m10

Will Tirone

Q1)

Looking at the summary output below, the columns `age`, `nodegree`, `married`, `re74`, and `re75` appear the most unbalanced. Really, the only balanced characteristic is `educ`.

```
m = matchit(treat ~ age + educ + married +
              nodegree + re74 + re75,
            data = data,
            method = NULL,
            distance = "glm")

summary(m)
```

```
Call:
matchit(formula = treat ~ age + educ + married + nodegree + re74 +
    re75, data = data, method = NULL, distance = "glm")

Summary of Balance for All Data:
         Means Treated Means Control Std. Mean Diff. Var. Ratio eCDF Mean
distance        0.3951        0.2609          0.9490     0.7550    0.2314
age            25.8162       28.0303         -0.3094     0.4400    0.0813
educ           10.3459       10.2354          0.0550     0.4959    0.0347
married         0.1892        0.5128         -0.8263          .    0.3236
nodegree        0.7081        0.5967          0.2450          .    0.1114
re74         2095.5737     5619.2365         -0.7211     0.5181    0.2248
re75         1532.0553     2466.4844         -0.2903     0.9563    0.1342
         eCDF Max
distance   0.3700
age        0.1577
```

```
educ       0.1114
married    0.3236
nodegree   0.1114
re74       0.4470
re75       0.2876

Sample Sizes:
          Control Treated
All           429     185
Matched       429     185
Unmatched       0       0
Discarded       0       0
```

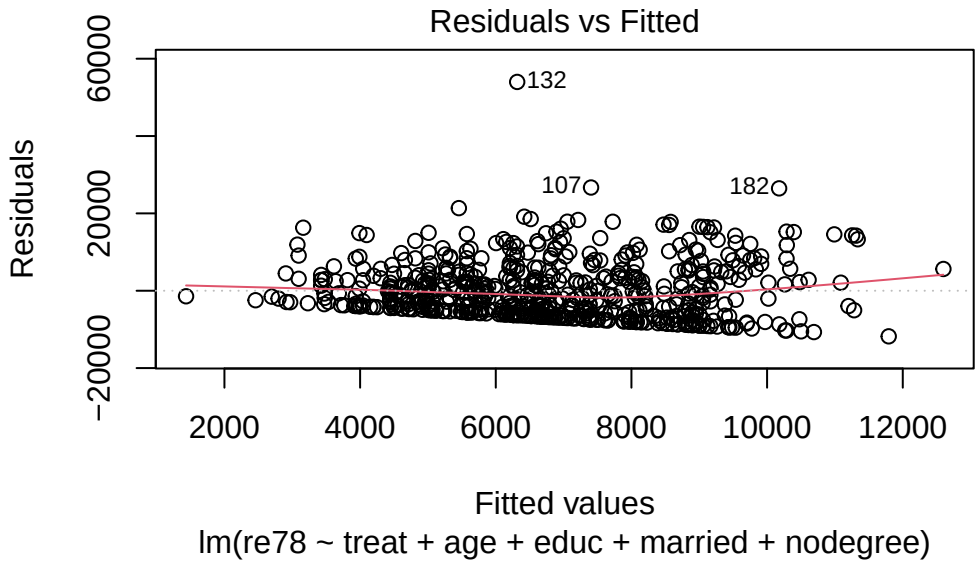
Q2)

Looking at the initial fit of the model, we see evidence of heteroskedasticity and maybe slight deviance from normality in the QQ plot. The deviance from normality is probably less of a concern, but the errors fan out pretty significantly and have structure.

However, we can't use log transforms on most of the variables since they're either binary or contain 0 values. However, removing the variables `re74` and `re75` seem to help in removing structure in the residuals. I imagine the high number of 0 values for these two income covariates are hurting the model.

```
m2 = lm(re78 ~ treat + age + educ + married + nodegree,
        data = data)

plot(m2, which=1)
```



Coefficient estimates for covariates (including treatment) are below:

```
coef(summary(m2))
```

```
              Estimate Std. Error    t value     Pr(>|t|)
(Intercept) -879.68944 2461.92564 -0.3573176 0.7209780239
treat        207.34950  676.46413  0.3065196 0.7593140105
age           53.48967   32.78567  1.6314952 0.1033036834
educ         510.22998  162.77213  3.1346275 0.0018036618
married     2369.16510  670.79021  3.5319017 0.0004438416
nodegree    -122.08533  884.01628 -0.1381030 0.8902047117
```

And the confidence intervals are here, for the treatment effect as well as other covariates.

```
confint(m2)
```

```
                 2.5 %    97.5 %
(Intercept) -5714.59970 3955.2208
treat       -1121.14041 1535.8394
age           -10.89724  117.8766
educ          190.56613  829.8938
married      1051.81805 3686.5121
nodegree    -1858.18138 1614.0107
```

Q3)

a)

Printing the `matches` object, we see that it is a 1:1 NN match without replacement, with the propensity scores estimated with logistic regression using all the covariates / background characteristics.

