Regression for causal inference

PSCI 2301: Quantitative Political Science II

Prof. Brenton Kenkel

brenton.kenkel@gmail.com

Vanderbilt University

February 24, 2025

Recap

Last week — matching in theory and in practice

- How to make causal inferences with observational data?
  - → Measure and control for confounding variables
  - → Ideal: compare obs that are same in all ways except treatment
- Subclassification
  - → Divide observations into subgroups based on confounder values
  - → Weighted mean of differences within subgroups
  - → Runs into curse of dimensionality w/ many covariates
- Other matching estimators
  - → Mahalanobis distance
  - → Propensity scores

Today's agenda

Using controlled regression to estimate average treatment effects

1. Reasons to include controls in a regression
    - Reduce standard errors by better isolating treatment effect
    - Contend with selection bias, similar to matching
2. Interpreting regression output
    - How the treatment effect is calculated when controls are present
    - Special considerations for a logged outcome variable
    - How to interpret coefficients on controls (spoiler: don't try)

Regression with controls

Bivariate regression: What we already know

Linear regression with a single covariate:

$$\mathbb{E}[Y_i \mid X_i = x] = \underbrace{\alpha}_{\text{intercept}} + \underbrace{\beta}_{\text{slope}} \cdot x$$

Formula for slope estimate:

$$\hat{\beta} = \frac{\mathbf{cov}[X_i, Y_i]}{\mathbf{var}[X_i]}$$

If X_i is binary, then $\hat{\beta} = \mathbf{avg}[Y_i \mid X_i = 1] - \mathbf{avg}[Y_i \mid X_i = 0]$

Regression with multiple covariates

Say we have K covariates, $X_{i1}, X_{i2}, \ldots, X_{iK}$

Linear regression model with many covariates:

$$\begin{aligned}\mathbb{E}[Y_i \mid X_{i1} = x_1, \ldots, X_{iK} = x_K] &= \alpha + \beta_1 x_1 + \cdots + \beta_K x_K \\ &= \alpha + \sum_{k=1}^{K} \beta_k x_k.\end{aligned}$$

β_k = change in predicted Y_i due to increasing X_{ik} by one unit, *holding all other covariates fixed*

→ Can only be interpreted as a causal effect under special circumstances — more on this to come

Why include more variables?

Assume your goal is to estimate the avg treatment effect of D_i:

$$\mathbb{E}[Y_i \mid D_i, X_i] = \underbrace{\alpha}_{\text{intercept}} + \overbrace{\tau D_i}^{\text{ATE}} + \underbrace{\sum_{k=1}^{K} \beta_k X_{ik}}_{\text{controls}}.$$

