PS4 Solution

Questions

Problem 1: Unweighted vs. Weighted Average

Using the Titanic dataset, answer the following parts. You can find information on the dataset variables here

a) Write down in expected value notation the estimator for a simple difference in means for this data set. Assume that we are interested in the effect of being in first class on survival.

$$E[Y|D = 1] - E[Y|D = 0]$$

where Y is equal to survival and D is whether a passenger was in first class

b) Using R, estimate the estimator that you wrote down in part a

```
titanic <- read_csv("titanic.csv")

### Get first class vs other
titanic <- titanic %>%
    mutate(pclass = if_else(pclass == 1, 1, 0))

### Naive estimator
difference_in_means(survived ~ pclass, data = titanic) %>%
    tidy()
```

```
##     term  estimate  std.error statistic      p.value  conf.low conf.high
## 1 pclass 0.3149354 0.03077636  10.23303 1.574796e-22 0.2544752 0.3753956
##        df  outcome
## 1 523.983 survived
```

c) Assume that assignment might be confounded by gender and age. Write down the new estimator that we are interested in to test the effect of being in first class on survival.

$$E[Y|D = 1, A, G] - E[Y|D = 0, A, G]$$

d) Calculate the weighted average treatment effect by doing the following. i) Define a variable `s` that takes on four values: 1 for a male child, 2 for a male adult, 3 for a female child, and 4 for a female adult. Define a child as an individual under 15. Compare the estimator in (b) to your weighted ATE. Which one do you think is more likely to be correct and why?

```
### make variable s
titanic <- titanic %>%
  mutate(age = if_else(age >= 15, 0, 1),
         sex = if_else(sex == "female", 1, 0),
         s = case_when(
           # male child
          sex == 0 & age == 1 ~1,
          # male adult
          sex == 0 & age == 0 ~2,
          # female child
          sex == 1 & age == 1~3,
          # female adult
          sex == 1 & age == 0~4
         )
  )

## Get weights
wt1 <- titanic %>%
  filter(s == 1, pclass == 0)%>%
  nrow(.)/nrow(titanic)

wt2 <- titanic %>%
  filter(s == 2, pclass == 0)%>%
  nrow(.)/nrow(titanic)

wt3 <- titanic %>%
  filter(s == 3, pclass == 0)%>%
  nrow(.)/nrow(titanic)

wt4 <- titanic %>%
  filter(s == 4, pclass == 0)%>%
  nrow(.)/nrow(titanic)


## Within strata DM
### Treateds
get_treat <- function(df, s){
  ## you can do this by hand but we can wrap it in a function
  treat <- mean(df$survived[df$s == s & df$pclass == 1], na.rm = T)
  control <- mean(df$survived[df$s == s & df$pclass == 0], na.rm = T)
  return(treat - control)
}

diff1 <- get_treat(titanic, s = 1)
diff2 <- get_treat(titanic, s = 2)
diff3 <- get_treat(titanic, s = 3)
diff4 <- get_treat(titanic, s = 4)

wate <- diff1 *wt1 + diff2 * wt2 + diff3*wt3 + diff4*wt4
wate
```

```
## [1] 0.1377635
```

The latter is likely to be a more accurate estimate because we are including other factors that we think would affect treatment.

Problem 2

Read Hyde (2007). Then answer the following questions:

a) Based on lectures and readings about natural experiments, evaluate the plausibility of this research design as a natural experiment.

This is a natural experiment. Hyde does not control the assignment mechanism of election monitors. The *outcome is electoral fraud. The treatment is election monitors. The external mechanism was a list of stations for each monitor team. The entity making the list did not possess information about polling station attributes other than geographic logistics and were designed to not overlap with other teams.*

How plausible this design is depends on whether or not this assignment is considered random as opposed to haphazard. There isn't an explicit balance check in the article, but there is a test that round 1 treatment should be equal between polling stations that were monitored in the second round and those that were not. The design passes this check.

