Precept 7

Summary of R Topics Covered in the Tutorial

We will mostly be using `lm` to run multiple regression. We will also introduce `confint` to create confidence intervals for all the coefficients of a model.

Can Television be Educational?

In this precept we're going to 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

Read the data into an object named `electric`. Fit a linear regression of reading score on grade. (We'll look at treatment effects later.)

What sort of variable has R assumed grade is? Under what circumstances would this be a reasonable modeling choice?

Answer 1

```r
electric <- read.csv("data/electric-company.csv")

mod <- lm(post.score ~ grade, data = electric)
summary(mod)
```

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

```
Residuals:
    Min      1Q  Median      3Q     Max
-35.360  -5.692   0.652   7.783  29.040

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  67.2343     2.1755   30.91   <2e-16 ***
grade        12.3256     0.8217   15.00   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.05 on 190 degrees of freedom
Multiple R-squared:  0.5422,    Adjusted R-squared:  0.5398
F-statistic:   225 on 1 and 190 DF,  p-value: < 2.2e-16
```

Question 2

Let's not make that assumption. Adjust the model so that grade is treated as a nominal variable. Hint: it will be helpful to make a new grade variable as a factor - maybe call it `grade.nom`.

Now refit and interpret the regression. What do each of the coefficients mean?

Answer 2

```r
electric$grade.nom <- as.factor(electric$grade)

mod <- lm(post.score ~ grade.nom, data = electric)
summary(mod)
```

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

Residuals:
    Min      1Q  Median      3Q     Max
-30.991  -3.530   2.198   5.729  35.660

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   72.940      1.696   43.01   <2e-16 ***
grade.nom2    24.451      2.157   11.34   <2e-16 ***
grade.nom3    33.402      2.428   13.76   <2e-16 ***
grade.nom4    39.271      2.398   16.38   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10.99 on 188 degrees of freedom
Multiple R-squared:  0.623, Adjusted R-squared:  0.617
F-statistic: 103.6 on 3 and 188 DF,  p-value: < 2.2e-16
```

Question 3

Now let's consider the effect of treatment. First, fit a regression of post.score on just the treatment variable. Now fit a model that contains the treatment variable and your nominal version of grade.

Summarise both models and compare them: Let's start with the coefficient on treatment. Are the estimates for this coefficient *different* in the two models? Are we more or less *certain* about the value of the coefficient in second model (with grade) compared to the first? Why do you think that is?

Answer 3

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

summary(mod)
```

```
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 ***
treatmentT     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)
```

```
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 ***
treatmentT     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 4

In question above the models agreed about the coefficient estimate. Weird, no? This is quite a rare thing in general, but it happens in experiments when, unlike in observation data, two variables are perfectly *independent* of each other. 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. Here's grade 1, for instance:

```r
cor(electric$grade == 1, electric$treatment == "T")
```

```
[1] 0
```

Let's turn now to our uncertainty about the true effect of treatment. One measure of this is a confidence interval. How would you compute a 95% confidence interval for the effect of treatment from each summary table?

Answer 4

coef +/- 1.96 * SE for each, basically.

Question 5

In practice we don't usually construct intervals by hand, but rather use the `confint` function, which takes a fitted model and returns a data frame of confidence intervals for each of the coefficients. (Try it)

Turning to testing, can we reject the hypothesis that the treatment effect is 0 in both models? What do the p-values for this test mean?

Although both models agree about the (im)plausibility of the null hypothesis, why do you think the p-values and intervals are numerically different?

Answer 5

```r
summary(mod)
```

```

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 ***
treatmentT     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
confint(mod)
```

```
                 2.5 %   97.5 %
(Intercept) 90.7822555 97.85941
treatmentT   0.6529869 10.66160
```

```r
summary(mod_grade)
```

```

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 ***
treatmentT     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
```

```r
confint(mod_grade)
```

```
               2.5 %    97.5 %
(Intercept) 66.53620 73.687465
treatmentT   2.62759  8.686993
grade.nom2  20.33127 28.570135
grade.nom3  28.76464 38.039404
grade.nom4  34.69095 43.851906
```

We can reject the null of no treatment effect. The p value is the chance of seeing a t-statistic at least as extreme as the one we see if there to be in reality no treatment effect.

Different models mean different assumptions about the generating process which mean different sampling distributions which mean different intervals and p values. In short, p values and intervals are conditional on the model.

Question 6

Now let's consider the effect of treatment within in each grade. We can use the `lm` function's `subset` argument to fit the model on just a subset of all the rows in the data set. For example, we can fit a model of the relationship of `post.score` to `treatment` just in grade 2 like this:

```
mod <- lm(post.score ~ treatment, data = electric, subset = (grade == 2))
```

This is equivalent to

```
electric_grade2 <- electric[electric$grade == 2, ]
mod <- lm(post.score ~ treatment, data = electric_grade2)
```

but a bit shorter to type.

Fit a regression model for the effect of `treatment` on `post.score` for each grade. You can use either of the strategies above. There are now *four* treatment effects. How do they differ as grade increases? Are these ATEs? If so, which population are they ATEs for? What do we call ATEs for specific values of pre-treatment variables?

Answer 6

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

summary(mod1)
```

```

Call:
lm(formula = post.score ~ treatment, data = electric, subset = grade ==
    1)

Residuals:
    Min      1Q  Median      3Q     Max
-32.890 -13.190   2.060   7.685  31.510

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   68.790      3.268  21.047   <2e-16 ***
treatmentT     8.300      4.622   1.796   0.0801 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 14.98 on 40 degrees of freedom
Multiple R-squared:  0.0746,    Adjusted R-squared:  0.05146
F-statistic: 3.224 on 1 and 40 DF,  p-value: 0.0801
```

```r
summary(mod2)
```

```

Call:
lm(formula = post.score ~ treatment, data = electric, subset = grade ==
    2)
```

```
Residuals:
    Min      1Q  Median      3Q     Max 
-35.171  -6.796   2.509   9.299  17.088 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   93.212      1.907   48.88  < 2e-16 ***
treatmentT     8.359      2.697    3.10  0.00285 ** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 11.12 on 66 degrees of freedom
Multiple R-squared:  0.1271,	Adjusted R-squared:  0.1138 
F-statistic: 9.607 on 1 and 66 DF,  p-value: 0.002848
```

```r
summary(mod3)
```

```
Call:
lm(formula = post.score ~ treatment, data = electric, subset = grade == 
    3)

Residuals:
    Min      1Q  Median      3Q     Max 
-17.610  -3.525   2.740   4.900   9.125 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  106.175      1.663  63.858   <2e-16 ***
treatmentT     0.335      2.351   0.142    0.887    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.436 on 38 degrees of freedom
Multiple R-squared:  0.0005339,	Adjusted R-squared:  -0.02577 
F-statistic: 0.0203 on 1 and 38 DF,  p-value: 0.8875
```

```r
summary(mod4)
```

```
Call:
lm(formula = post.score ~ treatment, data = electric, subset = grade == 
    4)

Residuals:
    Min      1Q  Median      3Q     Max 
-16.357  -1.489   1.093   3.918   7.933 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  110.357      1.299   84.98   <2e-16 ***
treatmentT     3.710      1.837    2.02   0.0501 .  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 5.951 on 40 degrees of freedom
Multiple R-squared:  0.09255,	Adjusted R-squared:  0.06987 
F-statistic:  4.08 on 1 and 40 DF,  p-value: 0.05014
```

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.

Question 7

How confident would you be that each of them is real, i.e. non-zero, on the basis of these models? Are we less confident about each effect? If so, why do you think that is? How many data points are used in each model?

Answer 7

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

```
 1  2  3  4 
42 68 40 42 
```

Quite a bit less certain because only 1/4 of the data is being used in each model.

Question 8

In precept 3 we found that `pre.score` - the scores at the beginning of the year - were very predictive of `post.scores`. Add `pre.score` to each of the models you fitted in Question 6. Do we become more or less sure about the value of the treatment after adding `pre.score`? Why do you think that is?

What are the advantages and disadvantages of these multiple models over fitting just one model?