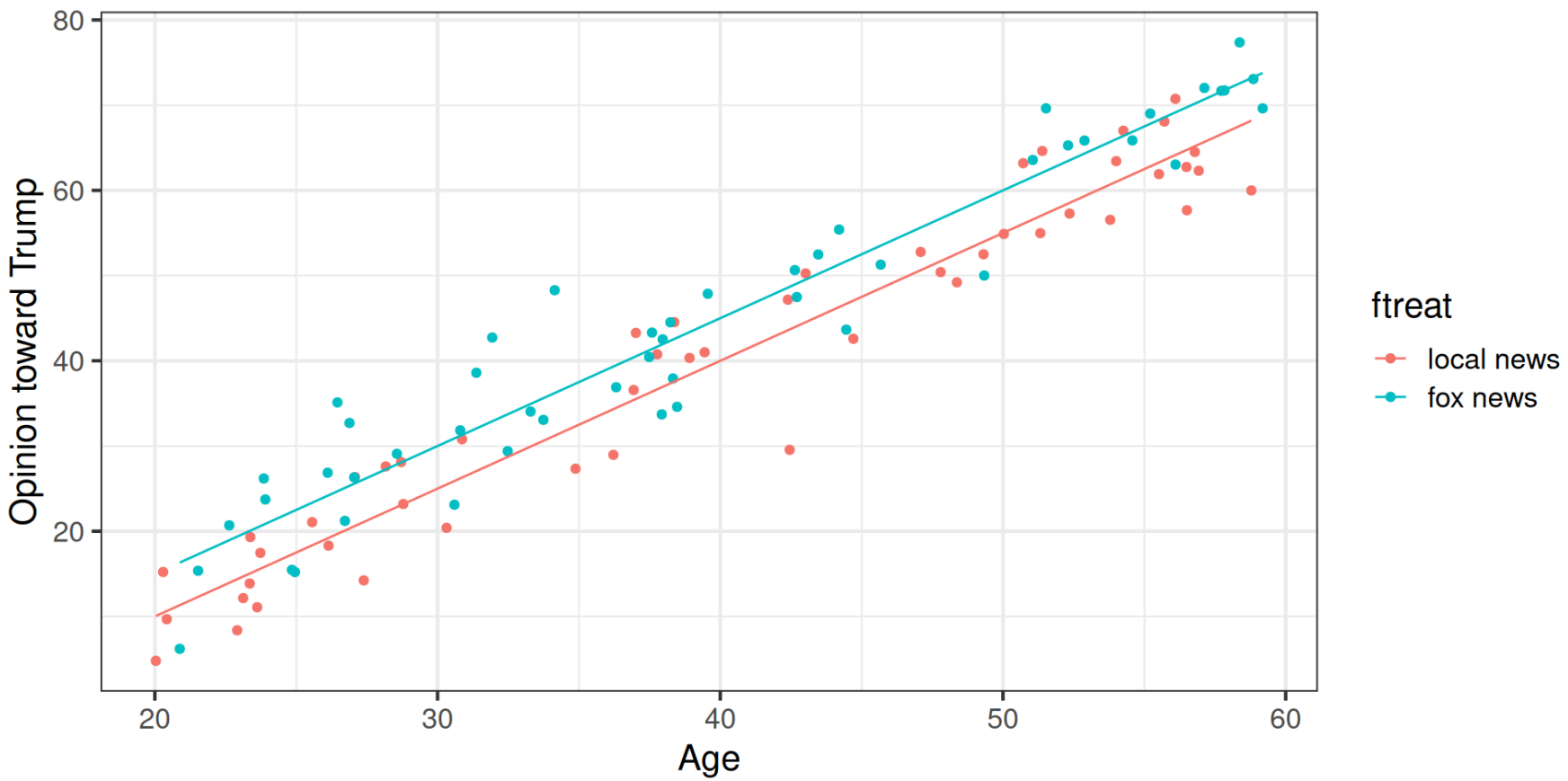
What is the point of including the controls?

1. Variance reduction
    - Isolate the effect of treatment more cleanly
    - Beneficial for both randomized experiments and observational data
2. Control selection bias — akin to matching

Controls for variance reduction

Hypothetical example: news watching experiment

Outcome mainly dependent on age, but treatment has a small effect



Controls for variance reduction

True ATE in simulation is $+5$ points

```
fit_no_controls <- lm(y ~ treat, data = df_fox)
tidy(fit_no_controls)
```

```
# A tibble: 2 × 5
  term  estimate std.error statistic  p.value
  <chr>    <dbl>     <dbl>     <dbl>    <dbl>
1 (Int…    39.8       2.70     14.8  1.24e-26
2 treat     3.93      3.81      1.03 3.05e- 1
```