For the next set of questions, use the `HydeData.csv` dataset.

b) Replicate Table 1

```
### Table 1

hyde <- read_csv(file = "HydeData.csv")

hyde_sub1 <- hyde %>%
    filter(O1 == 1 | O4 == 1)
hyde_sub2 <- hyde %>%
    filter(mon_voting == 0)
hyde_sub3 <- hyde %>%
    filter(mon_voting_R2 == 0)
hyde_sub4 <- hyde %>%
    filter(mon_voting == 1)

# Should also have this but these columns were not included
# in your dataset by mistake
hyde_sub5 <- hyde %>%
    filter(O3 == 1 | O4 == 1)


get_dm <- function(Y, D, data) {
    formula <- reformulate(termlabels = D, response = Y)
    out <- difference_in_means(formula, data = data) %>%
        tidy() %>%
        mutate(estimate = round(abs(estimate), 3), statistic = abs(statistic),
            p.value = round(p.value, 4)) %>%
        select(estimate, t = statistic, p.value)
    return(out)
}

row1 <- get_dm("kocharian", "mon_voting", data = hyde)
row2 <- get_dm("KocharianR2", "mon_votingR2", data = hyde)
row3 <- get_dm("KocharianR2", "O4", data = hyde_sub1)
row4 <- get_dm("AveKocharian", "O4", data = hyde_sub1)
row5 <- get_dm("AveKocharian", "O9", data = hyde)
row6 <- get_dm("KocharianR2", "mon_voting", data = hyde_sub3)
row7 <- get_dm("KocharianR2", "mon_votingR2", data = hyde_sub2)
# This row is going to be a bit off because your instructor
# didn't include all the appropriate rows in the dataset
row8 <- get_dm("KocharianR2", "O4", hyde_sub5)
row9 <- get_dm("KocharianR2", "O4", hyde_sub4)
row10 <- get_dm("KocharianR2", "R1R2only", hyde)

### Get all the rows into one data frame
```

Table 2: Table 3

term	estimate	std.error	statistic	p.value	conf.low	conf.high	df	outcome
(Intercept)	0.6376183	0.0042210	151.05703	0.000000	0.6293396	0.6458971	1762	turnout
mon_voting_R2	0.0173486	0.0122907	1.41152	0.158268	-0.0067573	0.0414544	1762	turnout

term	estimate	std.error	statistic	p.value	conf.low	conf.high	df	outcome
(Intercept)	0.2758383	0.0043310	63.6894685	0.0000000	0.2673439	0.2843327	1762	demirchian
mon_voting_R2	0.0024161	0.0068338	0.3535548	0.7237149	-0.0109870	0.0158193	1762	demirchian

term	estimate	std.error	statistic	p.value	conf.low	conf.high	df	outcome
(Intercept)	0.2758383	0.0185614	14.8608802	0.0000012	0.2321416	0.3195350	7.156637	demirchian
mon_voting_R2	0.0024161	0.0247880	0.0974711	0.9255662	-0.0584269	0.0632591	5.924072	demirchian

```r
bind_rows(row1, row2, row3, row4, row5, row6, row7, row8, row9,
    row10) %>%
    mutate(estimate = estimate * 100) %>%
    knitr::kable(., caption = "Difference of Means Tests Comparing 'Treatment and 'Control' Groups",
        col.names = c("Difference (Percentage)", "t-statistic",
            "p value"))
```

Table 1: Difference of Means Tests Comparing ‘Treatment and ’Control’ Groups

Difference (Percentage)	t-statistic	p value
5.9	5.9910595	0.0000
2.0	2.4693548	0.0137
4.5	4.5120857	0.0000
5.8	5.4937577	0.0000
4.6	5.6231980	0.0000
4.4	4.3799638	0.0000
2.0	1.6904033	0.0917
2.5	1.9108051	0.0566
0.1	0.0941106	0.9250
2.4	1.8223494	0.0690

c) Replicate Table 3

The standard errors are wrong for reasons we've mentioned in class

```r
# Technically you'd need to cluster this by region We'll
# ignore that because the original dataset does not include
# it There's an updated version of the dataset online if
# you want to compare
lm1 <- lm_robust(turnout ~ mon_voting_R2, data = hyde) %>%
    tidy()
lm2 <- lm_robust(demirchian ~ mon_voting_R2, data = hyde) %>%
    tidy()
lm3 <- lm_robust(demirchian ~ mon_voting_R2, data = hyde, clusters = regionmarzes) %>%
    tidy()
knitr::kable(list(lm1, lm2, lm3), caption = "Table 3")
```

d) Replicate Table 4

The standard errors are wrong for reasons we've mentioned in class

Table 3: Table 4

term	estimate	std.error	statistic	p.value	conf.low	conf.high	df	outcome
(Intercept)	0.6326962	0.0113343	55.821602	0.0000000	0.6104662	0.6549263	1760	kocharian
mon_voting	-0.0304835	0.0092300	-3.302657	0.0009769	-0.0485863	-0.0123806	1760	kocharian
totalvoters	-0.0002781	0.0000183	-15.161262	0.0000000	-0.0003141	-0.0002421	1760	kocharian
total	0.0003194	0.0000296	10.789031	0.0000000	0.0002613	0.0003774	1760	kocharian

