002: Regression and matching

EC 607

Solutions

01. Download and load two datasets: (1) data from the randomized employment program (we'll call this the **NSW data**) and (2) data on 2,490 potential 'control' individuals from the PSID (Panel Study of Income Dynamics) (we'll call this the **PSID data**).

Answer:

```r
# Load packages
library(pacman)
p_load(
  ggplot2, gridExtra, ggthemes, latex2exp, kableExtra,
  dagitty, ggdag,
  tidyverse, broom, knitr, haven,
  estimatr, pdist, StatMatch, purrr,
  huxtable, magrittr
)
# Load LaLonde's experimental data
nsw_df = read_dta("http://users.nber.org/~rdehejia/data/nsw.dta")
# Load PSID potential controls
psid_df = read_dta("http://users.nber.org/~rdehejia/data/psid_controls.dta")
```

The last page of the problem set describes the variables in these data.

Note Questions 02–07 use the *NSW data*.

02. Regress real earnings in 1975 (the year before treatment) on treatment (and an intercept, which we will always assume should be included unless otherwise stated). Why/how is this regression (and its outcome) informative? What does it tell us?

Answer:

```r
lm_robust(re75 ~ treat, data = nsw_df) %>% huxreg()
```

	(1)
(Intercept)	3026.683 ***
	(252.298)
treat	39.415
	(379.037)
N	722
R2	0.000

*** p < 0.001; ** p < 0.01; * p < 0.05.

The results of this regression tell us whether there were significant differences between the treatment and control groups before the program began. Because the program (at some level) was randomized, we should find no significant differences. We find no significant difference in earnings before the program.

03. The program rolled out in 1976 and ended (at least for our purposes) in 1978, so we'll use earnings in 1978 to estimate whether the program had any sustained effect on earnings.

Regress 1978 earnings on treatment. What do you find?

Answer:

```
lm_robust(re78 ~ treat, data = nsw_df) %>% huxreg()
```

	(1)
(Intercept)	5090.048 ***
	(277.368)
treat	886.304
	(488.205)
N	722
R2	0.005
*** p < 0.001; ** p < 0.01; * p < 0.05.	

Answer We find marginally significant evidence (*p*-value of approximately 0.07) that earnings in 1978 (earnings at the end of the program) were higher for program participants. The estimated effet of the program is approximately $886 with a 95% confidence interval [-$72, $1,844].

04. What is required for us to interpret the estimated in **03.** as causal? Does our setting meet this requirement?

Answer: To interpret these estimates as *causal*, we must believe that treatment was randomly assigned. In other words, we must believe that potential outcomes Y_{0i} and Y_{1i} are uncorrelated with treatment.

Note: Because we are not conditioning on covariates, we must believe that treatment is exogenous (rather than conditionally exogenous).

05. Add controls for age, education, race (black and Hispanic). How does your estimated treatment effect and its standard change. Why do you think this happense?

Answer: Our point estimate decreases slightly—and is less significant (*p*-value of approx. 0.1).

```
lm_robust(
  re78 ~ treat + age + education + black + hispanic + married + nodegree,
  data = nsw_df
) %>% huxtable::huxreg()
```

	(1)
(Intercept)	4268.577
	(2631.003)
treat	793.609
	(487.142)
age	20.105
	(35.062)
education	205.879
	(166.752)
black	-1765.638 *
	(776.401)
hispanic	-133.947
	(992.171)
married	540.991
	(681.557)
nodegree	-522.315
	(757.487)
N	722
R2	0.025

*** p < 0.001; ** p < 0.01; * p < 0.05.

06. What is a "bad control"? Are any of the controls we added in **05.** "bad"? Briefly explain.

Answer: A "bad control" is a covariate that is affected by treatment. The best controls are "fixed" prior to treatment, which seems to be the case for all of our control (with one potentical exception for marriage).

07. Since we have an experiment, can we interpret the coefficient on `nodegree` (not having a high-school diploma) as causal? What about its interaction with treatment? Briefly explain.

