Precept 8

Summary of R Topics Covered in the Tutorial

We will mostly be using `lm` to run multiple regression. However, we'll be making its formula interface do more work for us.

Can Television be Educational?

In this precept we're going to re-revisit The Electric Company on children's reading ability. For more details of the experiment see Cooney (1976).

The dataset `electric-company.csv` in the `data` folder contains the following variables:

Name	Description
`pair`	The index of the treated and control pair (ignored here).
`city`	The city: Fresno ("F") or Youngstown ("Y")
`grade`	Grade (1 through 4)
`supp`	Whether the program replaced ("R") or supplemented ("S") a reading activity
`treatment`	"T" if the class was treated, "C" otherwise
`pre.score`	Class reading score *before* treatment, at the beginning of the school year
`post.score`	Class reading score at the end of the school year

As a reminder, every observation is a class of students, which was either *treated*, if the program was shown to them, or *control* if the program was not shown as part of their studies. The outcome of interest, our 'dependent variable', is the class's average score on a reading test at the end of the year. We've called that `post.score`. Every observation in our data is a separate class, so no class got the treatment more than once.

Question 1

Back in the previous precept we say that we could get a pretty precise idea of the treatment effects that might be expected in each grade. But we had to fit four separate models. Here's the code we used back then. We'll make a nominal (categorical) version of grade too.

```
electric <- read.csv("data/electric-company.csv")
electric$grade.nom <- as.factor(electric$grade)

mod1 <- lm(post.score ~ treatment + pre.score, data = electric,
           subset = grade == 1)
mod2 <- lm(post.score ~ treatment + pre.score, data = electric,
           subset = grade == 2)
mod3 <- lm(post.score ~ treatment + pre.score, data = electric,
           subset = grade == 3)
mod4 <- lm(post.score ~ treatment + pre.score, data = electric,
           subset = grade == 4)
```

This worked fine, but was a lot of typing. It also had a couple of other disadvantages.

How many classes were used to fit these models? What variable effects did we implicity treat as potentially differing between models?

Answer 1

Everything was allowed to vary, i.e. treatment and pre.score. Nothing learned about one grade was available to tell us anything about another grade.

Question 2

Now let's try to learn about separate grade effects in a single model. One way to do this is to *interact* treatment with grade. Here's a general modeling principle:

> If you think the *effect* of variable A varies according to the *values* of variable B, then you should think of *adding an interaction* between A and B in your model

Reminder: In the `lm` formula interface this amounts to adding an `A:B` term. For example, if A and B interact to predict Y then the formula would be

```
Y ~ A + B + A:B
```

which would fit the model

$$Y_i = \beta_0 + A_i\beta_A + B_i\beta_B + (A_i \times B_i)\beta_{AB} + \epsilon_i$$

Another way to fit this model is to use `A*B` to interact `A` and `B`. Since we always want to have `A` and `B` if we have an `A:B` term, this notation makes sure we don't forget any of them. So to fit the model above using this notation the formula is

```
Y ~ A * B
```

which is the same model as before because `A * B` is exactly `A + B + A:B`.

Now fit a model of all the grades that includes `pre.score`, `treatment`, and `grade.nom`, and also interacts `treatment` and `grade.nom`. Summarize the results.

Answer 2

```
modint <- lm(post.score ~ treatment + grade.nom + treatment:grade.nom +
                     pre.score, data = electric)
summary(modint)
```

```
Call:
lm(formula = post.score ~ treatment + grade.nom + treatment:grade.nom +
    pre.score, data = electric)

Residuals:
     Min       1Q   Median       3Q      Max
-27.4940  -2.4504  -0.1819   2.9730  27.8431

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)          56.25984    1.80771  31.122  < 2e-16 ***
treatmentT            8.37638    2.24218   3.736  0.00025 ***
```

```
grade.nom2            -19.75811    3.66615  -5.389 2.16e-07 ***
grade.nom3            -26.91855    5.00126  -5.382 2.23e-07 ***
grade.nom4            -29.50395    5.41175  -5.452 1.60e-07 ***
pre.score               0.80202    0.05558  14.429  < 2e-16 ***
treatmentT:grade.nom2  -4.17864    2.86683  -1.458  0.14667
treatmentT:grade.nom3  -6.19673    3.21265  -1.929  0.05530 .
treatmentT:grade.nom4  -7.12257    3.17577  -2.243  0.02611 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.265 on 183 degrees of freedom
Multiple R-squared:  0.8396,    Adjusted R-squared:  0.8326
F-statistic: 119.8 on 8 and 183 DF,  p-value: < 2.2e-16
```

Question 3.1

Let's start with our estimated coefficients.

