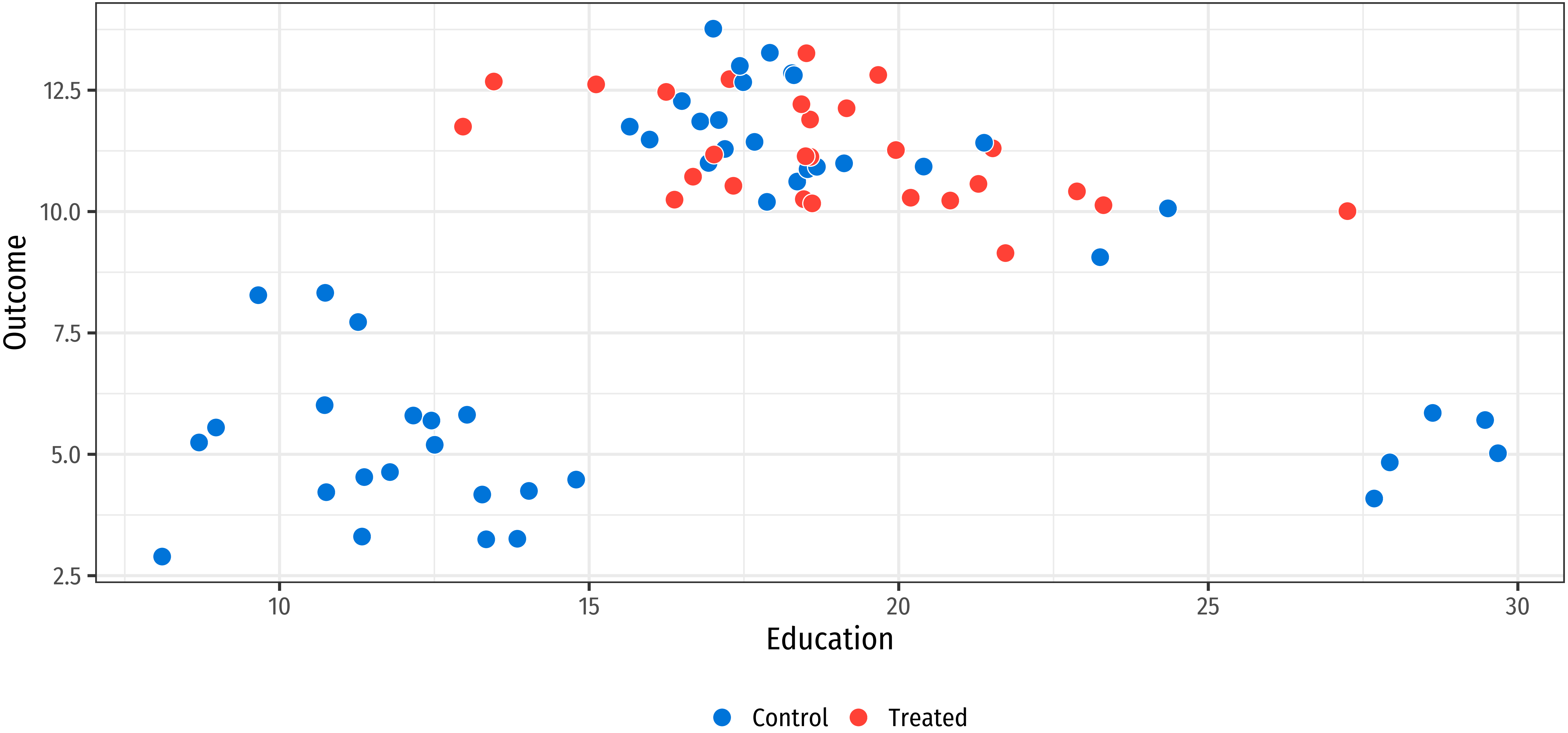
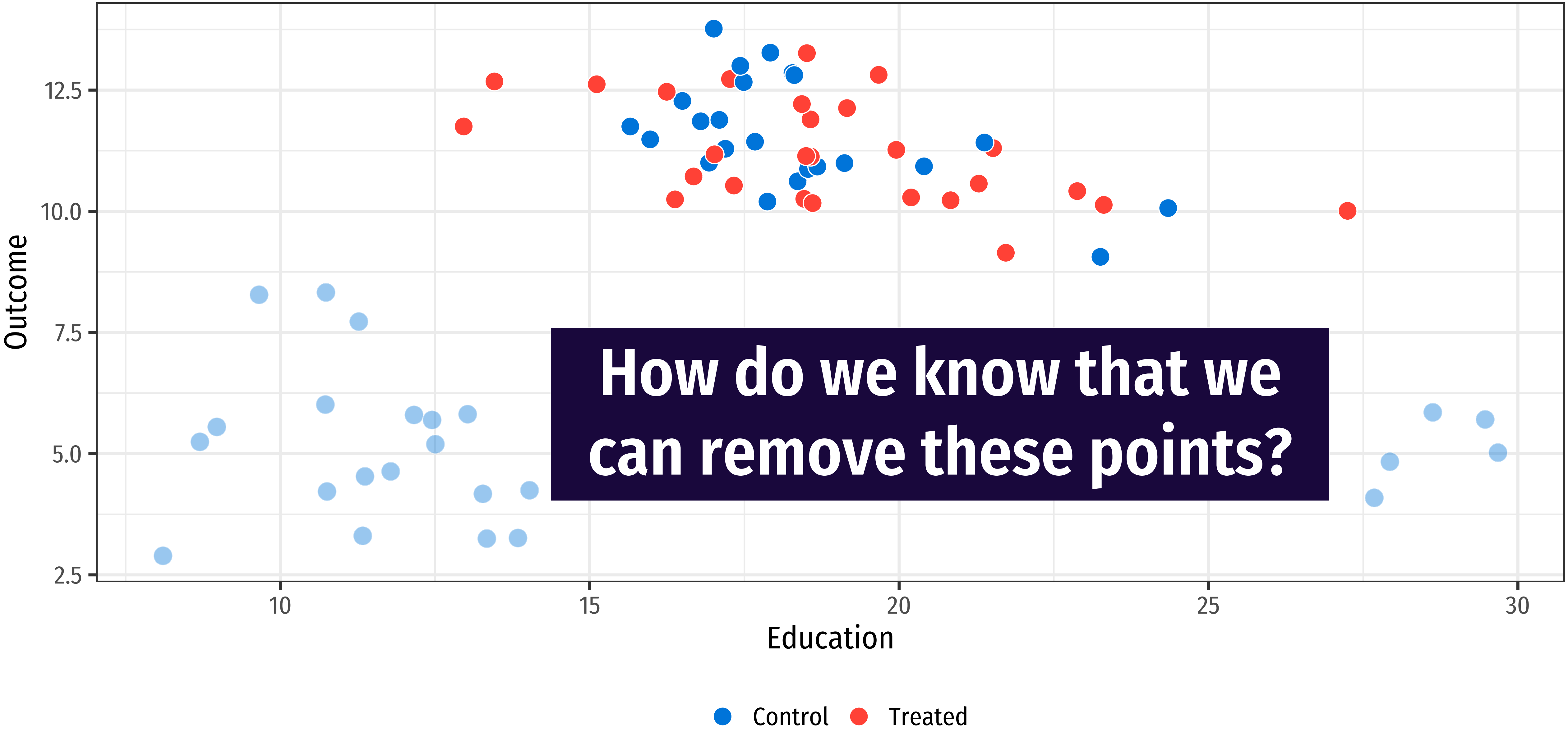
Causation, regression, and matching