Answer: Nope. We would need degree status to be as-good-as randomly assigned (uncorrelated with potential outcomes), which is likely not the case. Lots of things are correlated with degree status (think omitted-variable bias). Same issue with the interaction.

08. Compare a simple difference in means to your results in the regression results in **03.**.

Hint The `dplyr` functions `group_by()` and `summarize()` could be helpful.

Answer:

```
nsw_df %>%
  group_by(Trt = treat) %>%
  summarize("Mean '78 Earnings" = mean(re78)) %>%
  hux(add_colnames = T) %>%
  set_number_format(0) %>%
  set_bold(1, everywhere, T) %>%
  set_bottom_border(1, everywhere)
```

Trt	**Mean '78 Earnings**
0	5090
1	5976

The difference in means here is exactly equal to the point estimate from 03.

09. Create a new dataset that combines **treated individuals from the *NSW data*** and **control individuals from the *PSID data***. We'll refer to this dataset as our **mixed dataset**.

Hint Remember our old friends `filter()` and `bind_rows()` from `dplyr`.

Answer:

```
mixed_df = bind_rows(
  filter(nsw_df, treat == 1),
  psid_df
)
```

Note **Questions from 10–13 use this *mixed dataset***, focusing on earnings in 1978.

10. Compare the difference in means from the **mixed dataset** to the difference from the **NSW dataset**.

Answer:

```r
# Group means
group_means = left_join(
  nsw_df %>% group_by(treat) %>% summarize(NSW = mean(re78)),
  mixed_df %>% group_by(treat) %>% summarize(Mixed = mean(re78)),
  by = "treat"
) %>% rename(Trt = treat) %>% as.data.frame()
# Difference
group_means[3,] = apply(X = group_means, FUN = diff, MARGIN = 2)
group_means %<>% mutate("Comparison" = c("Ctrl", "Trt", "Diff"))
# Results
group_means %>%
  select(-1) %>%
  hux(add_colnames = T) %>%
  set_number_format(everywhere, 1:2, fmt_pretty(digits = 0)) %>%
  set_bottom_border(1, everywhere) %>%
  set_bold(c(1,4), everywhere, T)
```

NSW	Mixed	Comparison
5,090	21,554	Ctrl
5,976	5,976	Trt
886	**-15,578**	**Diff**

In the *mixed* dataset, the treatment group mean is much smaller than that the control group—the difference is negative, as opposed the positive difference between treatment and control in the NSW dataset.

11. Use our potential-outcomes (Rubin causal model) notation to explain how the difference with the mixed dataset may be biased. Does the sign of the difference across the two differences-in-means match what you would expect from our model of selection bias? Briefly explain.

Answer: We should be concerned that program participation correlates with potential outcomes (since the program targeted individuals with low expected employment outcomes). So we're concerned that D_i (participation in the program) correlates negatively with Y_{0i}, meaning individuals who are not in the program likely have higher expected Y_{0i} outcomes than folks who are in the program. Thus, we will have selection bias equal to

$$E[\mathrm{Y}_{0i}|\mathrm{D}_i = 1] - E[\mathrm{Y}_{0i}|\mathrm{D}_i = 0] < 0$$

If the selection bias is sufficiently larget, it will flip the sign of a positive treatment effect, as we likely observe in the *mixed* dataset.

12. Time for nearest-neighbor matching. Use all six covariates. **You must write write the code for this matching yourself.** You *can* use basic pre-written functions (*e.g.*, `mean`), but do not use matching-estimation packages that do all of the matching for you.

> **12A.** Estimate the average treatment effect on the treated by matching treated individuals to their nearest neighbor using a **Euclidean** metric.