Since I'm not using all the covariates, I'm only including the ones used in the final model from Q2.

```
matches = matchit(treat ~ age + educ + married +
                    nodegree,
                  data = data,
                  method = "nearest",
                  distance = "glm")

matches
```

```
A matchit object
 - method: 1:1 nearest neighbor matching without replacement
 - distance: Propensity score
             - estimated with logistic regression
 - number of obs.: 614 (original), 370 (matched)
 - target estimand: ATT
 - covariates: age, educ, married, nodegree
```

b)

Here I summarize by including the "matched" and "unmatched" background covariate summaries. The difference is enormous! Matching makes a huge difference and we can see the background characteristics are significantly more balanced after matching.

```
bind_rows(
  data.frame(summary(matches)$sum.matched)[,1:3] |>
    mutate(type = "matched"),
  data.frame(summary(matches)$sum.all)[,1:3] |>
    mutate(type = "unmatched"),
) |> kable()
```

	Means.Treated	Means.Control	Std..Mean.Diff.	type
distance...1	0.3772215	0.3730423	0.0322534	matched
age...2	25.8162162	24.7513514	0.1488277	matched
educ...3	10.3459459	10.5891892	-0.1209774	matched
married...4	0.1891892	0.1891892	0.0000000	matched
nodegree...5	0.7081081	0.6702703	0.0832272	matched
distance...6	0.3772215	0.2685642	0.8385695	unmatched
age...7	25.8162162	28.0303030	-0.3094453	unmatched
educ...8	10.3459459	10.2354312	0.0549647	unmatched
married...9	0.1891892	0.5128205	-0.8263093	unmatched
nodegree...10	0.7081081	0.5967366	0.2449702	unmatched

c)

Here using the data from part b) and the formula from the slides, we get a confidence interval of [-1276.74, 1680]. Since this covers 0, we're not confident that the treatment has any effect.

```
matched_data = match.data(matches)

treat_sum = matched_data |>
  filter(treat == 1) |>
  group_by() |>
  summarize(mean = mean(re78),
            std = var(re78)/n())

control_sum = matched_data |>
  filter(treat == 0) |>
  group_by() |>
  summarize(mean = mean(re78),
            std = var(re78)/n())

treat_SE = treat_sum |> pull(std)
contr_SE = control_sum |> pull(std)

y_bar_t = treat_sum |> pull(mean)
y_bar_c = control_sum |> pull(mean)

tau_hat = y_bar_t - y_bar_c

val = 1.96 * sqrt(treat_SE + contr_SE)
```

```
cat("Point estimate for treatment effect: ", tau_hat, "\n")
```

```
Point estimate for treatment effect:  201.6332
```

```
cat("95% Conf. Int for treatment effect: ", "[", tau_hat - val,
    ",", tau_hat + val, "]")
```

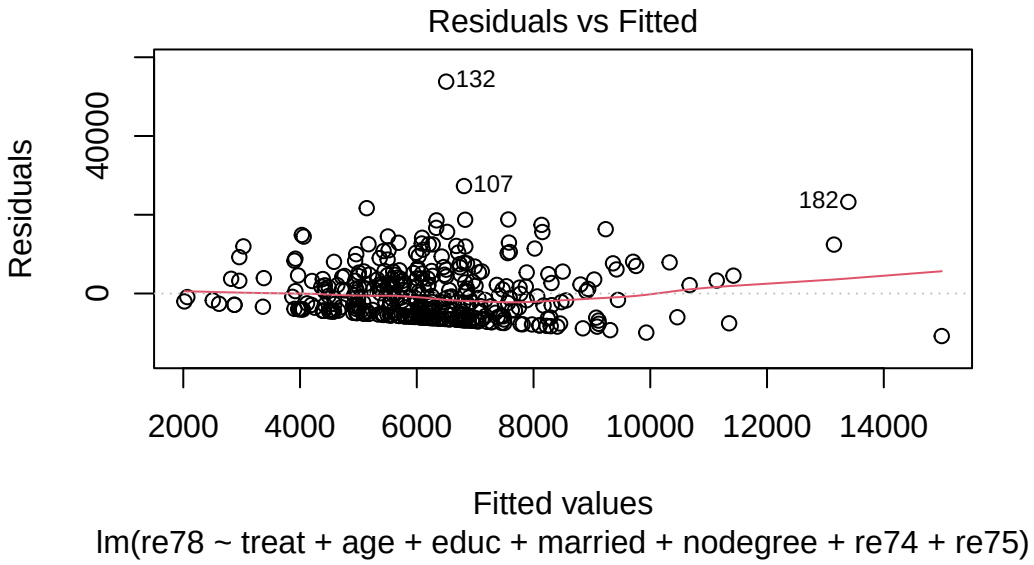
```
95% Conf. Int for treatment effect:  [ -1276.74 , 1680.006 ]
```