On a piece of paper or in a file, write out the mathematical form of the regression equation of the model we just fitted, filling in the βs with their estimated values from the summary in the previous question. (Remember that `grade.nom` turns into a dummy/indicator variable for each of grades 2 through 4.)

Answer 3.1

Roughly:

expected post.score = 56 + treat x 8 + grade2 x -20 + grade3 x -27 + grade4 x -30 + pre.score 0.8 + (treat x grade2) x -4 + (treat x grade3) x -6 + (treat x grade4) x -7

Question 3.2

In this model, what is the expected value of `post.score` for a *first* grade class with a `pre.score` of 100, who did *not* see the television program? Now compute the expected value of `post.score` for this same class if they *had* seen the television program. (You can round up the numbers a bit if it makes the math easier). The difference is the treatment effect in grade 1. What is it? Does ths quantity correspond to a coefficient?

Answer 3.2

From above, 56 + 80 and 56 + 8 + 80. So a difference (treatment effect) of 8. And it corresponds to the treatment coefficient.

Question 3.3

Now do the same calculation but for a class in grade 2. Now the difference is the treatment effect in grade 2. What is it? Does it correspond to a coefficient?

Describe in substantive rather than statistical terms what the test that `summary` performs on the interaction term `treatmentT:grade.nom2` actually tests. Can we reject the null?

Answer 3.3

From above 56 + 80 - 19 and 56 + 8 + 80 - 19 - 4. So a difference (treatment effect) of 8 - 4 = 4. It corresponds to two coefficients: treatment and treatment:grade.nom2

The test on the interaction term asks whether the treatment effect in grade 2 is different to the treatment effect in grade 1. The null hypothesis is that it is not. Apparently given this model, we cannot reject this null.

Question 3.4

Now do the same calculation but for a class in grade 3. Now the difference is the treatment effect in grade 3. What is it?

What happens to our view of the treatment if we set the pre.score for these classes to 50 instead of 100?

Answer 3.4

From above 56 + 80 - 26 and 56 + 8 + 80 - 26 - 6. So a difference (treatment effect) of 8 - 6 = 2.

Nothing, because pre.score does not affect treatment anywhere in the model so we just add 40 rather than 80 to treatment and control classes, so it makes no difference to the difference.

