20201010-p8157_hw2_jsg2145

Jared Garfinkel

10/10/2020

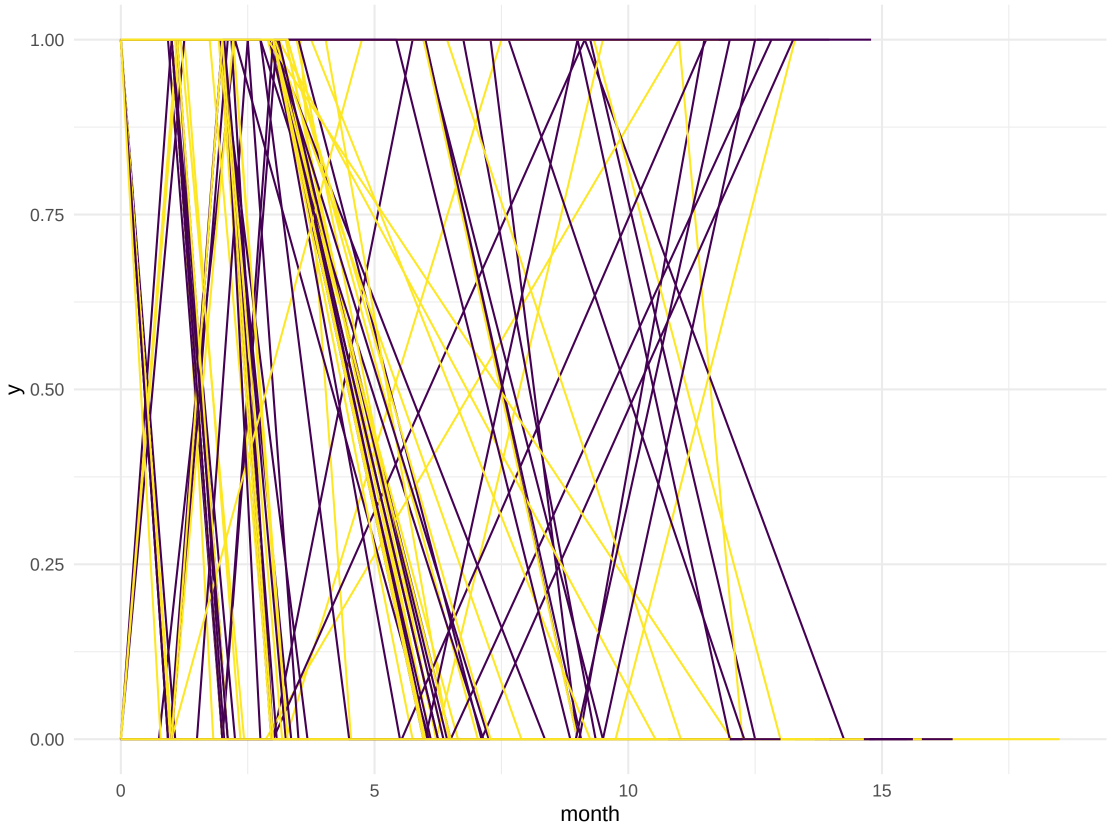
```
# data = read.delim(file = "./data/toenail.txt", sep = "", col.names = c("id", "y", "treatment", "month

data = read_delim(file = "./data/toenail.txt", delim = " ", col_names = c("id", "y", "treatment", "montl

data
```

```
## # A tibble: 1,908 x 5
##       id     y treatment  month visit
##    <dbl> <dbl>     <dbl>  <dbl> <dbl>
##  1     1     1         1  0         1
##  2     1     1         1  0.857     2
##  3     1     1         1  3.54      3
##  4     1     0         1  4.54      4
##  5     1     0         1  7.54      5
##  6     1     0         1 10.0       6
##  7     1     0         1 13.1       7
##  8     2     0         0  0         1
##  9     2     0         0  0.964     2
## 10     2     1         0  2         3
## # ... with 1,898 more rows
```

```
data %>%
  ggplot() +
  geom_path(aes(x = month, y = y, group = id, color = treatment)) +
  theme(legend.position = "none")
```



```
gee1 = geepack::geeglm(y ~ month * treatment, data = data, id = id, family = binomial(link = "logit"), c
summary(gee1)
```

```
##
## Call:
## geepack::geeglm(formula = y ~ month * treatment, family = binomial(link = "logit"),
##     data = data, id = id, corstr = "exchangeable")
##
##  Coefficients:
##                 Estimate  Std.err   Wald     Pr(>|W|)
## (Intercept)     -0.58192  0.17206 11.439     0.000719 ***
## month           -0.17128  0.03000 32.596 0.0000000113 ***
## treatment        0.00718  0.25949  0.001     0.977924
## month:treatment -0.07773  0.05411  2.064     0.150862
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Correlation structure = exchangeable
## Estimated Scale Parameters:
##
##             Estimate Std.err
## (Intercept)    1.088  0.5013
##   Link = identity
##
## Estimated Correlation Parameters:
##       Estimate Std.err
## alpha   0.4218  0.2119
## Number of clusters:   294  Maximum cluster size: 7
```

```
L = matrix(0, ncol = 4, nrow = 1)

L[1, c(4)] = c(1)

L
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    0    0    0    1
```

```
esticon(gee1, L=L, joint.test = TRUE)
```

```
##   X2.stat DF Pr(>|X^2|)
## 1   2.064  1     0.1509
```

```
gee2 = geepack::geeglm(y ~ month + treatment, data = data, id = id, family = binomial(link = "logit"), 
summary(gee2)
```

```
##
## Call:
## geepack::geeglm(formula = y ~ month + treatment, family = binomial(link = "logit"),
##     data = data, id = id, corstr = "exchangeable")
##
##  Coefficients:
##             Estimate Std.err  Wald          Pr(>|W|)
## (Intercept)  -0.6104  0.1777 11.80           0.00059 ***
## month        -0.2051  0.0259 62.66 0.0000000000000024 ***
## treatment     0.0402  0.2532  0.03           0.87388
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Correlation structure = exchangeable
## Estimated Scale Parameters:
##
##             Estimate Std.err
## (Intercept)     1.09   0.423
##   Link = identity
##
## Estimated Correlation Parameters:
##       Estimate Std.err
## alpha    0.424   0.182
## Number of clusters:   294  Maximum cluster size: 7
```