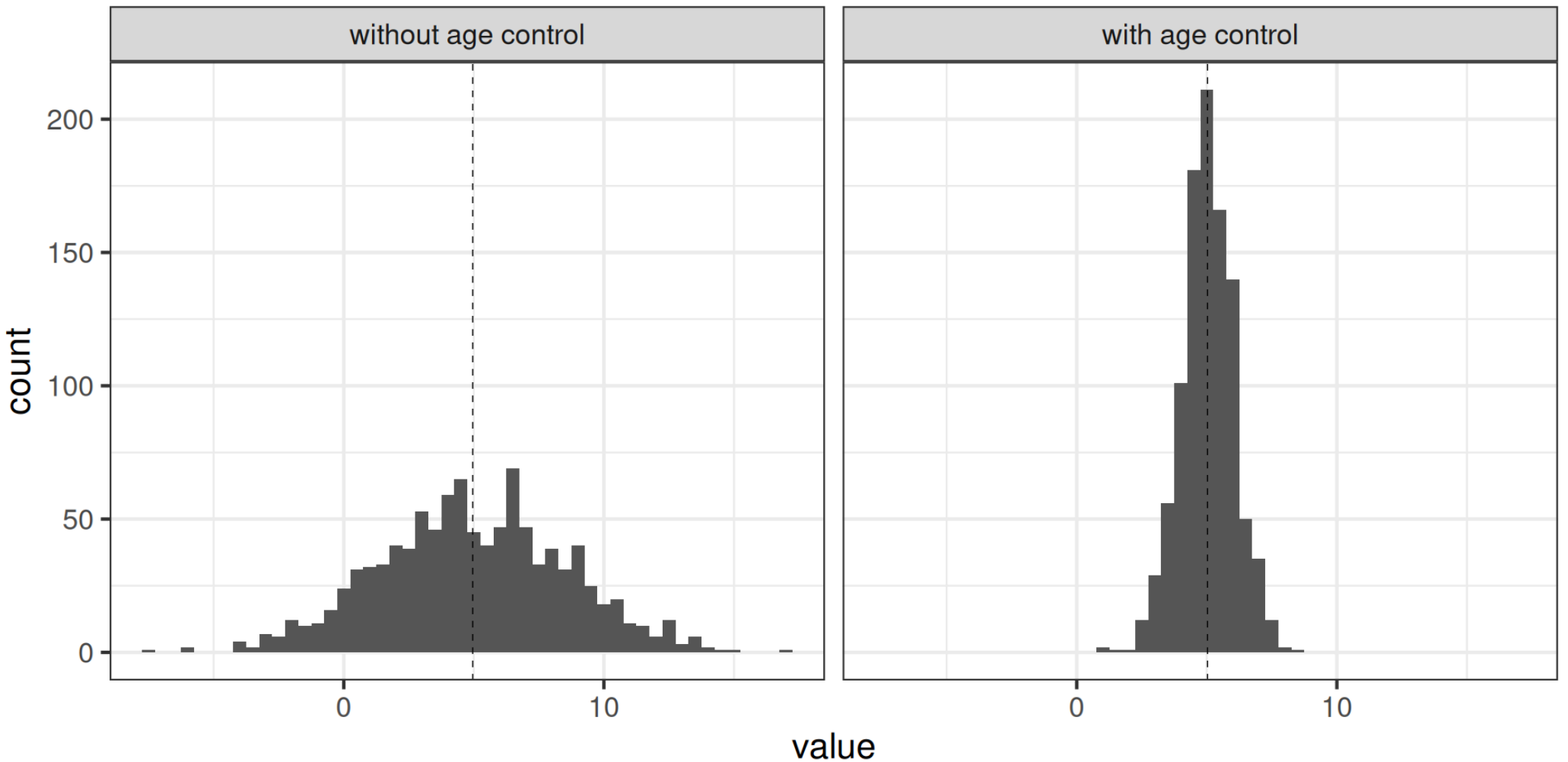
```
fit_controls <- lm(y ~ treat + age, data = df_fox)
tidy(fit_controls)
```

```
# A tibble: 3 × 5
  term  estimate std.error statistic  p.value
  <chr>    <dbl>     <dbl>     <dbl>    <dbl>
1 (Int…   -19.6     1.79      -10.9  1.34e-18
2 treat     4.81    1.01        4.76 6.67e- 6
3 age       1.49    0.0414     36.1  5.05e-58
```

