Problem set 1: Part I

Alin Ierima

Effects of Educational Television

In this exercise we're going to look at the effect of a educational television program The Electric Company that ran from 1971-77 on children's reading scores. We will investigate what reading gains, if any, were made by the 1st through 4th grade classes as part of a randomized experiment.

This exercise is based on:

> Joan G. Cooney (1976) The Electric Company: Television and Reading, 1971-1980: A Mid-Experiment Appraisal. Children's Television Network Report.

The data comes from a two location trial in which treatment was randomized at the level of school classes.[1] Each class was either treated (to watch the program) or control (to not watch the program). The outcome of interest is the score on a reading test administered at the end of each year called `post.score`. Note that these are distinct classes from all four years.

The variables are:

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 (15 points)

Read the data into a data frame named `electric`.

- What sort of variable (e.g. integer, factor, character etc.) has R assumed `grade` is? **Hint:** You may find that `summary` illuminates the new data set. How will the `grade` variable be treated in a linear model if we use it as an independent variable? Discuss under what circumstances would that be reasonable and when would that be unreasonable? *(8 points)*
- Make a new variable from `grade` that is a factor. How will a linear model treat this new variable? Elaborate this in the text. *(5 points)*
- Finally, overwrite the existing treatment variable so that it is numerical: 1 when the class is treated and 0 when not. *(2 points)*

[1] Classes were paired, but we will ignore that in the analysis

Answer 1

```
# Load Data
electric <- read.csv("data/electric-company.csv")
```

grade variable

```
class(electric$grade)
```

```
## [1] "integer"
```

```
electric$grade_factor = as.factor(electric$grade)
electric$treatment =ifelse(electric$treatment == "T", 1, 0)
```

Summary of all variables in the dataset after the transformations asked in question 1

```
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_factor
##   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
```

Response: As far as I understand, it is good that R assumed ”grade” is an integer, because this might be what we are looking for. Our expectation might be that there is a direct relationship between the grade and the reading score. In this case, it makes more sense to have grade as an integer because the linear model will show a continuous variable in a lm.

However, this would be unreasonable if there would be a non-linear relationship, because the linear model would not help us understand the effects that the grade the pupil is from. Additionally, it might normally make more sense to have grade as a categorical variable if we are interested in exploring how belonging to different grades impacts reading scores. This is exactly what as.factor() prepares.

Question 2 (15 points)

Let’s now consider the treatment effect

- First, fit a linear model that predicts `post.score` with just the treatment variable. *(2 points)*
- Then fit a model, which uses your factor version of `grade`, as well as the treatment variable. *(3 points)*

- Summarise both models in terms of how much of the variance in post.score they "explain" and the median size of their errors (the median of the residuals). (**Hint**, you can find information on the R^2 and the median residual if you use the summary() function on your model, e.g. summary(model1), where model1 is the object to which you assigned the output of your linear model.) *(5 points)*
- Now, consider each model's treatment coefficient. *(5 points)*
  - Are the estimates of this coefficient *different* in the two models?
  - Why do you think that is?

Answer 2

Linear model output

```r
summary(lm(post.score ~ treatment, data = electric))
```

```
##
## 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
```

Linear model output

```r
summary(lm(post.score ~ treatment + grade_factor, data = electric))
```

```
##
## Call:
## lm(formula = post.score ~ treatment + grade_factor, 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_factor2   24.451      2.088  11.709  < 2e-16 ***
## grade_factor3   33.402      2.351  14.209  < 2e-16 ***
## grade_factor4   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
```

Response: *The estimates of the two linear models are identical for the treatment (5.657), indicating that a one-unit increase in treatment (in this case, jumping from control to treated – 0-1) leads to an increase on the post.score of 5.657 units.*

Even though this is identical for both lm's, the second one is an improvement of the model, as the adjusted R^2 shows that it can explain 64% of the variance, as compared to only about 2% for the first model. This is further proved by the median of the residuals, which show that the errors are not as dispersed.

Question 3 (10 points)

- 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`. *(5 points)*
- Is this model better? If so, in what ways? Provide your answer in the text. (Hint: you can consider the model fit, as well as the size of the residuals.) *(5 points)*

Answer 3

Linear model output

```r
summary(lm(post.score ~ treatment + grade_factor + pre.score, data = electric))
```

```
##
## Call:
## lm(formula = post.score ~ treatment + grade_factor + 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 ***
## treatment       4.05175    1.06278   3.812 0.000187 ***
## grade_factor2 -21.72234    3.50360  -6.200 3.56e-09 ***
## grade_factor3 -29.84558    4.66632  -6.396 1.26e-09 ***
## grade_factor4 -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
```

Response: *In this model, the advantage is that 83% of the variance can be explained by including grade and pre.score, and the residuals get ever so smaller. However, it might be difficult to interpret some of the results. For example, the coefficient for grade_factor2 shows that a one-unit increase leads to the scores being worse by -21, which seems counterintuitive, as a bigger grade should mean more knowledge. Or it could perhaps be explained by the increasing test difficulty as one gets in bigger grades.*

Question 4 (40 points)

