Quasi-experimental designs: Regression discontinuity design (RDD)

Julia Brillinger (adapted from Jay Brophy)

EBOH, McGill University

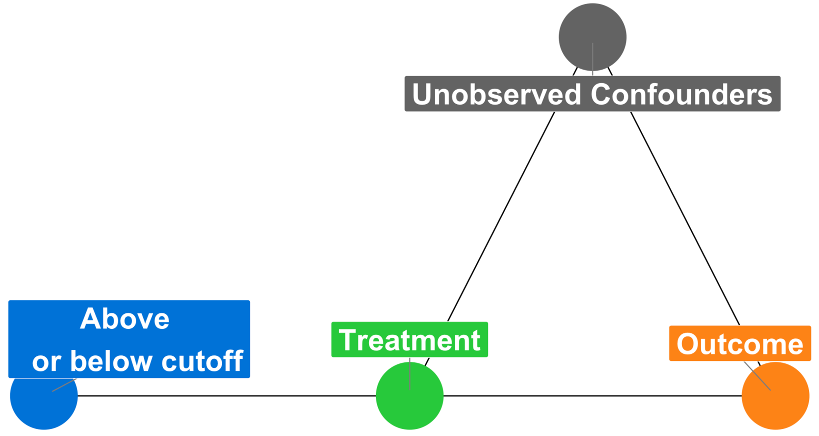
2025-11-23 (updated: 2025-12-01)

What are quasi-experimental methods?

- Quasi-experimental methods take advantage of natural experiments
- Exploit situations where treatment assignment is as good as random
- **Key benefit:** Exchangeability. Balances measured and unmeasured baseline covariates (by design)
- Common quasi-experimental methods: interrupted time series, instrumental variables, difference-in-differences, regression discontinuity

