Some guidance on R commands
for repeated measures

The three main statistical commands used in the course materials (which are written for Stata) for the repeated measures portion of the course are xtgee, mixed, and meglm (with some specializations such as melogit). These fit Generalized Estimating Equations (GEE - xtgee), mixed models for approximately normally distributed data (mixed) and mixed models for other distributions (meglm). The routines in R are not nearly as good as those in Stata, but, fortunately, R will handle most of the standard situations. Below is an identification of how to translate the Stata code into R.

The corresponding R commands are:

Stata command	**R command, library**
xtgee	glmgee, library(glmtoolbox)
mixed	lmer, library(lme4)
meglm	glmer, library(lme4)

Note: I prefer the glmtoolbox library over more common libraries like gee since its ability to conduct omnibus tests of interactions and categorical variables is more robust

xtgee

The syntax for xtgee in Stata is

```
xtgee outcome_variable predictor_variables, i(cluster_variable)
family(distribution) link(link_function) corr(cov_structure)
```

with or without the "robust" option.

example:
```
xtgee bweight birthord initage, i(momid) family(gaussian)
link(identity) corr(exchangeable)
```

The syntax is similar in R

```
glmgee(outcome_variable + predictor_variable, id=cluster_variable,
family = distribution(link= "link_function"), corstr =
"cov_structure")
```

example:
```
GEE_fit <- glmgee(bweight ~ birthord+initage, id=momid,
family=gaussian(link="identity"), data=GAbabies,
corstr="exchangeable")
```

Note: R automatically fits the model with and without the robust option.

Here are two other examples for fitting binary outcome models, a logistic and a relative risk regression.

Logistic regression model

Stata syntax:

```
xtgee lowbirth birthord initage, i(momid) family(binomial)
link(logit) corr(exch) robust
```

R syntax:

```
GEE_fit_logit <- glmgee(lowbirth ~ birthord+initage, id=momid,
family=binomial(link="logit"), data=GAbabies, corstr="exchangeable")
```

Relative risk model

Stata syntax:
```
xtgee lowbirth birthord initage, i(momid) family(binomial)
link(log) corr(exch) robust
```

R syntax:
```
GEE_fit_log <- glmgee(lowbirth ~ birthord+initage, id=momid,
family=binomial(link="log"), data=GAbabies, corstr="exchangeable")
```

mixed

Stata general syntax:
```
mixed outcome_variable predictor_variables ||
random_factor:, reml
```

R general syntax:
```
Mix_model <- lmer(outcome_variable ~ predictor_variables
+(1|random_factor), data=dataset)
```

Notes:

1) R automatically uses the REML option, so you do not need to specify it.
2) Stata places random intercepts after ||.
3) R denotes random intercepts with a 1 (for intercept) followed by a | and the clustering variable.

Example:

Stata syntax:
```
mixed bweight birthord initage || momid:, reml
```

R syntax:
```
mix_model <- lmer(bweight ~ birthord + initage +(1|momid),
data=GAbabies)
```

meglm

Stata general syntax:
```
meglm outcome_variable predictor_variables ||
random_factor:, family(distribution) link(link_function)
```

R general syntax:
```
meglm_model <- glmer(outcome_variable ~
predictor_variables +(1|random_factor), nAGQ=20, data=dataset, family
= distribution(link= “link_function”))
```

Note: I like to specify the option “nAGQ=20” forces the model to use maximum likelihood, which is a more appropriate fitting method. But it then can only handle a single level of clustering. The value can be set higher than 20 to get higher precision, which sometimes helps with fits that will not converge. The value can be set lower to speed up the fitting, which can be slow with large datasets.

Logistic regression example

Stata:
```
meglm lowbirth birthord initage || momid:, family(binomial)
link(logit)
```

R:
```
mixed_logit <- glmer(lowbirth ~ birthord+initage+(1|momid), nAGQ=20,
family=binomial(link="logit"), data=GAbabies)
```

Spaghetti plot:

```
> GAbabies_sub <- GAbabies[GAbabies$momid < 2500,]
> p_sub<-ggplot(data=GAbabies_sub, aes(x=birthord, y=bweight, group=momid))
> p_sub+geom_line()
```