Gelman & Hill Chapter 9 exercises

```
library(ggplot2)
library(cowplot)
```

```
##
## Attaching package: 'cowplot'
```

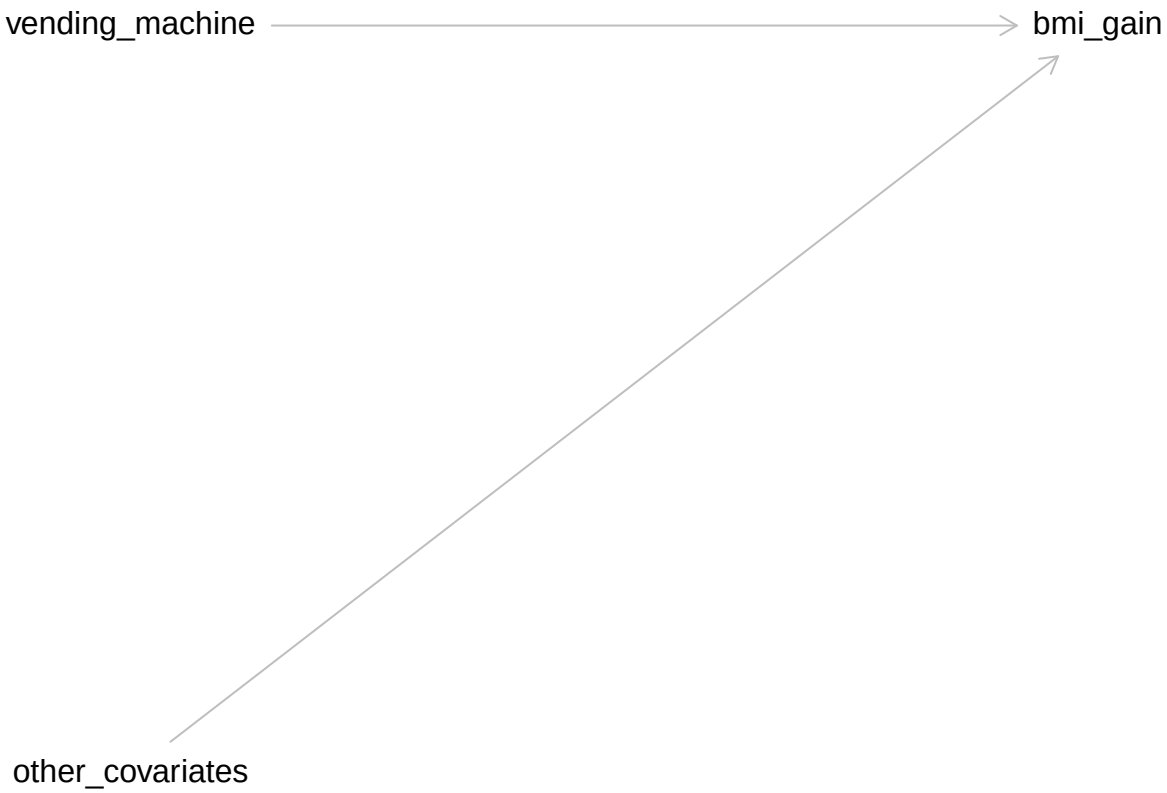
```
## The following object is masked from 'package:ggplot2':
##
##     ggsave
```

```
library(dagitty)
library(data.table)
```

1. Vending machines

Suppose you are interested in the effect of the presence of vending machines in schools on childhood obesity. What randomized experiment would you want to do (in a perfect world) to evaluate this question?

```
g1 <- dagitty( 'dag {
  other_covariates [pos="0,1"]
  vending_machine [exposure, pos="0,0"]
  bmi_gain [outcome, pos="1,0"]

  vending_machine -> bmi_gain
  other_covariates -> bmi_gain

}')

plot(g1)
```



A: Select 100 schools. At random select 50 of them, put a vending machine there. At the start, measure the weight/BMI of all the children. Maybe also measure a few other variables of each child to increase precision. After that, every year, measure the weight/BMI of all the children.

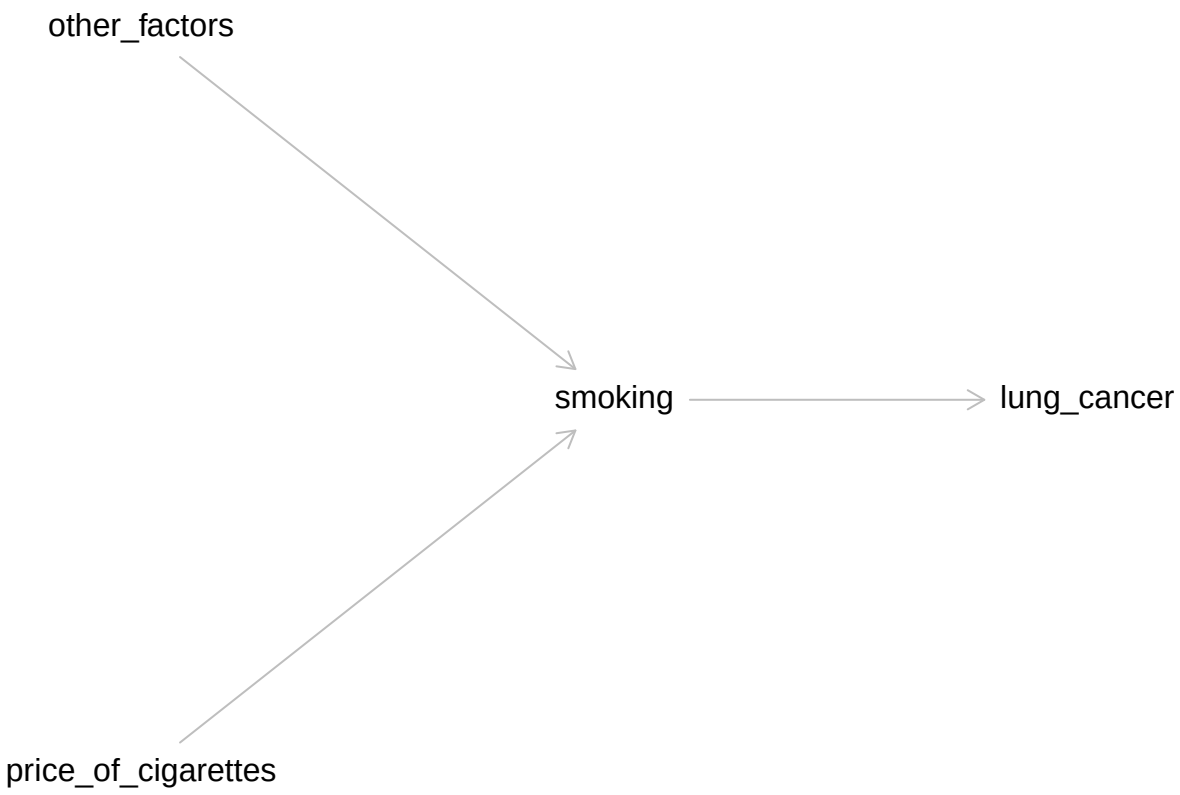
2. Smoking and lung cancer

Suppose you are interested in the effect of smoking on lung cancer. What experiment could you plausibly perform (in the real world) to evaluate this effect?

A: We cannot force people to smoke or not. We can influence their decision to smoke and how much by changing the price of cigarettes. Or make smoking less attractive by disallowing smoking in public places. If we would do this locally, then we could randomize this per geographic region. We would still not know WHICH people would react to the policy changes. If people with a higher susceptibility for lung cancer would not react to the policy changes, we would not get the proper causal effect. With proper i mean for the original population of smokers.

```
g1 <- dagitty( 'dag {
  price_of_cigarettes [exposure, pos = "0,2"]
  lung_cancer [outcome, pos = "2,1"]
  other_factors[pos = "0,0"]
  smoking[pos = "1,1"]

  smoking -> lung_cancer
  other_factors -> smoking
  price_of_cigarettes -> smoking
```

```
} ')

# stochastic layout
plot(g1)
```



3. Teacher quality and test scores

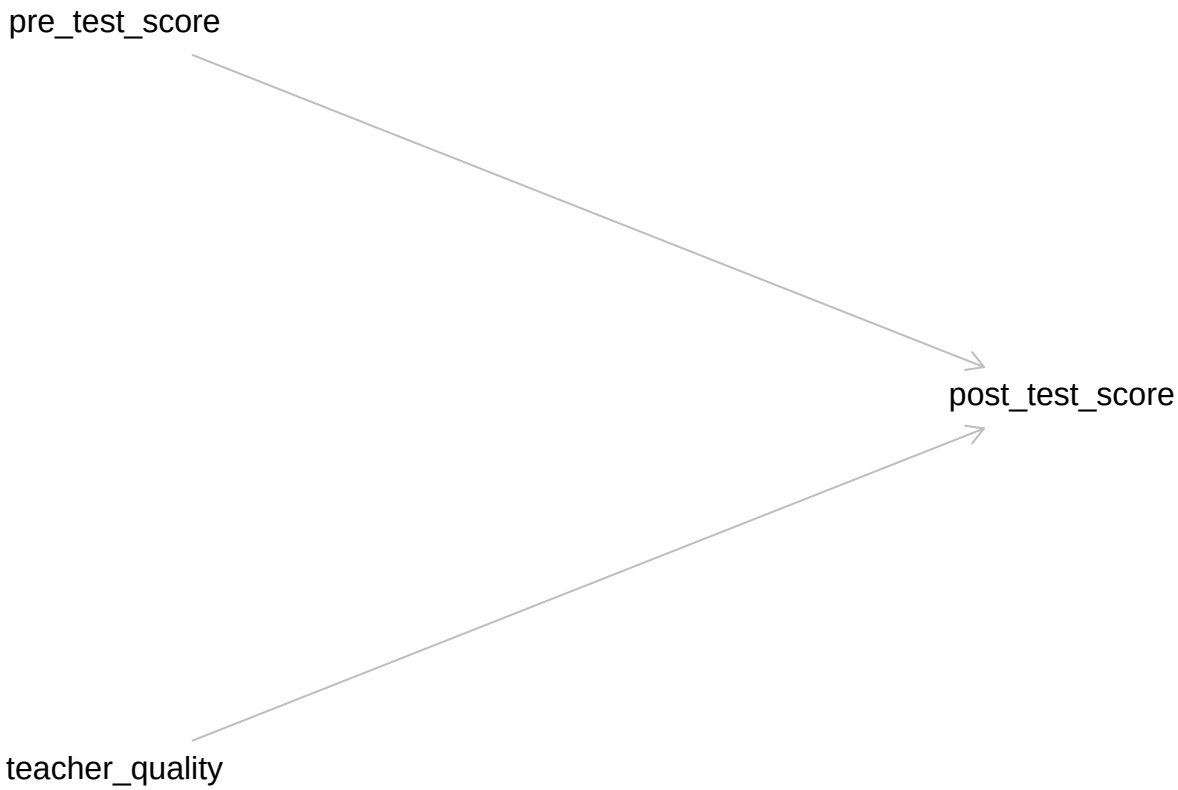
Suppose you are a consultant for a researcher who is interested in investigating the effects of teacher quality on student test scores. Use the strategy of mapping this question to a randomized experiment to help define the question more clearly. Write a memo to the researcher asking for needed clarifications to this study proposal.

