Notes on GEE application using GEEPACK

This note describes the use of `geepack` with `R` to fit a GEE model to clustered data. The basic design is one in which subjects read transcripts of two-person interactions, are told something about the persons in the transcripts, and make a rating about the person. This structure creates clustered observations with within-cluster structure.

Specifically, suppose that there are $rs = 4 \times 2 = 8$ conditions defined by the factors `r` and `s`. Subjects, identified by variable `id`, are is shown two dialog scripts (variable `script`) and one of the rs combinations of `r` and `s` is assigned to each four roles (variable `target`). A rating `y` is obtained. Each subject receives the full set of four levels of `r`, combined with two instances of the two levels of `s`. To balance the procedure, each subject is paired (variable `pairs`) with a second subject who receives the other four conditions assigned to two different dialogs with the opposite assignment of `s`. The order of the `r` conditions for each pair of subjects is are governed by a Latin square (variable `square`), of which only one is used in this example. To unconfound `s` from `target`, the order of the `s` conditions is reversed for alternate pair. These manipulations assure that each target (dialog role) appears once at every level of `r` and twice at every level of `s`.

Data were simulated using the program

```
make.y <- function (d,diag=F, mu=10, r.effect=c(3,0,-1,-2), s.effect=c(1,-1),
    rs.int=matrix(c(1,0,-1,0,-1,0,1,0),4),
    s.script = 0.5, s.target=1, s.id=1.5, s.error=0.5)
{id <- as.numeric(d$id)
 id.effect <- round(rnorm(max(id),0,s.id),3)
 script <- as.numeric(d$script)
 script.effect <- round(rnorm(max(script),0,s.script),3)
 target <- as.numeric(d$target)
 target.effect <- round(rnorm(max(target),0,s.target),3)
 rs.effect <- outer(r.effect,s.effect,'+') + rs.int
 error <- round(rnorm(length(d$y),0,s.error),3)
if (diag){
    print(mu)
    print(id); print(id.effect); print(id.effect[id])
    print(script); print(script.effect); print(script.effect[script])
    print(target); print(target.effect); print(target.effect[target])
    print(rs.effect)
    print(error)}
 y <- numeric(length(d$y))
 for (i in 1:length(y)) {y[i] <- mu + rs.effect[d$r[i],d$s[i]] +
   id.effect[id[i]] - script.effect[script[i]] +
   target.effect[target[i]] + error[i]}
 round(y,2)
```

A data set obtained from one run of this program is given in Table 1.

Correlations among the subject responses are induced by subject differences and by the two levels of `script`. Assuming that the subjects and the scripts are interchangeable, the

	square	pair	id	obs	script	target	r	s	y
1	1	1	1	1	1	1	1	1	14.97
2	1	1	1	2	1	2	2	2	10.17
3	1	1	1	3	2	3	3	1	5.63
4	1	1	1	4	2	4	4	2	6.48
5	1	1	2	1	3	5	1	2	10.77
6	1	1	2	2	3	6	2	1	13.37
7	1	1	2	3	4	7	3	2	10.12
8	1	1	2	4	4	8	4	1	8.65
9	1	2	3	1	1	1	2	2	8.25
10	1	2	3	2	1	2	3	1	9.58
11	1	2	3	3	2	3	4	2	3.94
12	1	2	3	4	2	4	1	1	14.54
13	1	2	4	1	3	5	2	1	8.27
14	1	2	4	2	3	6	3	2	8.53
15	1	2	4	3	4	7	4	1	8.68
16	1	2	4	4	4	8	1	2	8.08
17	1	3	5	1	1	1	3	1	9.71
18	1	3	5	2	1	2	4	2	8.11
19	1	3	5	3	2	3	1	1	13.43
20	1	3	5	4	2	4	2	2	9.15
21	1	3	6	1	3	5	3	2	9.13
22	1	3	6	2	3	6	4	1	11.67
23	1	3	6	3	4	7	1	2	11.95
24	1	3	6	4	4	8	2	1	9.46
25	1	4	7	1	1	1	4	2	8.73
26	1	4	7	2	1	2	1	1	17.97
27	1	4	7	3	2	3	2	2	7.69
28	1	4	7	4	2	4	3	1	11.11
29	1	4	8	1	3	5	4	1	5.41
30	1	4	8	2	3	6	1	2	9.19
31	1	4	8	3	4	7	2	1	8.75
32	1	4	8	4	4	8	3	2	3.94