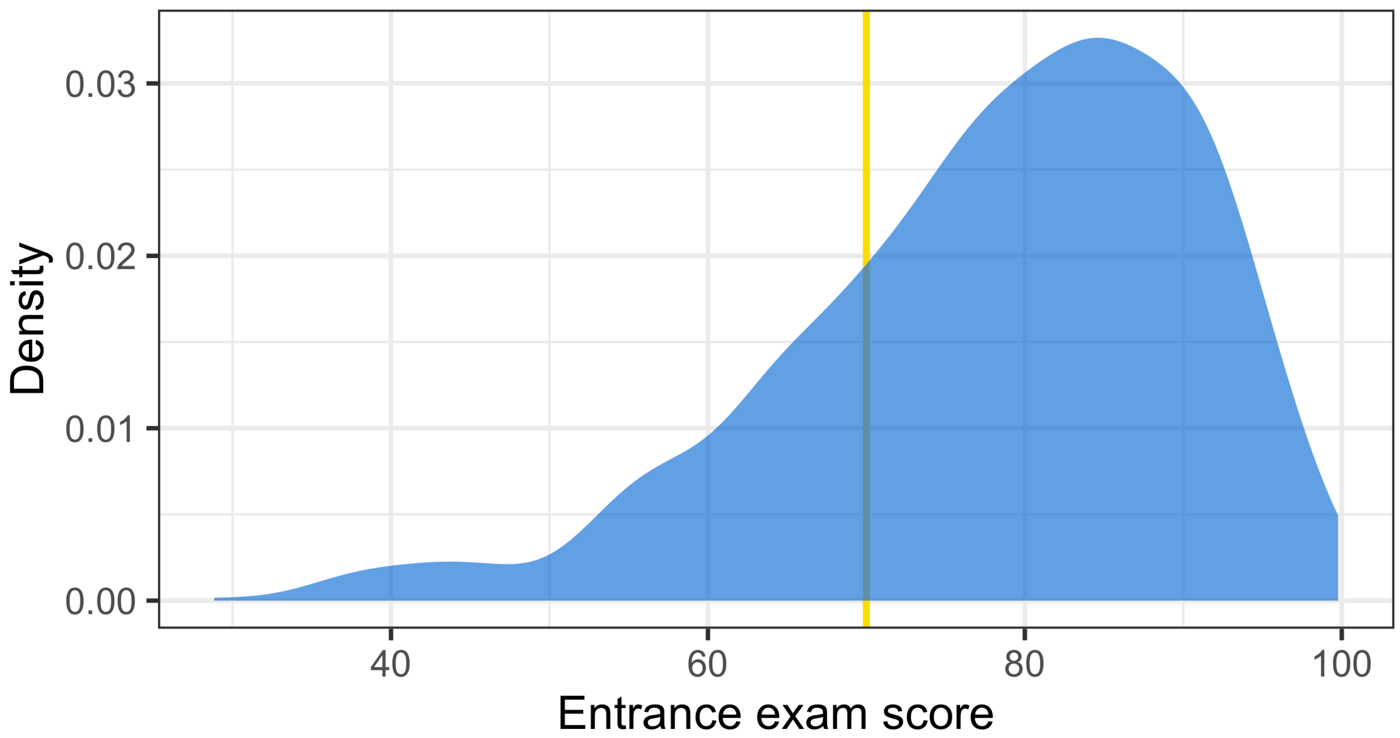
Regression discontintuity design

- Participants are assigned to treatment based on whether a measure falls above or below a certain **cut-point**/**cutoff**/**threshold**
- That measure that determines treatment/eligibility is called a **running variable**/**forcing variable**/**assignment variable**
- The running variable must be continuous at cutoff