Now let's consider the effect of treatment *within* 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` and `pre.score` just in grade 2 like this:

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

Fit a linear model predicting `post.score` using treatment and `pre.score` for each grade. Create a function to do so. For this purpose you need to follow the following procedures:

1. Define a function named `fit_reg` that returns the coefficient on treatment. The function should have two arguments: the entire data (`data_all`) and the grade (`grade_subset`). *(15 points)*
   - Within the function run a linear model using the arguments you created
   - Hint: do not forget to use `return()` within the function to return the coefficients from the fitted linear model.
2. After you are done with the function use a `for loop` to call the `fit_reg()` function for each grade (`1` to `4`).Inside the for loop: *(15 points)*
   - Store what the `fit_reg()` function returns in a variable.
   - Print out the coefficient on treatment using the `print()` function.
3. Briefly comment on the result. There are now *four* treatment effects. How do they differ as grade increases? *(10 points)*

Answer 4

```
fit_reg = function(data_all, grade_subset) {
  data_subset = data_all[data_all$grade == grade_subset,]
  model = lm(post.score ~ treatment + pre.score, data = data_subset)
  return(coef(model)["treatment"])
}
```

Print out from the loop

```
for (i in 1:4) {
  effect = fit_reg(data_all = electric, grade_subset = i )
  print(paste("Treatment for grade", i))
  print(effect)[i]
}
```

```
## [1] "Treatment for grade 1"
## treatment
##  8.786518
## [1] "Treatment for grade 2"
## treatment
##  4.265765
## [1] "Treatment for grade 3"
## treatment
##  1.909726
## [1] "Treatment for grade 4"
## treatment
##  1.701436
```

Response: *It appears that as grade increases, the treament effect decreases, perhaps indicating that it is better to start with the alternative kind of education as early as possible for the children to reap most benefits of the treatment. It could be that the earlier grades are the moments the children soak the most information.*

Question 5 (20 points)

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

$$Y_i = \beta_0 + \beta_1 A_i + \beta_2 B_i + \beta_3 (A_i \times B_i) + \epsilon_i$$

An alternative syntax to fit this model is `A*B`. So to fit the model above using this notation the formula is

```
Y ~ A * B
```

Since we always want to have `A` and `B` if we have an `A:B` term, the `*` notation makes sure we don't forget any of them. But they are equivalent.

- Fit a model of all the grades that includes `pre.score`, `treatment`, `grade` (factor version), the factor version of grade interacted with `treatment`, and the factor version of grade interacted with `pre.score` (this is called a fully interacted model). *(5 points)*
- How would you construct grade-specific treatment effects from these coefficients? In your answer show an example for grade 2. There are two ways to do so.
  - The easiest is to use the `predict()` function. *(5 points)*
  - Another way is to directly interpret the linear model coefficients and think about which coefficient or combination of coefficients will give you the treatment effect for grade 2. Refer to the correct coefficient or combination of coefficients in the text. *(5 points)*
  - Use both approaches in your answer and **elaborate these**. *(5 points for the explanation of each approach)*

Answer 5

Linear model output

```
(interacted_model = lm(post.score ~  treatment * grade_factor + grade_factor * pre.score, data = electri
```

```
##
## Call:
## lm(formula = post.score ~ treatment * grade_factor + grade_factor *
##     pre.score, data = electric)
##
## Coefficients:
##            (Intercept)               treatment            grade_factor2
##                -11.023                   8.787                   48.452
##          grade_factor3           grade_factor4                pre.score
##                 51.607                  53.018                    5.108
## treatment:grade_factor2  treatment:grade_factor3  treatment:grade_factor4
##                 -4.521                  -6.877                   -7.085
## grade_factor2:pre.score  grade_factor3:pre.score  grade_factor4:pre.score
##                 -4.320                  -4.424                   -4.453
```

Sample average treatment effect (SATE) for grade 2

```
coef(interacted_model)["treatment"] + coef(interacted_model)["treatment:grade_factor2"]
```

```
## treatment
##  4.265765
```

```
# OR

diff(predict(interacted_model,
        data.frame(treatment = c(0,1),
            grade_factor = factor(2),
            pre.score = mean(electric$pre.score)
                                  )))
```

```
##        2
## 4.265765
```

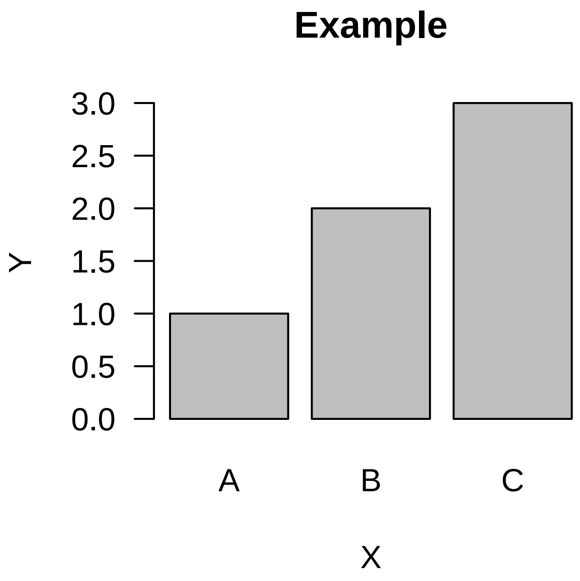
Response: *In the first approach, I took the coefficient "treatment" out of the model, which is the average treatment effect overall, then I added the effect of the 2nd grade, resulting in 4.26. In the second approach, I have used predict() to predict the scores of 2nd grade pupils for the control (0) and treatment(1) groups, and then the different of these two gave the treatment effect.*