Answer:

```r
# Create covariate matrices of treatment and control
trt = mixed_df %>% filter(treat == 1)
ctrl = mixed_df %>% filter(treat == 0)
trt_cov = trt %>% select(age, education, black, hispanic, married, nodegree)
ctrl_cov = ctrl %>% select(age, education, black, hispanic, married, nodegree)
# Calculate Euclidean distances using pdist()
dist_mat = pdist(X = trt_cov, Y = ctrl_cov) %>% as.matrix()
# Calculate individual-level treatment effects from nearest neighbor(s)
trt_12a = map_dbl(1:nrow(trt), function(i) {
  # Find which individuals are the nearest neighbors
  i_nn = dist_mat[i,] == min(dist_mat[i,], na.rm = T)
  # Return the difference (allow for ties; take the mean)
  trt$re78[i] - mean(ctrl$re78[i_nn], na.rm = T)
})
```

Based upon Euclidean-distance-based nearest neighbor matching, our estimate for the treatment effect on the treated is approximately -6,781.

> **12B.** Estimate the average treatment effect on the treated by matching treated individuals to their nearest neighbor using a **Mahalanobis** metric. *Hint:* The `StatMatch` package has a function named `mahalanobis.dist()` that finds the *Mahalanobis distance* between observations of two datasets.

Answer:

```r
# Calculate Mahalanobis distances
mdist_mat = mahalanobis.dist(data.x = trt_cov, data.y = ctrl_cov) %>% as.matrix()
# Calculate individual-level treatment effects from nearest neighbor(s)
trt_12b = map_dbl(1:nrow(trt), function(i) {
  # Find which individuals are the nearest neighbors
  i_nn = mdist_mat[i,] == min(mdist_mat[i,], na.rm = T)
  # Return the difference (allow for ties; take the mean)
  trt$re78[i] - mean(ctrl$re78[i_nn], na.rm = T)
})
```

Based upon Mahalanobis-distance-based nearest neighbor matching, our estimate for the treatment effect on the treated is approximately -3,986.

> **12C.** How do your estimates in **12A.** and **12B.** compare to your previous estimates?

Answer: Our experiment-based estimate was a positive (approximately) $886. The simple difference in means on the *mixed data* resulted in -$15,578. Matching on covariates via the Euclidean distance moved us to -$6,781, and matching on Mahalanobis distance moved us to -$3,986. So things seem to be getting better (if we take the NSW experiment-based estimate as truth), but we're still getting estimates with the wrong magnitude and sign—still likely biased from selection.

> **Extra credit** Use kernel matching (any kernel) to estimate the treatment effect.

13. Now for propensity-score methods. **Again: Do not use pre-written propensity-score packages. Write the code for the steps.** You *can* use the `glm()` function to estimate a logit regression (more below in **13A**).

> **13A.** Estimate the propensity score for each treated individual using the covariates using a logit model that is linear in the covariates. Which variables are predictive of treatment?
>
> *Hint:* The function `glm()` with `family = binomial` estimates a logit model.

Answer:

```
# Estimate logit model for propensity scores, linear in covariates
pscore_logit = glm(
  treat ~ age + education + black + hispanic + married + nodegree,
  family = "binomial",
  data = mixed_df
)
pscore_logit %>% huxreg()
```

	(1)
(Intercept)	0.643
	(0.872)
age	-0.101 ***
	(0.012)
education	-0.044
	(0.054)
black	2.170 ***
	(0.227)
hispanic	2.404 ***
	(0.383)
married	-2.567 ***
	(0.190)
nodegree	1.302 ***
	(0.254)
N	2787
logLik	-445.193
AIC	904.385

*** p < 0.001; ** p < 0.01; * p < 0.05.

Each covariate, excluding education, appears to be significantly predictive of treatment.

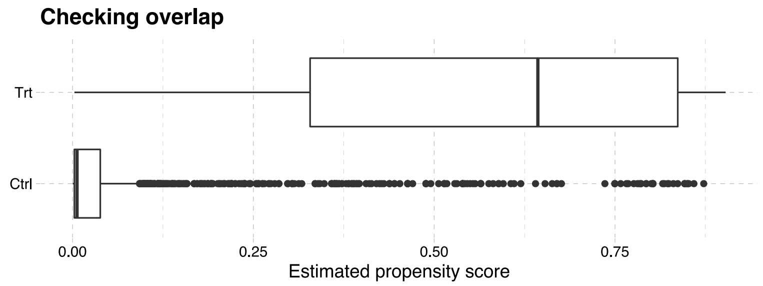
13B. Add the estimated propensity scores $(\hat{p}_i)$ to the mixed dataset. Is there overlap? Explain.