A: We would like to randomize teacher quality for example in three levels. low quality, medium quality and high quality. We would select 60 classes, and with probability 1/3 for each class, assign the teacher quality level to that class. Then we would measure average pre-treatment test scores and post-treatment test scores (Say after a year of being exposed to the teacher). Then we would measure the increase in test-scores, for the three groups of classes and compare differences.

So now we can ask the researcher: Is there a treatment that influences teacher quality that we can randomize? For example a training? Or can we measure teacher quality and then randomize their placement over classes/schools?

```
g1 <- dagitty( 'dag {
  pre_test_score [pos = "0,0"]
```

```
  teacher_quality [exposure, pos = "0,2"]
  post_test_score [outcome, pos = "1,1"]
  pre_test_score -> post_test_score
  teacher_quality -> post_test_score

}')

# stochastic layout
plot(g1)
```

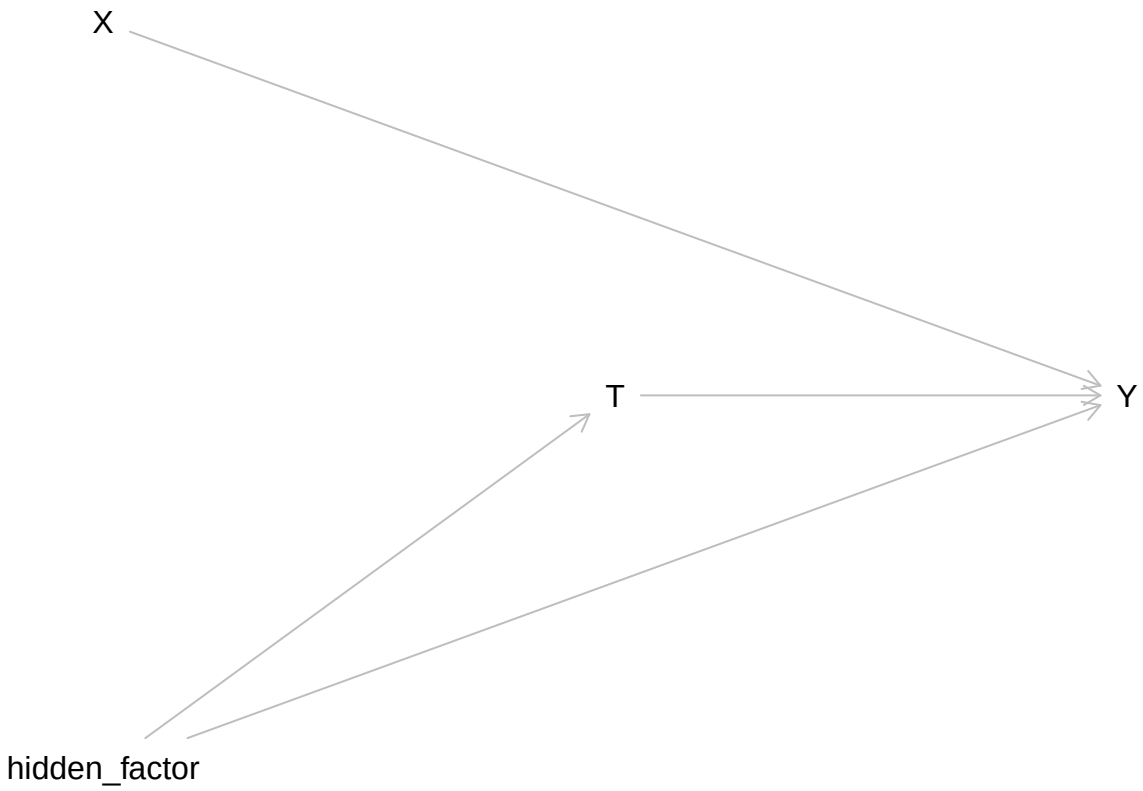


4. Potential outcomes

The table below describes a hypothetical experiment on 2400 persons. Each row of the table specifies a category of person, as defined by his or her pre-treatment predictor x, treatment indicator T , and potential outcomes $y0, y1$. (For simplicity, we assume unrealistically that all the people in this experiment fit nto these eight categories.)

```
g1 <- dagitty( 'dag {
  T [exposure, pos = "1,1"]
  Y [outcome, pos="2,1"]
  X[pos="0,0"]
  hidden_factor[pos = "0,2"]
  X -> Y
```

```
  T -> Y
  hidden_factor -> Y
  hidden_factor -> T

}')

# stochastic layout
plot(g1)
```



Category	# persons in category	x	T	y0	y1
1	300	0	0	4	6
2	300	1	0	4	6
3	500	0	1	4	6
4	500	1	1	4	6
5	200	0	0	10	12
6	200	1	0	10	12
7	200	0	1	10	12
8	200	1	1	10	12

In making the table we are assuming omniscience, so that we know both $y0$ and $y1$ for all observations. But the (nonomniscient) investigator would only observe x, T , and y^T for each unit. (For example, a person in category 1 would have $x = 0$, $T = 0$, $y = 4$, and a person in category 3 would have x=0, T =1, y=6.)

(a) What is the average treatment effect in this population of 2400 persons?

2, since in all categories the differences $y1 - y0$ equals 2.

(b) Is it plausible to believe that these data came from a randomized experiment? Defend your answer.

No, because people with $x = 1$ and potential outcomes $y0 = 4, y1 = 6$, we observe a ratio of treated/non-treated of 500 over 300, whereas for people with $x = 1$ and potential outcomes $y0 = 10, y1 = 12$ we observe a ratio of treated/non-treated of 200 over 200. Whereas in a randomized experiment, we expect both ratio's to be similar.

Thus, we conclude that there is a hidden factor that "causes" T and y.

(c) Another population quantity is the mean of y for those who received the treatment minus the mean of y for those who did not. What is the relation between this quantity and the average treatment effect?

Here this is (1000 x 6 + 400 x 12)/1400, minus (600 x 4 + 400 x 10)/1000.

```
mean_treated <- (1000 * 6 + 400 * 12)/1400
mean_not_treated <- (600 *  4 + 400 * 10)/1000

mean_treated - mean_not_treated
```

```
## [1] 1.314286
```

(d) For these data, is it plausible to believe that treatment assignment is ignorable given sex? Defend your answer.

It is not plausible, since we know that there exists a hidden factor that influences treatment assignment. Since we do not know how sex is distributed across categories, it would be a strong additional assumption that it is equally distributed for each category.

5. Estimate and standard error of treatment effect

For the hypothetical study in the previous exercise, figure out the estimate and the standard error of the coefficient of T in a regression of y on T and x.

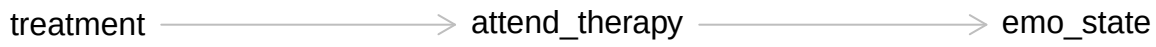
```
# actually make this dataset and then run a regression y ~ x + T

# or use formulas from http://faculty.cas.usf.edu/mbrannick/regression/Reg2IV.html
```

6. Intermediate outcomes

You are consulting for a researcher who has performed a randomized trial where the treatment was a series of 26 weekly therapy sessions, the control was no ther- apy, and the outcome was self-report of emotional state one year later. However, most people in the treatment group did not attend every therapy session. In fact there was a good deal of variation in the number of therapy sessions actually attended. The researcher is concerned that her results represent watered down estimates because of this variation and suggests adding in another predictor to the model: number of therapy sessions attended. What would you advise her?

A: This can be considered a post-treatment variable, that lies on the path between treatment and outcome. It can also be thought of as a mediating variable or an intermediate outcome. If we control for attend_therapy then we undo the randomization of treatment.

```
g1 <- dagitty( 'dag {
  treatment [exposure, pos = "0,0"]
  attend_therapy[pos="1,0"]
  emo_state [outcome, pos = "2,0"]

  treatment -> attend_therapy -> emo_state
}')

# stochastic layout
plot(g1)
```



7. Gain-score models

Gain-score models: in the discussion of gain-score models in Section 9.3, we noted that if we include the pre-treatment measure of the outcome in a gain score model, the coefficient on the treatment indicator will be the same as if we had just run a standard regression of the outcome on the treatment indicator and the pre-treatment measure. Show why this is true.

A: Grab all the equations. Plug 'm in. Reorder to recover earlier equation.

$g_i = y_i - x_i$

$g_i = \alpha + \theta T_i + error_i$

$y_i = \alpha + \theta T_i + \beta x_i + error_i$

$g_i = \alpha + \theta T_i + \beta x_i + error_i$

$y_i - x_i = \alpha + \theta T_i + \beta x_i + error_i$

$y_i = \alpha + \theta T_i + (\beta + 1)x_i + error_i$

$y_i = \alpha + \theta T_i + \beta' x_i + error_i$

This is identical to the original model.

