Repeated Measures and Mixed Models

Contents

Introduction

Some are still trained in statistics with a heavy focus on so-called analysis of variance (ANOVA) techniques, but without delving too much (or at least, enough) into their role as special cases of regression, and mixed models more generally. This is despite the fact that data collected is typically richer than a couple of tightly controlled treatment conditions with a continuous target variable, rendering these techniques too restrictive for many common situations. The following uses a very small dataset to demonstrate the connections to standard linear and mixed models, with a hope that some will consider using the more general approaches after realizing they are only gaining in flexibility while maintaining the same analyses of interest they are already using.

Data Setup

Let's look at some simple data in which the effect is *eyeballable*. We will use it in both wide format, where a row would represent a single observational unit, and long format, where multiple rows belong to the observational unit.

```
library(dplyr)
treat  = rep(c('treat', 'control'), e=5)
pre = c(20,10,60,20,10,50,10,40,20,10)
post = c(70,50,90,60,50,20,10,30,50,10)

df = data.frame(id=factor(1:10), treat, pre, post)
change = post-pre

dflong = tidyr::gather(df, key=time, value=score, pre:post) %>% arrange(id)
head(df)
```

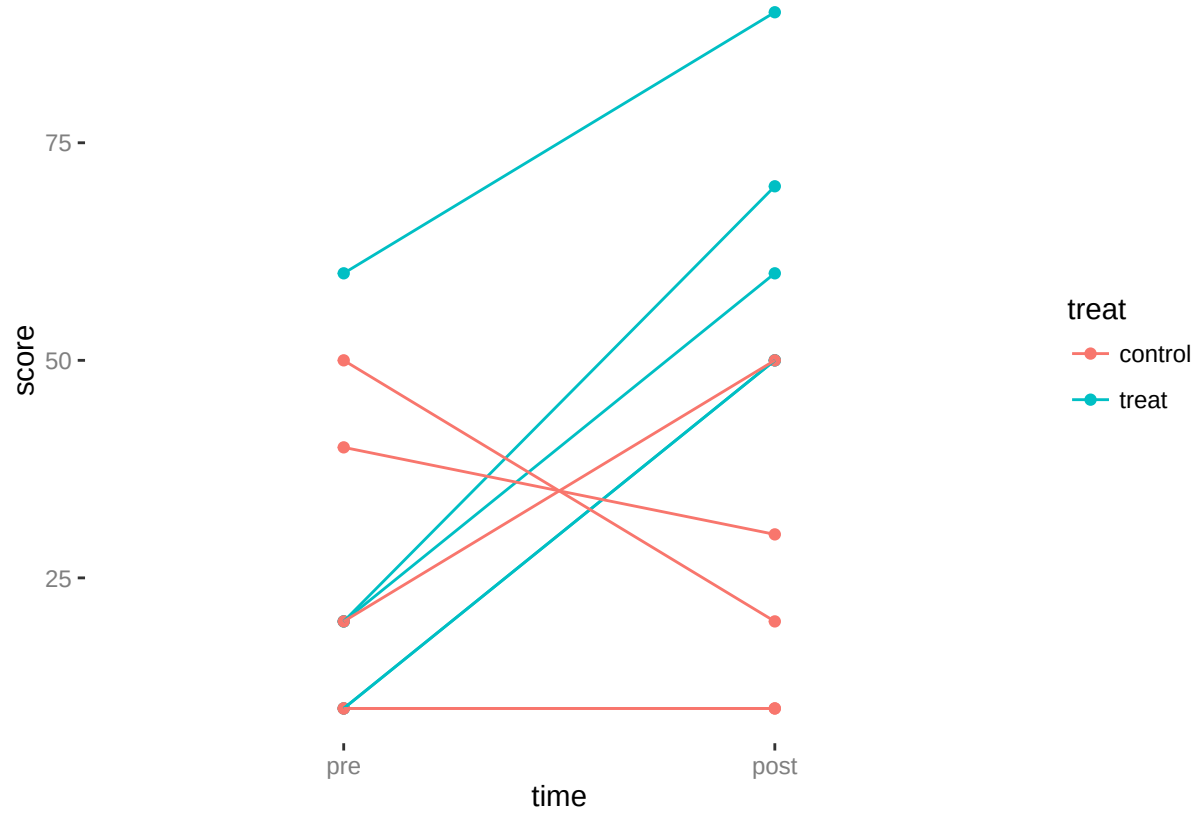
```
##   id   treat pre post
## 1  1   treat  20   70
## 2  2   treat  10   50
## 3  3   treat  60   90
## 4  4   treat  20   60
```

```
## 5  5   treat  10   50
## 6  6 control  50   20
```

```
head(dflong)
```

```
##   id treat time score
## 1  1 treat  pre    20
## 2  1 treat post    70
## 3  2 treat  pre    10
## 4  2 treat post    50
## 5  3 treat  pre    60
## 6  3 treat post    90
```