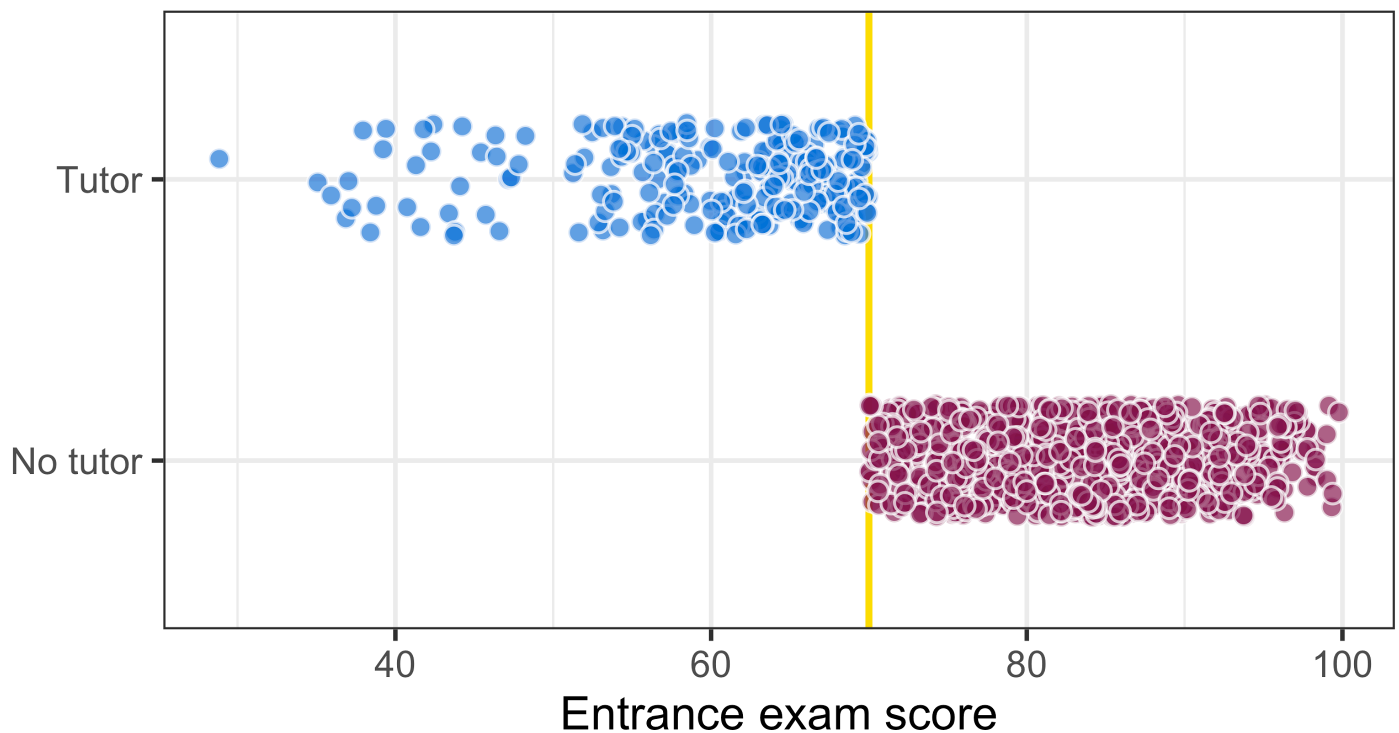
Example: Hypothetical tutoring program

- Students take an exam at the beginning of the year (entrance exam) and at the end of the year (exit exam)
- A tutoring program was introduced to improve test scores
- Students who score 70 or lower on the entrance exam get a free tutor for the year



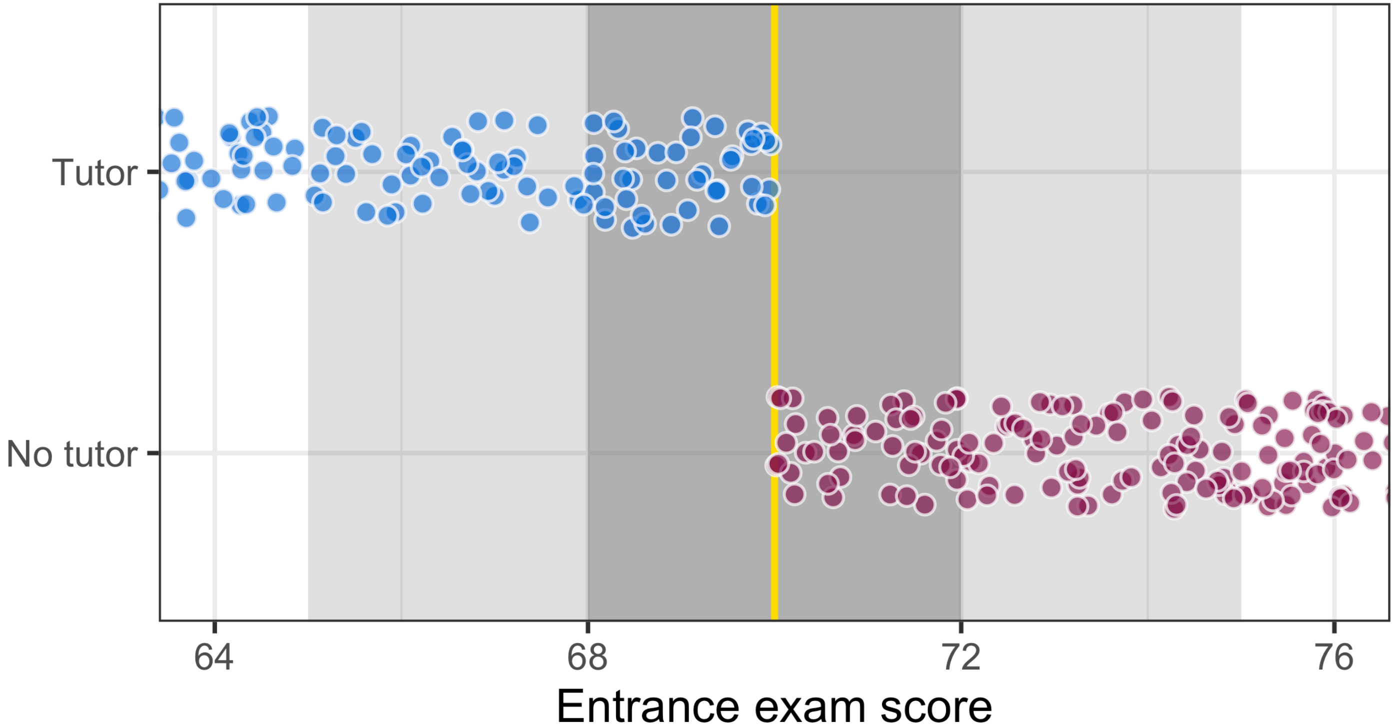
Example: Hypothetical tutoring program

- We want to know the **effect of a tutoring program** on a student's **exit exam score** at the end of the year ($ATE = \mathbb{E}[Y(X=1)] - \mathbb{E}[Y(X=0)]$, where Y is the exit exam score and X is having a tutor)
- Do we have exchangeability?