8. Fake data pt 1

Assume that linear regression is appropriate for the regression of an outcome, y, on treatment indicator, T, and a single confounding covariate, x.

```
g1 <- dagitty( 'dag {
  x [pos="0,1"]
  T [exposure, pos="0,0"]
  y [outcome, pos="1,0"]

  T -> y
  x -> T
x -> y

}')

plot(g1)
```

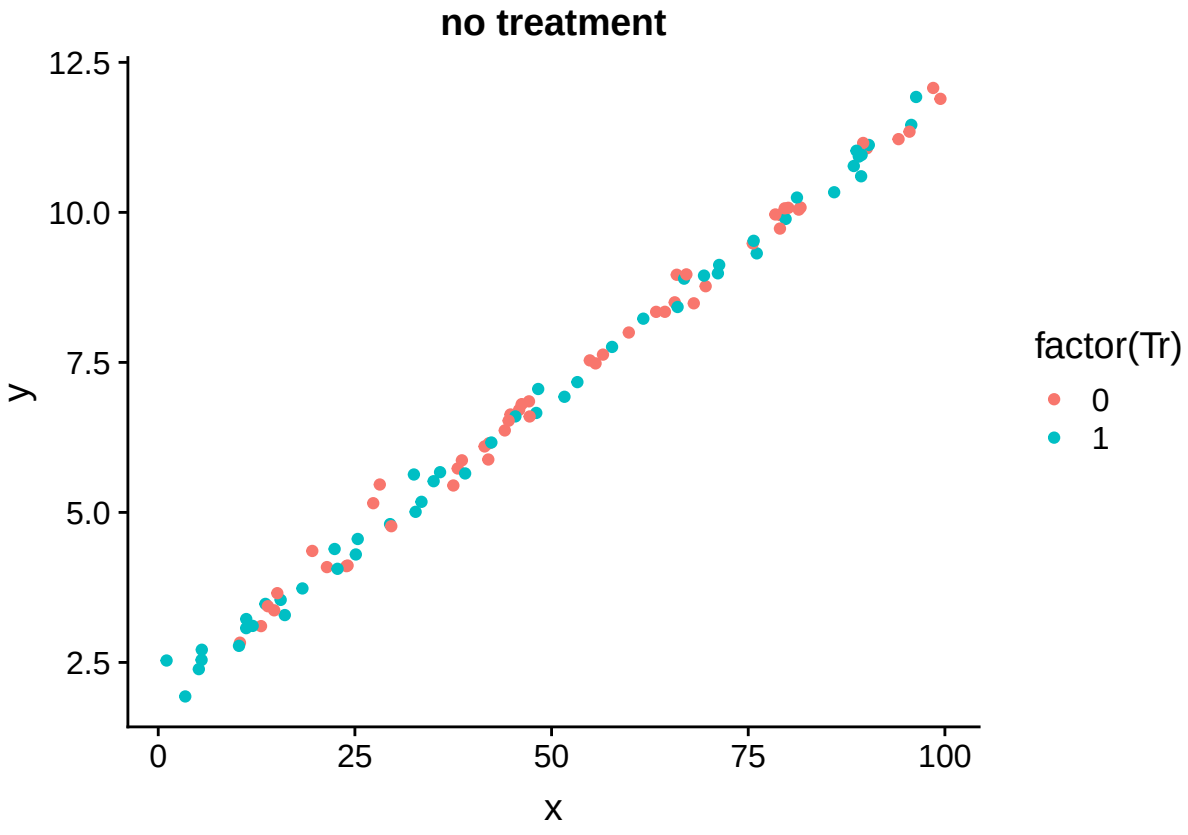
Sketch hypothetical data (plotting y versus x, with treated and control units indicated by circles and dots, respectively) and regression lines (for treatment and control group) that represent each of the following situations:

$y_i = \alpha + \theta T_i + \beta x_i + \gamma * T_i x_i + error_i$

```
# function to simulate datasets according to linear model
generate_data <- function(alpha, theta, beta, gamma, eps_sd, N){
  x <- runif(N, min = 1, max = 100)
  Tr <- rbinom(N, 1, prob = 0.5)
  y <- rnorm(N, mean = alpha + beta * x + Tr * theta + Tr * gamma * x, sd = eps_sd)
  return(data.frame(y, Tr, x))
}
```

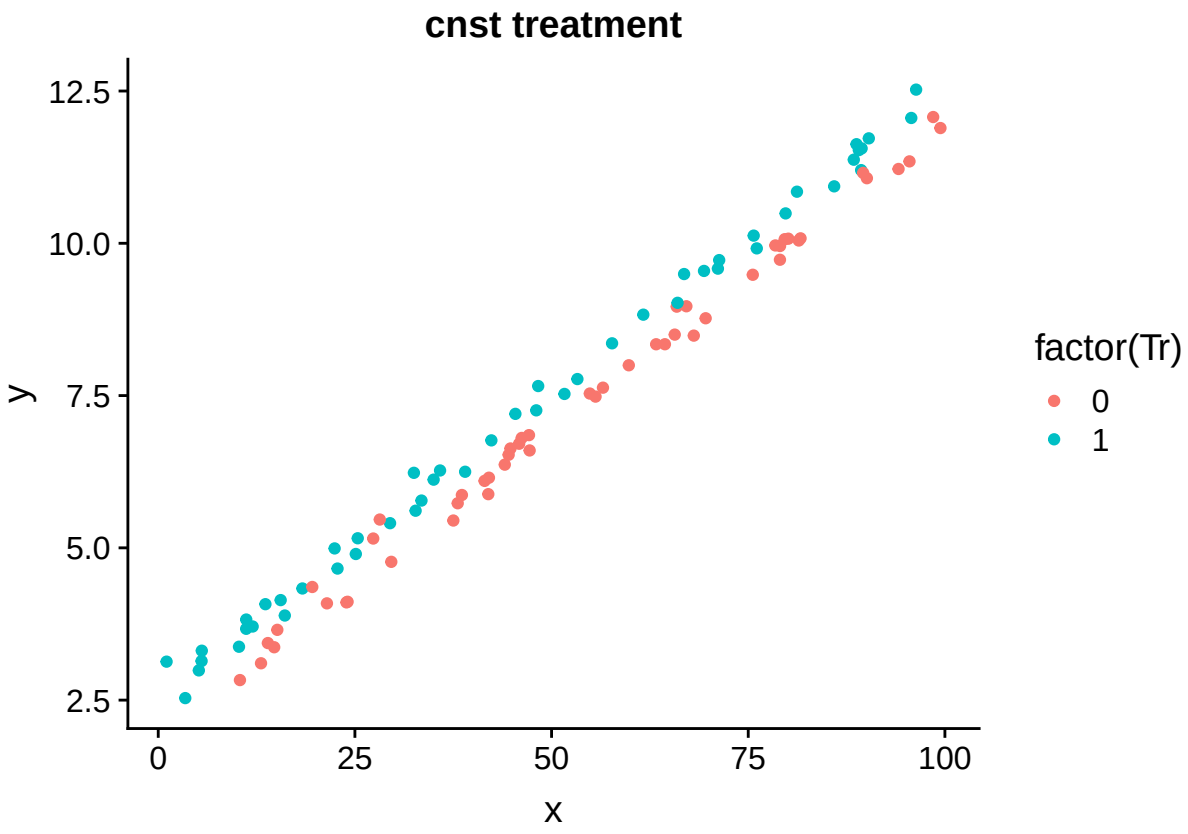
*(a) No treatment effect,

```
set.seed(123)
df1 <- generate_data(alpha = 2, theta = 0,
                     beta = 0.1, gamma = 0, eps_sd = 0.2, N = 100)

gp1 <- ggplot(df1, aes(x = x, y = y, group = Tr, col = factor(Tr))) + geom_point()
gp1 + ggtitle("no treatment")
```



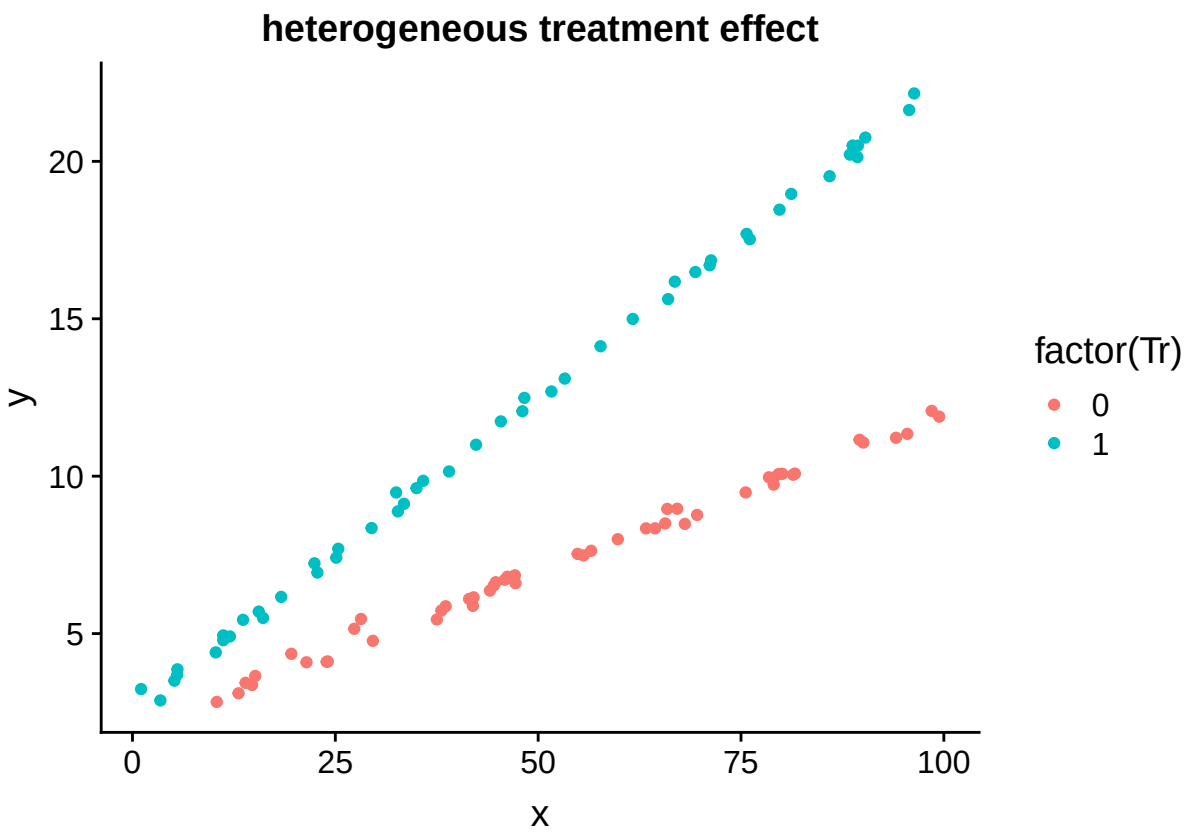
*(b) Constant treatment effect,

```r
set.seed(123)
df2 <- generate_data(alpha = 2, theta = 0.6,
                     beta = 0.1, gamma = 0, eps_sd = 0.2, N = 100)

gp2 <- ggplot(df2, aes(x = x, y = y, group = Tr, col = factor(Tr))) + geom_point()
gp2 + ggtitle("cnst treatment")
```



*(c) Treatment effect increasing with x (heterogeneous treatment effect)

```r
set.seed(123)
df3 <- generate_data(alpha = 2, theta = 0.6,
                     beta = 0.1, gamma = 0.1, eps_sd = 0.2, N = 100)


gp3 <- ggplot(df3, aes(x = x, y = y, group = Tr, col = factor(Tr))) + geom_point()
gp3 + ggtitle("heterogeneous treatment effect")
```



9. Fake data pt2

Consider a study with an outcome, y, a treatment indicator, T, and a single confounding covariate, x.