Outcome
12.5
10.0
7.5
5.0
2.5
10
15
20
25
30
Education
● Control
● Treated



How do we know that we can remove these points?

General process for matching

1. Preprocess data

Do something to guess or model the assignment to treatment

Use what you know about the DAG to inform this!

2. Estimation

Use the new trimmed/preprocessed data to build a model, calculate difference in means, etc.

Different methods

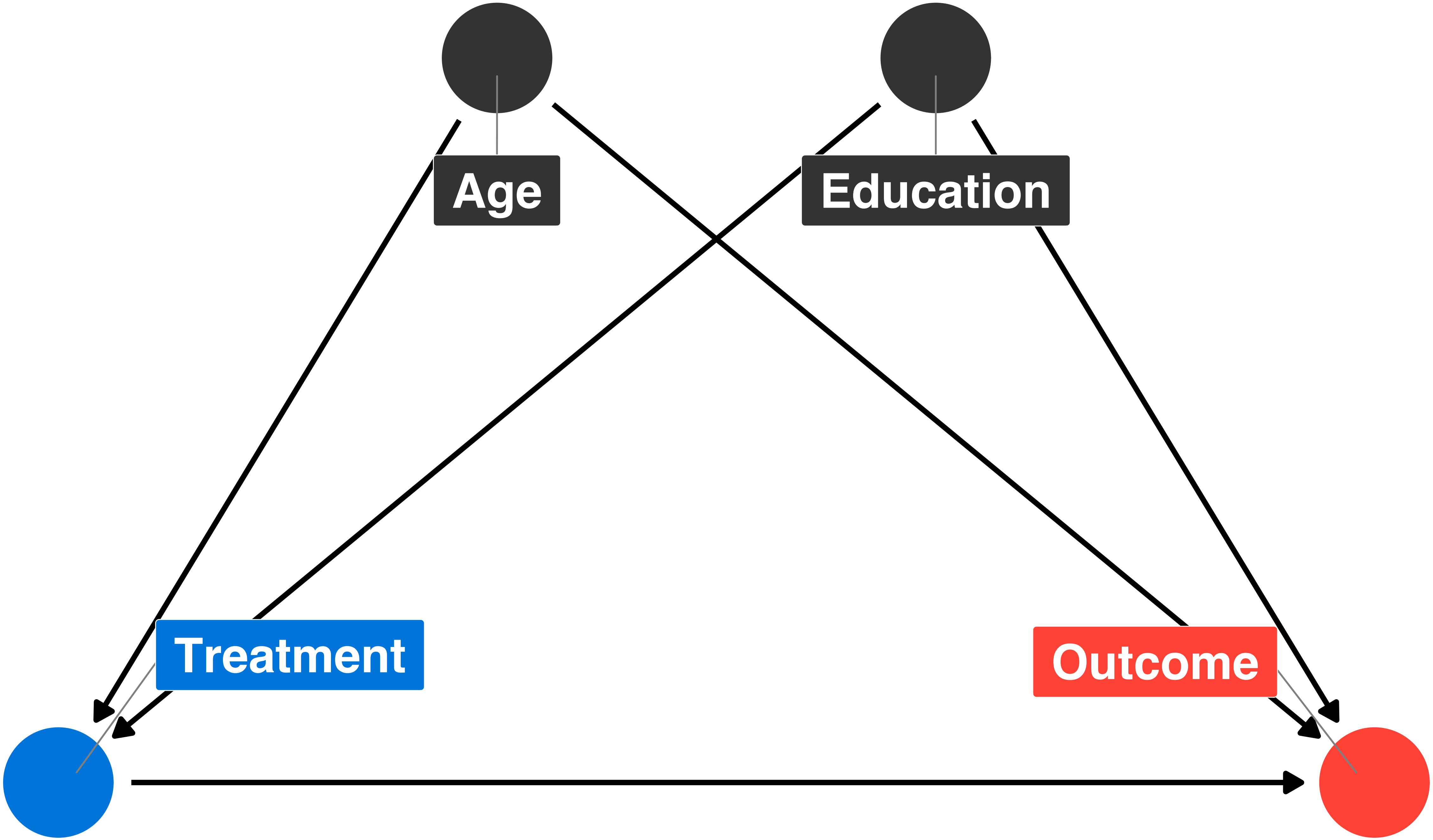
Nearest neighbor matching (NN)