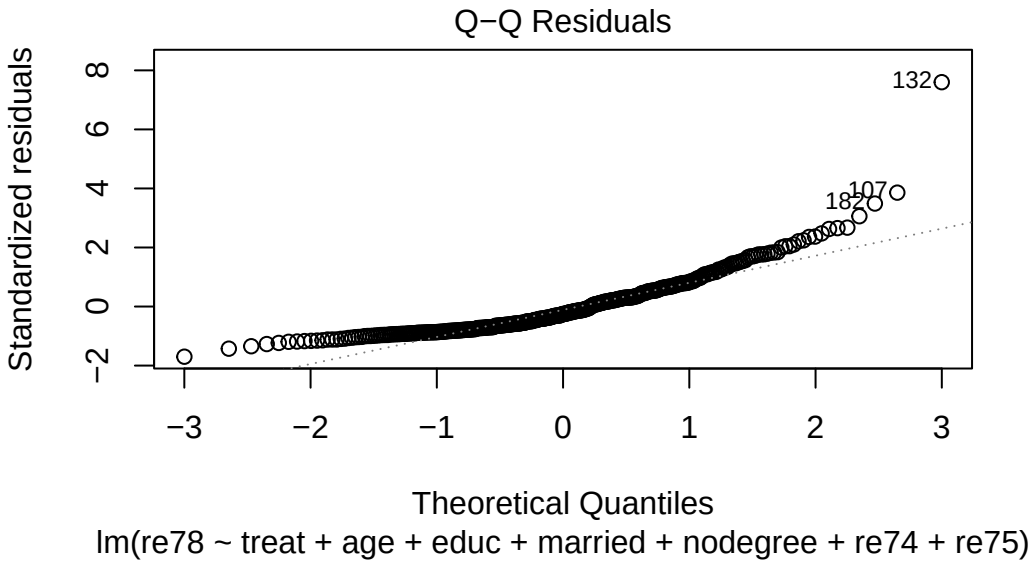
Q4)

Fitting the model again, this time it looks like `re74` and `re75` don't hurt the distribution of the errors nearly as much. However, we have a slightly worse looking normality plot. I've tried a handful of transforms and excluding different variables, but it doesn't seem like there's much we can do to improve this, it's probably just structure in the data.

```
m4 = lm(re78 ~ treat + age + educ + married + nodegree + re74 + re75,
        data = matched_data)

plot(m4, which = c(1,2))
```





Coefficient estimates for covariates (including treatment) are below:

```
coef(summary(m4))
```

```
                Estimate   Std. Error    t value   Pr(>|t|)
(Intercept) -879.4425237 3.522280e+03 -0.2496799 0.80297654
treat        468.8263015 7.546813e+02  0.6212243 0.53484305
age           16.5746948 4.799364e+01  0.3453519 0.73003045
educ         520.9247975 2.413505e+02  2.1583750 0.03155559
married      676.0180776 1.056538e+03  0.6398427 0.52267958
nodegree     351.0912084 1.140959e+03  0.3077158 0.75847556
re74           0.0601832 9.220672e-02  0.6526986 0.51436487
re75           0.2870522 1.601336e-01  1.7925799 0.07387496
```

And the confidence intervals are here, for the treatment effect as well as other covariates.

```
confint(m4)
```

```
                   2.5 %       97.5 %
(Intercept) -7.806144e+03 6047.2585989
treat       -1.015284e+03 1952.9362806
age         -7.780667e+01  110.9560595
educ         4.629975e+01  995.5498419
married     -1.401705e+03 2753.7408002
```

```
nodegree    -1.892650e+03 2594.8319486
re74        -1.211449e-01    0.2415113
re75        -2.785666e-02    0.6019611
```