Hint: You can access predictions from a model using `$fitted.values`.
Another hint: Try histograms grouped/filled by treatment status.

Answer:

We do not have overlap: The minimum p in control is less than the minimum in the treatment group. The maximum p in the treatment group exceeds the maximum in the control group. That said, it's not too bad.

```r
# Add propensity scores
mixed_df$p_score = pscore_logit$fitted.values
# Plot overlap
ggplot(
  data = mixed_df,
  aes(x = factor(treat, labels = c("Ctrl", "Trt")), y = p_score)
) +
geom_boxplot() +
theme_pander() +
xlab("") +
ylab("Estimated propensity score") +
ggtitle("Checking overlap") +
coord_flip()
```



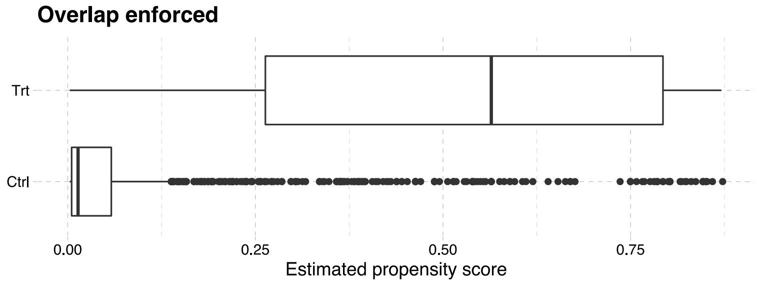
```r
# Summary table
mixed_df %>%
  group_by(treat) %>%
  summarize(min(p_score), max(p_score)) %>%
  hux(add_colnames = T)
```

treat	min(p_score)	max(p_score)
0	0.000262	0.873
1	0.00261	0.903

13C. Enforce overlap using the minimum $\hat{p}_i$ observed in the treated group and the maximum $\hat{p}_i$ observed in the control group.

Answer:

```r
# Find min p-score in trt and max in control
lo_pscore = filter(mixed_df, treat == 1)$p_score %>% min()
hi_pscore = filter(mixed_df, treat == 0)$p_score %>% max()
# New dataset with overlap enforced
overlap_df = filter(mixed_df, between(p_score, lo_pscore, hi_pscore))
# Plot overlap
ggplot(
  data = overlap_df,
  aes(x = factor(treat, labels = c("Ctrl", "Trt")), y = p_score)
) +
geom_boxplot() +
theme_pander() +
xlab("") +
ylab("Estimated propensity score") +
ggtitle("Overlap enforced") +
coord_flip()
```



13D. Estimate the treatment effect using a regression that conditions on $\hat{p}_i$. What happens if you also include $\hat{p}_i$ interacted with treatment?

Answer:

```
# Conditioning on estimated propensity score
reg_13d_1 = lm_robust(re78 ~ treat + p_score, data = overlap_df)
# Conditioning on estimated propensity score and interacted trt.
reg_13d_2 = lm_robust(re78 ~ treat * p_score, data = overlap_df)
# Results
huxreg(reg_13d_1, reg_13d_2)
```

	(1)	(2)
(Intercept)	20737.722 ***	21190.568 ***
	(333.337)	(346.729)
treat	-7039.786 ***	-13660.094 ***
	(730.901)	(959.454)
p_score	-14388.692 ***	-20829.885 ***
	(1100.947)	(1395.096)
treat:p_score		18313.371 ***
		(1995.966)
N	2119	2119
R2	0.139	0.151

*** p < 0.001; ** p < 0.01; * p < 0.05.

We're back to large, negative, and statistically significant estimates (for both specifications).

13E. Now estimate the treatment effect by blocking on $\hat{p}_i$.