term	estimate	std.error	statistic	p.value	conf.low	conf.high	df	outcome
(Intercept)	0.6482301	0.0112069	57.842013	0.000000	0.6262498	0.6702103	1761	kocharian
mon_voting	-0.0190915	0.0096639	-1.975556	0.048361	-0.0380454	-0.0001376	1761	kocharian
totalvoters	-0.0000994	0.0000077	-12.934179	0.000000	-0.0001145	-0.0000843	1761	kocharian

term	estimate	std.error	statistic	p.value	conf.low	conf.high	df	outcome
(Intercept)	0.5917988	0.0119521	49.514226	0.0000000	0.5683570	0.6152405	1761	kocharian
mon_voting	-0.0387457	0.0101552	-3.815352	0.0001407	-0.0586633	-0.0188282	1761	kocharian
total	-0.0000771	0.0000148	-5.194570	0.0000002	-0.0001063	-0.0000480	1761	kocharian

```r
lm4 <- lm_robust(kocharian ~ mon_voting + totalvoters + total,
    data = hyde) %>%
    tidy()
lm5 <- lm_robust(kocharian ~ mon_voting + totalvoters, data = hyde) %>%
    tidy()
lm6 <- lm_robust(kocharian ~ mon_voting + total, data = hyde) %>%
    tidy()
knitr::kable(list(lm4, lm5, lm6), caption = "Table 4")
```

e) Replicate Table 5

For each table, explain in your own words what the table means and how a reader should evaluate it in context of Hyde's research design and claims.

```r
t51 <- difference_in_means(kocharian ~ mon_voting, data = hyde %>%
    filter(urban == 1)) %>%
    tidy()
t52 <- difference_in_means(kocharian ~ mon_voting, data = hyde %>%
    filter(urban == 0)) %>%
    tidy()
t53 <- difference_in_means(kocharian ~ mon_voting, data = hyde %>%
    filter(nearNagorno == 1)) %>%
    tidy()
t54 <- difference_in_means(kocharian ~ mon_voting, data = hyde %>%
    filter(nearNagorno == 0)) %>%
    tidy()

bind_rows(t51, t52, t53, t54) %>%
    select(estimate, statistic, p.value) %>%
    mutate(across(estimate:statistic, abs), estimate = estimate *
        100) %>%
    knitr::kable(., caption = "Difference of Means Tests of Round 1 Kocharian Vote Share",
        col.names = c("Difference (Percentage)", "t-statistic",
            "p value"))
```

Table 5: Part B

term	estimate	std.error	statistic	p.value	conf.low	conf.high	df	outcome
treatment	20.42817	3.081757	6.628739	0	14.37137	26.48496	440	primary

term	estimate	std.error	statistic	p.value	conf.low	conf.high	df	outcome
treatment	-0.4684782	1.086343	-0.4312434	0.6665826	-2.60578	1.668823	319	secondary

Table 4: Difference of Means Tests of Round 1 Kocharian Vote Share

Difference (Percentage)	t-statistic	p value
7.823496	6.4727343	0.0000000
2.781078	1.7209967	0.0857334
3.388373	0.9595517	0.3400328
6.110441	6.2322731	0.0000000

The t statistics are a bit different than the article due to the method of calculation.

Problem 3

Read Jakiela (2021) available here. The dataset for this problem is `JakielaData.csv`.

a) What are Jakiela's two diagnostics? In your own words, why are they helpful in assessing the potential bias of the two-way fixed effects estimator?