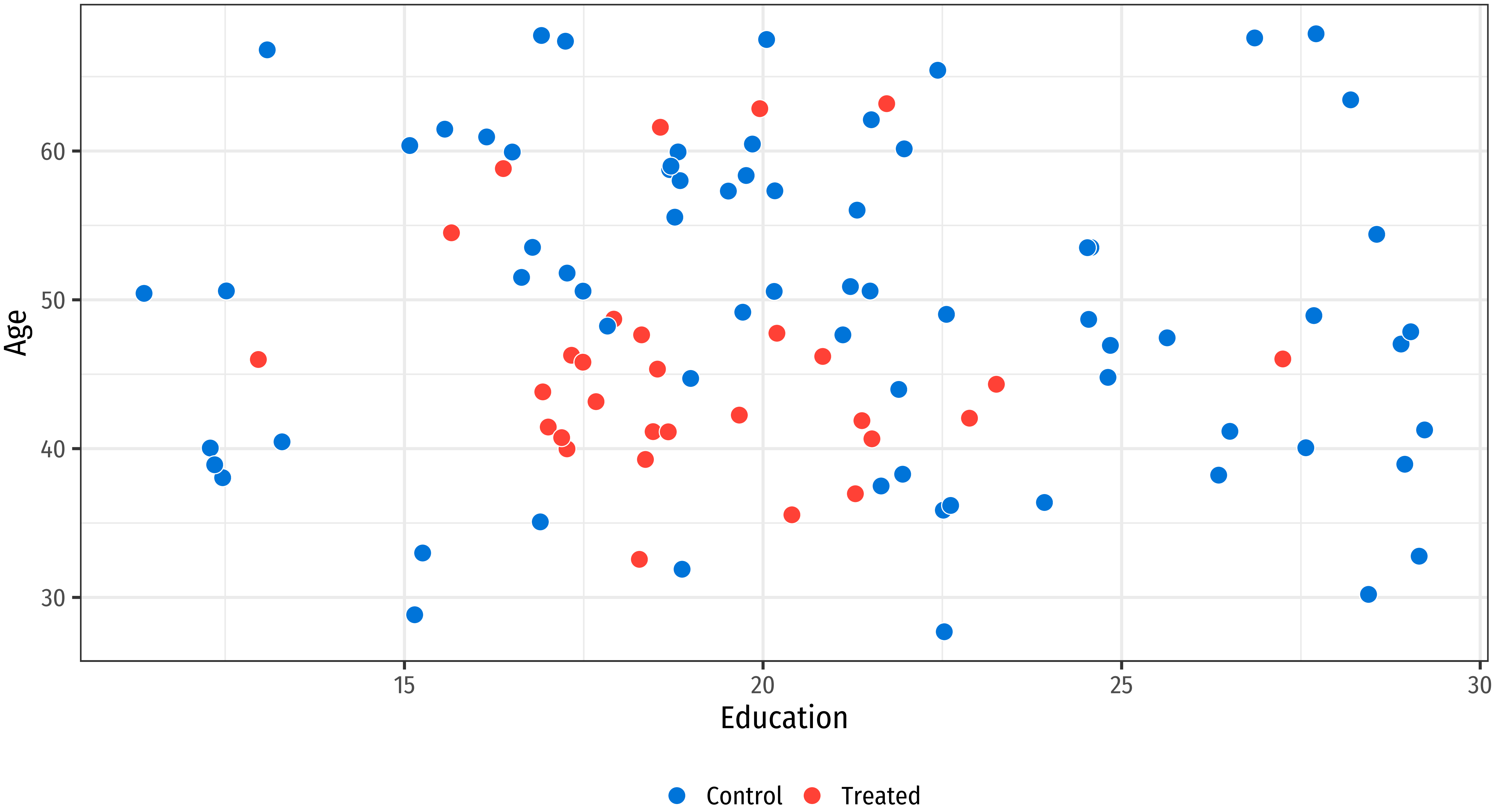
Mahalanobis distance / Euclidean distance

