

12.3: Programming for Steer Example

✓ Using SAS

The SAS code given below will run a repeated measures ANCOVA in SAS for the Neutral Detergent Fiber levels in steer example in section 12.2.

Code

12.2.

```
data steer;
input PER SEQ DIET $ STEER NDF x1 x2;
datalines;
1 1 A 1 50 0 0
1 1 A 2 55 0 0
1 2 B 1 44 0 0
1 2 B 2 51 0 0
1 3 C 1 35 0 0
1 3 C 2 41 0 0
1 4 A 1 54 0 0
1 4 A 2 58 0 0
1 5 B 1 50 0 0
1 5 B 2 55 0 0
1 6 C 1 41 0 0
1 6 C 2 46 0 0
2 1 B 1 61 1 0
2 1 B 2 63 1 0
2 2 C 1 42 0 1
2 2 C 2 45 0 1
2 3 A 1 55 -1 -1
2 3 A 2 56 -1 -1
2 4 C 1 48 1 0
2 4 C 2 51 1 0
2 5 A 1 57 0 1
2 5 A 2 59 0 1
2 6 B 1 56 -1 -1
2 6 B 2 58 -1 -1
3 1 C 1 53 0 1
3 1 C 2 57 0 1
3 2 A 1 57 -1 -1
3 2 A 2 59 -1 -1
3 3 B 1 47 1 0
3 3 B 2 50 1 0
3 4 B 1 51 -1 -1
3 4 B 2 54 -1 -1
3 5 C 1 51 1 0
3 5 C 2 55 1 0
3 6 A 1 58 0 1
```

```
3      6      A      2      61      0      1
;
run;

/*Obtaining fit Statistics*/
proc mixed data=steer;
  class per seq diet steer;
  model ndf = per diet seq x1 x2/ddfm=kr;
  repeated per / subject=steer(seq) type=cs rcorr;
  ods output FitStatistics=FitCS (rename=(value=CS)) FitStatistics=FitCSp;
  title 'Compound Symmetry';
run;

proc mixed data=steer;
  class per seq diet steer;
  model ndf = per diet seq x1 x2/ddfm=kr;
  repeated per / subject=steer(seq) type=AR(1) rcorr;
  ods output FitStatistics=FitAR1 (rename=(value=AR1)) FitStatistics=FitAR1p;
  title 'Autoregressive Lag 1';
run;

proc mixed data=steer;
  class per seq diet steer;
  model ndf = per diet seq x1 x2/ddfm=kr;
  repeated per / subject=steer(seq) type=UN rcorr;
  ods output FitStatistics=FitUN (rename=(value=UN)) FitStatistics=FitUNp;
  title 'Unstructured';
run;

proc mixed data=steer;
  class per seq diet steer;
  model ndf = per diet seq x1 x2/ddfm=kr;
  repeated per / subject=steer(seq) type=CSH rcorr;
  ods output FitStatistics=FitCSH (rename=(value=CSH)) FitStatistics=FitCSHp;
  title 'HETEROGENOUS COMPOUND SYMMETRY';
run;

data fits;
  merge FitCS FitAR1 FitUN FITCSH;
  by desc;
run;
ods listing; title 'Summerized Fit Statistics'; run;
proc print data=fits; run;

/* Model Adjusting for carryover effects */
proc mixed data= steer;
  class per seq diet steer;
```

```

model ndf = per diet seq x1 x2/ddfm=kr;
repeated per / subject=steer(seq) type=csh;
store out_steer;
run;

proc plm restore=out_steer;
  lsmeans diet / adjust=tukey plot=meanplot cl lines;
  ods exclude diffs diffplot;
run;

/* Reduced Model, Ignoring carryover effects */
proc mixed data= steer;
  class per seq diet steer;
  model ndf = per diet seq/ddfm=kr;
  repeated per / subject=steer(seq) type=csh;
  lsmeans diet / pdiff adjust=tukey;
run;