Draw a scatterplot of treatment and control observations that demonstrates each of the following:

$y_i = \alpha + \theta T_i + \beta x_i + \gamma T_i x_i + error_i$

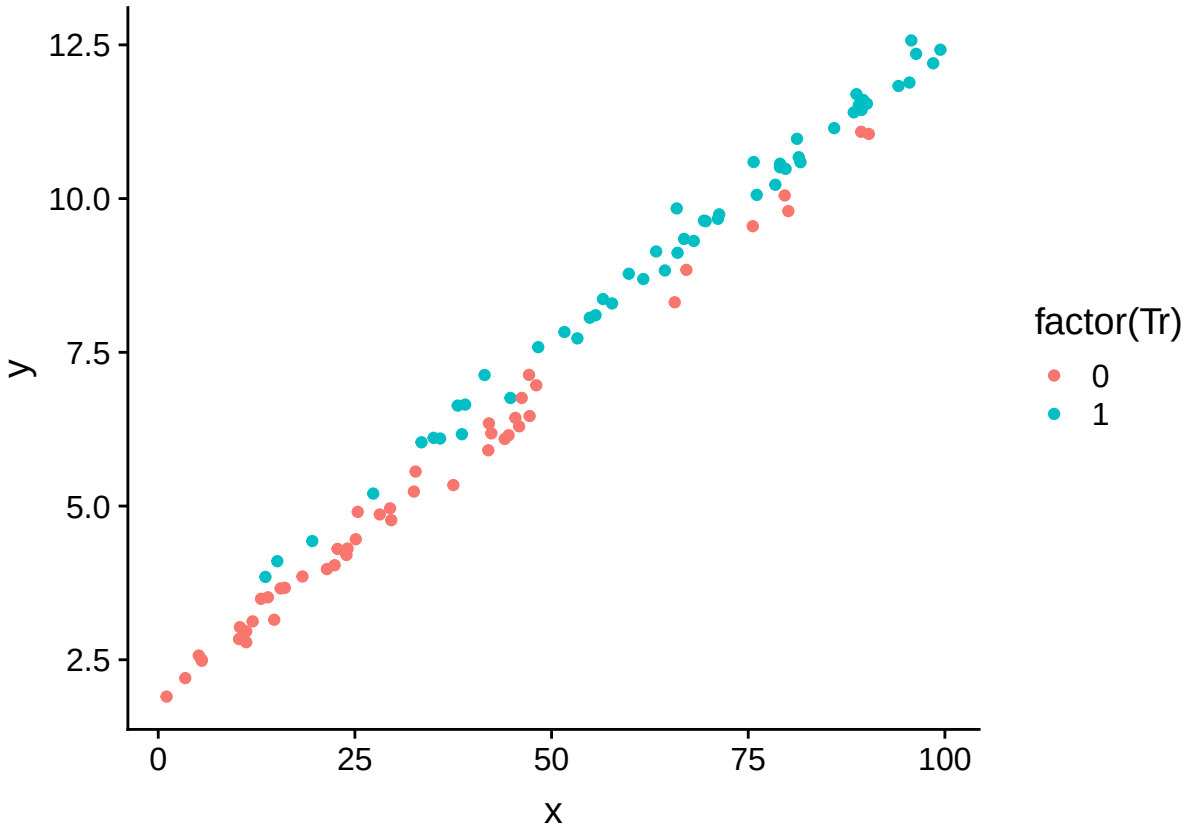
```
# function to generate Fake-data
generate_data <- function(alpha, theta, beta, gamma, kappa, eps_sd, N, prob_vec){
  x <- runif(N, min = 1, max = 100)
  x <- sort(x)
  Tr <- rbinom(N, 1, prob = prob_vec)
  y <- rnorm(N, mean = alpha + beta * x + Tr * theta + Tr * gamma * x + kappa * x^2, sd = eps_sd)
  return(data.frame(y, Tr, x))
}
```

(a) A scenario where the difference in means estimate would not capture the true treatment effect but a regression of y on x and T would yield the correct estimate.

A: If we make P(Tr) a linear function of x then we get the desired effect.

```
set.seed(123)
df <- generate_data(alpha = 2, theta = 0.6,
```

```
                     beta = 0.1, gamma = 0, kappa = 0,
                    eps_sd = 0.2, N = 100,
                    prob_vec = seq(from = 0, to = 1, length.out = 100)
                    )

ggplot(df, aes(x = x, y = y, group = Tr, col = factor(Tr))) + geom_point()
```



Calculate causal effect by a) group means and b) using regression

True treatment effect is 0.6

```
# A
mean(subset(df, Tr == 1)$y) - mean(subset(df, Tr == 0)$y)
```

```
## [1] 3.876104
```

```
lm(y ~ Tr, data = df)
```

```
##
## Call:
## lm(formula = y ~ Tr, data = df)
##
## Coefficients:
## (Intercept)           Tr
##       5.278        3.876
```

```r
# B
lm(y ~ Tr + x, data = df)
```

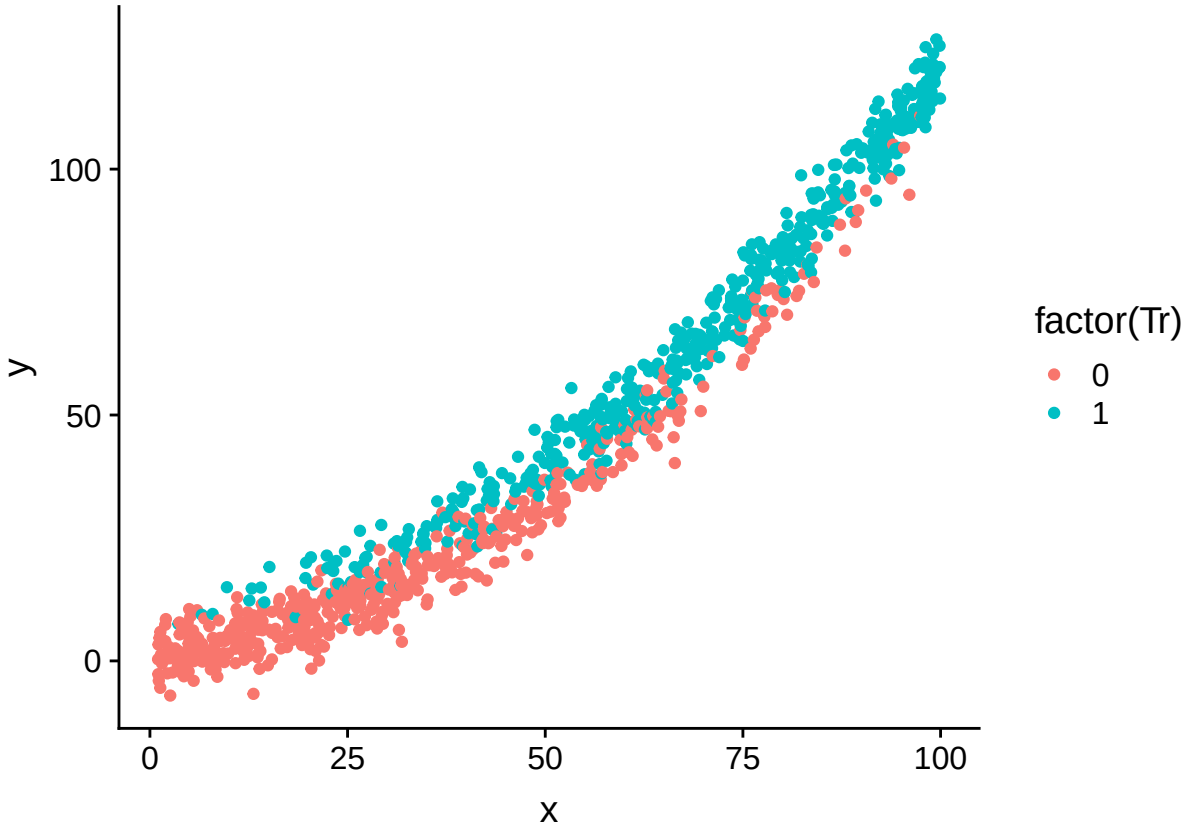
```
##
## Call:
## lm(formula = y ~ Tr + x, data = df)
##
## Coefficients:
## (Intercept)           Tr            x
##      1.9335       0.6443       0.1004
```

(b) A scenario where a linear regression would yield the wrong estimate but a nonlinear regression would yield the correct estimate.