Question 2

Read in data

Format the data for analysis.

```r
df = read_delim(file = "./data/skin.txt", delim = " ", col_names = c("id", "center", "age", "skin", "ge
  mutate(tr = as_factor(tr),
         year = as.numeric(year),
         y = as.numeric(y),
         age = as.numeric(age),
         gender = as_factor(gender),
         skin = as_factor(skin),
         exposure = as.numeric(exposure),
         current_age = age + year - 1)

df
```

```
## # A tibble: 7,081 x 10
##    id       center     age skin    gender exposure     y tr     year current_age
##    <chr>    <chr>    <dbl> <fct>   <fct>     <dbl> <dbl> <fct> <dbl>       <dbl>
##  1 "    10~ "      ~    51 "     ~ "    ~        4     0 "   ~     1          51
##  2 "    10~ "      ~    51 "     ~ "    ~        4     1 "   ~     2          52
##  3 "    10~ "      ~    51 "     ~ "    ~        4     1 "   ~     3          53
##  4 "    10~ "      ~    51 "     ~ "    ~        4     1 "   ~     4          54
##  5 "    10~ "      ~    51 "     ~ "    ~        4     0 "   ~     5          55
##  6 "    10~ "      ~    68 "     ~ "    ~        2     0 "   ~     1          68
##  7 "    10~ "      ~    68 "     ~ "    ~        2     0 "   ~     2          69
##  8 "    10~ "      ~    68 "     ~ "    ~        2     0 "   ~     3          70
##  9 "    10~ "      ~    68 "     ~ "    ~        2     0 "   ~     4          71
## 10 "    10~ "      ~    68 "     ~ "    ~        2     0 "   ~     5          72
## # ... with 7,071 more rows
```