Example: Hypothetical tutoring program

- Probably not for the entire sample
- But the people right before and right after the cutoff are essentially the same
- They are similar on observed and unobserved pre-treatment covariates, like in an RCT

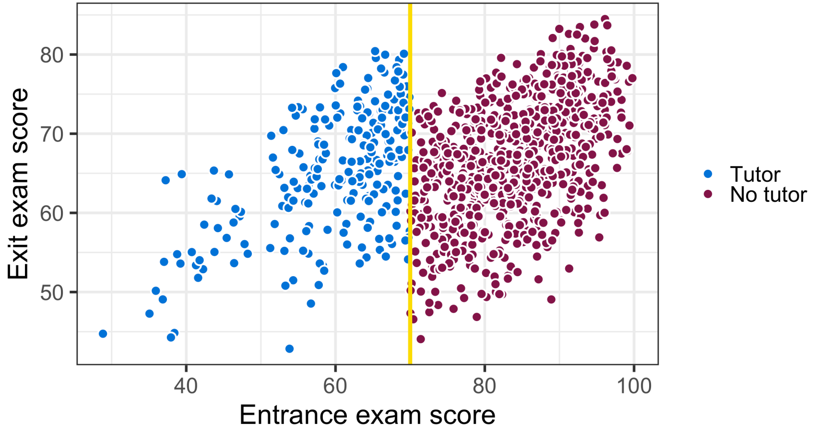


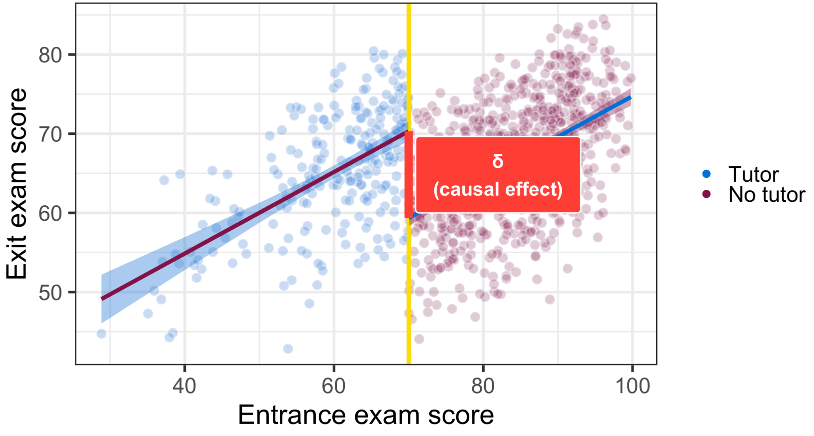
The running variable

- The running variable is subject to random variation (measurement error, sampling variability, chance factors)
- The arbitrary cutoff creates random variation in treatment assignment
- Other examples of running variables: CD4 count for initiation of ART, low birth weight for intensive interventions, income eligibility for benefits

Causal inference intuition

- Compare **outcomes** for those right before/after
- Measure the gap (difference) in outcome for people on both sides of the cutoff point
- Magnitude of this difference is the **local average treatment effect (LATE)** , the causal estimate under **perfect compliance**
- $\text{LATE} = \lim_{z \uparrow c} \mathbb{E}[Y_i \mid Z_i = z] - \lim_{z \downarrow c} \mathbb{E}[Y_i \mid Z_i = z]$, where Z_i is the running variable and c the the cutoff