Answer 8

```r
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)

summary(mod1)
```

```
Call:
lm(formula = post.score ~ treatment + pre.score, data = electric, 
    subset = grade == 1)

Residuals:
    Min      1Q  Median      3Q     Max 
-19.360  -5.059   0.445   5.640  15.349 
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -11.0229     8.7860  -1.255  0.21709
treatmentT    8.7865     2.6118   3.364  0.00173 **
pre.score     5.1084     0.5498   9.292 1.96e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 8.461 on 39 degrees of freedom
Multiple R-squared:  0.712, Adjusted R-squared:  0.6973
F-statistic: 48.22 on 2 and 39 DF,  p-value: 2.863e-11
```

```r
summary(mod2)
```

```
Call:
lm(formula = post.score ~ treatment + pre.score, data = electric,
    subset = grade == 2)

Residuals:
     Min       1Q   Median       3Q      Max
-15.8446  -3.4414   0.3449   3.8631  11.2716

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 37.42877    3.99098   9.378 1.07e-13 ***
treatmentT   4.26577    1.35896   3.139  0.00255 **
pre.score    0.78891    0.05486  14.382  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.479 on 65 degrees of freedom
Multiple R-squared:  0.7913,    Adjusted R-squared:  0.7848
F-statistic: 123.2 on 2 and 65 DF,  p-value: < 2.2e-16
```

```r
summary(mod3)
```

```
Call:
lm(formula = post.score ~ treatment + pre.score, data = electric,
    subset = grade == 3)

Residuals:
    Min      1Q  Median      3Q     Max
-5.2063 -1.7614  0.3153  1.7005  6.9502

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 40.58424    3.72776   10.89  4.3e-13 ***
treatmentT   1.90973    0.77616    2.46   0.0187 *
pre.score    0.68466    0.03849   17.79  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 2.438 on 37 degrees of freedom
Multiple R-squared:  0.8953,    Adjusted R-squared:  0.8897
F-statistic: 158.3 on 2 and 37 DF,  p-value: < 2.2e-16
```

```r
summary(mod4)
```

```
Call:
lm(formula = post.score ~ treatment + pre.score, data = electric,
    subset = grade == 4)

Residuals:
    Min      1Q  Median      3Q     Max
-5.3504 -1.0094  0.0801  0.7166  7.0962

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 41.99473    4.28162   9.808 4.42e-12 ***
treatmentT   1.70144    0.68535   2.483   0.0175 *
pre.score    0.65583    0.04082  16.066  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.184 on 39 degrees of freedom
Multiple R-squared:  0.8809,    Adjusted R-squared:  0.8748
F-statistic: 144.2 on 2 and 39 DF,  p-value: < 2.2e-16
```

More certainty because pre.score is a good predictor. More models allow all explanatory variables - specifically pre.score - to vary within grade which is should lead to a better fit. However, we are always less certain about the models because each uses less data. (A bias-variance tradeoff). Also we do a lot of hypothesis testing with lots of models, so we risk fooling ourselves that way.

Question 9

(Hard!) In the previous precept using this data we constructed a new variable, `score.diff` which was defined as `post.score` - `pre.score`. (This is called a change score, and is widely used in educational applications).

Back then we were trying to take into account the role of prior ability as reflected in `pre.score` by asking not simply whether treatment raised `post.scores`, but rather whether treatment increased reading performance over what we would expect on the basis of `pre.scores`. But in the question above we just added `pre.score` as a predictor in regression models of `post.score`. That was a whole lot easier.

Let's compare these approaches: Remake the `score.diff` variable and regress it against treatment and the nominal version of grade. In a separate model regress `post.score` against treatment and nominal grade. Do you see different results?

The model with `score.diff` is a *special case* of the model that uses post.score and has `pre.score` as an explanatory variable. What would the coefficient on `pre.score` be in the second model to get the same results as the first model. Hint: write down the equation of each model first and move `pre.score` to the right hand side.

Which model would you prefer, and why?

Answer 9

```r
electric$score.diff <- electric$post.score - electric$pre.score

mod_diff <- lm(score.diff ~ treatment + grade.nom, data = electric)
mod_pre <- lm(post.score ~ treatment + grade.nom + pre.score, data = electric)

summary(mod_diff)
```

```
## 
## Call:
## lm(formula = score.diff ~ treatment + grade.nom, data = electric)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -24.0893  -3.5292  -0.2286   2.8018  29.3107 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)   55.539      1.287  43.152  < 2e-16 ***
## treatmentT     3.650      1.091   3.347 0.000988 ***
## grade.nom2   -33.276      1.483 -22.441  < 2e-16 ***
## grade.nom3   -45.672      1.669 -27.361  < 2e-16 ***
## grade.nom4   -50.921      1.649 -30.885  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 7.556 on 187 degrees of freedom
## Multiple R-squared:  0.8612,    Adjusted R-squared:  0.8582 
## F-statistic: 290.1 on 4 and 187 DF,  p-value: < 2.2e-16
```

```r
summary(mod_pre)
```

```
## 
## Call:
## lm(formula = post.score ~ treatment + grade.nom + pre.score, 
##     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 ***
## treatmentT    4.05175    1.06278   3.812 0.000187 ***
## 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 ***
## pre.score     0.79986    0.05535  14.450  < 2e-16 ***
## ---
## 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
```

The change score model is equivalent to forcing a pre.score coefficient on the right hand side of the model to exactly 1. (Add pre.score to both sides) Since it probably is not exactly 1, the change score models is a more constrained model than one that lets it be estimated to any number. Here it makes little difference because the estimate happens to be quite close to 1.

References

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