Part 1

Set up the model

Using only year and treatment group, set up a GEE model for rate of skin cancers.

```r
gee_model1 = geepack::geeglm(y ~ year + tr , data = df, id = id, family = poisson(link = "log"), corstr

summary(gee_model1)
```

```
##
## Call:
## geepack::geeglm(formula = y ~ year + tr, family = poisson(link = "log"),
##     data = df, id = id, corstr = "exchangeable")
##
##  Coefficients:
##             Estimate Std.err   Wald            Pr(>|W|)
## (Intercept)  -1.4123  0.1080 171.10 <0.0000000000000002 ***
## year          0.0173  0.0247   0.49                0.48
## tr    1       0.1478  0.1094   1.83                0.18
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation structure = exchangeable
## Estimated Scale Parameters:
```

```
##
##             Estimate Std.err
## (Intercept)     2.65   0.374
##   Link = identity
##
## Estimated Correlation Parameters:
##       Estimate Std.err
## alpha    0.378   0.111
## Number of clusters:   1683  Maximum cluster size: 5
```

Part 2

Interpret the coefficients

None of the covariates in the model appear to be significant.

However, the interpretation would be that the log of the rate ratio of new skin cancer growths increases by 0.0173 each year holding treatment group constant.

Further, the log of the rate ratio of new skin cancer growths in treatment 1 is 0.1478 times greater compared to those in the control group at each time point.

Based on these results, it appears that beta-carotene does not improve health outcomes, since those in the treatment group had a higher log of the rate ratio of new skin cancer growths as mentioned above.

Part 3

```
gee_model2 = geepack::geeglm(y ~ skin + age + year + tr + exposure, data = df, id = id, family = poisso

summary(gee_model2)
```

Part 4

Adjusting for skin type, age, and exposure to previous skin cancers, the coefficients have different interpretations.

This data show that having skin that is not burned is associated with a lower log of the rate ratio for new cancer growths by 0.162 times compared to those with burned skin holding other covariates constant.

A one unit increase in the patient's age at randomization is associated with a log of the rate ratio of new skin cancer of 0.01496 holding other covariates constant.

There is an increase in the log of the rate ratio associated with each annual check of 0.01759 holding other covariates constant.

Those in the treatment group had a log of the rate ratio of 0.124 times those not in the treatment group holding other covariates constant.

Each one unit increase in exposure is associated with a 0.139 increase in the log of the rate ratio of new cancer holding other covariates constant.

Since the treatment effect is not significant in the model, there is no evidence to suggest that the treatment of beta carotene improves skin cancer outcomes.