A: add quadratic term to parametric equation for y

$y_i = \alpha + \theta T_i + \beta x_i + \kappa x_i^2 + error_i$

```r
set.seed(1234)
df <- generate_data(alpha = 2, theta = 8,
                     beta = 0.1, gamma = 0, kappa = 0.01,
                    eps_sd = 4, N = 1000,
                    prob_vec = seq(from = 0, to = 1, length.out = 1000)
                    )

ggplot(df, aes(x = x, y = y, group = Tr, col = factor(Tr))) + geom_point()
```



```r
# A linear
summary(lm(y ~ Tr + x, data = df))
```

```
##
## Call:
## lm(formula = y ~ Tr + x, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -19.355  -6.683  -1.402   6.640  22.293
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -15.9832     0.5465  -29.25   <2e-16 ***
## Tr            6.8490     0.6803   10.07   <2e-16 ***
## x             1.1382     0.0118   96.49   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.468 on 997 degrees of freedom
## Multiple R-squared:  0.9449, Adjusted R-squared:  0.9448
## F-statistic:  8547 on 2 and 997 DF,  p-value: < 2.2e-16
```

```r
# B nonlinear
summary(lm(y ~ Tr + x + I(x^2), data = df))
```

```
##
## Call:
## lm(formula = y ~ Tr + x + I(x^2), data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.6896  -2.5594  -0.0207   2.6382  12.5535
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.9587846  0.3907123   5.013 6.33e-07 ***
## Tr          7.8312872  0.3158903  24.791  < 2e-16 ***
## x           0.1024284  0.0180181   5.685 1.72e-08 ***
## I(x^2)      0.0100189  0.0001661  60.335  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.927 on 996 degrees of freedom
## Multiple R-squared:  0.9882, Adjusted R-squared:  0.9881
## F-statistic: 2.771e+04 on 3 and 996 DF,  p-value: < 2.2e-16
```

Hmm, the nonlinear model is more accurate (lower std error), but the linear model is not really bad. True treatment effect is 8.

10. Sesame Street experiment

The folder sesame contains data from an experiment in which a randomly selected group of children was encouraged to watch the television program Sesame Street and the randomly selected control group was not.

(a) The goal of the experiment was to estimate the effect on child cognitive development of watching more Sesame Street. In the experiment, encouragement but not actual watching was randomized. Briefly explain why you think this was done. (Hint: think of practical as well as statistical reasons.)

A. Children are either encouraged to watch Sesame street or not. Possible the effect of forcing a child to watch Sesame street has adverse effects on learning / cognitive development. Also, a practical reason is that if actual watching sesame street is randomized, children that want to watch it were not allowed to watched it, which might be unethical / or would cause parents to not cooperate with the experiment, or they would allow it anyway and bias the results.

(b) Suppose that the investigators instead had decided to test the effectiveness of the program simply by examining how test scores changed from before the intervention to after. What assumption would be required for this to be an appropriate causal inference? Use data on just the control group from this study to examine how realistic this assumption would have been.

A. The assumption is that all changes in test scores are then due to the intervention.

We find that in the control group, there is a substantial change in scores pre and post for numbers and letters. So this is not a realistic assumption.

```
library(foreign)
sesame <- read.dta("../../datasets/ARM_Data\\sesame\\sesame.dta")
sesame <- data.table(sesame)
# recode viewenc into treatment 1/0
sesame <- sesame[, viewenc := 1 -(viewenc - 1) ]
# DOH! already have this variable (encour)
# calculate
res <- sesame[, .(avg_prelet = mean(prelet),
                    avg_prenumb = mean(prenumb),
                    avg_postlet = mean(postlet),
                  avg_postnumb = mean(postnumb),
                  N = .N), .(viewenc)]
# control group
res[viewenc == 0]
```

```
##    viewenc avg_prelet avg_prenumb avg_postlet avg_postnumb  N
## 1:       0   17.04545    20.63636    24.92045     28.60227 88
```

```
res
```

```
##    viewenc avg_prelet avg_prenumb avg_postlet avg_postnumb   N
## 1:       1   15.29605    20.98026    27.79605     30.82237 152
## 2:       0   17.04545    20.63636    24.92045     28.60227  88
```

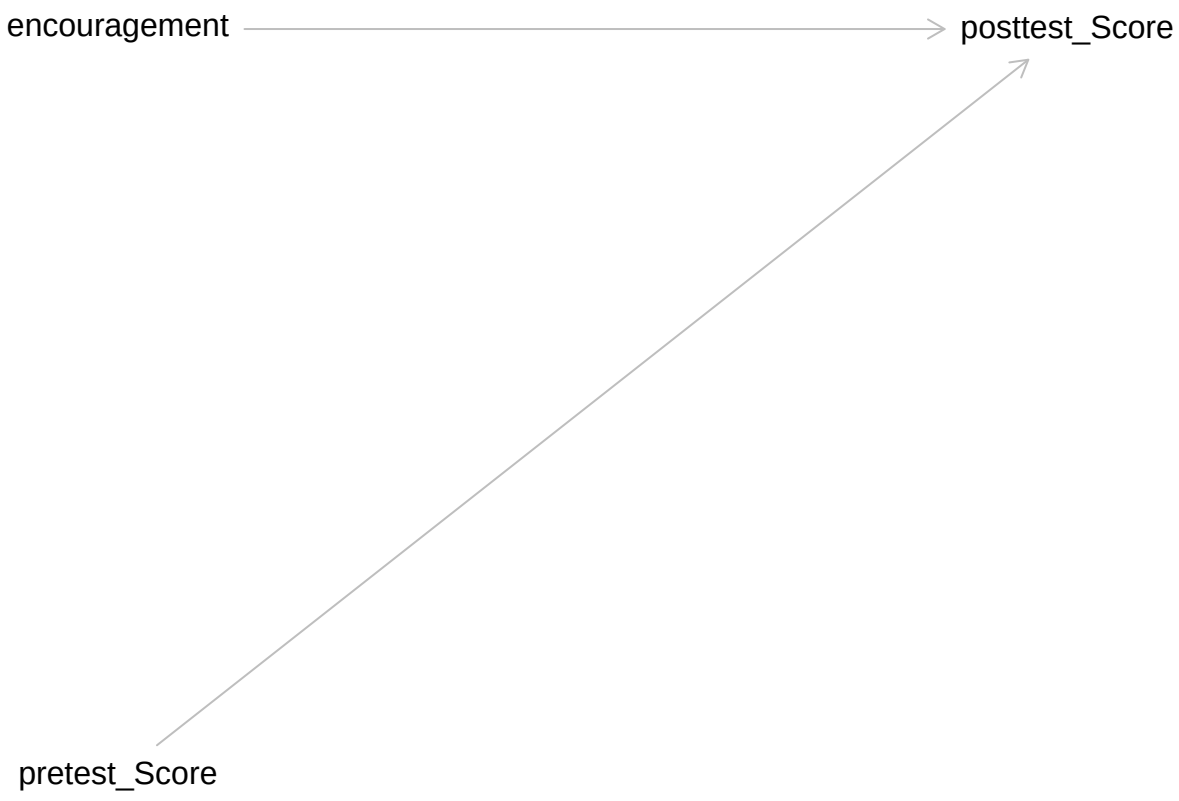
11 Sesame street pt2

Return to the Sesame Street example from the previous exercise.

(a) Did encouragement (the variable viewenc in the dataset) lead to an increase in post-test scores for letters (postlet) and numbers (postnumb)? Fit an appropriate model to answer this question.

First draw causal model.

```
g1 <- dagitty( 'dag {
  pretest_Score [pos="0,1"]
  encouragement [exposure, pos="0,0"]
  posttest_Score [outcome, pos="1,0"]

  pretest_Score -> posttest_Score
  encouragement -> posttest_Score

}')

plot(g1)
```



Fit models.

```r
lmfit <- lm(postlet ~ prelet + viewenc, data = sesame)
summary(lmfit)
```

```
##
## Call:
## lm(formula = postlet ~ prelet + viewenc, data = sesame)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -28.600  -7.954  -2.412   7.232  32.396
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  9.63831    1.83971   5.239 3.56e-07 ***
## prelet       0.89655    0.08341  10.749  < 2e-16 ***
## viewenc      4.44403    1.47441   3.014  0.00286 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.95 on 237 degrees of freedom
## Multiple R-squared:  0.335,  Adjusted R-squared:  0.3294
## F-statistic: 59.69 on 2 and 237 DF,  p-value: < 2.2e-16
```

```r
lmfit <- lm(postnumb ~ prenumb + viewenc, data = sesame)
summary(lmfit)
```

```
##
## Call:
## lm(formula = postnumb ~ prenumb + viewenc, data = sesame)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -37.879  -5.878  -0.478   5.612  23.140
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 11.94771    1.55532   7.682 4.12e-13 ***
## prenumb      0.80705    0.05734  14.076  < 2e-16 ***
## viewenc      1.94255    1.26863   1.531    0.127
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.47 on 237 degrees of freedom
## Multiple R-squared:  0.4591, Adjusted R-squared:  0.4546
## F-statistic: 100.6 on 2 and 237 DF,  p-value: < 2.2e-16
```

It lead to an increase that is stat sig for letters but not for numbers. However, both effect estimates are actually pretty uncertain: 4 +/- 3 points for letters, and 2 +/- 2.6 points for numbers.

(b) We are actually more interested in the effect of watching Sesame Street regularly (regular) than in the effect of being encouraged to watch Sesame Street. Fit an appropriate model to answer this question.