Controlled regression gets closer + has much smaller std error

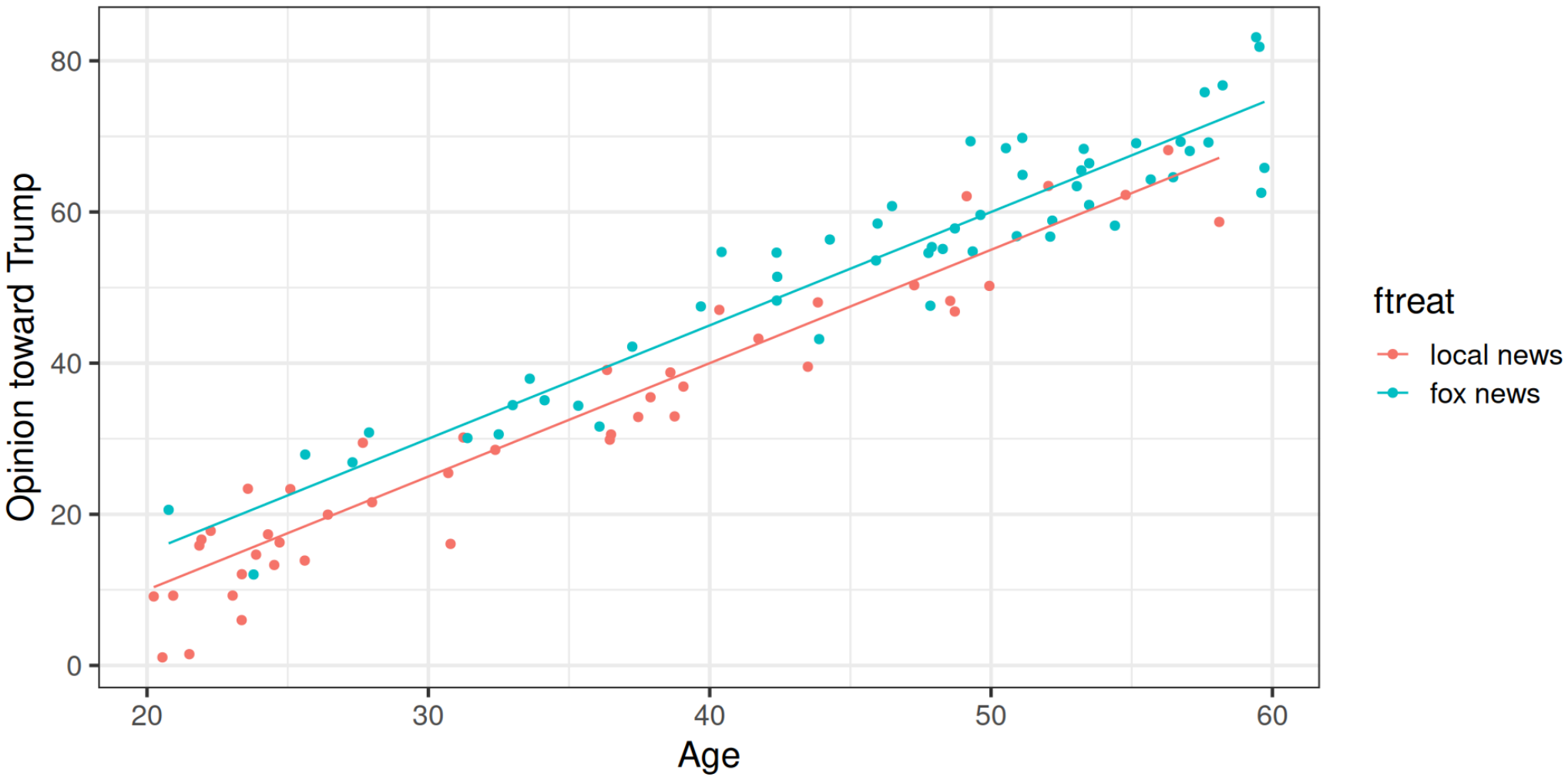
Controls for variance reduction

Distribution of ATE estimates across 1000 simulated experiments



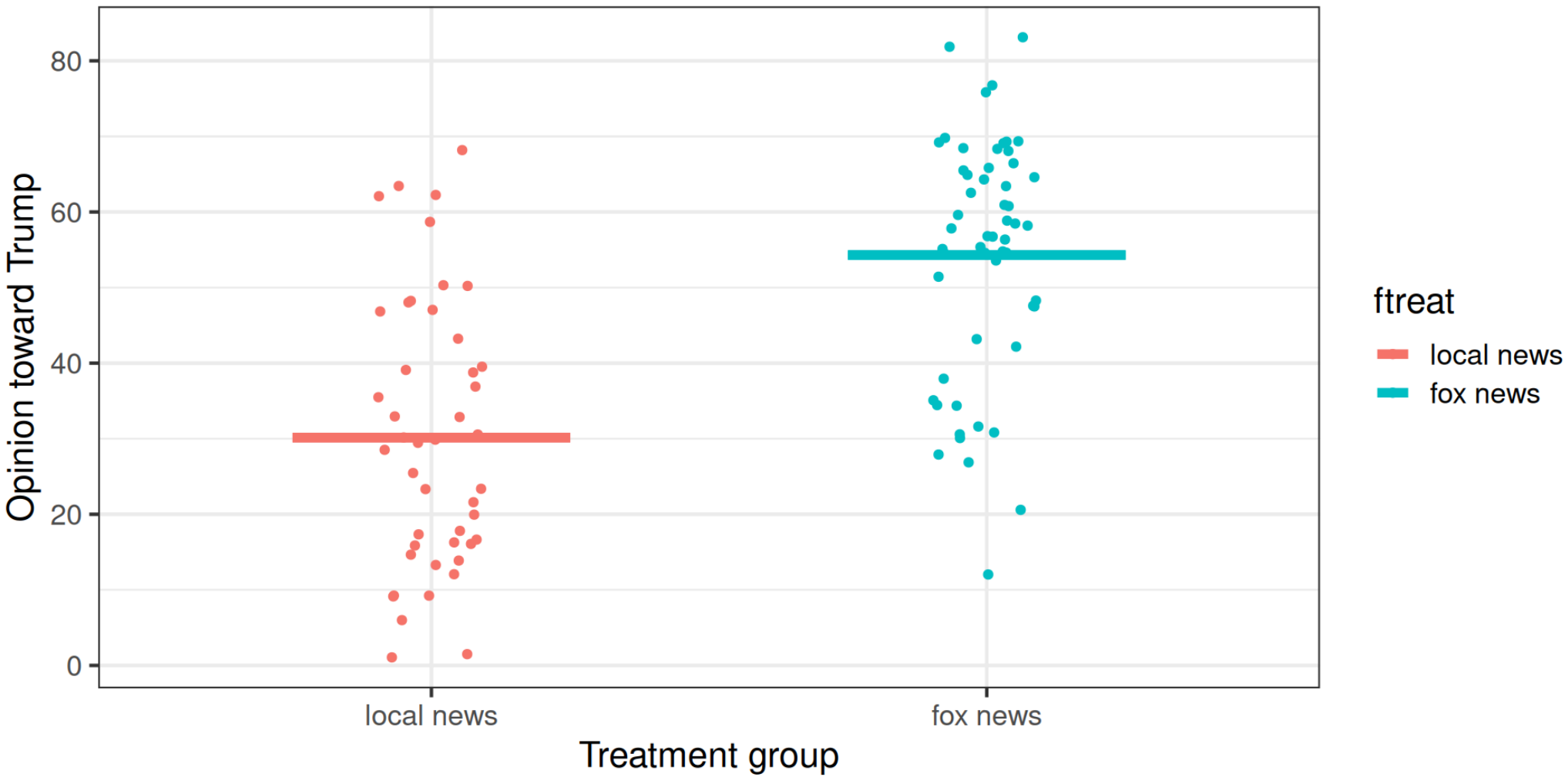
Controls to reduce selection bias

Imagine similar data, but where Pr(treat) increases with age



Controls to reduce selection bias

With selection bias, the raw difference of means vastly overstates ATE



Controls to reduce selection bias

```r
fit_no_control <- lm(y ~ treat, data = df_fox_obs)
tidy(fit_no_control)
```

```
# A tibble: 2 × 5
  term  estimate std.error statistic  p.value
  <chr>    <dbl>     <dbl>     <dbl>    <dbl>
1 (Int...     30.1      2.51     12.0  5.75e-21
2 treat     24.2      3.38      7.14 1.68e-10
```

```r
fit_control <- lm(y ~ treat + age, data = df_fox_ob
tidy(fit_control)
```