Part 5

```
gee_model3 = geepack::geeglm(y ~ skin + age + year + tr + exposure, data = df, id = id, family = poisson

summary(gee_model3)
```

```
## 
## Call:
## geepack::geeglm(formula = y ~ skin + age + year + tr + exposure, 
##     family = poisson(link = "log"), data = df, id = id, corstr = "unstructured")
## 
##  Coefficients:
##             Estimate  Std.err   Wald            Pr(>|W|)    
## (Intercept) -2.88147  0.31903  81.57 <0.0000000000000002 ***
## skin     0  -0.18398  0.10808   2.90              0.0887 .  
## age          0.01527  0.00513   8.88              0.0029 ** 
## year         0.01637  0.02469   0.44              0.5072    
## tr    1      0.11595  0.09772   1.41              0.2354    
## exposure     0.13806  0.01016 184.49 <0.0000000000000002 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
## 
## Correlation structure = unstructured 
## Estimated Scale Parameters:
## 
##             Estimate Std.err
## (Intercept)     1.64  0.0776
##   Link = identity 
## 
## Estimated Correlation Parameters:
##           Estimate Std.err
## alpha.1:2    0.164  0.0353
## alpha.1:3    0.178  0.0365
## alpha.1:4    0.199  0.0572
## alpha.1:5    0.186  0.0513
## alpha.2:3    0.197  0.0479
## alpha.2:4    0.181  0.0436
## alpha.2:5    0.150  0.0457
## alpha.3:4    0.317  0.0884
## alpha.3:5    0.312  0.0773
## alpha.4:5    0.245  0.0686
## Number of clusters:   1683  Maximum cluster size: 5
```

```
gee_model4 = geepack::geeglm(y ~ skin + age + year + tr + exposure, data = df, id = id, family = poisson

summary(gee_model4)
```

```
## 
## Call:
## geepack::geeglm(formula = y ~ skin + age + year + tr + exposure, 
##     family = poisson(link = "log"), data = df, id = id, corstr = "AR1")
## 
```

```
##  Coefficients:
##              Estimate  Std.err   Wald            Pr(>|W|)
## (Intercept) -2.88267  0.31844  81.95 <0.0000000000000002 ***
## skin     0  -0.16191  0.11079   2.14              0.1439
## age           0.01496  0.00525   8.12              0.0044 **
## year          0.01759  0.02521   0.49              0.4854
## tr    1       0.12357  0.09941   1.55              0.2139
## exposure      0.13899  0.01055 173.42 <0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation structure = exchangeable ar1 unstructured userdefined fixed
## Estimated Scale Parameters:
##
##              Estimate Std.err
## (Intercept)      1.64  0.0769
##   Link = identity
##
## Estimated Correlation Parameters:
##        Estimate Std.err
## alpha    0.209  0.0262
## Number of clusters:   1683  Maximum cluster size: 5
```

```
# str(gee_model3)
a = gee_model4$geese$alpha
a2 = gee_model3$geese$alpha


M = matrix(data = 0, nrow = 5, ncol = 5, byrow = TRUE)
M[1,] = c(1, a, a^2, a^3, a^4)
M[2,] = c(a, 1, a, a^2, a^3)
M[3,] = c(a^2, a, 1, a, a^2)
M[4,] = c(a^3, a^2, a, 1, a)
M[5,] = c(a^4, a^3, a^2, a, 1)
M
```

```
##          [,1]    [,2]   [,3]    [,4]    [,5]
## [1,] 1.00000 0.20870 0.0436 0.00909 0.00190
## [2,] 0.20870 1.00000 0.2087 0.04356 0.00909
## [3,] 0.04356 0.20870 1.0000 0.20870 0.04356
## [4,] 0.00909 0.04356 0.2087 1.00000 0.20870
## [5,] 0.00190 0.00909 0.0436 0.20870 1.00000
```

```
N = matrix(data = c(1, a2[1], a2[2], a2[3], a2[4],
                    a2[1], 1, a2[5], a2[6], a2[7],
                    a2[2], a2[5], 1, a2[8], a2[9],
                    a2[3], a2[6], a2[8], 1, a2[10],
                    a2[4], a2[7], a2[9], a2[10], 1),
           nrow = 5, ncol = 5, byrow = TRUE)
N
```

```
##       [,1]  [,2]  [,3]  [,4]  [,5]
## [1,] 1.000 0.164 0.178 0.199 0.186
## [2,] 0.164 1.000 0.197 0.181 0.150
## [3,] 0.178 0.197 1.000 0.317 0.312
## [4,] 0.199 0.181 0.317 1.000 0.245
## [5,] 0.186 0.150 0.312 0.245 1.000
```

Part 6

The beta estimates of the parameters appear to be similar across correlation structures with slight differences in the p-values.

However, the correlation parameters appear to decrease as the lag increases in the AR1 correlation structure.

This indicates that there may be overdispersion.