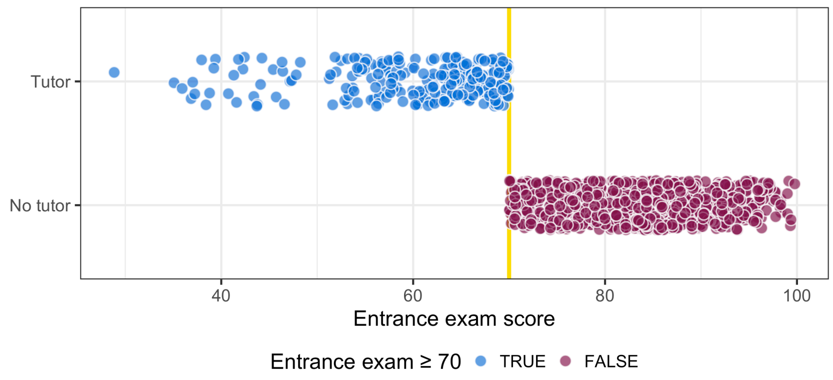




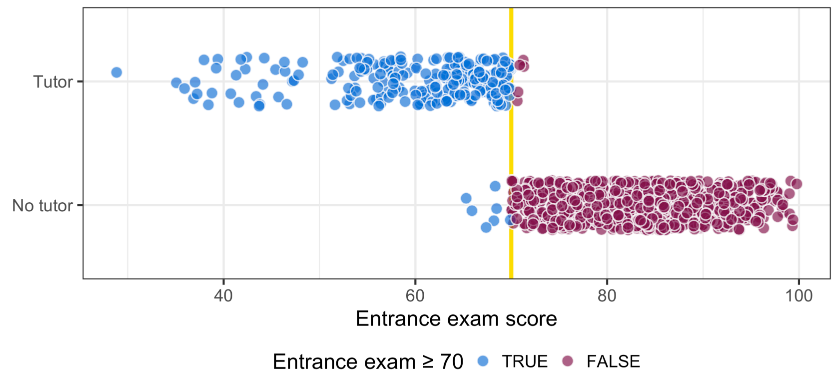
Noncompliance

- People on the margin of the cutoff might end up in/out of the program
- Sharp vs. fuzzy discontinuities

Sharp discontinuity - perfect compliance



Fuzzy discontinuity - imperfect compliance



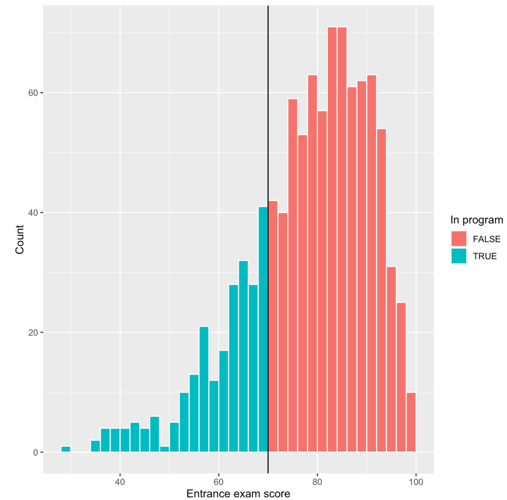
Key conditions for causal inference:

1. The decision rule exists and the cutoff c is known
   - Probability of treatment must change **discontinuously** at c of the running variable Z
   - $\lim_{Z\uparrow c} \Pr[X_i = 1 \mid Z_i = z] \neq \lim_{Z\downarrow c} \Pr[X_i = 1 \mid Z_i = z]$
   - Check this by plotting the running variable Z against treatment X
2. The running variable Z is continuous at the cutoff
   - Check covariate balance and potential manipulation
3. The relationship between Z_i and the potential outcomes $Y_i(0), Y_i(1)$ is continuous at c

Back to our tutoring example

Step 1: Verify the decision rule exists and the cutoff is known (done)
Step 2: Determine if the design is fuzzy or sharp (done)
Step 3: Check for discontinuity in running variable around cutoff (i.e make sure there's no unexpected variation, like many people clustered just below the cutoff to get tutoring)

```
 <- ggplot(tutoring, aes(x = entrance_exam,
geom_histogram(binwidth = 2, color = "white"
geom_vline(xintercept = 70) +
labs(x = "Entrance exam score", y = "Count",
```



Here it doesn't look like there's a jump around the cutoff but use `rddensity::rddensity()` to do a formal statistical test.