When I meant the effect is eyeballable, everyone in the treatment group improves from pre to post, while controls are mixed at best, and this effect is visually obvious. I should note that when patterns are this obvious, the compulsion to do statistical tests should be stifled. We only do so here for expository reasons.



Between Groups Approach

t-test

Ignoring the correlated nature of the data for now, we will show the simple regression of post score on treatment to show how the different approaches can produce identical results.

```
ttestModel = t.test(post ~ treat, data=df, var.equal=T)
ttestModel
```

```
##
##  Two Sample t-test
##
## data:  post by treat
## t = -3.7796, df = 8, p-value = 0.005391
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -64.40445 -15.59555
## sample estimates:
## mean in group control   mean in group treat
##                    24                    64
```

```
ttestModel$statistic^2
```

```
##        t
## 14.28571
```

ANOVA

Now for the ANOVA, which is identical.

```
anovaModelPostOnly = aov(post ~ treat, df)
summary(anovaModelPostOnly)
```

```
##             Df Sum Sq Mean Sq F value  Pr(>F)
## treat        1   4000    4000   14.29 0.00539 **
## Residuals    8   2240     280
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
coef(anovaModelPostOnly)
```

```
## (Intercept)  treattreat
##          24          40
```

$t^2 = F$, with identical p-values.

Standard Regression

Now for the standard linear model (SLiM).

```
lmModelPostOnly = lm(post ~ treat, df)
summary(lmModelPostOnly)
```

```
##
## Call:
## lm(formula = post ~ treat, data = df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##    -14    -14     -4      6     26
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   24.000      7.483   3.207  0.01248 *
## treattreat    40.000     10.583   3.780  0.00539 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.73 on 8 degrees of freedom
## Multiple R-squared:  0.641,  Adjusted R-squared:  0.5962
## F-statistic: 14.29 on 1 and 8 DF,  p-value: 0.005391
```

```
coef(lmModelPostOnly)
```

```
## (Intercept)  treattreat
##          24          40
```

The intercept is the mean for the reference group, while the coefficient for group is the difference between their respective means. The t for the coefficient (and associated p-value) is identical to the t from the t-test (sign is arbitrary depending on how the labels are used), while the overall model F in the regression is the same as in the ANOVA. This equivalence is explicit in R, as the `aov` function actually calls the `lm` function, and the class of an aov object is both 'aov' and 'lm'. I guess I should point out that SAS and SPSS ANOVA tables default to a sums of squares approach that is not equivalent to a SLiM in the presence of interactions (and which some would say doesn't make much sense). See the help file for `Anova` in the car package for details. The point here is simply that we have run the same model three different ways, but produced identical results in each case.

Pre-Post Design

Now let's incorporate the pre-post nature of the data. We'll start with a t-test on the change from pre to post. Our target variable is the change score (sometimes called the gain or difference score).

t-test

```
ttestChange = t.test(change ~ treat, df, var.equal=T)
ttestChange
```

```
##
##  Two Sample t-test
##
## data:  change by treat
## t = -4.1184, df = 8, p-value = 0.003351
## alternative hypothesis: true difference in means is not equal to 0
```

```
## 95 percent confidence interval:
##  -65.51672 -18.48328
## sample estimates:
## mean in group control   mean in group treat
##                    -2                     40
```

```
ttestChange$statistic^2
```

```
##        t
## 16.96154
```

ANCOVA

An alternative approach is ANCOVA, which changes the substantive emphasis to examining post-test scores while controlling for pre-test. The target variable is thus fundamentally different than in the previous model (raw vs. change). However, note that an ANCOVA is a sequential regression model that examines the treatment effect while controlling for pretest scores.

```
ancova = aov(post~ pre + treat)
ancovalm = lm(post~ pre + treat)
rbind(coef(ancova), coef(ancovalm))
```

```
##      (Intercept)       pre treattreat
## [1,]    10.31579 0.5263158   41.05263
## [2,]    10.31579 0.5263158   41.05263
```

Again the outcomes are identical, as `aov` uses `lm` under the hood.