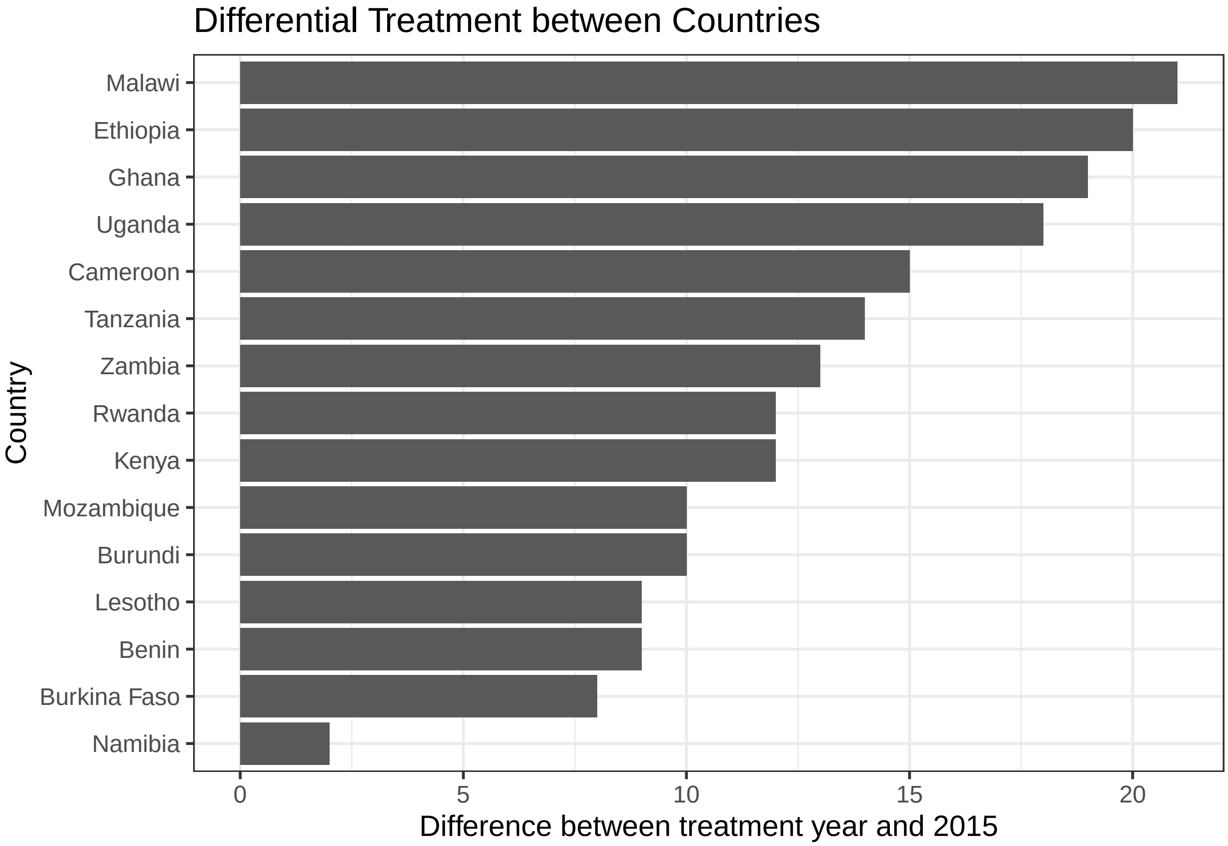
The first diagnostic is to test whether any treated units get negative weights. The second is to test the homogeneity assumption directly.

b) Estimate two TWFE models. In both models, use country and year fixed effects and cluster by country. In your first model, regress the number of enrollees in primary school on the treatment. In your second model, regress the number of enrollees in secondary school on the treatment.

```r
data <- read_csv("JakielaData.csv")

primary <- data %>%
    filter(!is.na(primary))

secondary <- data %>%
    filter(!is.na(secondary))

m1 <- lm_robust(primary ~ treatment, fixed_effects = ~country +
    year, data = primary) %>%
    tidy()

m2 <- lm_robust(secondary ~ treatment, fixed_effects = ~country +
    year, data = secondary) %>%
    tidy()


knitr::kable(list(m1, m2), caption = "Part B")
```

c) Did these countries pass laws at different times? To assess this, create a new variable in your dataset called `lengthTreat` that is the number of years between the first year of implementation and the last year of the dataset for each country. Make a bar graph (`geom_col`) of this variable.

```
data %>%
    mutate(lengthTreat = max(year) - fpe_year) %>%
    distinct(country, lengthTreat) %>%
    arrange(desc(lengthTreat)) %>%
    ggplot(aes(x = reorder(country, lengthTreat), y = lengthTreat)) +
    geom_bar(stat = "identity") + coord_flip() + ylab("Difference between treatment year and 2015") +
    xlab("Country") + theme_bw() + ggtitle("Differential Treatment between Countries")
```



d) Subset your data into two data frames, one for primary schools and one for secondary schools. For each data frame:

- *i*) Get the residuals of the regression of treatment on the fixed effects.
- *ii*) Append the residuals to the data frame. Add a column for treatment weights calculated as in Equation (2) of Jakiela's paper.
- *iii*) Show that the coefficient of treatment is equal to the sum of the outcome multiplied by treatment weights
- *iv*) Make a histogram (`geom_histogram()`) of the weights. Are any weights negative?
- *v*) Run a regression of the outcome variable on the fixed effects. Get the residuals of this regression. Plot the residuals of the treatment against the residuals of the outcome. Use the color argument in `aes()` to differentiate the points based on whether they are treated or untreated.
- *vi*) Statistically test if the slopes of the two groups are the same by running a regression of the outcome residuals on the treatment residuals and the interaction of the treatment residuals and treatment. For either model, is the interaction term significant? If it is, what does that mean, according to Jakiela?

```
## Parts i and ii
m3 <- lm_robust(treatment ~ country + factor(year), data = primary)
residuals <- primary$treatment - m3$fitted.values
primary <- primary %>%
    mutate(residuals = residuals, weight = residuals/sum(residuals^2))

m4 <- lm_robust(treatment ~ country + factor(year), data = secondary)
residuals_s <- secondary$treatment - m4$fitted.values
secondary <- secondary %>%
    mutate(residuals = residuals_s, weight = residuals/sum(residuals^2))
```

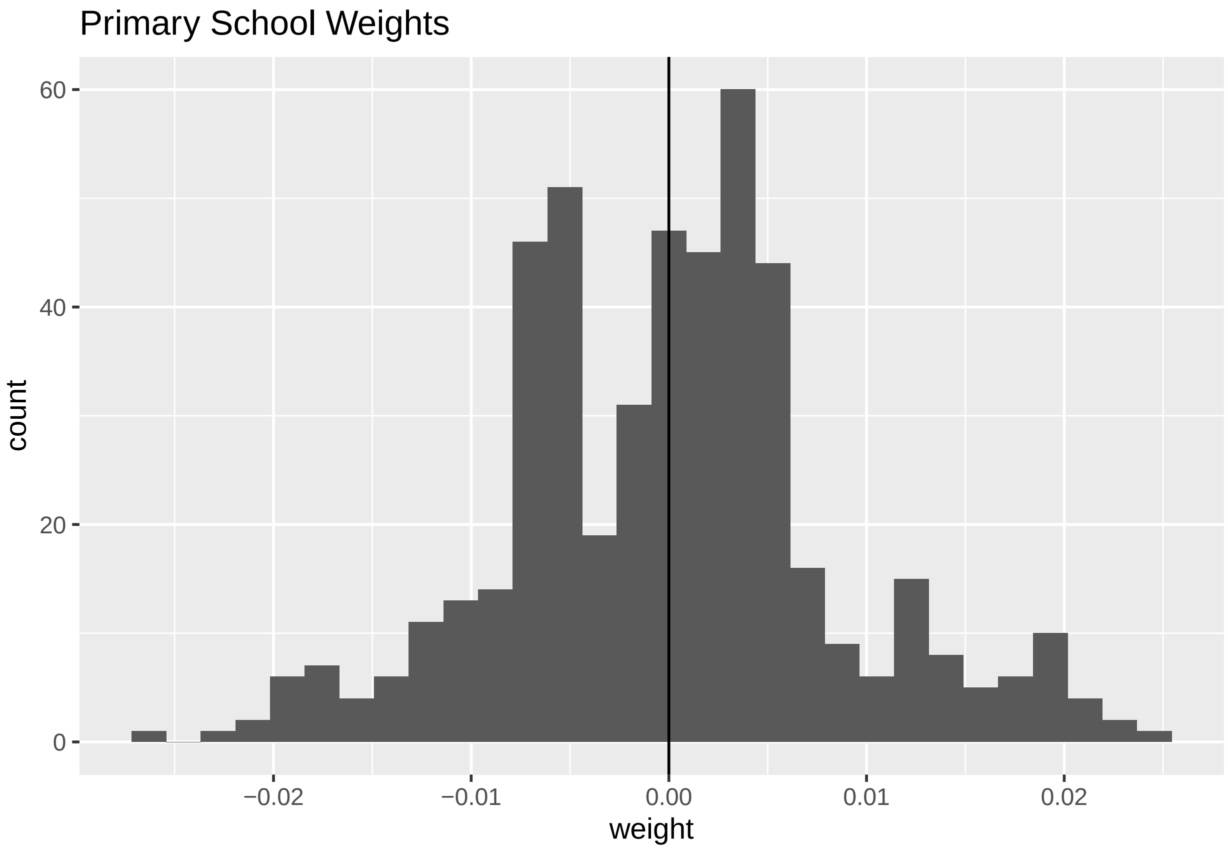
```
### Part iii
primary %>%
    summarise(beta = sum(primary * weight))
```

```
## # A tibble: 1 x 1
##    beta
##   <dbl>
## 1  20.4
```