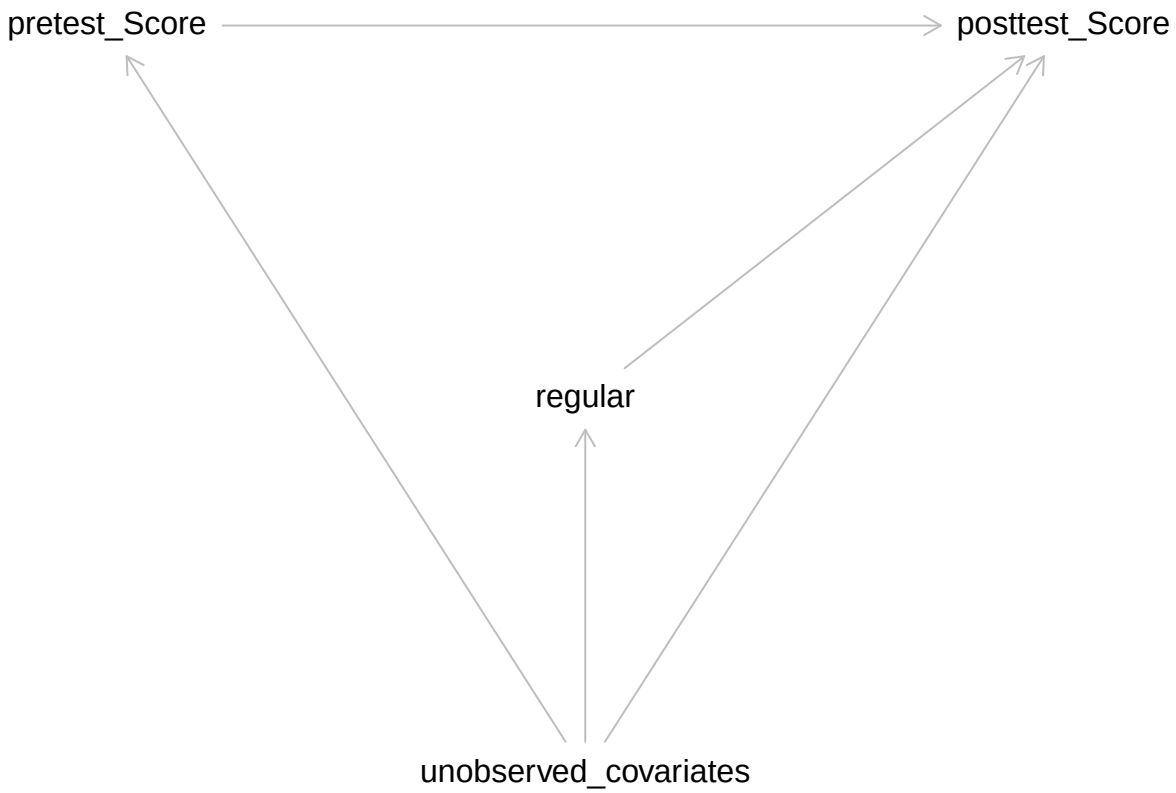
```
lmfit <- lm(postlet ~ prelet + regular, data = sesame)
summary(lmfit)
```

```
##
## Call:
## lm(formula = postlet ~ prelet + regular, data = sesame)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -31.862  -7.516  -1.558   5.948  26.630
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.08255    1.77511   2.863  0.00457 **
## prelet       0.81969    0.07728  10.607  < 2e-16 ***
## regular     11.09062    1.57640   7.035 2.12e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.15 on 237 degrees of freedom
## Multiple R-squared:  0.4288, Adjusted R-squared:  0.424
## F-statistic: 88.96 on 2 and 237 DF,  p-value: < 2.2e-16
```

(c) Comment on which of the two previous estimates can plausibly be interpreted causally.

A: Only the first one, because encouragement was randomized. We can expect many unobserved covariates to differ between regular and non-regular watchers. And some of these unobserved covariates will likely also influence the change in test scores.

```
g1 <- dagitty( 'dag {
  pretest_Score [pos="-1,0"]
  regular [exposure, pos="0,1"]
unobserved_covariates [pos="0,2"]
  posttest_Score [outcome, pos="1,0"]

  pretest_Score -> posttest_Score
  regular -> posttest_Score
unobserved_covariates -> regular
unobserved_covariates -> posttest_Score
unobserved_covariates -> pretest_Score

}')

plot(g1)
```



12. Messy randomization

(https://andrewgelman.com/2007/11/29/those_darn_cows/)

The folder cows contains data from an agricultural experiment that was conducted on 50 cows to estimate the effect of a feed additive on six outcomes related to the amount of milk fat produced by each cow.

Four diets (treatments) were considered, corresponding to different levels of the additive, and three variables were recorded before treatment assignment: lactation number (seasons of lactation), age, and initial weight of cow.

Cows were initially assigned to treatments completely at random, and then the distributions of the three covariates were checked for balance across the treatment groups; several randomizations were tried, and the one that produced the "best" balance with respect to the three covariates was chosen.

The treatment assignment is ignorable (because it depends only on fully observed covariates and not on unrecorded variables such as the physical appearances of the cows or the times at which the cows entered the study) but unknown (because the decisions whether to rerandomize are not explained).

We shall consider different estimates of the effect of additive on the mean daily milk fat produced.

```
dir.create("../../datasets/ARM_Data/cows")

download.file("http://www.stat.columbia.edu/~gelman/arm/examples/cows/cow.dat",
              "../../datasets/ARM_Data/cows/cow.dat")
download.file("http://www.stat.columbia.edu/~gelman/arm/examples/cows/readme.R",
              "../../datasets/ARM_Data/cows/readme.R")
```

```r
# measure manually in rstudio
sep_pos <- c(4,7,11,17,25,33,39,45,51,57)
sep_pos - shift(sep_pos)
```

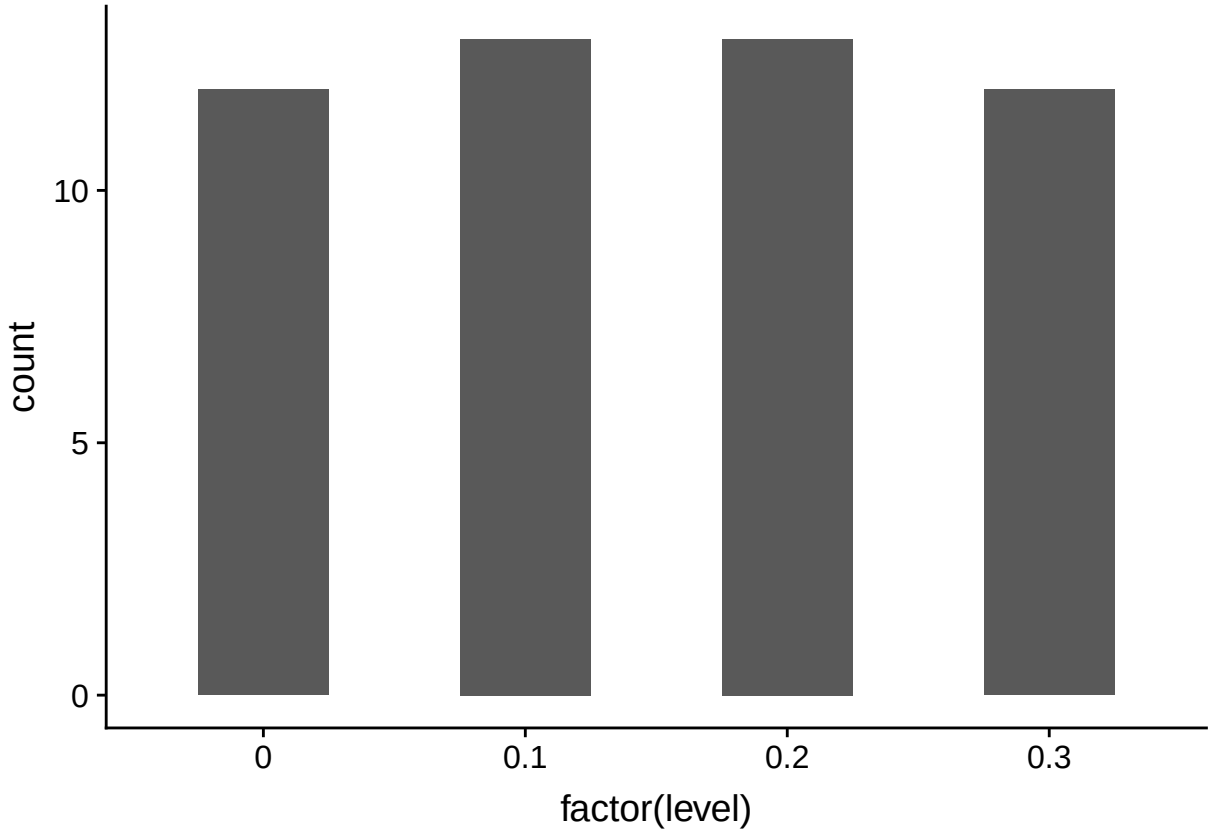
```
##  [1] NA  3  4  6  8  8  6  6  6  6
```

```r
cows <- read.fwf("../../datasets/ARM_Data/cows/cow.dat", skip = 5,
                 widths = c(4, 3,  4,  6,  8,  8,  6,  6,  6,  6))
colnames(cows) <- c("level","lactation","age","initial.weight","dry", "milk","fat","solids","final.weigh

cows$mean_daily_milk_fat <- cows$milk * cows$fat
```

(a) Consider the simple regression of mean daily milk fat on the level of additive. Compute the estimated treatment effect and standard error, and explain why this is not a completely appropriate analysis given the randomization used.

