871    2.423
## Number of obs: 199, groups:  id, 78
##
## Fixed effects:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -2.6142     1.0227  -2.556  0.01058 *
## baselineGood        2.8084     0.9990   2.811  0.00494 **
## txtIntervention     3.4540     1.0919   3.163  0.00156 **
## time3               0.4390     0.5592   0.785  0.43243
## time4               0.3546     0.6212   0.571  0.56806
## agegroup25-34       2.2779     1.0248   2.223  0.02623 *
## agegroup35+         1.9878     1.3960   1.424  0.15446
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) bslnGd txtInt time3  time4  a25-34
## baselineGod -0.671
## txtIntrvntn -0.673  0.456
## time3       -0.320  0.089  0.114
## time4       -0.230  0.023  0.057  0.420
## agegrp25-34 -0.661  0.386  0.402  0.067  0.015
## agegroup35+ -0.445  0.277  0.209  0.021 -0.004  0.392
```

Note that the coefficients between the GEE and the GLMM models shall not be compared, as the former is on the population level and the latter is on the subject level. The GLMM model fits random intercepts per individual, which adds or subtracts from the fixed effect marginal intercept β_0. A GLMM model is inherently different from a GEE model, because it estimates its covariance model, and not presume it under some structure. Furthermore, there is an added random factor with respect to each subject at the cost of computation power.

```
random.effects(glmm_fit)
```

```
## $id
##     (Intercept)
## 101  0.26955372
## 102 -0.76222542
## 103  0.60941107
## 104  0.03540812
## 105 -0.32144511
## 106  2.09061966
## 107  1.51432407
## 109  0.03540812
## 110  1.75639618
## 111  0.03540812
## 112 -2.34516556
## 113 -0.58147296
## 114  0.39929206
## 116  0.48651928
## 117 -2.47119883
## 118 -0.92735696
## 119  0.30609132
## 120 -1.59110159
## 121 -0.58147296
## 122  0.39929206
## 123  0.57906666
## 124 -2.78772464
## 125  0.20442269
## 126  1.29416594
## 127  0.26955372
## 128 -2.77400155
## 129  1.37787703
## 130 -0.58147296
## 131 -2.81263283
## 132 -0.58147296
## 133  1.37787703
## 134 -0.58147296
## 135 -4.46986571
## 136  1.18290863
## 137  0.26955372
## 138  1.82565052
## 139 -3.84571226
## 140  1.75639618
## 141 -1.99802581
## 142  0.30609132
## 143 -1.68610509
## 145  0.30250211
## 201  1.16545575
## 202  0.60941107
## 203 -1.68610509
## 204  0.88671151
## 205  0.20442269
## 206 -0.58147296
## 207  2.28086128
## 208  0.03540812
## 209  0.20178800
## 210 -1.59110159
## 211  0.30609132
## 213  1.29416594
## 601 -1.86094379
## 602  0.60941107
## 603 -1.37702527
## 604  0.26955372
## 605 -0.76222542
## 606 -5.56869974
## 607  0.61578065
## 608  0.39929206
## 609  0.60941107
## 610  0.39929206
## 611  1.18290863
## 612 -1.72050125
## 613  0.20178800
## 614  0.19132583
## 615  0.39929206
## 616  0.60941107
## 617  0.60941107
## 618  1.37787703
## 619 -0.26905010
## 620  0.26955372
## 621  0.03540812
## 622 -0.30163868
## 624 -0.50113922
## 625  0.39929206
##
## with conditional variances for "id"
```