Lab 5

Psychology 319 (GCM)

Instructions. Work through the lab, saving the output as you go. If you work in Microsoft Word, you can easily copy any graph to Word via the clipboard. Numerical output may also be copied easily by highlighting, moving it to the clipboard, then copying into Word. However, you should format R output in TrueType Courier New font so that it is *monospaced*. Output from this lab is to be handed in by Monday, March 1. Your output file should be named `LAST_FIRST_LAB5.DOC`, where `LAST` is your last name, and `FIRST` is your first name. Any additional files should have the same naming scheme, except the file extension should be correct. You may add any description text you wish after `LAB5` in the file name.

Preamble. In today's lab, we examine the effect of some assumptions on some commonly computed statistics in longitudinal multilevel modeling.

1 Introduction

Multilevel mixed modeling is employed widely in the analysis of longitudinal data. We've just spent several weeks gaining some facility in this kind of modeling. Now that we are no longer ignorant about the basics, we might start asking some questions about the procedures we've learned how to use.

Recall that the regression analysis procedures we employ rest on several assumptions. One frequently forgotten assumption is that the X scores are fixed. However, in many applications, the covariates are random variables. Another hidden assumption is that the measures being employed are perfectly reliable. In some cases this is nearly true, but in others it is obviously false.

2 Unreliable Data

In Lab 2, we learned how to generate artificial data for a simple multilevel model in which the relationship between X and Y varied randomly across schools. I want you to go back to your code for Lab 2, and modify it in the following way. Suppose that X is measured with perfect reliability, but Y is not. In other words, your Y data produced in the previous lab represents the

true scores that you would obtain if they were actually available, but these Y scores are not actually available, because they are perturbed by random error. Suppose that the Y values are measured with reliability .70 in all schools. That is, the Y scores you observe are not the Y scores produced in Lab 2, but are rather those scores perturbed by enough normally distributed random error that the true Y values account for only 80% of the variance of the observed Y values. Create these new Y values, and repeat your previous analysis and compare it with the same analysis on the new unreliable data. What is the effect of the unreliability?

3 The Fixed X Assumption

Suppose we have just two variables, X and Y, with population means of 0 and standard deviations of 1, and they have a bivariate normal distribution with correlation .6. So the population regression coefficient between the variables is 0.60. You can create bivariate normal data with these characteristics in a variety of ways. One way is to load the Steiger Library Functions code from Lab 2 and use the `MultivariateNormalSample` function.

Let's do that.

```
> mu <- c(0,0)
> Sigma <- matrix(c(1,.6,.6,1),2,2)
> source("Steiger R Library Functions.R")
> set.seed(12345)
> data <- MultivariateNormalSample(mu,Sigma,50)
> y <- data[,1]
> x <- data[,2]
```

We can then fit the simple bivariate regression model.

```
> fit <- lm(y~x)
```

We can look at the estimates and their standard errors.

```
> summary(fit)

Call:
lm(formula = y ~ x)

Residuals:
    Min      1Q  Median      3Q     Max
-1.7998 -0.4990 -0.0689  0.6022  1.6412
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.06436    0.13190   0.488    0.628
x            0.58090    0.11203   5.185 4.28e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8901 on 48 degrees of freedom
Multiple R-squared: 0.359,        Adjusted R-squared: 0.3457
F-statistic: 26.89 on 1 and 48 DF,  p-value: 4.277e-06
```

You can "extract" the estimate for the slope, and its standard error, as follows:

```
> beta <- summary(fit)$coefficients[2,1]
> se.beta <- summary(fit)$coefficients[2,2]
> beta
[1] 0.5809044
> se.beta
[1] 0.1120298
```

How accurate are the estimates and standard errors?

I'd like you to check this out with a brief Monte Carlo study. There are, of course, many things you *could* look at, but try doing the following:

1. Do one experiment with $n = 5$, and another with $n = 50$.
2. For each experiment do 1,000 replications.
3. Create a function called `sampleit` which takes as its arguments `mu`, `Sigma`, and `n`, and returns a vector with the sample `beta` and `se.beta` (that is, the estimate and its *estimated* standard error) in it.
4. Use the replicate function to run 1,000 reps (test it first with, say, 10 reps!) and save the results in a matrix called `outcome`.
5. You'll probably have to transpose `outcome` to get the data into columns instead of rows. Use the R function `t(outcome)` to do this.

Once you have your data, here is what I want you to do. Your data should have 1000 rows and 2 columns. You can get the column means with the function call `apply(outcome,2,mean)` which applies the `mean` function to the second dimension (i.e., the column) of the `outcome` matrix. You now have the mean estimated coefficient, and the mean of the estimated standard errors. Compare the mean of the estimated coefficients with the true value (.60). But what about the standard error estimates? We have their mean. But what is the true standard error? We can estimate the true standard error by taking the standard deviation of our first column of data. Use the call `sd(outcome)` and look at the first standard deviation. Is the standard error systematically biased?

To be more certain of your conclusion, do 25,000 replications. This may take some time, depending on your computer, so plan accordingly!!

Then, try the same experiment with $n = 50$.