Since Lord's Paradox was introduced, there has been some back and forth among various papers ever since debating whether one should use ANCOVA vs. t-test on change scores. Aside from it not really being a paradox (both statisticians are correct), it's unclear to me that the debate is very useful for applied researchers for various reasons. For one, I've never seen a study in which I'd be interested in the results of a t-test. At best it tells an overly simplistic story, and we have far better, and as easily implemented, tools to help us understand the complexities of nature. As such, a t-test on gain scores is never going to be an option outside of a perhaps misguided use as a descriptive statistic. Some are concerned about statistical power, but basing your assessment of treatment viability based solely on an arbitrary p-value cutoff is borderline unethical for some lines of research (in my opinion), so that shouldn't be the overriding concern either. In addition, whether observational or experimental, a statistical result will not tell you about causal relations. You can, however, specify those causal effects that you think exist and run a model appropriate to the situation, focusing on the specific effect of interest. In this case the t-test and ANCOVA can be seen as different effects, *total* and *direct* respectively, from the same causal model (see Pearl, 2014).

Statistically speaking, as we will see, both approaches are special cases of more flexible models that can better handle the nuances of the typical, more complex data situation in which there are multiple predictors, interactions of interest, nonlinear relationships, or more than two time points, etc., such that neither would be appropriate. But statistically, ANCOVA is no different than the SLiM, so it only has the issues that the SLiM has. A t-test on change scores is part of a repeated measures ANOVA result that is a special case of mixed models, which themselves generalize the SLiM.

Finally, there is actually a way to convert the t-test on change scores to the ANCOVA result, such that, *if there is a strong correlation between pre and post*, one can find an equivalence (using the squared t vs. the ANCOVA F) with increasing sample size. The code in the Appendix demonstrates this.

Repeated Measures Anova

Now that we've gone through that, let's do a repeated measures ANOVA with treatment as the between subjects effect and score as the repeated measure (pre-post).

```
anovaModelRM = aov(score ~ treat*time + Error(id), dflong)
summary(anovaModelRM)
```

```
##
## Error: id
##           Df Sum Sq Mean Sq F value Pr(>F)
## treat      1   1805    1805   3.406  0.102
## Residuals  8   4240     530
##
## Error: Within
##            Df Sum Sq Mean Sq F value  Pr(>F)
## time        1   1805    1805   13.88 0.00582 **
## treat:time  1   2205    2205   16.96 0.00335 **
## Residuals   8   1040     130
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Note that the interaction F-value here is the squared value of the t-test on the change scores (-4.122). In other words, a t-test on the change score is the interaction term from the repeated measures ANOVA on the same data.

Mixed Model

From the mixed model, we also use the long form of data. We'll use an R package specifically for mixed models, lme4, but R comes with the nlme package that could also be used.

```
library(lme4)
lmeModel = lmer(score ~ treat*time + (1|id), data=dflong)
anova(lmeModel)
```

```
## Analysis of Variance Table
##            Df  Sum Sq Mean Sq F value
## treat       1  442.74  442.74  3.4057
## time        1 1805.00 1805.00 13.8846
## treat:time  1 2205.00 2205.00 16.9615
```

While the estimation approaches are different, the end results are the same as with the ANOVA. However, now we are using a tool that can handle additional time points, continuous covariates with possibly nonlinear relationships, different types of outcome variables, other correlational structures among the observations, etc. In fact, if one looks at the help file for `aov`, one will note that it suggests using `lme` for unbalanced designs and other situations (see the appendix for additional examples).

Summary

In general, standard ANOVA techniques are special cases of modeling approaches that are far more flexible, extensible, and often just as easy to use. R is useful because just a simple inspection of the help file for the

aov function makes clear the ties to linear and linear mixed models. If one has a straightforward experimental design and is not interested in exploring complex interactions or dealing with data nuances, the 'sums of squares' approach may be adequate. What's happened in the past (e.g. within psychology departments), is that people were taught statistics with a strong focus on ANOVA, but who were not likely going to do experiments. This resulted in students forcing their data to conform to the technique (e.g. target variable transforms, categorizing numeric covariates), rather than finding a suitable techniques for how their data is represented. While such teaching seems far less the case these days (thankfully!), some are still taught that way or are not given enough time to learn during their short statistical exposure how to generalize beyond simple settings. Hopefully this document can help a bit in that case.

Appendix

t-test ANCOVA equivalence

Set **n** to very large and the estimates will converge. The values may be pretty close even without the transform, which just makes it exact with large N. Again, this assumes a notable correlation between pre and post, but that's typically the case. Note also, that this assumes zero correlation between treatment condition and pre-test score, which should be the case in the experimental setting. The issue with regard to Lord's Paradox revolves around the situation in which groups are different at baseline/pre.