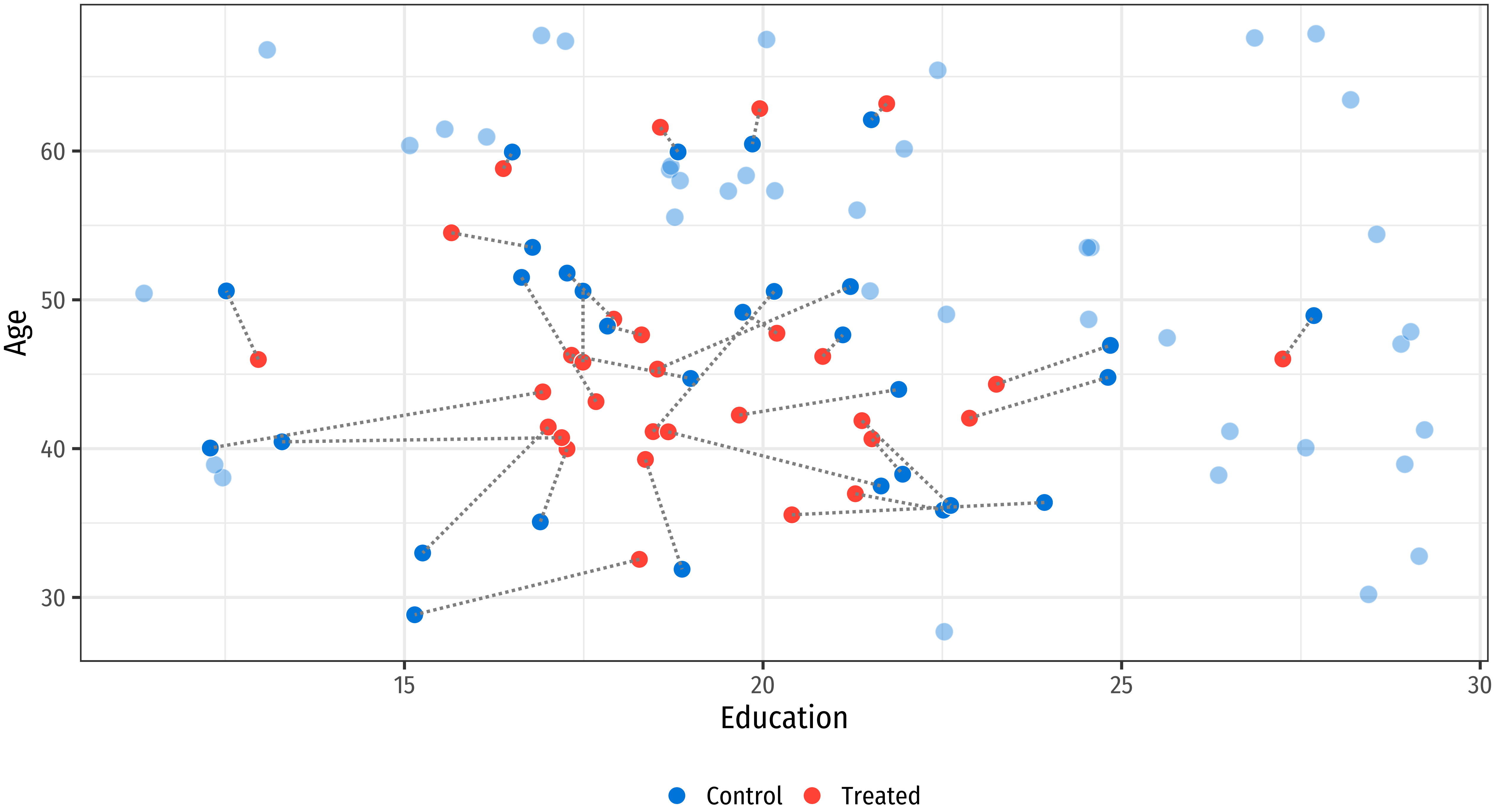
Inverse probability weighting (IPW)

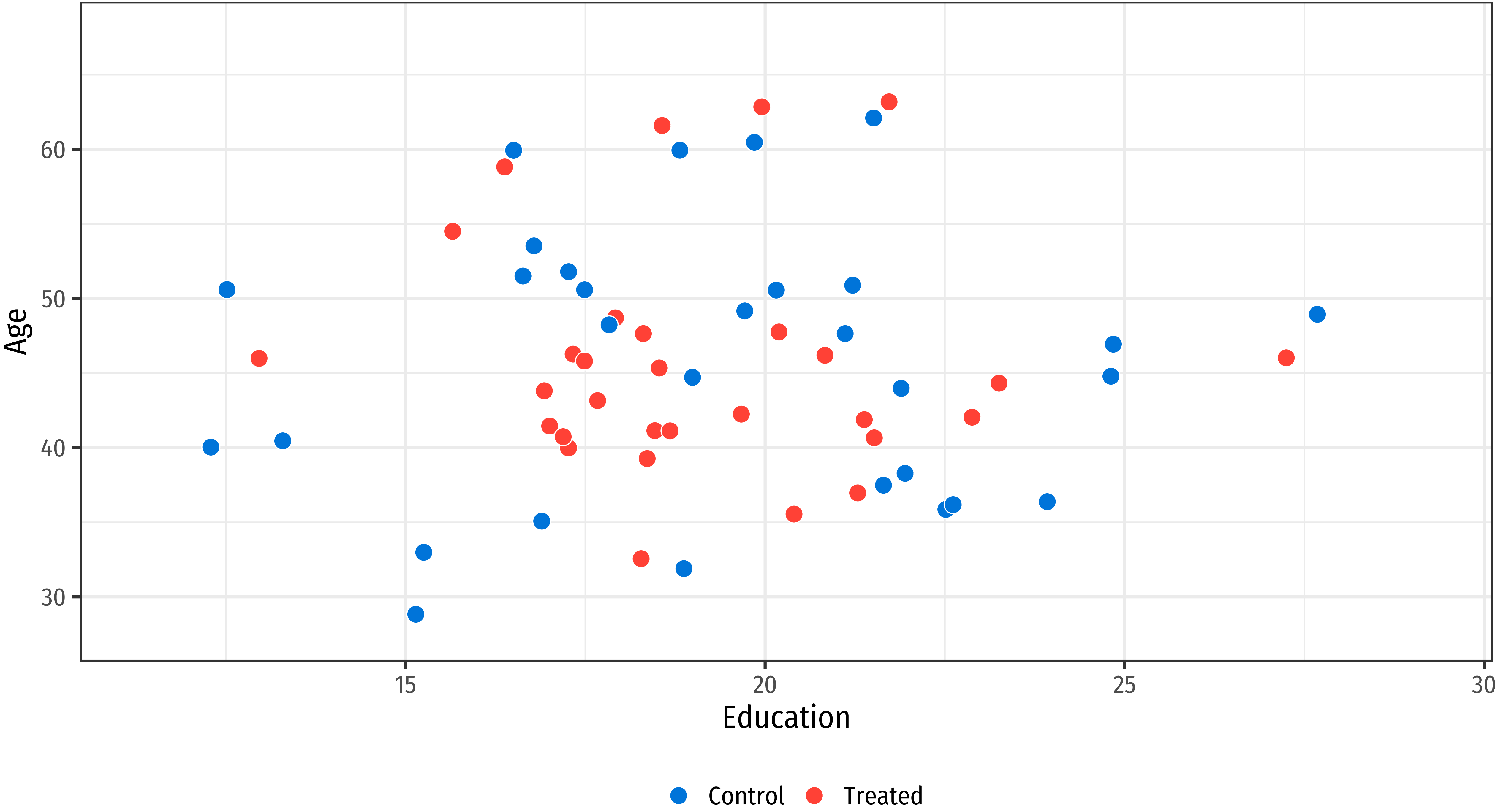
Nearest neighbor matching

Find control observations that are very close/similar to treatment observations based on confounders


