Effects of Educational Television

In this exercise we're going to look at the effect on reading scores of a educational television program The Electric Company that ran from 1971-77.

It was a fairly expensive show. We'll ask what reading gains, if any, were made by the 1st through 4th grade classes that were randomized to watch it as part of their school program. This exercise is based on Cooney (1976).[1]

The data is from a randomized trial. Here we're looking at a two location trial that randomized at the level of school classes. (Actually it paired classes, but we'll ignore that for now). Each class was either treated (watch the program) or control (did not watch the program). The outcome of interest - our dependent variable - will be the score on a reading test at the end of the year called `post.score`. Our variables are therefore:

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 (randomized)
`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

Question 1

Read the data into an data frame named `electric`. What sort of variable has R assumed grade is? How will it be treated in a linear model? Under what circumstances would that be reasonable or unreasonable?

Make a new grade variable that is a factor. How will a linear model treat this new grade variable? Hint: You may find that `summary` illuminates the new data set.

Finally, overwrite the existing treatment variable so that it is numerical: 1 when the class is treated and 0 when not.

Answer 1

```
electric <- read_csv("electric-company.csv")
electric <- mutate(electric,
```

[1] Cooney, Joan G. 1976. "The Electric Company: Television and Reading,1971-1980: A Mid-Experiment Appraisal." New York: Children's Television Network.

```r
                    grade_nom = factor(grade),
                    treatment = ifelse(treatment == "T", 1, 0))
summary(electric)
```

```
      pair            city               grade            supp          
 Min.   : 1.00   Length:192         Min.   :1.000   Length:192        
 1st Qu.:24.75   Class :character   1st Qu.:2.000   Class :character  
 Median :48.50   Mode  :character   Median :2.000   Mode  :character  
 Mean   :48.50                      Mean   :2.427                     
 3rd Qu.:72.25                      3rd Qu.:3.000                     
 Max.   :96.00                      Max.   :4.000                     
   treatment     pre.score        post.score     grade_nom
 Min.   :0.0   Min.   :  8.80   Min.   : 44.20   1:42     
 1st Qu.:0.0   1st Qu.: 52.50   1st Qu.: 86.95   2:68     
 Median :0.5   Median : 80.75   Median :102.30   3:40     
 Mean   :0.5   Mean   : 72.22   Mean   : 97.15   4:42     
 3rd Qu.:1.0   3rd Qu.:100.62   3rd Qu.:111.00            
 Max.   :1.0   Max.   :119.80   Max.   :122.00            
```

Question 2

Let's now consider the effect of treatment. First, fit a linear model that predict `post.score` with just treatment. Now fit a model uses your factor version of `grade` as well as treatment.

Summarise both models in terms of how much of the variance in post.score they 'explain' and the average size of their errors.

Now, consider each model's treatment coefficient. Are the estimates of this coefficient *different* in the two models? Why do you think that is?

Answer 2

```r
mod <- lm(post.score ~ treatment, data = electric)
mod_grade <- lm(post.score ~ treatment + grade_nom, data = electric)

summary(mod)       # R_squared is 0.02
```

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

Residuals:
    Min      1Q  Median      3Q     Max 
-55.778  -9.935   4.872  13.397  23.679 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   94.321      1.794   52.58   <2e-16 ***
treatment      5.657      2.537    2.23   0.0269 *  
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 17.58 on 190 degrees of freedom
Multiple R-squared:  0.0255,	Adjusted R-squared:  0.02037 
F-statistic: 4.973 on 1 and 190 DF,  p-value: 0.02692
```

```r
summary(mod_grade) # R_squared is 0.65!
```

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

Residuals:
    Min      1Q  Median      3Q     Max 
-33.820  -5.282   1.774   6.547  32.831 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   70.112      1.813  38.682  < 2e-16 ***
treatment      5.657      1.536   3.684 0.000301 ***
grade_nom2    24.451      2.088  11.709  < 2e-16 ***
grade_nom3    33.402      2.351  14.209  < 2e-16 ***
grade_nom4    39.271      2.322  16.914  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10.64 on 187 degrees of freedom
Multiple R-squared:  0.6485,	Adjusted R-squared:  0.641 
F-statistic: 86.26 on 4 and 187 DF,  p-value: < 2.2e-16
```