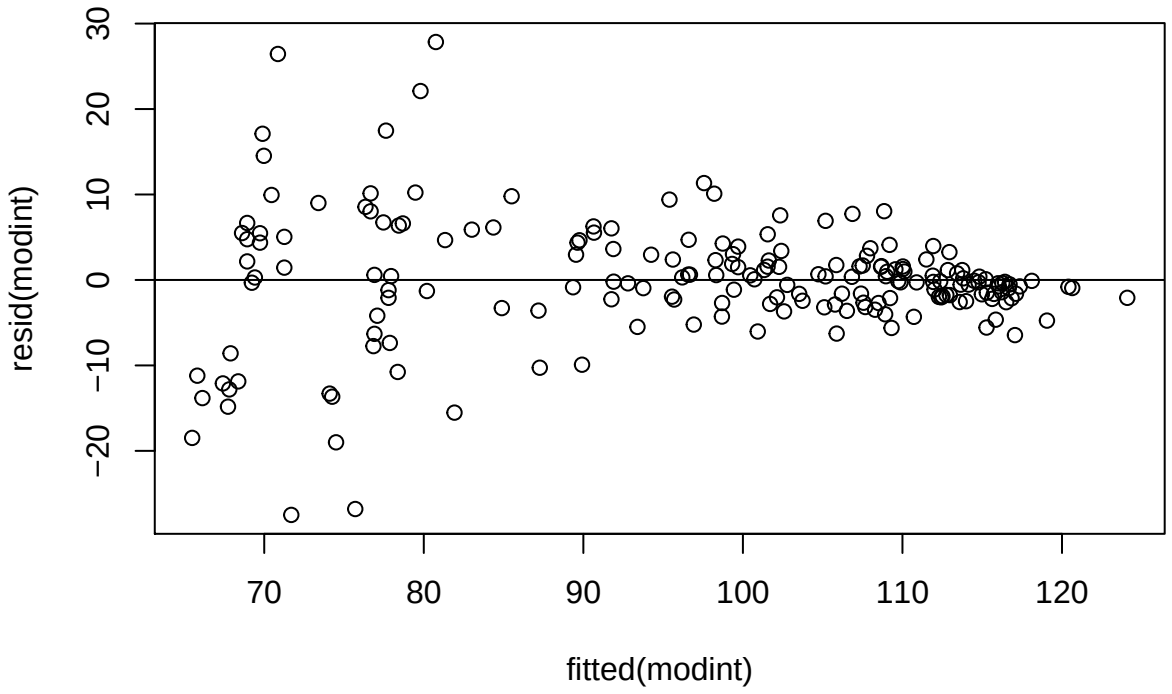
Question 4.1

Let's see what we might be missing in this model by examining the *residuals* (estimated error terms for each class). Extract the residuals from the model and plot them against the model's *fitted* values, marking 0 with a horizontal line. Hint, the functions `resid` and `fitted` will be helpful here.

What pattern do you see?

Answer 4.1

```
plot(fitted(modint), resid(modint))
abline(h = 0)
```



classes with lower predicted scores are predicted less well - have larger residuals - than classes with higher predicted scores.

Question 4.2

Let's investigate how this pattern might relate to other variables. Remake the plot in the previous question, but use different colors for the different grades. Now do remake the plot with treated classes a different color from control classes.

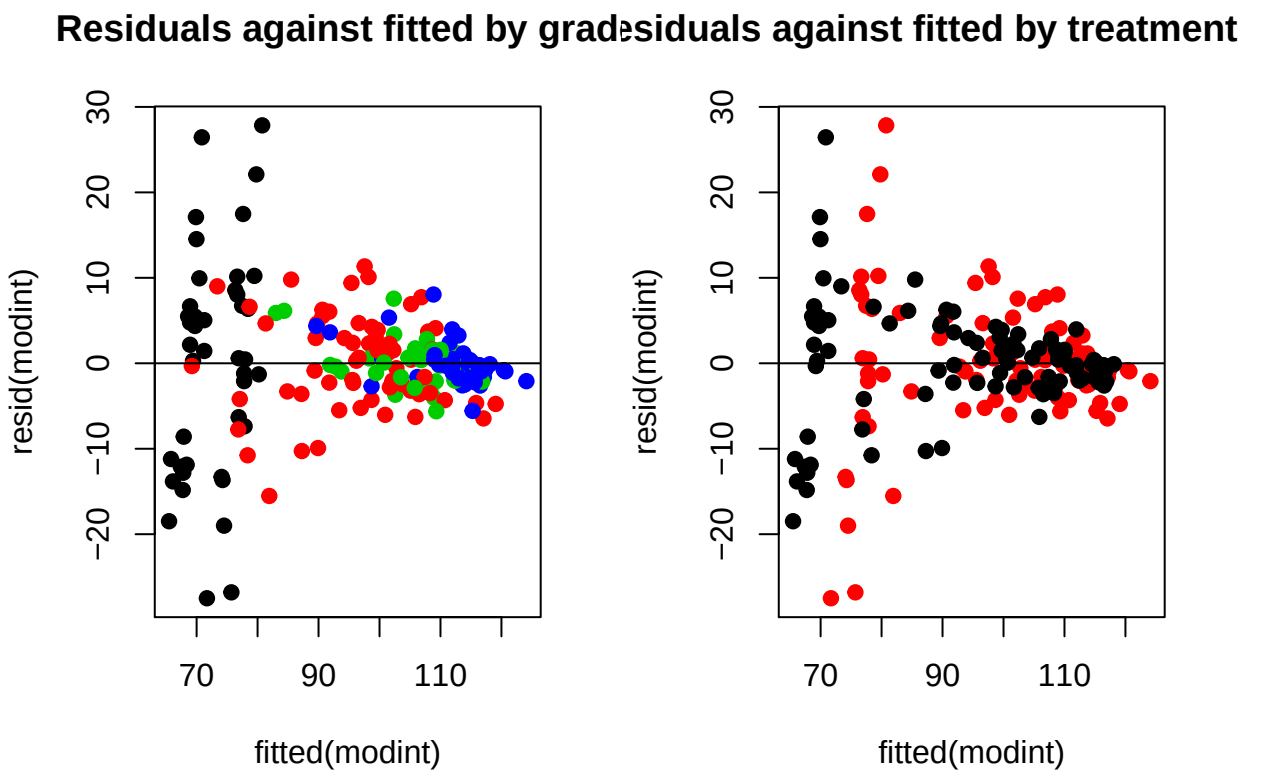
What do you conclude from these plots?