Age
Education
Treatment
Outcome


Age
60
50
40
30
15
20
25
30
Education
● Control ● Treated


Age
60
50
40
30
Education
15
20
25
30
● Control ● Treated


Age
60
50
40
30
Education
15
20
25
30
● Control
● Treated

Propensity scores

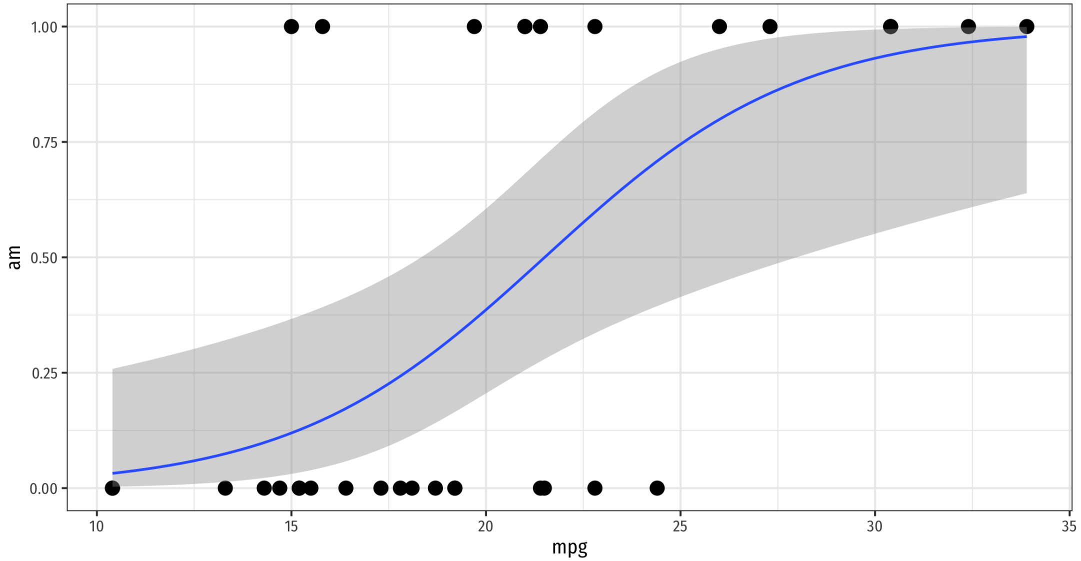
Predict the probability of assignment to treatment using a model

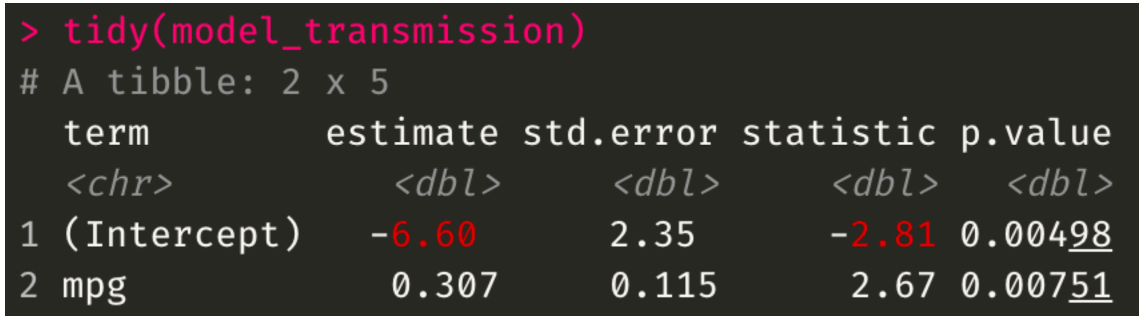
Logistic regression, probit regression, machine learning

$$\log \frac{p_{\text{Treatment}}}{1 - p_{\text{Treatment}}} = \beta_0 + \beta_1 \text{Education} + \beta_2 \text{Age}$$

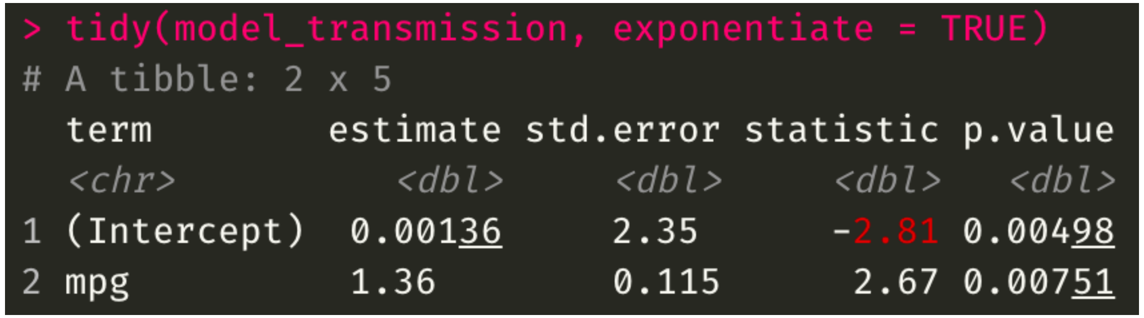
$$\log \frac{p_{\text{Manual}}}{1 - p_{\text{Manual}}} = \beta_0 + \beta_1 \text{MPG}$$

```r
model_transmission <- glm(am ~ mpg, data = mtcars, family = binomial(link = "logit"))
```