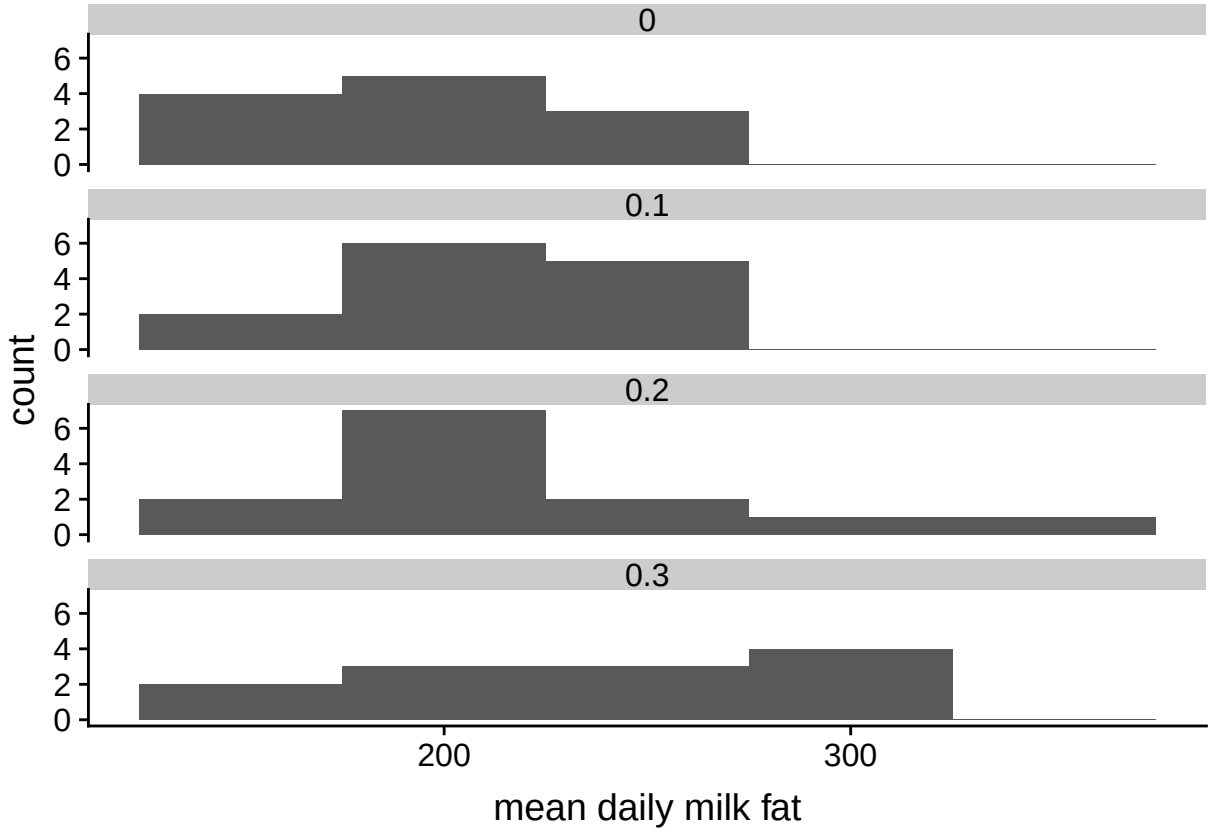
Check number of cows per treatment level. There are evenly spread.

```r
ggplot(cows, aes(x = factor(level))) + geom_bar(width = 0.5)
```



Plot the data distribution (daily milk fat) by treatment level.

```
ggplot(cows, aes(x = mean_daily_milk_fat)) + geom_histogram(binwidth = 50) +
  facet_wrap(. ~ factor(level), ncol = 1) + xlab("mean daily milk fat") + ylab("count")
```



Why is this not completely appropriate? Because treatment assignment was in fact partly determined by patterns in the measured covariates, those patterns could be influenced by unobserved covariates, that also affect outcome. This is the messy part.

```
lmfit <- lm(mean_daily_milk_fat ~ level, data = cows)
summary(lmfit)
```

```
##
## Call:
## lm(formula = mean_daily_milk_fat ~ level, data = cows)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -73.370 -29.086  -9.407  34.246 109.289
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   194.32      10.44  18.615   <2e-16 ***
## level         125.09      56.12   2.229   0.0305 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 43.65 on 48 degrees of freedom
## Multiple R-squared:  0.0938, Adjusted R-squared:  0.07492
## F-statistic: 4.969 on 1 and 48 DF,  p-value: 0.03053
```

A 0.1 increase in additive level leads to an increase of 12.5 gr fat.

(b) Add more predictors to the model. Explain your choice of which variables to include. Compare your estimated treatment effect to the result from (a).

There are three pretreatment predictors. For these variables, we expect them to influence the outcome (amount of milk and percentage fat) but we expect them not to lie on the causal path between level of additive and outcomes.

```
lmfit <- lm(mean_daily_milk_fat ~ level + lactation + age + initial.weight, data = cows)
summary(lmfit)
```

```
##
## Call:
## lm(formula = mean_daily_milk_fat ~ level + lactation + age +
##     initial.weight, data = cows)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -64.283 -25.939   1.157  20.402  82.329
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     46.16059   44.64933   1.034  0.30673
## level          105.47935   46.83330   2.252  0.02923 *
## lactation       33.01528   14.33531   2.303  0.02595 *
## age             -2.29849    1.08812  -2.112  0.04024 *
## initial.weight   0.13468    0.04522   2.978  0.00466 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 35.58 on 45 degrees of freedom
## Multiple R-squared:  0.4354, Adjusted R-squared:  0.3852
## F-statistic: 8.676 on 4 and 45 DF,  p-value: 2.8e-05
```

Surprisingly, it does not really increase the precision of the estimate.

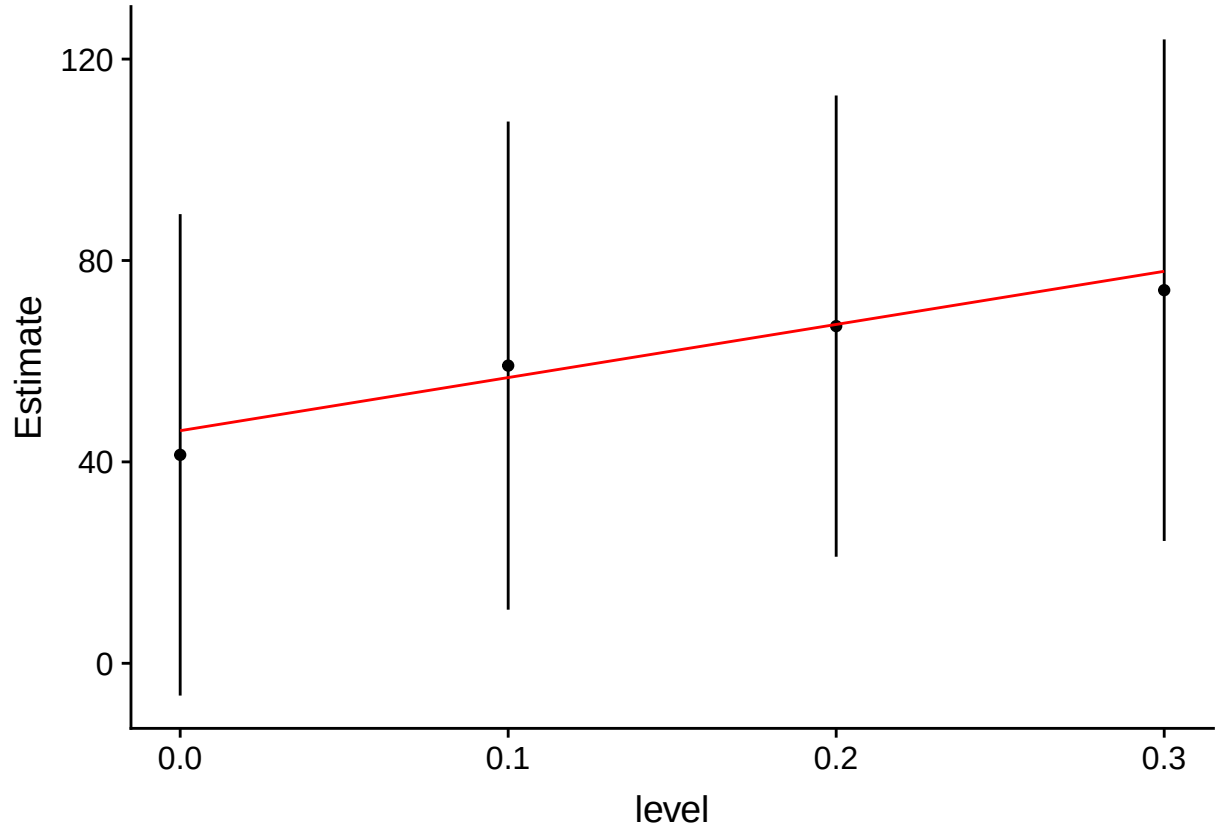
(c) Repeat (b), this time considering additive level as a categorical predictor with four letters. Make a plot showing the estimate (and standard error) of the treatment effect at each level, and also showing the inference from the model fit in part (b).

We fit without intercept to get proper std errors for the group means, while taking into account the covariates. This will likely break other stuff in the regression, but we don't care for that now.

```
lmfit_cat <- lm(mean_daily_milk_fat ~ factor(level) +
                  lactation + age + initial.weight -1, data = cows)
summary(lmfit_cat)
```

```
##
## Call:
## lm(formula = mean_daily_milk_fat ~ factor(level) + lactation +
##     age + initial.weight - 1, data = cows)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -61.757 -24.320   0.196  20.591  81.013
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## factor(level)0   41.40128   47.80440   0.866  0.39127
## factor(level)0.1 59.12348   48.47149   1.220  0.22920
## factor(level)0.2 66.96685   45.80249   1.462  0.15099
## factor(level)0.3 74.10386   49.80390   1.488  0.14407
## lactation        33.05706   14.63546   2.259  0.02903 *
## age              -2.28114    1.11004  -2.055  0.04598 *
## initial.weight    0.13519    0.04764   2.838  0.00691 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 36.28 on 43 degrees of freedom
## Multiple R-squared:  0.9761, Adjusted R-squared:  0.9723
## F-statistic: 251.3 on 7 and 43 DF,  p-value: < 2.2e-16
```

```
res <- coef(summary(lmfit_cat))
res <- data.frame(res)
res$varname <- rownames(res)
res <- res[1:4,]
res$level <- c(0, 0.1, 0.2, 0.3)
#(Intercept)     46.16059   44.64933
#level          105.47935   46.83330
res$est_cont <- 46.2 + 105.5 * res$level

ggplot(res, aes(x = level, y = Estimate)) + geom_point() +
  geom_linerange(aes(ymin = Estimate - Std..Error, ymax = Estimate + Std..Error )) +
  geom_line(aes(y = est_cont), col = "red")
```



(This does not work, because we do not share power for covariates. Need to estimate simultaneously.)

```
library(arm)
treatment_levels <- unique(cows$level)
for(k in 1:4){
  display(lm(mean_daily_milk_fat ~ lactation + age + initial.weight,
             data = cows,
             subset = (level == treatment_levels[k])))
}
```