```

The results of the fit statistics are as follows:

Obs	Descr	CS	AR1	UN	CSH
1	-2 Res Log Likelihood	148.3	147.2	121.6	122.5
2	AIC (Smaller is Better)	152.3	151.2	133.6	130.5
3	AICC (Smaller is Better)	152.8	151.7	138.6	132.6
4	BIC (Smaller is Better)	153.2	152.1	136.5	132.5

Based on the fit statistics AIC (and also AICC and BIC), the covariance structure heterogeneous compound symmetry (type=CSH) was shown to be better compared to UN or CS or AR(1). Similar to the covariance structure CS, the CSH covariance structure, also has a constant correlation in the off-diagonal elements. However, the diagonal elements (the variance at each time point), can be different.

Here is the output that is generated for the full model:

Type 3 Tests of Fixed Effects				
Effect	Num DF	Den DF	F Value	Pr > F
PER	2	5.09	10.86	0.0146
DIET	2	11.6	188.52	< .001>
SEQ	5	10.9	31.96	< .0001
x1	1	11.2	17.03	0.0016
x2	1	11.2	78.85	< .0001

The Type 3 tests shown above are "model dependent," meaning that the sum of squares for each of the effects are adjusted for the other effects in the model. In this case, we have adjusted for the presence of carry-over effects. As the diet is significant, it is appropriate to generate LSmeans and the Tukey-Kramer mean comparisons for the diet factor.

DIET Least Squares Means								
DIET	Estimate	Standard Error	DF	t Value	Pr > t	Alpha	Lower	upper
A	57.8092	1.6046	6.412	36.03	< 0.001	0.05	53.9432	61.6752
B	50.8134	1.6046	6.412	31.67	< 0.001	0.05	46.9474	54.6794
C	48.3774	1.6046	6.412	30.15	< 0.001	0.05	44.5114	52.2434

To see the adjustment on the treatment means, we can compare the LSmeans for a reduced model that does not contain the carry-over covariates.

LSmeans		
Full Model with Covariates		
Effect	DIET	Estimate
DIET	A	57.8092
DIET	B	50.8134
DIET	C	48.3774
Reduced Model (without carry-over covariates)		
Effect	DIET	Estimate
DIET	A	57.3941
DIET	B	50.9766
DIET	C	48.6292

Although the differences in the LSmeans between the two models are small in this particular example, these carry-over effect adjustments can be very important in many research situations.

✓ Using R

- Load the Steer Data.
- Run the analysis by using different covariance structures and obtain fit statistics.

Code

- Load the Steer data, run the analysis by using different covariance structures and obtain fit statistics by using the following commands:

```
setwd("~/path-to-folder/")
steer_data <- read.table("steer_data.txt", header=T)
attach(steer_data)
library(nlme)
```

```
model_CS<-glms(NDF ~ factor(PER) + DIET + factor(SEQ) + x1 + x2, steer_data, correlat:
model_AR<-glms(NDF ~ factor(PER) + DIET + factor(SEQ) + x1 + x2, steer_data, correlat:
Model_Selection <- data.frame(
  c("", "-2LogLik", "AIC", "BIC"),
  c("CS", round(-2*summary(model_CS)$logLik, 2), round(summary(model_CS)$AIC, 2), round(s
  c("AR1", round(-2*summary(model_AR)$logLik, 2), round(summary(model_AR)$AIC, 2), round
  stringsAsFactors = FALSE)
names(Model_Selection) <- c( " ", "", "", "")
print(Model_Selection)
#1          CS    AR1
#2 -2LogLik 140.75 141.11
#3      AIC 168.75 169.11
#4      BIC 185.25 185.61
detach(steer_data)
```

This page titled [12.3: Programming for Steer Example](#) is shared under a [CC BY-NC 4.0](#) license and was authored, remixed, and/or curated by [Penn State's Department of Statistics](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.