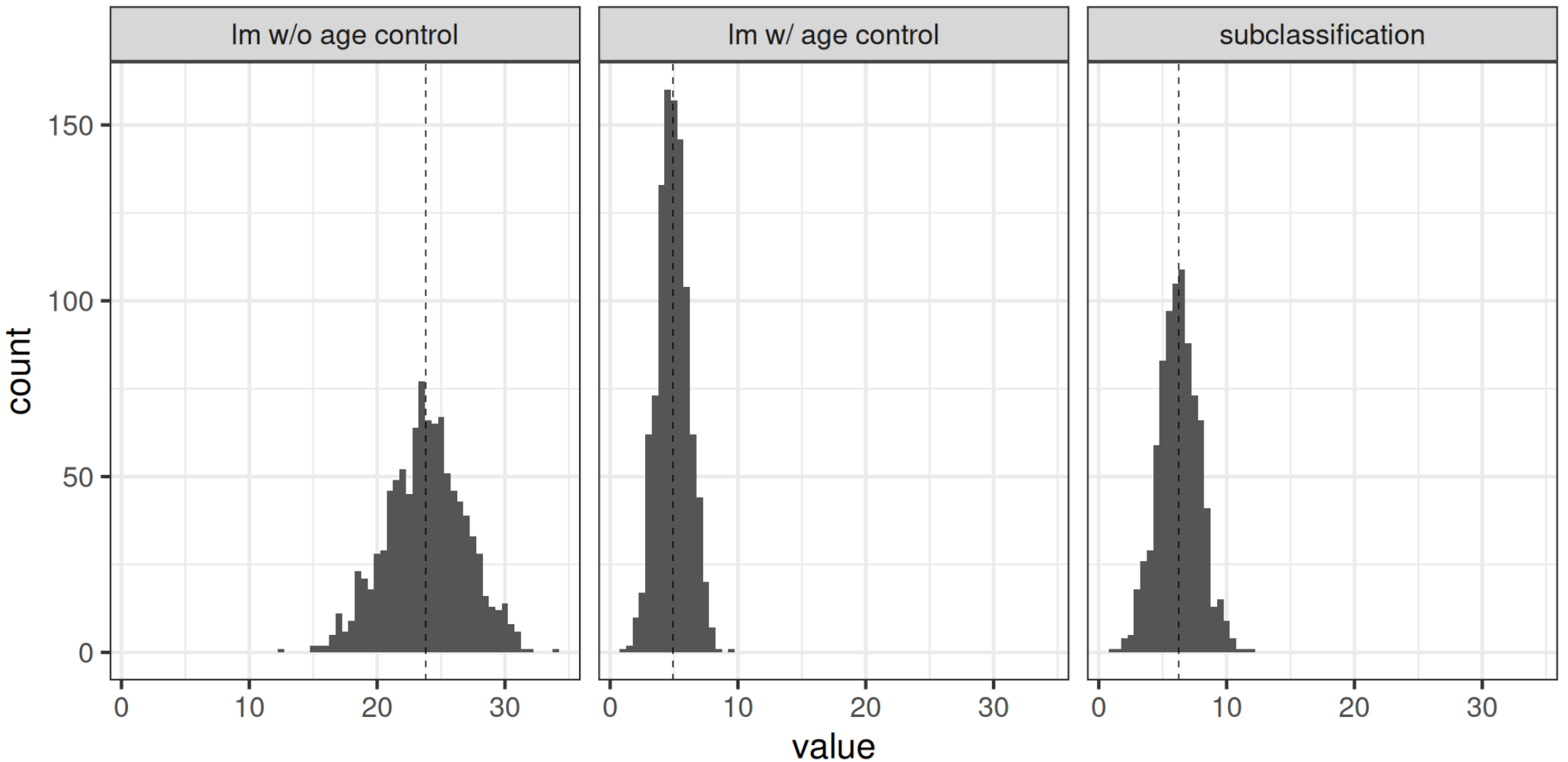
```
# A tibble: 3 × 5
  term  estimate std.error statistic  p.value
  <chr>    <dbl>     <dbl>     <dbl>    <dbl>
1 (Int...   -20.7     1.83      -11.3  2.03e-19
2 treat      6.02    1.20        5.03 2.24e- 6
3 age        1.49    0.0486     30.7  9.78e-52
```

```r
df_fox_obs |>
  mutate(age_group = ntile(age, 5)) |>
  group_by(age_group) |>
  summarize(diff = mean(y[treat == 1]) - mean(y[tre
            n = n()) |>
  summarize(ate = weighted.mean(diff, n))
```

```
# A tibble: 1 × 1
    ate
  <dbl>
1  7.41
```

Controls to reduce selection bias

Distribution of ATE estimates across 1000 simulated experiments



Comparison of treatment effect estimators

Regression

- Biased unless all confounders are measured and included
- Always uses all the data
- Lower standard errors *if* linear approximation is good
- Biased if linear approx. is bad
- Easier to use (IMO)

Matching

- Biased unless all confounders are measured and included
- May throw away some data
- Higher standard errors
- Unbiased, no linearity needed
- Harder to use — lots of choices

My bottom line: Best to use both, see if conclusions change much

... but even better not to have to rely on measuring all confounders

Interpreting regression output

Recap: Gilligan & Sergenti data

- Unit of observation: Conflict event
- Outcome: Length of peace, `ln_peace_duration`
- Treatment variable: UN intervention, `intervention`
- Various country/conflict-level confounders

```
df_gs
```

```
# A tibble: 87 × 11
  id    ethfrac country   intervention ln_peace_duration ln_deaths ln_wardur ln_population
  <chr>   <dbl> <chr>            <dbl>             <dbl>     <dbl>     <dbl>         <dbl>
1 41_2     1.36 Haiti                0              2.40      0            9          8.78
2 41_3     1.36 Haiti                1              4.96      5.52        12          8.81
3 52_2    55.8  Trinidad...           0              5.07      3.40         1          7.10
4 70_2    30.5  Mexico                0              3.40      4.98         1         11.4
5 70_3    30.5  Mexico                0              4.42      0            4         11.4
# i 82 more rows
# i 3 more variables: ln_military <dbl>, ln_gdppc <dbl>, polity <dbl>
```