13. Election outcomes

The folder congress has election outcomes and incumbency for U.S. congressional election races in the 1900s.

```
#http://www.stat.columbia.edu/~gelman/book/data/incumbency/house.doc
congress <- vector ("list", 49)
for (i in 1:49){
  year <- 1896 + 2*(i-1)
  file <- paste ("../../datasets/ARM_Data/congress/cong3/", year, ".asc", sep="")
  data.year <- matrix (scan (file, quiet = T), byrow=TRUE, ncol=5)
  data.year <- cbind (rep(year, nrow(data.year)), data.year)
  congress[[i]] <- data.year
}
```

```
# Each file is formatted identically with blanks between columns and with the
# following fields: (1) State #, (2) District #, (3) Incumbency Code, (4) #
# Democratic votes, and (5) # Republican votes.

# States are numbered using the standard ICPSR code

# we pick 1990 (so we also need 1988)

df <- data.frame(congress[[47]])
df <- rbind(df,
            data.frame(congress[[48]]))

colnames(df) <- c("year", "state_id", "district_id",
                  "incumb_code", "dem_votes", "rep_votes")

df <- data.table(df)

df <- df[, dem_share := dem_votes / (dem_votes + rep_votes)]

# if not contested then no voting takes place, coded as -9
df <- df[dem_votes == -9, dem_share := NA]
df <- df[rep_votes == -9, dem_share := NA]
# check state - district ids
#unique(paste(df$state_id, df$district_id))
```

(a) Take data from a particular year, t, and estimate the effect of incumbency by fitting a regression of $v_{i,t}$, the Democratic share of the two-party vote in district i, on $v_{i,t-2}$ (the outcome in the previous election, two years earlier), I_{it} (the incumbency status in district i in election t, coded as 1 for Democratic incumbents, 0 for open seats, -1 for Republican incumbents), and P_{it} (the incumbent party, coded as 1 if the sitting congressmember is a Democrat and -1 if he or she is a Republican).

In your analysis, include only the districts where the congressional election was contested in both years, and do not pick a year ending in "2." (District lines in the United States are redrawn every ten years, and district election outcomes v_{it} and $v_{i,t-2}$ are not comparable across redistrictings, for example, from 1970 to 1972.)

```
# dem vote share panel
df_lm <- dcast(df[, .(year, state_id, district_id, dem_share)],
               state_id + district_id ~ year)
```

```
## Using 'dem_share' as value column. Use 'value.var' to override
```

```
df_inc <- dcast(df[, .(year, state_id, district_id, incumb_code)],
               state_id + district_id ~ year)
```

```
## Using 'incumb_code' as value column. Use 'value.var' to override
```

```
# Arkansas 1988
# https://en.wikipedia.org/wiki/United_States_House_of_Representatives_elections,_1988#Arkansas
df[state_id == 42 & district_id == 1]
```

```
##    year state_id district_id incumb_code dem_votes rep_votes dem_share
## 1: 1988       42           1           1        -9        -9        NA
## 2: 1990       42           1           1    101026     56071 0.6430804
```

```r
# incumbent was unopposed!
df[state_id == 82 & district_id == 1]
```

```
##    year state_id district_id incumb_code dem_votes rep_votes dem_share
## 1: 1988       82           1          -1     76394     96848 0.4409670
## 2: 1990       82           1           0     97622     62982 0.6078429
```

```r
# https://en.wikipedia.org/wiki/United_States_House_of_Representatives_elections,_1990#Hawaii
# in 1990 rep incumbent (elected 1986) retired
```

```r
# Create dataset for regression
setnames(df_lm, "1988", "dem_share_1988")
setnames(df_lm, "1990", "dem_share_1990")
# take df_lm, join incumb_code from 1990 on state and district id
df_incum <- df[year == 1990, .(state_id, district_id, incumb_code)]
setkey(df_incum, state_id, district_id)
setkey(df_lm, state_id, district_id)

df_lm <- df_incum[df_lm]
# from dem_share in 1988, derive incumbent party in 1990
df_lm <- df_lm[, incum_party := 0]
df_lm <- df_lm[dem_share_1988 > 0.5, incum_party := 1]
```

```r
# subset on all variables present
df_lm <- na.omit(df_lm[, .(dem_share_1990, dem_share_1988, incumb_code, incum_party, state_id, district_
# remove uncontested seats (Assume this is when vote share is 0 or 1)
df_lm <- df_lm[dem_share_1990 >0 & dem_share_1990 < 1,]
df_lm <- df_lm[dem_share_1988 >0 & dem_share_1988 < 1,]
# run regression
lmfit <- lm(dem_share_1990 ~ dem_share_1988 + as.factor(incumb_code) + incum_party,
            data = df_lm)

summary(lmfit)
```

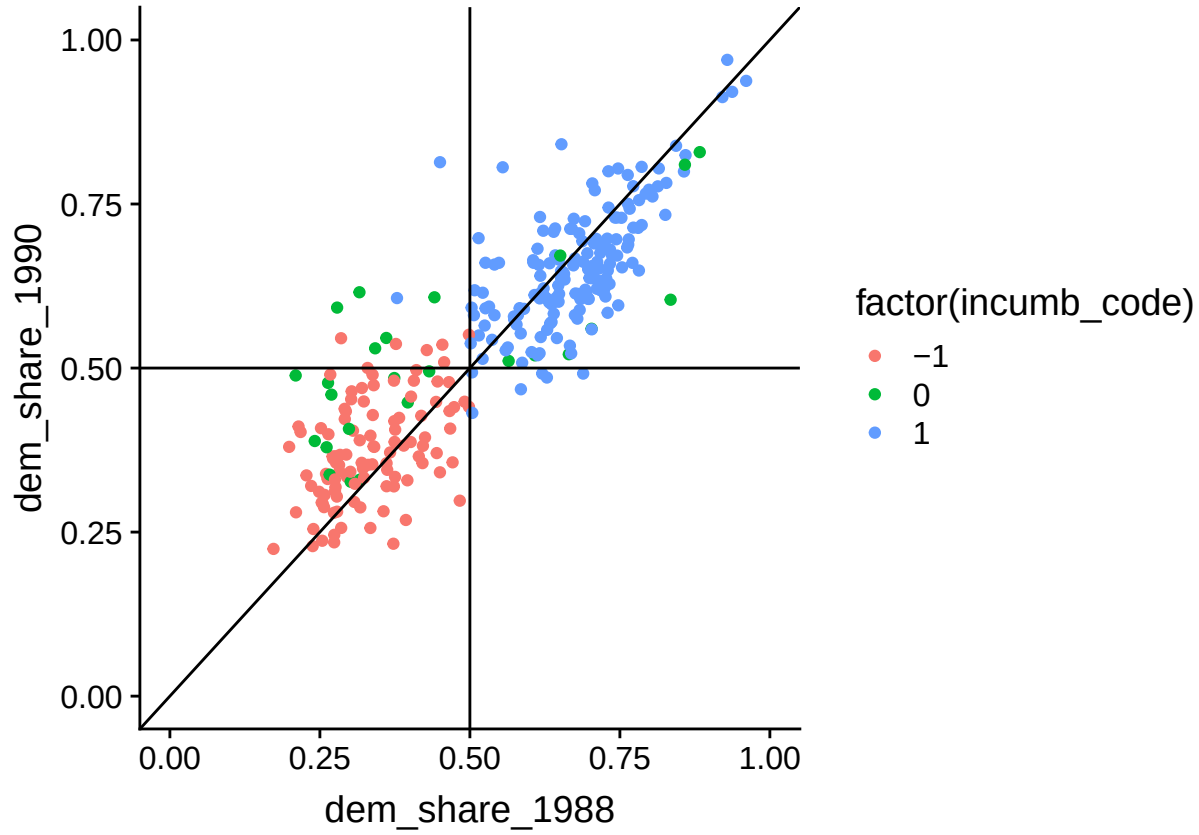
```
##
## Call:
## lm(formula = dem_share_1990 ~ dem_share_1988 + as.factor(incumb_code) +
##     incum_party, data = df_lm)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.174503 -0.047238 -0.006533  0.038879  0.232263
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)                0.15178    0.01701   8.923  < 2e-16 ***
## dem_share_1988             0.66343    0.04670  14.205  < 2e-16 ***
## as.factor(incumb_code)0    0.11537    0.01748   6.599 1.92e-10 ***
## as.factor(incumb_code)1    0.18861    0.02604   7.242 3.86e-12 ***
## incum_party               -0.13471    0.03006  -4.482 1.06e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.06971 on 295 degrees of freedom
## Multiple R-squared:  0.8123, Adjusted R-squared:  0.8098
## F-statistic: 319.3 on 4 and 295 DF,  p-value: < 2.2e-16
```

(b) Plot the fitted model and the data, and discuss the political interpretation of the estimated coefficients.

PM Plot 2 x 2 model fit lines from coef(lmfit)

```
ggplot(df_lm, aes(x = dem_share_1988, y = dem_share_1990, col = factor(incumb_code),
                  group = factor(incumb_code))) +
  geom_point() + geom_vline(xintercept = 0.5) +
  geom_hline(yintercept = 0.5) +
  geom_abline(slope = 1, intercept = 0) +
  xlim(0,1) + ylim(0,1)
```



(c) What assumptions are needed for this regression to give a valid estimate of the causal effect of incumbency? In answering this question, define clearly what is meant by incumbency as a "treatment variable."

See Erikson (1971), Gelman and King (1990), Cox and Katz (1996), Levitt and Wolfram (1997), Ansolabehere, Snyder, and Stewart (2000), Ansolabehere and Snyder (2002), and Gelman and Huang (2006) for further work and references on this topic.

14. Value of a statistical life

Causal inference based on data from individual choices: our lives involve tradeoffs between monetary cost and physical risk, in decisions ranging from how large a car to drive, to choices of health care, to purchases of safety equipment. Economists have estimated people's implicit balancing of dollars and danger by comparing different jobs that are comparable but with different risks, fftting regression models predicting salary given the probability of death on the job. The idea is that a riskier job should be compensated with a higher salary, with the slope of the regression line corresponding to the "value of a statistical life."

(a) Set up this problem as an individual choice model, as in Section 6.8. What are an individual's options, value function, and parameters?

(b) Discuss the assumptions involved in assigning a causal interpretation to these regression models.

See Dorman and Hagstrom (1998), Costa and Kahn (2002), and Viscusi and Aldy (2002) for different perspectives of economists on assessing the value of a life, and Lin et al. (1999) for a discussion in the context of the risks from radon exposure.