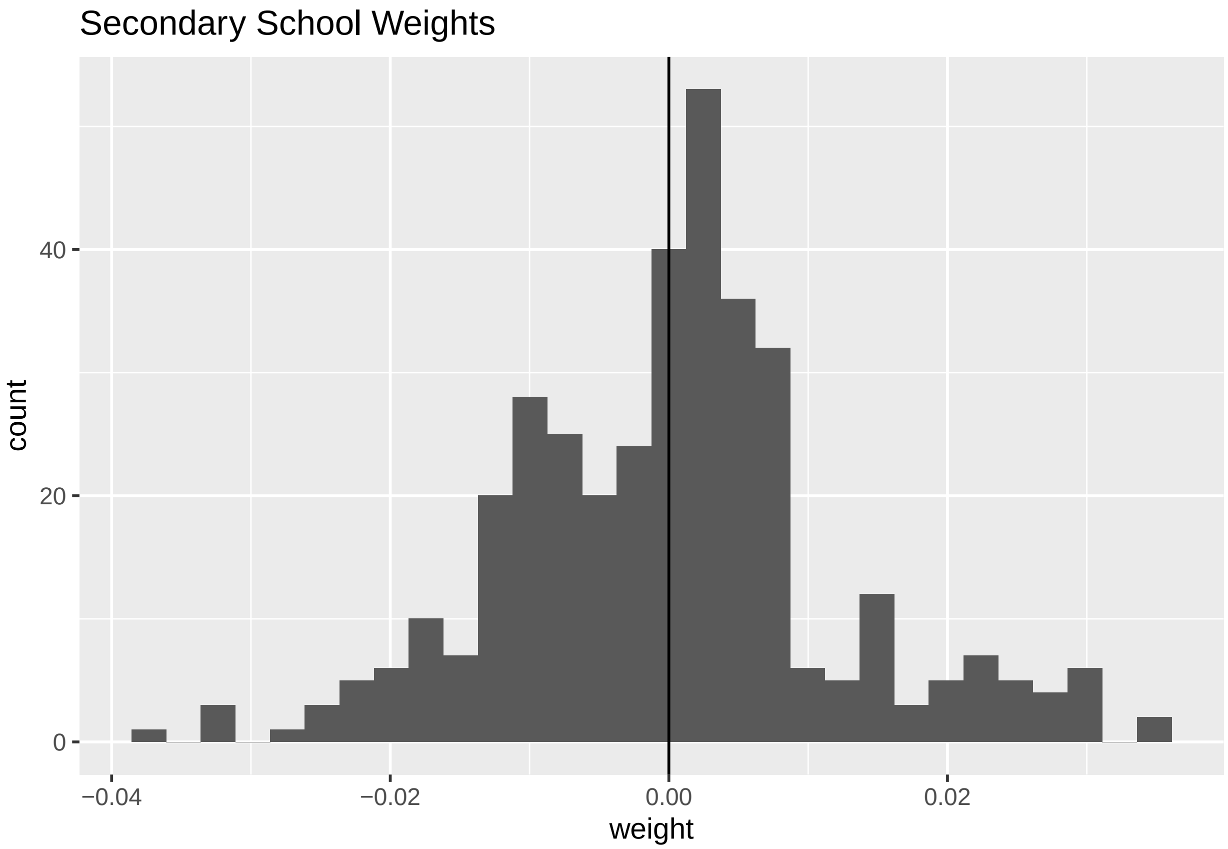
```
secondary %>%
    summarise(beta = sum(secondary * weight))
```

```
## # A tibble: 1 x 1
##     beta
##    <dbl>
## 1 -0.468
```

```
primary %>%
    ggplot(., aes(x = weight)) + geom_histogram() + geom_vline(xintercept = 0) +
    ggtitle("Primary School Weights")
```