Table 1: Simulated data for 32 observations obtained from a total of eight subjects.

correlation matrix of the four observations from each subject has the form

$$\begin{bmatrix} 1 & \alpha_1 & \alpha_2 & \alpha_2 \\ \alpha_1 & 1 & \alpha_2 & \alpha_2 \\ \alpha_2 & \alpha_2 & 1 & \alpha_1 \\ \alpha_2 & \alpha_1 & \alpha_1 & 1 \end{bmatrix}$$

Specification of this structure to `geepack` requires a matrix identifying which correlation parameter is attached to each observation. This matrix can be created by the function

```
make.cov <- function (d)
{zc <- genZcor(table(d$id),d$obs,'unstructured')
 z <- matrix(NA,nrow(zc),2)
 z[,1] <- apply(zc[,c(1,6)],1,sum)
 z[,2] <- apply(zc[,2:5],1,sum)
```

```
 z}
```

This routine was used to create the matrix `zbd` used below.

Two GEE models were fitted to the data in Table 1 using the routine `geese` in `geepack`. The first model uses the full factorial combination of `r` and `s`; the second uses an additive specification. The call and the summary output for the interactive model are

```
> gee1 <- geese(y~r*s,id=id,waves=obs,data=bd,zcor=zbd,corstr='userdefined')
> summary(gee1)

Call:
geese(formula = y ~ r * s, id = id, waves = obs, data = bd, zcor = zbd,
    corstr = "userdefined")

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

 Coefficients:
             estimate    san.se       wald            p
(Intercept) 15.230586 0.8471482 323.232059 0.000000e+00
r2          -5.270897 1.5289722  11.884198 5.661220e-04
r3          -6.226172 1.0726773  33.690235 6.462427e-09
r4          -6.625275 1.3262088  24.956519 5.863792e-07
s2          -5.233974 1.2659011  17.094780 3.555991e-05
r2:s2        4.091905 2.0195845   4.105132 4.275325e-02
r3:s2        4.160447 1.4489657   8.244490 4.087590e-03
r4:s2        3.441042 2.3343025   2.173029 1.404493e-01

Scale Model:
 Scale Link:                identity

 Estimated Scale Parameters:
            estimate    san.se     wald            p
(Intercept) 3.508068 0.5881323 35.57836 2.449935e-09

Correlation Model:
 Correlation Structure:     userdefined
 Correlation Link:          identity

 Estimated Correlation Parameters:
         estimate    san.se     wald           p
alpha:1 0.3083220 0.1171746 6.923758 0.008505804
alpha:2 0.3198232 0.1389752 5.295969 0.021374836

Returned Error Value:    0
Number of clusters:   8   Maximum cluster size: 4
```

The comparable analysis of the additive model gives

```
> gee2 <- geese(y~r+s,id=id,waves=obs,data=bd,zcor=zbd,corstr='userdefined')
> summary(gee2)

Call:
geese(formula = y ~ r + s, id = id, waves = obs, data = bd, zcor = zbd,
    corstr = "userdefined")

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

 Coefficients:
             estimate    san.se      wald            p
(Intercept) 13.758739 0.9024602 232.43477 0.000000e+00
r2          -3.213891 0.8766330  13.44085 2.462044e-04
r3          -4.125602 1.2059022  11.70444 6.235126e-04
r4          -4.895461 0.8679420  31.81309 1.697449e-08
s2          -2.310625 0.4751458  23.64858 1.156310e-06

Scale Model:
 Scale Link:                identity

 Estimated Scale Parameters:
            estimate    san.se     wald            p
(Intercept)  4.23747 0.7428962 32.53546 1.170374e-08

Correlation Model:
 Correlation Structure:     userdefined
 Correlation Link:          identity

 Estimated Correlation Parameters:
         estimate    san.se     wald          p
alpha:1 0.3323552 0.1562921 4.522001 0.03346167
alpha:2 0.2240311 0.1643292 1.858605 0.17278589

Returned Error Value:    0
Number of clusters:   8   Maximum cluster size: 4
```

Using these calls with `geeglm` gave the same result, but use of the `anova` method with them failed for unknown reasons.