Answer: I'm going to opt for 20 blocks

```r
# Define how many blocks we want
n_blocks = 20
# Find the spacing that will give us n_blocks blocks
block_step = (hi_pscore - lo_pscore) / n_blocks
# Create the blocks' breaks
block_breaks = seq(from = lo_pscore, to = hi_pscore, by = block_step)
# Cut the p-scores into 10, equally spaced blocks (using breaks above)
overlap_df %<>% mutate(
  block = cut(p_score, breaks = block_breaks, labels = F, include.lowest = T)
)
# Iterate over blocks, calculating (1) the trt. effect and (2) N
est_blocks = map_dfr(1:20, function(b) {
  # Subset to the block's data
  block_df = overlap_df %>% filter(block == b)
  # Estimate the block's treatment effect
  t_b = mean(filter(block_df, treat == 1)$re78) - mean(filter(block_df, treat == 0)$re78)
  # The number of individuals in block b
  n_b = nrow(block_df)
  # Return a data.frame of the two answers
  data.frame(est = t_b, n = n_b)
})
# Weighted average of the individual blocks' treatment effects
t_block = weighted.mean(
  x = est_blocks$est,
  w = est_blocks$n
)
```

Blocking on propensity scores (20 blocks), we estimate a treatment effect of approximately -$11,394.32.

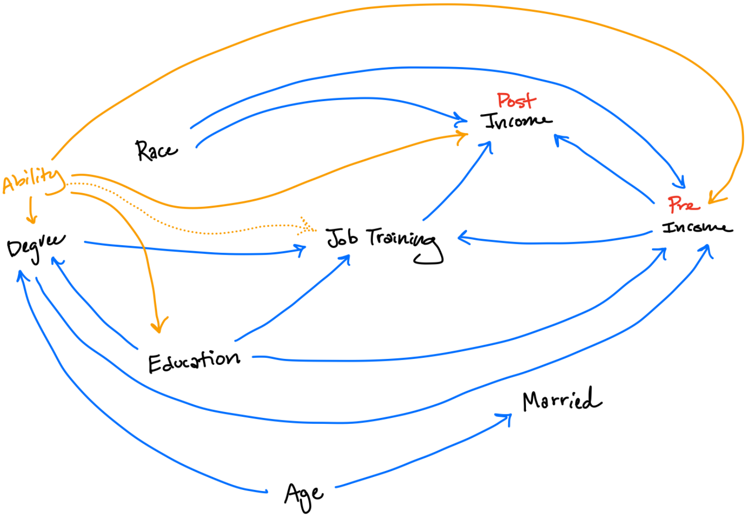
Extra credit Use the *doubly robust method* that combines regression and blocking.

14. Draw a DAG for this setting. Compare the various treatment effects that you've estimated in **10–13**. How do they compare to the effects you estimated **03.**? Which controls does your DAG suggest? (Explain.) Which estimates should we trust? Why?

Answer: Your call here... just make sure you justify your answer well.

An example: Here's a pretty complicated[†] DAG where I've connected all of our variables **plus** the classic omitted variable—ability. I dotted the line between ability and job training to highlight that this link will be important and is determined by "how random" treatment assignment is. There are a lot of decisions here that we would want to revisit/reconsider, *e.g.*, does race affect entry into the program, level of education, degree status, marital status, *etc.*?

```
knitr::include_graphics(
  "ex-dag.png"
)
```



[†]: It turns out that real life is complicated.

Data description

Variable	Description
`data_id`	Dataset identifier.
`treat`	Treatment indicator (select to be part of NSW).
`age`	Age (years).
`education`	Education (years).
`black`	Indicator for whether the individual is black.
`hispanic`	Indicator for whether the individual is Hispanic.
`married`	Indicator for whether the individual is married.
`nodegree`	Indicator for individuals without a high-school diploma.
`re74`	Real earnings in 1974 (1982 dollars).
`re75`	Real earnings in 1975 (1982 dollars).
`re78`	Real earnings in 1978 (1982 dollars).

Note: The NSW dataset does not include `re74`.