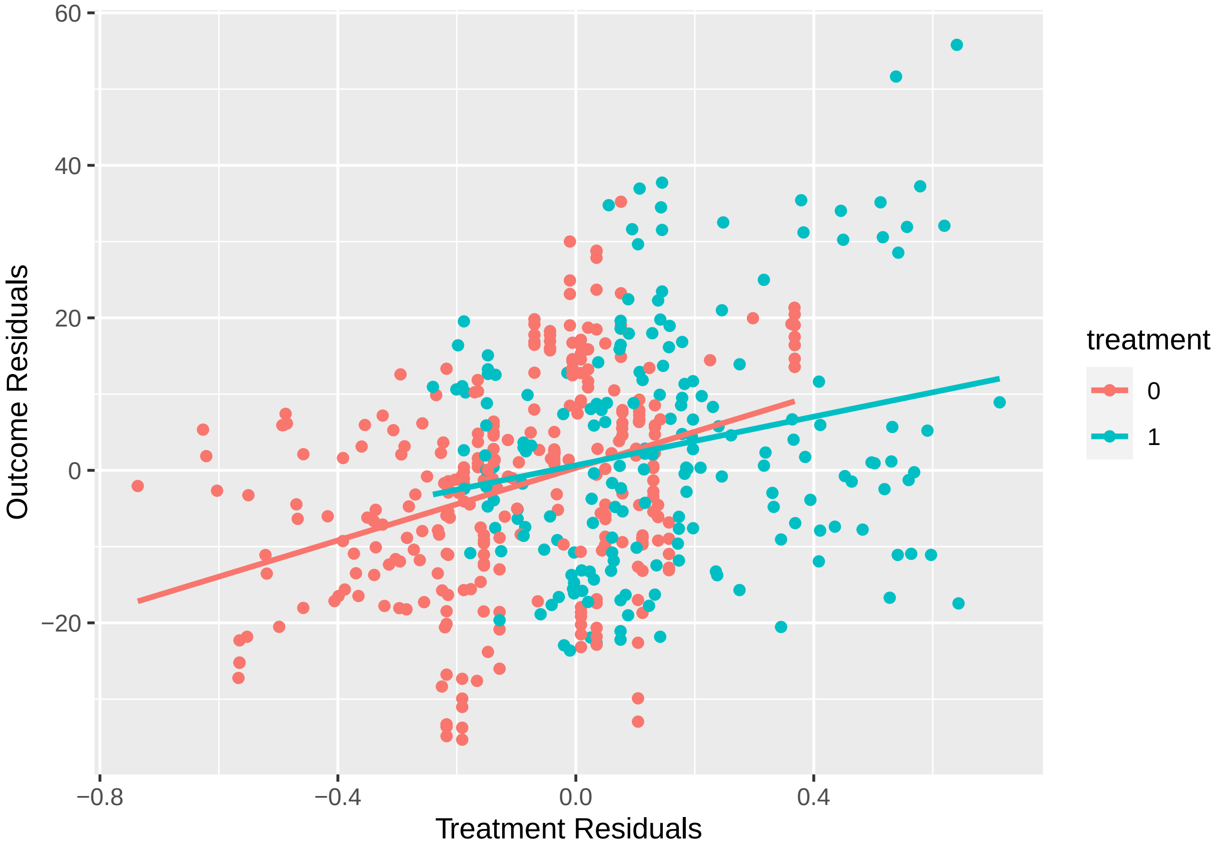


```
secondary %>%
    ggplot(., aes(x = weight)) + geom_histogram() + geom_vline(xintercept = 0) +
    ggtitle("Secondary School Weights")
```



```
outcomes_residuals_p <- lm_robust(primary ~ country + factor(year),
    data = primary)
outcomes_residuals_s <- lm_robust(secondary ~ country + factor(year),
    data = secondary)

out_resid_p <- primary$primary - outcomes_residuals_p$fitted.values
out_resid_s <- secondary$secondary - outcomes_residuals_s$fitted.values

## Join them up
primary <- primary %>%
    mutate(outcome_residuals = out_resid_p)

secondary <- secondary %>%
    mutate(outcome_residuals = out_resid_s)
```

```
## Plot the residuals against each other
primary %>%
    mutate(treatment = as.character(treatment)) %>%
    ggplot(aes(residuals, outcome_residuals, color = treatment)) +
    geom_point() + geom_smooth(method = "lm", se = F) + xlab("Treatment Residuals") +
    ylab("Outcome Residuals")
```