Question 3

(Optional). In the previous question we saw that the models agreed about the coefficient estimate. This is a very rare thing in observational data, but it happens in experiments when experimenters have carefully arranged features of the experiment to be 'balanced' with respect to treatment. For example, the experimental design of this study is to have equal number of classes in treatment and in control within each grade. This makes the treatment indicator and grade indicators independent and therefore uncorrelated.

To investigate this further, first compute the correlation between these grade and treatment assignment, and then make a table of these two variables. How does the table structure explain the correlation?

Compare this to the correlation of post.score and treatment.

Compute the average `post.score` for each grade. How does this this explain the correlation between post.score and treatment?

Why would it be helpful to 'balance' variables like grade with respect to treatment in this way?

Answer 3

```r
cor(electric$grade, electric$treatment)
```

```
[1] 0
```

```r
table(electric$grade, electric$treatment)
```

```

     0  1
  1 21 21
  2 34 34
  3 20 20
  4 21 21
```

```r
with(electric, prop.table(table(grade, treatment), 1))
```

```
     treatment
grade   0   1
    1 0.5 0.5
    2 0.5 0.5
    3 0.5 0.5
    4 0.5 0.5
```

```r
cor(electric$post.score, electric$treatment)
```

```
[1] 0.1596987
```

```r
summarize(group_by(electric, grade),
          mean_post = mean(post.score))
```

```
# A tibble: 4 x 2
  grade mean_post
  <dbl>     <dbl>
1     1      72.9
2     2      97.4
3     3     106.
4     4     112.
```

Question 4

Now make another model that uses the factor version of `grade` and `pre.score` (the reading score before the year begins) to predict `post.score`. Is this model better? If so, in what ways?

Answer 4

```r
mod_grade_pre <- lm(post.score ~ treatment + pre.score + grade_nom, data = electric)
summary(mod_grade_pre)
```

```
Call:
lm(formula = post.score ~ treatment + pre.score + grade_nom,
```

```
    data = electric)

Residuals:
     Min       1Q   Median       3Q      Max 
-25.3464  -2.7310   0.0292   2.7851  30.0153 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  58.45590    1.48549  39.351  < 2e-16 ***
treatment     4.05175    1.06278   3.812 0.000187 ***
pre.score     0.79986    0.05535  14.450  < 2e-16 ***
grade_nom2  -21.72234    3.50360  -6.200 3.56e-09 ***
grade_nom3  -29.84558    4.66632  -6.396 1.26e-09 ***
grade_nom4  -32.86980    5.24186  -6.271 2.45e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.323 on 186 degrees of freedom
Multiple R-squared:  0.8344,    Adjusted R-squared:   0.83 
F-statistic: 187.5 on 5 and 186 DF,  p-value: < 2.2e-16
```

Question 5

Now let's consider the effect of treatment *within* each grade. 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 the factor version of grade, interacted with `treatment`. There are now four treatment effects (but how would you construct them from the coefficients?). How do they differ as grade increases? And are these ATEs? If so, for which population are they ATEs for? What do we call ATEs for specific values of pre-treatment variables?

Answer 5

```
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 ***
treatment              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 ***
treatment:grade_nom2  -4.17864    2.86683  -1.458  0.14667
treatment:grade_nom3  -6.19673    3.21265  -1.929  0.05530 .
treatment: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
```

The effects appear large for the first two grades and negligible afterwards. They are ATEs but for classes in separate grades. We call these CATEs because they are ATEs conditional on grade, a.k.a heterogenous treatment effects.