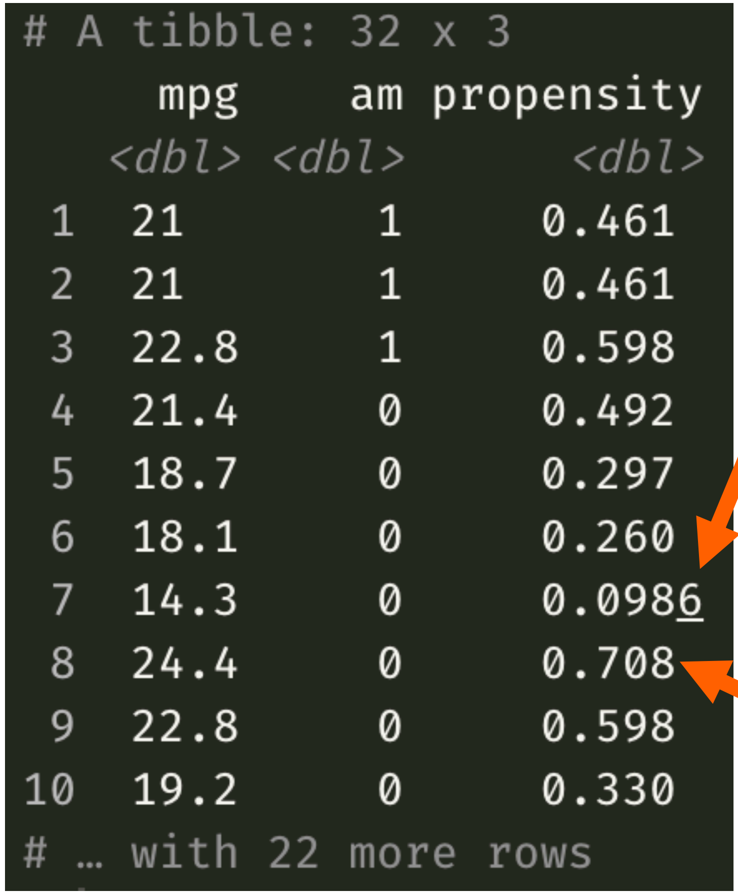
```
> tidy(model_transmission)
# A tibble: 2 x 5
  term        estimate std.error statistic p.value
  <chr>          <dbl>     <dbl>     <dbl>   <dbl>
1 (Intercept)   -6.60      2.35      -2.81 0.00498
2 mpg            0.307     0.115      2.67 0.00751
```



```
> tidy(model_transmission, exponentiate = TRUE)
# A tibble: 2 x 5
  term        estimate std.error statistic p.value
  <chr>          <dbl>     <dbl>     <dbl>   <dbl>
1 (Intercept)  0.00136     2.35      -2.81 0.00498
2 mpg          1.36        0.115      2.67 0.00751
```

Plug all the values of MPG into the model and find the predicted probability

```
augment(model_transmission, data = mtcars, type.predict ="response")
```



```
# A tibble: 32 x 3
     mpg    am propensity
   <dbl> <dbl>      <dbl>
 1  21       1     0.461
 2  21       1     0.461
 3  22.8     1     0.598
 4  21.4     0     0.492
 5  18.7     0     0.297
 6  18.1     0     0.260
 7  14.3     0     0.0986
 8  24.4     0     0.708
 9  22.8     0     0.598
10  19.2     0     0.330
# … with 22 more rows
```

Highly unlikely to be manual

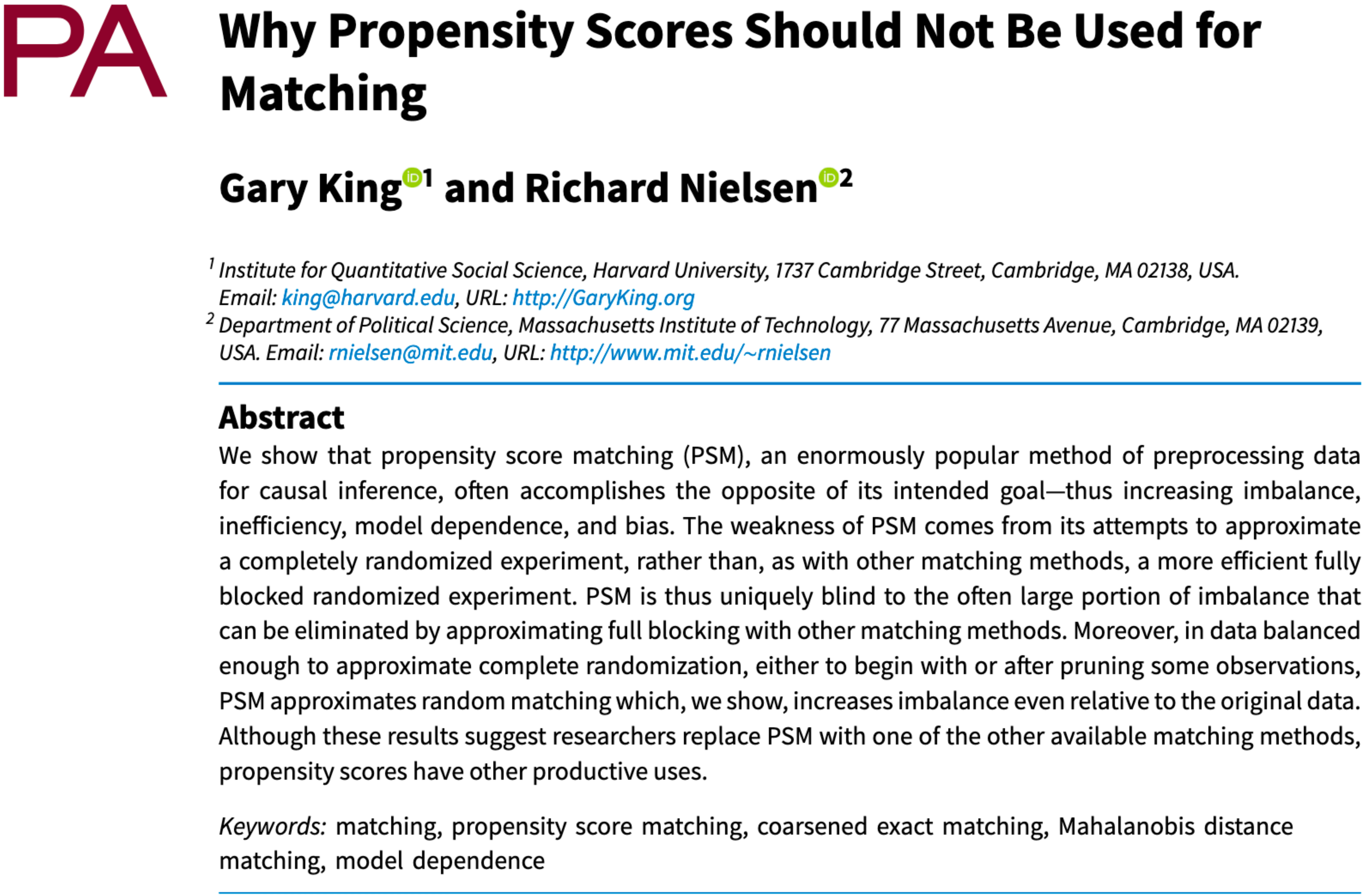
Highly likely to be manual (1)