Hint: R's first 4 default colors are black, red, green, and blue.

Answer 4.2

```
par(mfrow = c(1,2))
plot(fitted(modint), resid(modint), col = electric$grade,
     main = "Residuals against fitted by grade", pch = 19)
abline(h = 0)

plot(fitted(modint), resid(modint), col = electric$treatment,
     main = "Residuals against fitted by treatment status", pch = 19)
abline(h = 0)
```




```
par(mfrow = c(1,1))
```

Grade 1 is much worse predicted, in both treatment and control conditions.

Question 4.3

Let's take a look at the subset of data we've isolated as badly predicted.

First make two variables, one that is TRUE for all classes in a particular grade and another that is TRUE for classes that were treated. Using only the observations in the troublesome grade 1:

1. Plot `post.scores` against `pre.scores` for the control classes
2. Overlay this with the pre and post scores for the treated classes, coloring these in red.
3. Overlay the fitted values from the model against the `pre.scores` of each type of class as a line, using the same colors for treatment and control as before.

Hint: Don't refit the model, just subset to get what you want from it.

Now switch the value of the grade indicator variable you made and remake these plots for grade 2. What do you see? What have we learnt about `pre.score`? What can we do to pre.score in the model to reflect what we have just learned?

Answer 4.3

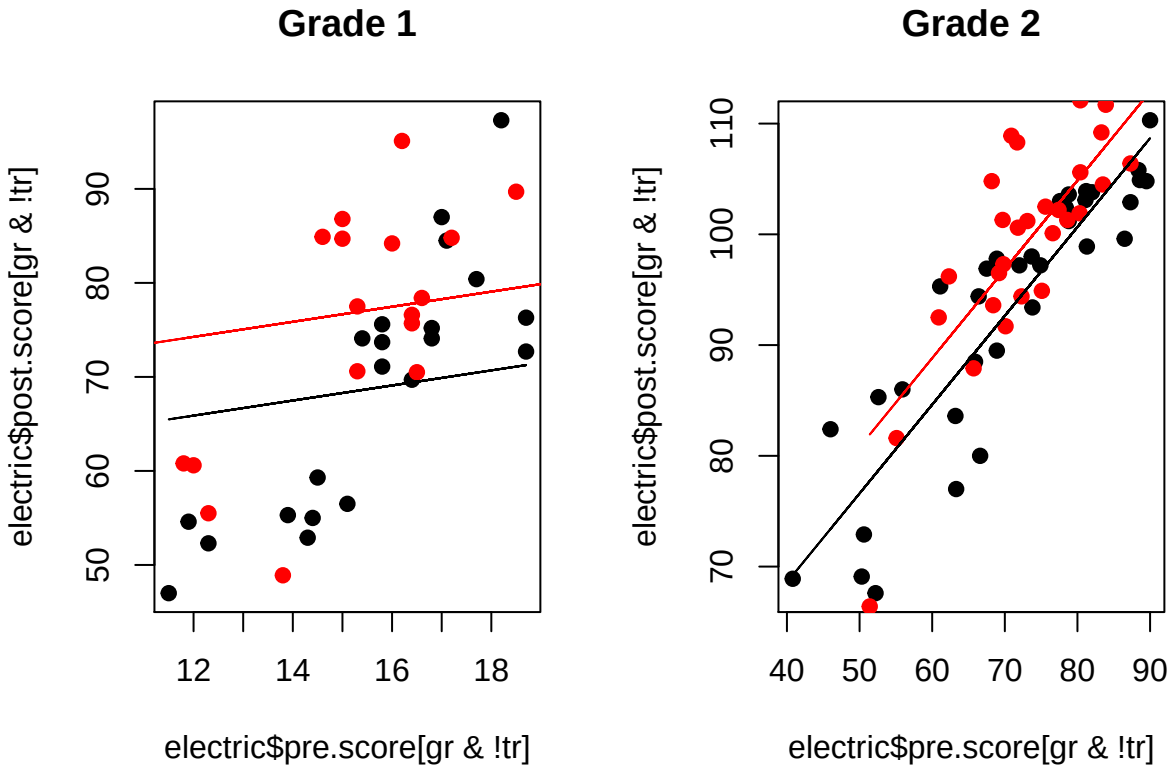
```
gr <- electric$grade == 1
tr <- electric$treatment == "T"

par(mfrow = c(1,2))
```

```
plot(electric$pre.score[gr & !tr], electric$post.score[gr & !tr],
     main = "Grade 1", pch = 19)
points(electric$pre.score[gr & tr], electric$post.score[gr & tr],
       col = "red", pch = 19)
lines(electric$pre.score[gr & !tr], fitted(modint)[gr & !tr])
lines(electric$pre.score[gr & tr], fitted(modint)[gr & tr],
      col = "red")

gr <- electric$grade == 2

plot(electric$pre.score[gr & !tr], electric$post.score[gr & !tr],
     main = "Grade 2", pch = 19)
points(electric$pre.score[gr & tr], electric$post.score[gr & tr],
       col = "red", pch = 19)
lines(electric$pre.score[gr & !tr], fitted(modint)[gr & !tr])
lines(electric$pre.score[gr & tr], fitted(modint)[gr & tr],
      col = "red")
```




```
par(mfrow = c(1,1))
```