Question 6 (bonus question 10 points)

Use a bar plot to visualize the grade-specific treatment effects that you calculated in the previous question. Briefly interpret the result. (10 points)

Hint: You can make a bar plot using a `barplot()` function (textbook p.81)

```
# Example
res <- data.frame(Y = c(1, 2, 3),
                  X = c("A", "B", "C"))
barplot(height = res$Y, names = res$X,
        xlab = "X", ylab = "Y", las = 1,
        main = "Example")
```



Answer 6

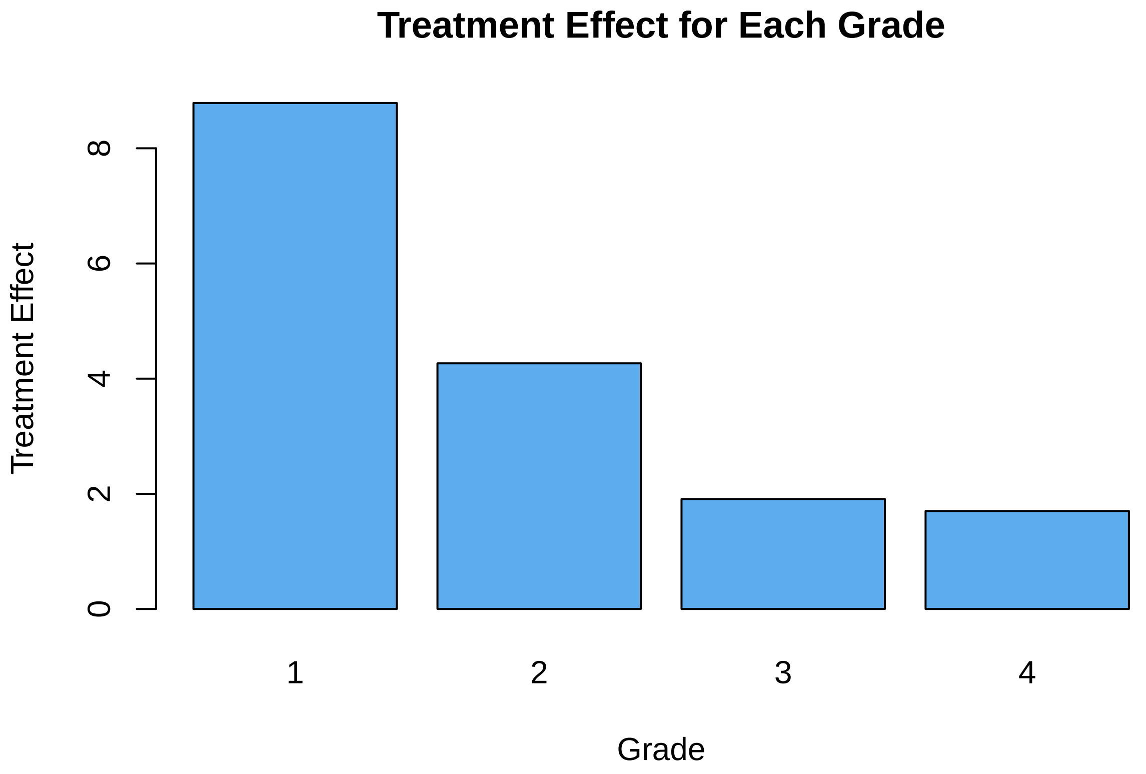
Sample average treatment effect (SATE) by grade

```r
SATE = data.frame(Effect = NA, Grade = unique(electric$grade))

for (i in SATE$Grade) {
  treatment_effect = coef(interacted_model)["treatment"] +
    coef(interacted_model)[paste0("treatment:grade_factor",i)]
  SATE[SATE$Grade == i, "Effect"] <- treatment_effect
}
SATE$Effect[SATE$Grade == 1] = coef(interacted_model)["treatment"]

SATE
```

```
##     Effect Grade
## 1 8.786518     1
## 2 4.265765     2
## 3 1.909726     3
## 4 1.701436     4
```

Barplot with the sample average treatment effect for each grade

```r
barplot(height = SATE$Effect,
        names.arg = SATE$Grade,
        xlab = "Grade",
        ylab = "Treatment Effect",
        main = "Treatment Effect for Each Grade",
        col = "steelblue2")
```



Response: *In this barplot we can see how the treatment effect varies across the grade, with the highest grade being the one where the effect is most noticeable. However, treatment effect goes down the latter grade the treatment is administered.*

Evaluation

- 5 questions for a total of 100 points
- 1 bonus question for a total of 10 points