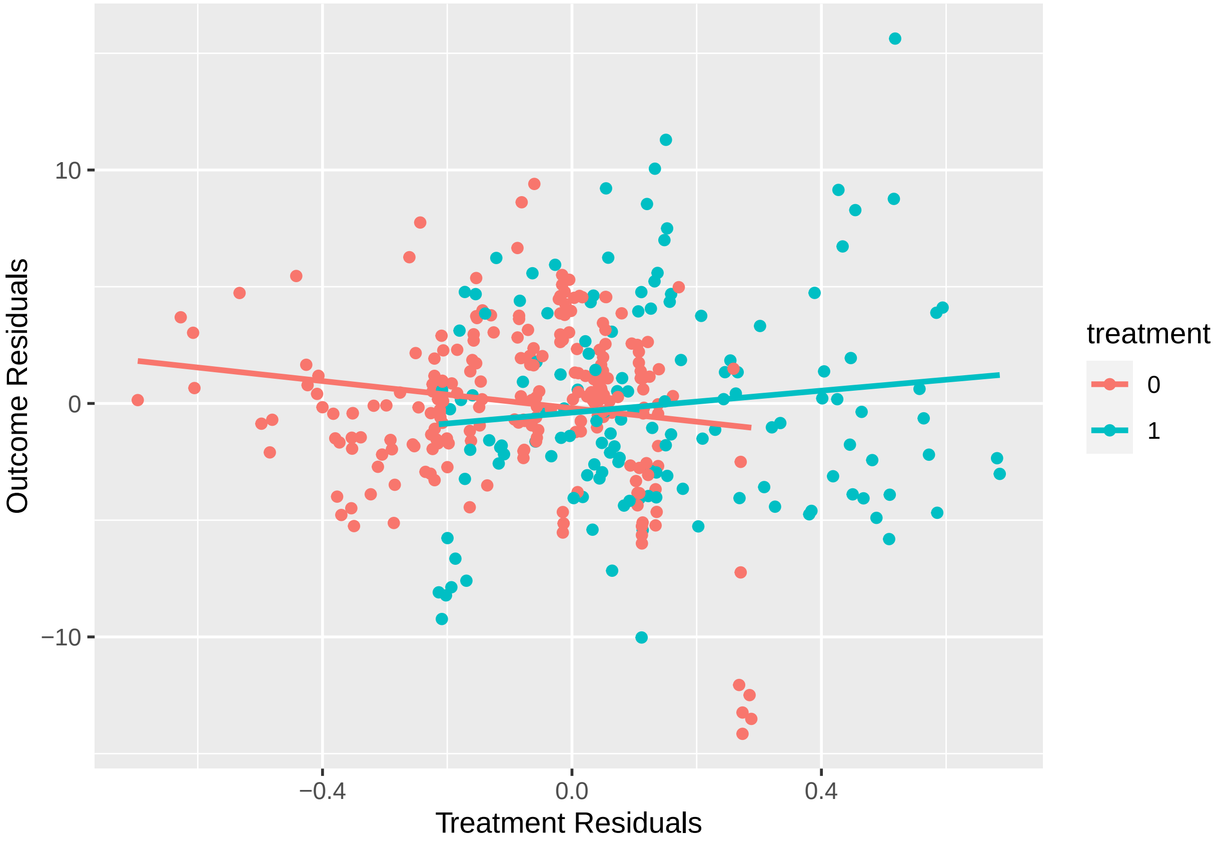


```
secondary %>%
    mutate(treatment = as.character(treatment)) %>%
    ggplot(aes(residuals, outcome_residuals, color = treatment)) +
    geom_point() + geom_smooth(method = "lm", se = F) + xlab("Treatment Residuals") +
    ylab("Outcome Residuals")
```

Table 6: Part v

term	estimate	std.error	statistic	p.value	conf.low	conf.high	df	outcom
(Intercept)	0.3196319	0.8442571	0.3785954	0.7051537	-1.339213	1.978477	486	outcom
residuals	23.7607617	3.4197550	6.9480888	0.0000000	17.041432	30.480092	486	outcom
factor(treatment)1	0.3406158	1.4475339	0.2353076	0.8140691	-2.503582	3.184813	486	outcom
residuals:factor(treatment)1	-7.8060217	6.7788921	-1.1515188	0.2500850	-21.125576	5.513533	486	outcom

term	estimate	std.error	statistic	p.value	conf.low	conf.high	df	outcon
(Intercept)	-0.2017444	0.3023574	-0.6672382	0.5050416	-0.7963256	0.3928368	365	outcon
residuals	-2.9020489	1.5611265	-1.8589454	0.0638395	-5.9719802	0.1678823	365	outcon
factor(treatment)1	-0.1888159	0.5319994	-0.3549175	0.7228564	-1.2349844	0.8573527	365	outcon
residuals:factor(treatment)1	5.2480474	2.4404743	2.1504211	0.0321782	0.4488924	10.0472025	365	outcon



```r
msp <- lm_robust(outcome_residuals ~ residuals * factor(treatment),
    data = primary) %>%
    tidy()

mss <- lm_robust(outcome_residuals ~ residuals * factor(treatment),
    data = secondary) %>%
    tidy()

knitr::kable(list(msp, mss), caption = "Part v")
```

Yes for the secondary schools there is a significant interaction. This suggests that there is a heterogeneous treatment effect for this subset of data and the treated country years with negative weights are biasing the result substantially. We should therefore be worried!

e) Conceptually, in your own words, what robustness checks does Jakiela recommend and why?

Exclude later years, limit the number of post-treatment years, exclude individual observations. A full answer would define these in the student's own words