Propensity score matching

Super popular method

There are mathy reasons why it's not great for matching

Propensity scores are fine!
Using them for matching isn't!



PA

Why Propensity Scores Should Not Be Used for Matching

Gary King[1] and Richard Nielsen[2]

[1] Institute for Quantitative Social Science, Harvard University, 1737 Cambridge Street, Cambridge, MA 02138, USA. Email: king@harvard.edu, URL: http://GaryKing.org

[2] Department of Political Science, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA 02139, USA. Email: rnielsen@mit.edu, URL: http://www.mit.edu/~rnielsen

Abstract

We show that propensity score matching (PSM), an enormously popular method of preprocessing data for causal inference, often accomplishes the opposite of its intended goal—thus increasing imbalance, inefficiency, model dependence, and bias. The weakness of PSM comes from its attempts to approximate a completely randomized experiment, rather than, as with other matching methods, a more efficient fully blocked randomized experiment. PSM is thus uniquely blind to the often large portion of imbalance that can be eliminated by approximating full blocking with other matching methods. Moreover, in data balanced enough to approximate complete randomization, either to begin with or after pruning some observations, PSM approximates random matching which, we show, increases imbalance even relative to the original data. Although these results suggest researchers replace PSM with one of the other available matching methods, propensity scores have other productive uses.

Keywords: matching, propensity score matching, coarsened exact matching, Mahalanobis distance matching, model dependence

https://www.youtube.com/watch?v=rBv39pK1iEs

Weighting in general

Make some observations more important than others

	Young	Middle	Old
Population	30%	40%	30%
Sample	60%	30%	10%

Weighting in general

Make some observations more important than others

	Young	Middle	Old
Population	30%	40%	30%
Sample	60%	30%	10%
Weight	30 / 60 = 0.5	40 / 30 = 1.333	30 / 10 = 3

Multiply weights by average values (or use in regression) to adjust for importance

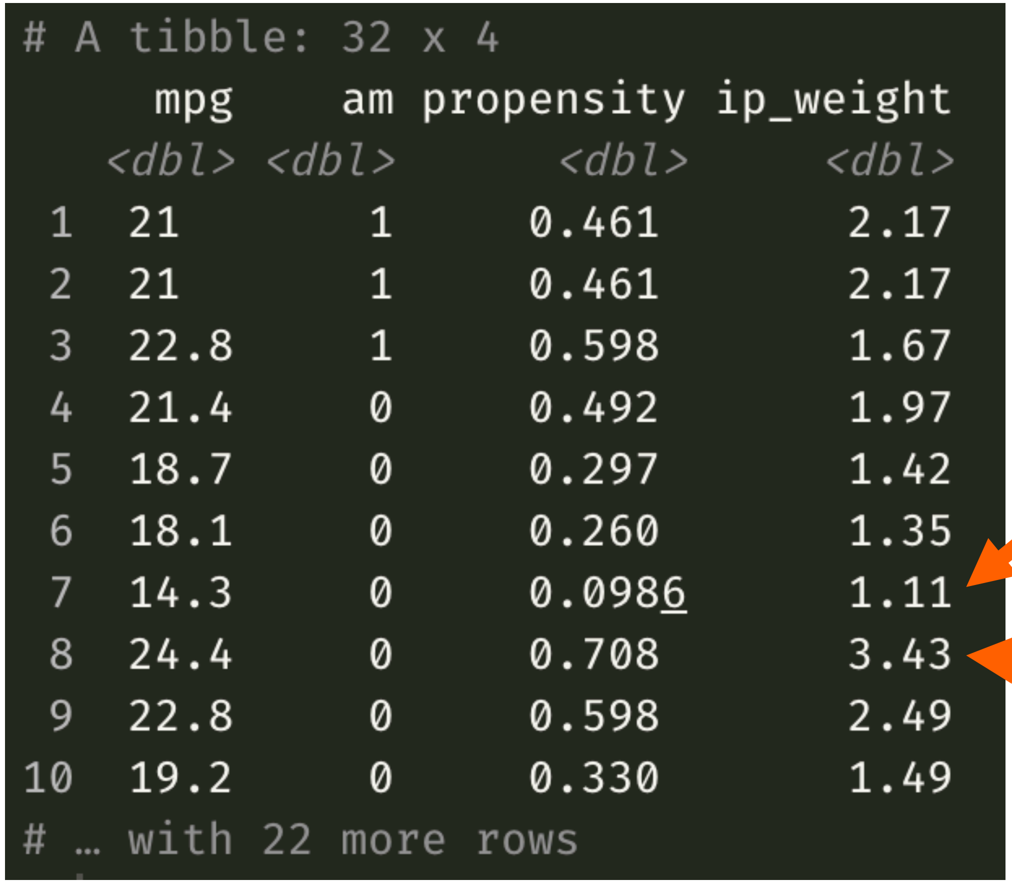
Inverse probability weighting

Use propensity scores to weight observations by how “weird” they are

Observations with high probability of treatment who don’t get it (and vice versa) have higher weight