The pre.score relationship is fit much less well in grade 1. It looks like the slope should be much steeper than it is currently estimated. From this we might conclude that there should not be just one pre.score term in the model. Which implies... an interaction of pre.score and grade.nom.

Question 5

Fit a model that interacts `treatment` with `grade.nom` *and* `pre.score` with `grade.nom`. Plot the residuals against the fitted values in this model to confirm it no longer has the problems we examined above.

Roughly speaking, what will `pre.score` be multiplied by in each grade? (You should now be able to read this from the `summary`). Is this a better model overall, statistically speaking? Has this model made us more confident about the effects of treatment?

Answer 5.1

```
modint2 <- lm(post.score ~ treatment + grade.nom + treatment:grade.nom +
                    pre.score + pre.score:grade.nom, data = electric)
summary(modint2)
```

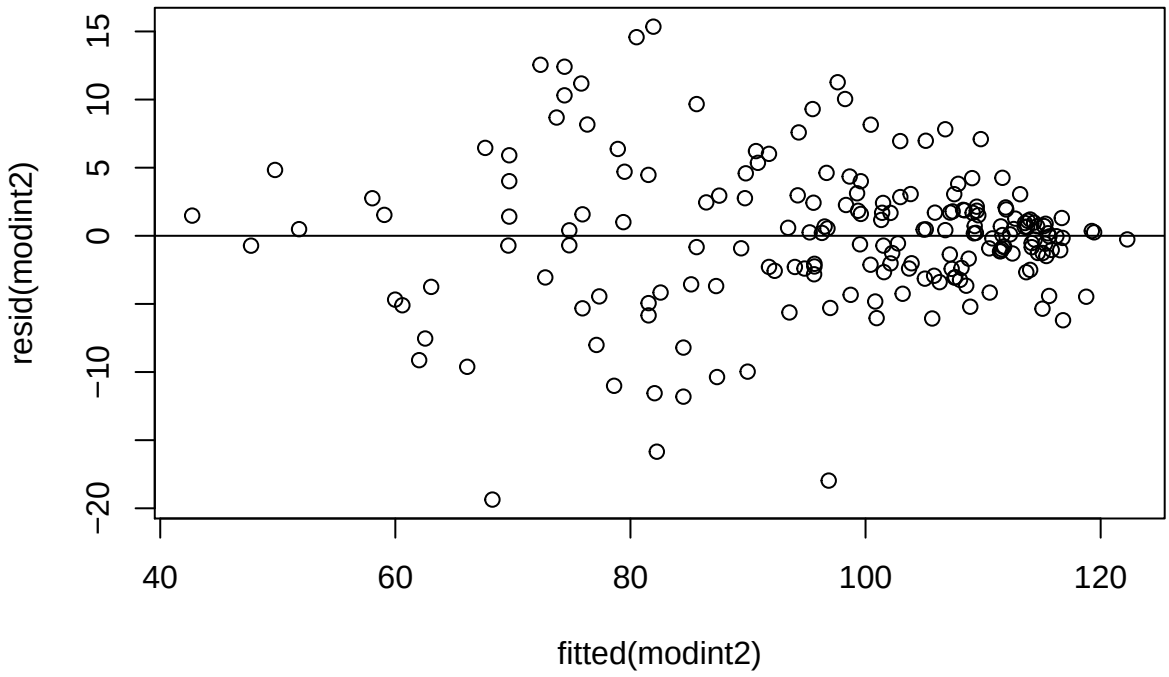
```
Call:
lm(formula = post.score ~ treatment + grade.nom + treatment:grade.nom +
    pre.score + pre.score:grade.nom, data = electric)

Residuals:
     Min       1Q   Median       3Q      Max
-19.3602  -2.5183   0.1998   2.1694  15.3492

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)           -11.0229     5.5538  -1.985   0.0487 *
treatmentT              8.7865     1.6509   5.322 3.02e-07 ***
grade.nom2             48.4516     6.7841   7.142 2.20e-11 ***
grade.nom3             51.6071     9.8845   5.221 4.87e-07 ***
grade.nom4             53.0176    11.8676   4.467 1.40e-05 ***
pre.score               5.1084     0.3475  14.699  < 2e-16 ***
treatmentT:grade.nom2  -4.5208     2.1179  -2.135   0.0342 *
treatmentT:grade.nom3  -6.8768     2.3715  -2.900   0.0042 **
treatmentT:grade.nom4  -7.0851     2.3545  -3.009   0.0030 **
grade.nom2:pre.score   -4.3195     0.3516 -12.284  < 2e-16 ***
grade.nom3:pre.score   -4.4238     0.3576 -12.369  < 2e-16 ***
grade.nom4:pre.score   -4.4526     0.3616 -12.313  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.349 on 180 degrees of freedom
Multiple R-squared:  0.9145,    Adjusted R-squared:  0.9093
F-statistic: 175.1 on 11 and 180 DF,  p-value: < 2.2e-16
```

```
plot(fitted(modint2), resid(modint2))
abline(h = 0)
```