Worked example (continued)

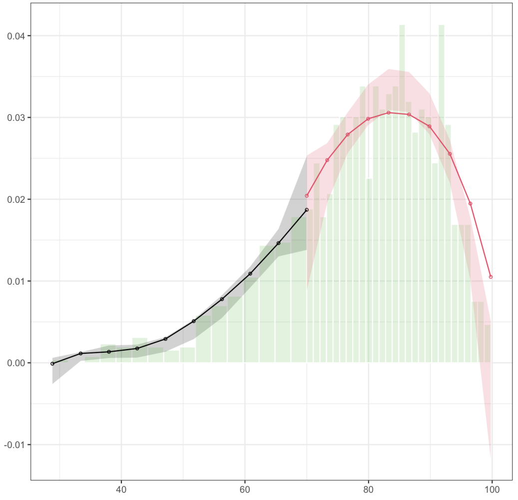
Formal statistical test

```r
test_density <- rddensity(tutoring$entrance_exam, c = 70)
output <- summary(test_density)
```

```
##
## Manipulation testing using local polynomial density estimation.
##
## Number of obs =       1000
## Model =               unrestricted
## Kernel =              triangular
## BW method =           estimated
## VCE method =          jackknife
##
## c = 70                Left of c           Right of c
## Number of obs         238                 762
## Eff. Number of obs    207                 523
## Order est. (p)        2                   2
## Order bias (q)        3                   3
## BW est. (h)           20.946              18.277
##
## Method                T                   P > |T|
## Robust                -0.4607             0.645
##
##
## P-values of binomial tests (H0: p=0.5).
```

Worked example (continued)

```
plot_density_test <- rdplotdensity(rdd = test_density,
                                   X = tutoring$entrance_exam,
                                   type = "both")  # This adds both points and lines
```

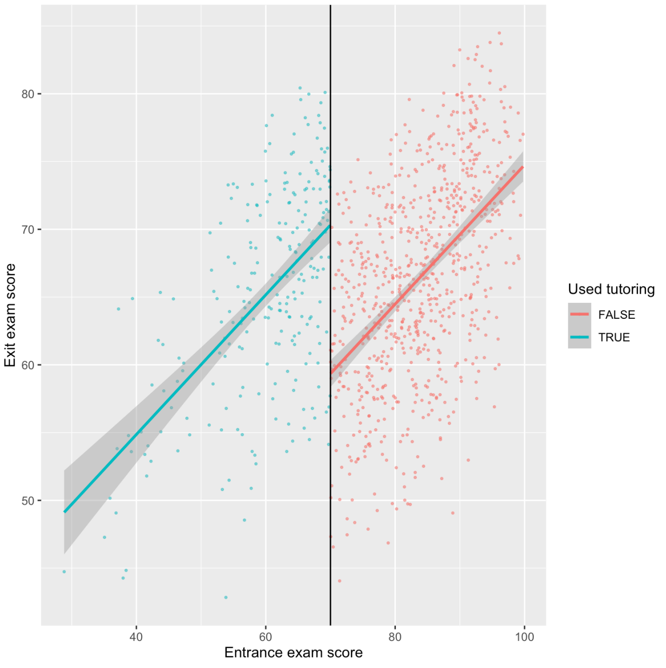


In the plot that the confidence intervals overlap substantially. Also the robust p-value > 0.05, so we have no evidence of a significant difference between the two lines. Based on this plot and the t-statistic, no evidence of manipulation or bunching.