$$\frac{\text{Treatment}}{\text{Propensity}} + \frac{1 - \text{Treatment}}{1 - \text{Propensity}}$$

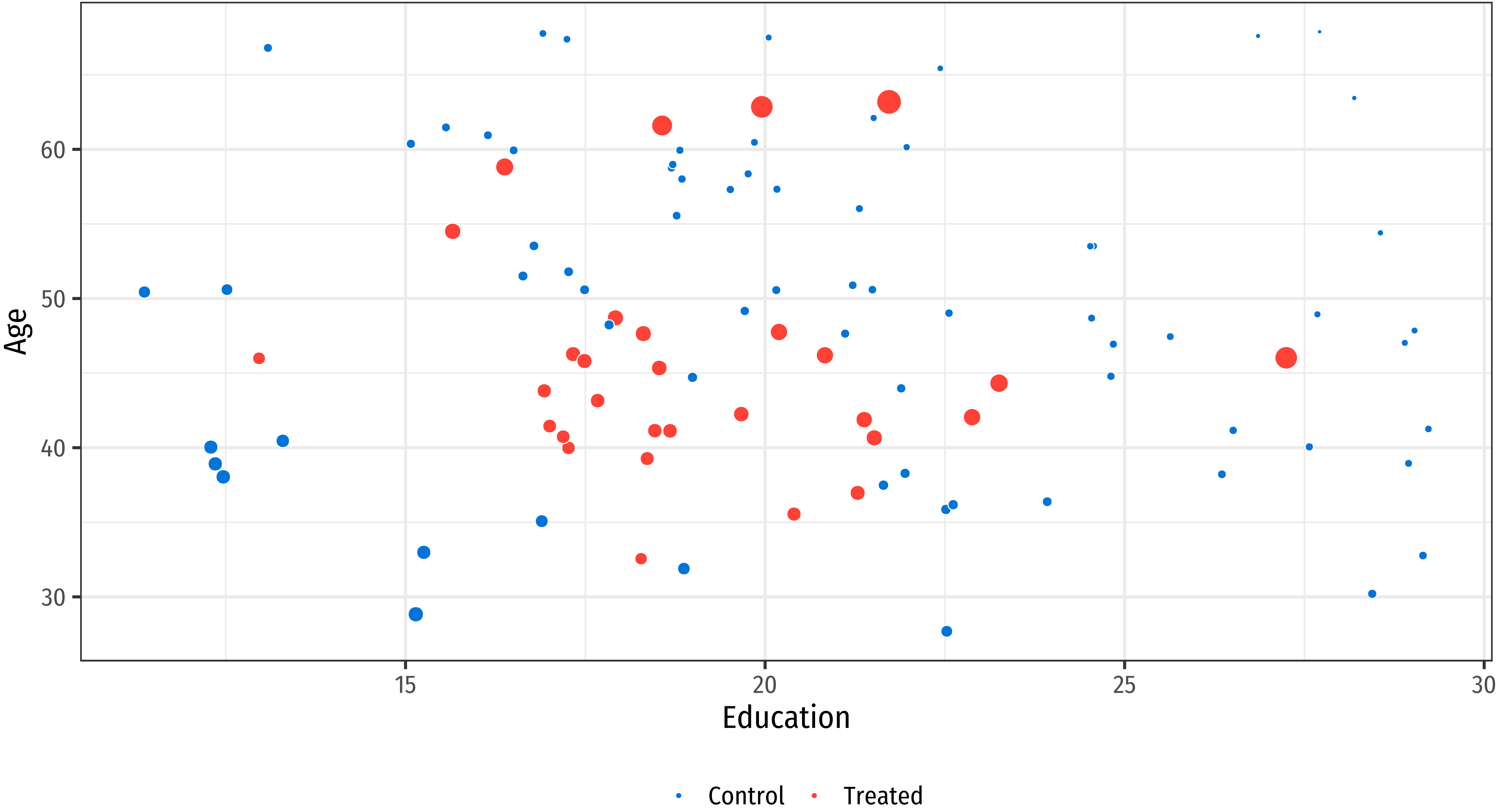
```r
augment(model_transmission, data = mtcars,
        type.predict = "response") %>%
  select(mpg, am, propensity = .fitted) %>%
  mutate(ip_weight = (am / propensity) +
                     ((1 - am) / (1 - propensity))
```



```
# A tibble: 32 x 4
     mpg    am propensity ip_weight
   <dbl> <dbl>      <dbl>     <dbl>
 1  21       1     0.461       2.17
 2  21       1     0.461       2.17
 3  22.8     1     0.598       1.67
 4  21.4     0     0.492       1.97
 5  18.7     0     0.297       1.42
 6  18.1     0     0.260       1.35
 7  14.3     0     0.0986      1.11
 8  24.4     0     0.708       3.43
 9  22.8     0     0.598       2.49
10  19.2     0     0.330       1.49
# … with 22 more rows
```

Unlikely to be manual and isn’t

Highly likely to be manual but isn’t. Weird!



Age
60
50
40
30
Education
15
20
25
30
● Control
● Treated

Our turn

Let’s close backdoors in R!