pre.score will be multiplied by about 5 in grade 1 but slightly less than 1 in the others. It's a much better model for all the reasons. we can think of at this point in the course. And we're quite a bit more confident about the effects of treatment (smaller standard errors, etc.)

Question 5.2

(Hard) One difficulty with this otherwise well fitting model is that it's hard to get a clear view of treatment effects in each grade and their associated uncertainty (confidence intervals don't add!) The four quantities that we'd really like to get confidence intervals for are (using the labels the model summary has given us):

1. `treatment` (treatment effect in grade 1)
2. `treatment` + `treatment:grade.nom2` (treatment effect in grade 2)
3. `treatment` + `treatment:grade.nom2` (treatment effect in grade 3)
4. `treatment` + `treatment:grade.nom4` (treatment effect in grade 4)

But we have to work a bit harder to get intervals for those - either with more math, or better, with a bootstrap. How would you go about doing that?

Answer 5.2

If you're feeling very adventurous, try the bootstrap. QIs are each of these effects, and you'll want to resample residuals not rows to respect the experimental design.

Question 6

```
y_sys <- fitted(modint2)
y_ran <- resid(modint2)

b <- 5000
set.seed(1234)
```

```
treat1 <- treat2 <- treat3 <- treat4 <- rep(NA, b)
for (i in 1:b) {
  new_y <- y_sys + sample(y_ran)
  mod <- lm(new_y ~ treatment + grade.nom + treatment:grade.nom +
                    pre.score + pre.score:grade.nom, data = electric)
  cc <- coef(mod)
  treat1[i] <- cc["treatmentT"]
  treat2[i] <- treat1[i] + cc["treatmentT:grade.nom2"]
  treat3[i] <- treat1[i] + cc["treatmentT:grade.nom3"]
  treat4[i] <- treat1[i] + cc["treatmentT:grade.nom4"]
}

res <- data.frame(grade = 1:4,
                  mean = c(mean(treat1), mean(treat2),
                           mean(treat3), mean(treat4)),
                  se = c(sd(treat1), sd(treat2),
                         sd(treat3), sd(treat4)))
res$lower <- res$mean - 1.96 * res$se
res$upper <- res$mean + 1.96 * res$se

res
```

```
  grade     mean       se     lower     upper
1     1 8.795582 1.584770  5.689433 11.901731
2     2 4.283207 1.292390  1.750123  6.816291
3     3 1.898277 1.621349 -1.279566  5.076120
4     4 1.684798 1.625265 -1.500721  4.870317
```

References

Cooney, Joan G. 1976. “The Electric Company: Television and Reading,1971-1980: A Mid-Experiment Appraisal.” New York: Children’s Television Network.