```
# setup
set.seed(12)
n = 1000
pre = rnorm(n)
treat = rep(0:1, e=n/2)
post = 1*pre + .25*treat + rnorm(n)
df = data.frame(treat, pre, post)
df %>% group_by(treat) %>% summarise(mean(pre), mean(post))
```

```
## Source: local data frame [2 x 3]
##
##   treat   mean(pre) mean(post)
##   (int)       (dbl)      (dbl)
## 1     0 -0.02285966 0.01079596
## 2     1 -0.03001336 0.25611457
```

```
# create transform value; see Knapp & Schafer 2009
# variances
varPre = var(pre)
varPost = var(post)

varWithinPre = df %>% group_by(treat) %>%  summarise(varr = var(pre))
varWithinPre = mean(varWithinPre$varr)
varWithinPost = df %>% group_by(treat) %>%  summarise(varr = var(post))
varWithinPost = mean(varWithinPost$varr)

# correlations
corPrePost = cor(pre, post); corPrePost
```

```
## [1] 0.6767847
```

```
corGroups = df %>% group_by(treat) %>%  summarise(corr = cor(pre,post))
corWithin = corGroups %$% sqrt(sum(corr^2)/2)

# transformation value
Anum =
  ( ((n-1)*varPost*(1-corPrePost^2))/
    ((n-2)*varWithinPost*(1-corWithin^2))  -1)
Adenom = ((n-1)*(varPre + varPost - 2*corPrePost*sqrt(varPre)*sqrt(varPost))) /
            ((n-2)*(varWithinPre + varWithinPost - 2*corWithin*sqrt(varWithinPre)*sqrt(varWithinPost)))

A = (n-3)/(n-2) * Anum/Adenom


# grab relevant statistics
ttestChange = t.test(post-pre~treat, data=df, var.equal=T)
ancova = aov(post~ pre + treat, data=df)
ancovaF = summary(ancova)[[1]]$F[2]
tF = ttestChange$statistic^2

c(tConverted = tF*A, ancovaF=ancovaF)
```

```
## tConverted.t      ancovaF
##     15.11989     15.11698
```

More mixed model, repeated measures anova equivalence

```
library(nlme)
fm1BW.lme <- lme(Reaction ~ Days, sleepstudy, random = ~ 1|Subject)
fm1BW.lmeSphere <- update(fm1BW.lme, correlation = corCompSymm(form = ~ Days|Subject))

anova(fm1BW.lme)
```

```
##             numDF denDF   F-value p-value
## (Intercept)     1   161 1087.9793  <.0001
## Days            1   161  169.4014  <.0001
```

```
anova(fm1BW.lmeSphere)
```

```
##             numDF denDF   F-value p-value
## (Intercept)     1   161 1087.9793  <.0001
## Days            1   161  169.4014  <.0001
```

```
summary(aov(Reaction ~ Days + Error(Subject), sleepstudy))
```

```
##
## Error: Subject
##           Df Sum Sq Mean Sq F value Pr(>F)
## Residuals 17 250618   14742
##
```

```
## Error: Within
##            Df Sum Sq Mean Sq F value Pr(>F)
## Days        1 162703  162703   169.4 <2e-16 ***
## Residuals 161 154634     960
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
data(Orthodont)

anova(lme(distance ~ age*Sex, random = ~ 1 | Subject, data = Orthodont))
```

```
##             numDF denDF  F-value p-value
## (Intercept)     1    79 4123.156  <.0001
## age             1    79  122.450  <.0001
## Sex             1    25    9.292  0.0054
## age:Sex         1    79    6.303  0.0141
```

```
summary(aov(distance ~ age*Sex + Error(Subject), data = Orthodont))
```

```
##
## Error: Subject
##           Df Sum Sq Mean Sq F value  Pr(>F)
## Sex        1  140.5  140.46   9.292 0.00538 **
## Residuals 25  377.9   15.12
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Error: Within
##           Df Sum Sq Mean Sq F value Pr(>F)
## age        1 235.36  235.36 122.450 <2e-16 ***
## age:Sex    1  12.11   12.11   6.303 0.0141 *
## Residuals 79 151.84    1.92
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```