Regression estimate of the intervention effect

```
fit_gs <- lm(
  ln_peace_duration ~ intervention + ln_deaths + ln_wardur + ln_population +
    ln_military + ln_gdppc + polity,
  data = df_gs
)
tidy(fit_gs)
```

```
# A tibble: 8 × 5
  term          estimate std.error statistic p.value
  <chr>            <dbl>     <dbl>     <dbl>   <dbl>
1 (Intercept)    3.60      1.57         2.30   0.0244
2 intervention   0.821     0.323        2.54   0.0129
3 ln_deaths     -0.0298    0.0530      -0.563  0.575
4 ln_wardur      0.00169   0.00193      0.877  0.383
5 ln_population -0.166     0.154       -1.08   0.285
6 ln_military    0.0415    0.143        0.290  0.773
7 ln_gdppc       0.199     0.142        1.40   0.165
8 polity         0.0204    0.0247       0.824  0.412
```

Calculating the controlled ATE

$$\hat{\tau} = \frac{\text{cov}[Y_i, D_i \mid X_{i1}, \ldots, X_{iK}]}{\text{var}[D_i \mid X_{i1}, \ldots, X_{iK}]}$$

```
# Variation in treatment assignment not explained by confounders:
resid_treat <- residuals(
  lm(intervention ~ ln_deaths + ln_wardur + ln_population + ln_military + ln_gdppc + polity,
     data = df_gs))

# Variation in outcome not explained by confounders:
resid_y <- residuals(
  lm(ln_peace_duration ~ ln_deaths + ln_wardur + ln_population + ln_military + ln_gdppc + polity,
     data = df_gs))

# Bivariate relationship between residuals
cov(resid_y, resid_treat) / var(resid_treat)
```

```
[1] 0.8207033
```

Interpreting regression with a logged outcome

Remember: $\ln z = a$ equivalent to $z = e^a$, where $e \approx 2.718$

Important property of exponents: $e^{a+b} = e^a \cdot e^b$

Rewriting the regression model with a logged outcome

$$\ln Y_i \approx \alpha + \tau D_i + \beta_1 X_{i1} + \cdots$$

$$Y_i \approx e^{\alpha + \tau D_i + \beta_1 X_{i1} + \cdots}$$

$$Y_i \approx \begin{cases} e^{\alpha + \beta_1 X_{i1} + \cdots} & \text{if } D_i = 0, \\ e^{\tau} \cdot e^{\alpha + \beta_1 X_{i1} + \cdots} & \text{if } D_i = 1. \end{cases}$$

$\rightsquigarrow$ Interpret as a *proportional* difference in the outcome due to treatment

Interpreting regression with a logged outcome

```
tidy(fit_gs) |> filter(term == "intervention")
```

```
# A tibble: 1 × 5
  term         estimate std.error statistic p.value
  <chr>           <dbl>     <dbl>     <dbl>   <dbl>
1 intervention    0.821     0.323      2.54  0.0129
```

```
exp(0.821)
```

```
[1] 2.272771
```

⇝ Expect peace to be 2.27x as long if UN intervenes, compared to if not

```
exp(0.821 + c(-2, 2) * 0.323)
```

```
[1] 1.191246 4.336207
```

⇝ Confidence interval: effect of 1.19x to 4.34x

Coefficients on control variables

`lm()` spits out coefficients/p-values for the treatment *and* each confounder

Best practice: **ignore** these for everything besides treatment variable

- Can't interpret their coefficients causally
  - → Coefficients give "all else equal" comparisons
  - → ... but all else is not equal when the variable affects treatment!
- High p-value does not mean the variable isn't a confounder

Wrapping up

What we did today

1. Worked through reasons to include controls
    - Variance reduction — isolate the treatment effect
    - Bias reduction — kill selection bias induced by confounders
2. Compared matching and regression
    - Regression is more precise *if* relationships are close to linear
    - Otherwise, matching is better at eliminating bias
3. Dealt with practical issues in regression
    - Proportional change interpretation w/ logged outcome
    - Ignore the control coefficients!

Next time: Estimating *heterogeneous effects* with regression