Constructing treatment effects from coefficients can be tricky. Let's take a different approach by creating some representative classes and plotting the difference treatment makes. First create 8 fictional classes: 4 treated and 4 untreated, each with an appropriate value of pre-score (for realism we can use the average `pre.score` in each grade, or for simplicity pick a single `pre.score`). Then, we get predictions from the most recent model for these classes, and plot them.

```
avs <- summarize(group_by(electric, grade_nom),
                 pre.score = mean(pre.score))
avs
```

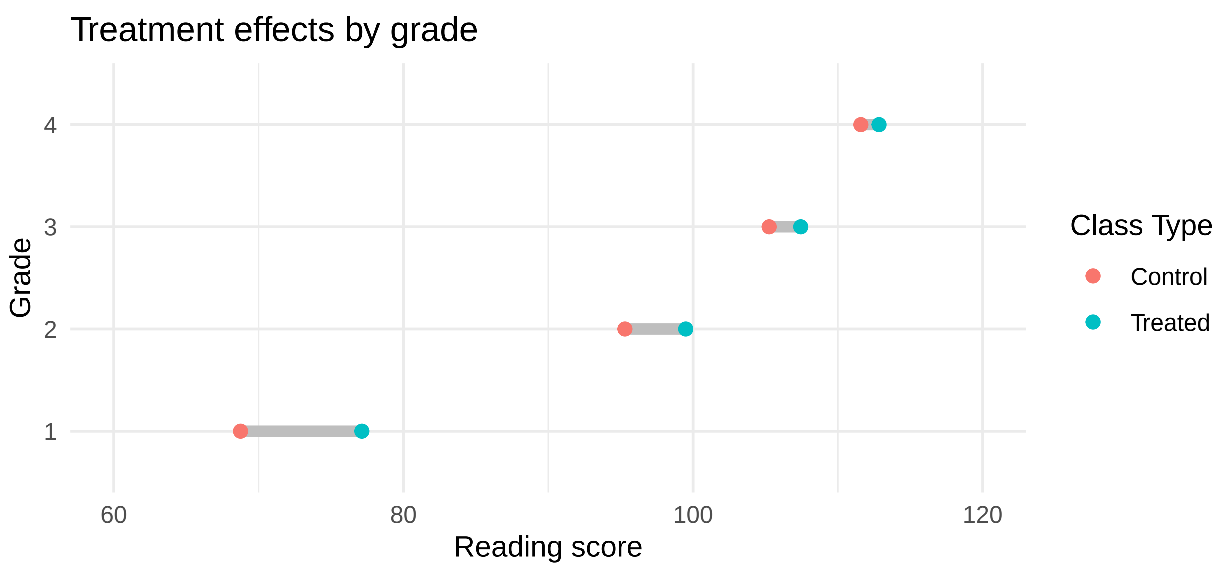
```
# A tibble: 4 x 2
  grade_nom pre.score
  <fct>         <dbl>
1 1              15.6
2 2              73.3
```

```
3 3              94.6
4 4             106.
```

```
rep_classes <- data.frame(treatment = rep(0:1, each = 4),
                          grade_nom = factor(c(avs$grade_nom, avs$grade_nom)),
                          pre.score = c(avs$pre.score, avs$pre.score))

preds <- mutate(rep_classes,
                pred = predict(modint, rep_classes),
                treatment = ifelse(treatment == 1, "Treated", "Control"))

# plot (note the coordinate flip)
# group aesthetic makes sure the lines join the right points (try
# mapping it to treatment for a different emphasis)
ggplot(preds, aes(x = grade_nom,
                  y = pred,
                  color = treatment,
                  group = grade_nom)) +
  geom_line(color = "grey", size = 2) +
  geom_point(size = 2) +
  ylim(60, 120) +
  coord_flip() +
  labs(color = "Class Type",
       y = "Reading score",
       x = "Grade",
       title = "Treatment effects by grade")
```



Alternatively we could plot ATE estimated against grades:

```
ates <- spread(preds, treatment, pred)
ates <- mutate(ates,
               ATE = Treated - Control)

ggplot(ates, aes(x = grade_nom, y = ATE)) +
```

```r
  geom_col(fill = "grey") +
  labs(x = "Grade",
       y = "Average Treatment Effect",
       title = "Treatment effects for each grade")
```