

# Biostat/Stat 571 Exercise #7

Answer Key

March, 2010

## Question 1

### Part g

The results from fitting a GEE with an independent working correlation matrix are given by:

Call:

```
geese(formula = score ~ time8 * tx + I(time8^2) * tx, id = id,  
      data = panss.gee, corstr = "independence")
```

Mean Model:

```
Mean Link:          identity  
Variance to Mean Relation: gaussian
```

Coefficients:

	estimate	san.se	wald	p
(Intercept)	81.48575463	4.2485255	367.8636938	0.000000000
time8	-0.92568018	1.3491495	0.4707624	0.492636644
tx1	-7.41394113	5.4415445	1.8563245	0.173049602
tx2	-9.56240231	4.7471880	4.0575246	0.043974918
I(time8^2)	0.04556935	0.1327651	0.1178089	0.731423162
time8:tx1	0.80971898	1.7988648	0.2026150	0.652618319
time8:tx2	2.28012905	1.4970882	2.3196587	0.127748053
tx1:I(time8^2)	0.24380535	0.1856510	1.7246131	0.189100319
tx2:I(time8^2)	0.41933057	0.1544089	7.3751050	0.006613293

Scale Model:

```
Scale Link:          identity
```

Estimated Scale Parameters:

	estimate	san.se	wald	p
(Intercept)	443.7252	24.74481	321.5584	0

and the results from fitting a GEE with an exchangeable working correlation matrix are given by:

```
Call:
geese(formula = score ~ time8 * tx + I(time8^2) * tx, id = id,
      data = panss.gee, corstr = "exchangeable")
```

```
Mean Model:
Mean Link:          identity
Variance to Mean Relation: gaussian
```

```
Coefficients:
      estimate   san.se      wald      p
(Intercept)  89.77430595 3.7908994 560.81496677 0.0000000000
time8        0.22749591 1.2537242  0.03292632 0.8560095809
tx1         -7.48732189 4.9229475  2.31314364 0.1282843706
tx2        -14.38869181 4.2824810 11.28890257 0.0007797176
I(time8^2)   0.05440923 0.1254858  0.18799917 0.6645871310
time8:tx1    0.51996269 1.6257422  0.10229184 0.7490964748
time8:tx2    1.68199559 1.3564708  1.53754835 0.2149830971
tx1:I(time8^2) 0.20566122 0.1715558  1.43712296 0.2306054756
tx2:I(time8^2) 0.42593672 0.1414776  9.06389608 0.0026070588
```

```
Scale Model:
Scale Link:          identity
```

```
Estimated Scale Parameters:
      estimate   san.se      wald p
(Intercept) 451.8062 25.43676 315.4865 0
```

```
Correlation Model:
Correlation Structure:  exchangeable
Correlation Link:      identity
```

```
Estimated Correlation Parameters:
      estimate   san.se      wald p
alpha 0.6709366 0.02889277 539.2432 0
```

We notice that while our parameter estimates and standard errors are different between the GEE models and the likelihood-based model from part (f) (which is entirely expected), our inference would not differ between the two.

## R Code

```
# Schizophrenia trial data: PANSS scores for patient in six treatment groups.
# Treatment codes: 1 = haloperidol 2 = placebo
# 3 = risperidone10 4 = risperidone16
# 5 = risperidone2 6 = risperidone6
# Variables in column order:
```

```

# group = treatment group
# score.m1 = score at time = -1
# score.0 = score at time = 0
# score.1 = score at time = 1
# score.2 = score at time = 2
# score.4 = score at time = 4
# score.6 = score at time = 6
# score.8 = score at time = 8
# Clinical References: Chouinard et al. (1993)
# Marder and Meibach (1994) Am J Psychiatry 1994 Jun;151(6):825-35
#

library(geepack)

panss.full<-read.table("panss.data", header=FALSE)
names(panss.full)<-c("group", "score.m1", "score.0", "score.1", "score.2", "score.4", "score.6",
"score.8")
panss.full$id<-1:517

# Only consider those in the following 3 treatment groups: placebo, haloperidol, risperidone
(risperidone 6 and 10)
panss.full$tx<-rep(9, 517)
panss.full$tx[panss.full$group == 2] <- 0
panss.full$tx[panss.full$group == 1] <- 1
panss.full$tx[panss.full$group == 3 | panss.full$group == 6] <- 2
panss.full <- panss.full[panss.full$tx != 9,]

# Investigate that monotonicity of the missingness; ie. if a subject drops out, they don't come back.
for(i in 1:344) {
nonmono <- FALSE
for(j in 3:8) {
if( is.na(panss.full[i,(j-1)]) == TRUE & is.na(panss.full[i,j])==FALSE) nonmono=TRUE
}
if(nonmono == TRUE) print( panss.full[i,] )
}

# Subjects 44, 68, 220 have non-monotone missingness
panss.full$score.4[44] <- NA
panss.full$score.4[68] <- NA
panss.full$score.1[220] <- NA

# Convert dataset into a long form and remove observations that have the outcome missing
(ie. post-dropout)
panss.long<-matrix(0, 344*7, 4)
for(i in 1:344) {
panss.long[(((i-1)*7) + 1:7),]<- cbind(rep(i,7), rep(panss.full$tx[i], 7), c(-1,0,1,2,4,6,8),
unlist(as.vector(panss.full[i,2:8]))) )
}

```

```

panss.long <- as.data.frame( panss.long )
names( panss.long ) <- c("id", "tx", "time","score")
panss<-na.omit( panss.long )

# part g
panss$time8<-panss$time - 8
panss$tx <- as.factor( panss$tx )
panss.gee <- panss[panss$time != -1,]
modg.ind<-geese(score ~ time8*tx + I(time8^2)*tx, id=id, data=panss.gee,corstr = "independence")
summary(modg.ind)

modg.exch<-geese(score ~ time8*tx + I(time8^2)*tx, id=id, data=panss.gee, corstr = "exchangeable")
summary(modg.exch)

```