Worked example (continued)

Step 4: Check for discontinuity in outcome across running variable

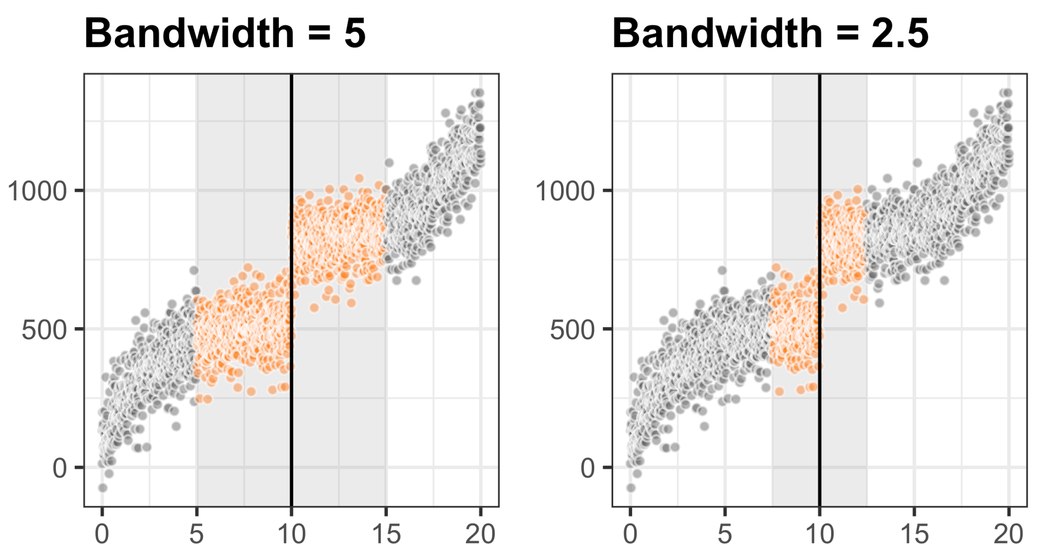
```
gg <- ggplot(tutoring, aes(x = entrance_exa
  geom_point(size = 0.5, alpha = 0.5) +
  # Add a line based on a linear model for
  geom_smooth(data = filter(tutoring, entra
  # Add a line based on a linear model for
  geom_smooth(data = filter(tutoring, entra
  geom_vline(xintercept = 70) +
  labs(x = "Entrance exam score", y = "Exit
```



Based on the clear discontinuity, participation in the tutoring program seems to boost final scores

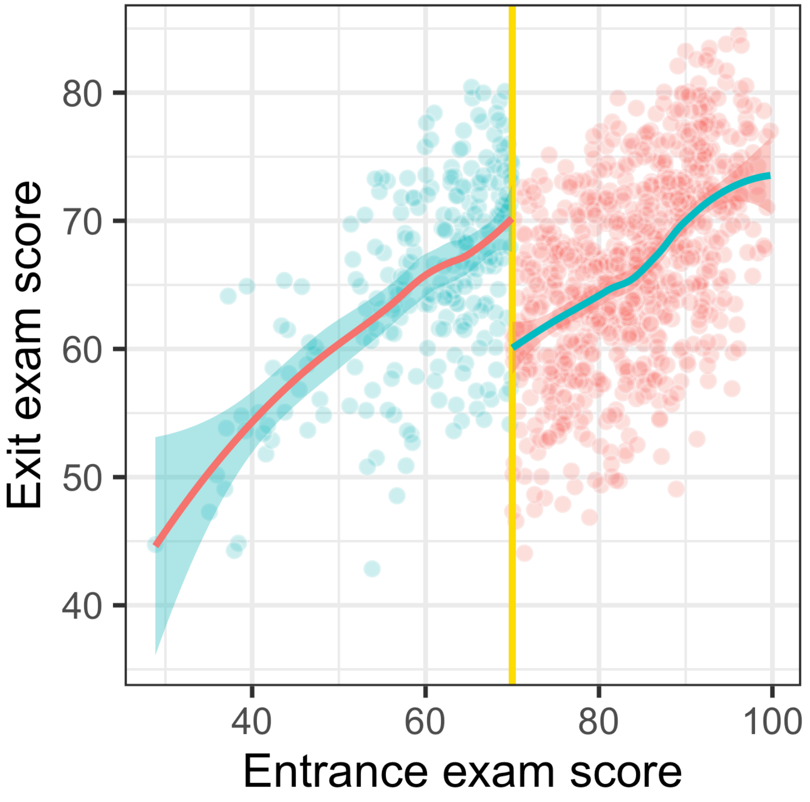
Step 5: Select a bandwidth and model

- Small bandwidths reduce potential bias from approximating outcome/exposure relationship using linear function
- However, larger bandwidth allows for greater power
- Algorithms exist to choose optimal width
- Also use common sense. Maybe ±5 for the entrance exam?
- For robustness, check what happens, if you double and halve the bandwidth



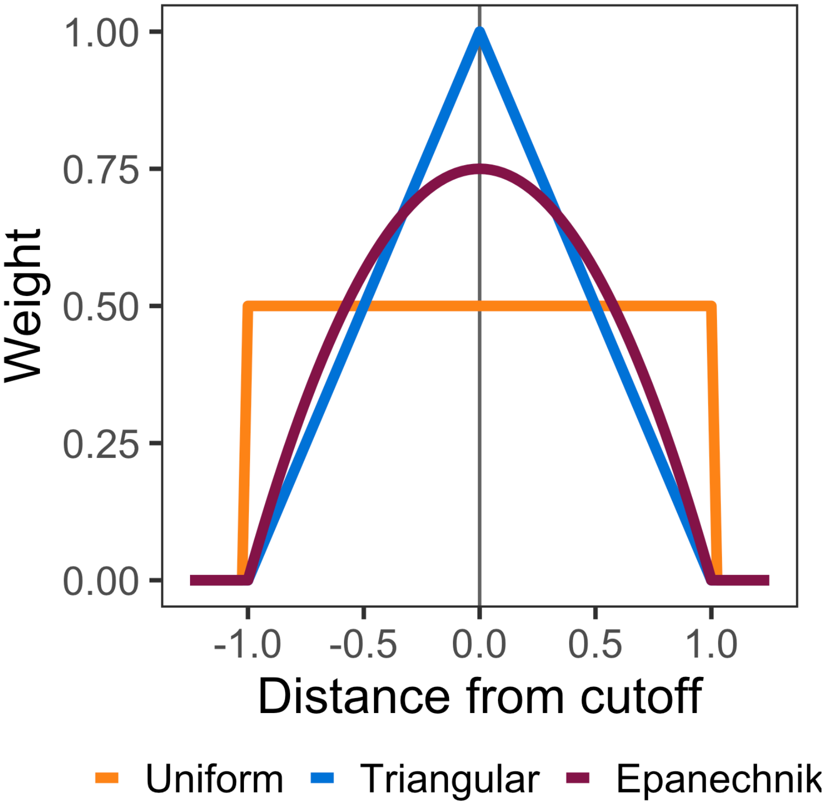
Modelling

- Simplest approach is linear model
- Can allow for the same or different slopes on either side of the cutoff
- Could also consider polynomial terms or use splines
- Could also use lines without parameters (use data to find the best line, often with windows and moving averages)
    - Locally estimated weighted scatter plot smoothing **(LOESS/LOWESS)** is a common method
- You can give greater weight to data closer to the cutoff using a **kernel**



Kernels

- Kernel = method for assigning importance to observations based on distance to the cutoff
- Because we care the most about observations right by the cutoff, give more distant ones less weight (weighted least squares)



Worked example (continued)

Step 6: Measure the size of the effect and its statistical significance

$$\text{Exit exam} = \beta_0 + \beta_1 \text{Entrance exam score}_{\text{centered}} + \beta_2 \text{Tutoring program} + \epsilon$$

```
tutoring_centered <- tutoring %>%
  mutate(entrance_centered = entrance_exam - 70)

model_simple <- lm(exit_exam ~ entrance_centered + tutoring, data = filter(tutoring_centered
tidy(model_simple)
```

```
## # A tibble: 3 × 5
##   term              estimate std.error statistic   p.value
##   <chr>                <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)         60.1       1.10      54.8  1.29e-117
## 2 entrance_centered    0.553     0.338      1.64 1.03e-  1
## 3 tutoringTRUE        10.2       1.93       5.26 3.81e-   7
```

Worked example (continued)

Interpretation:

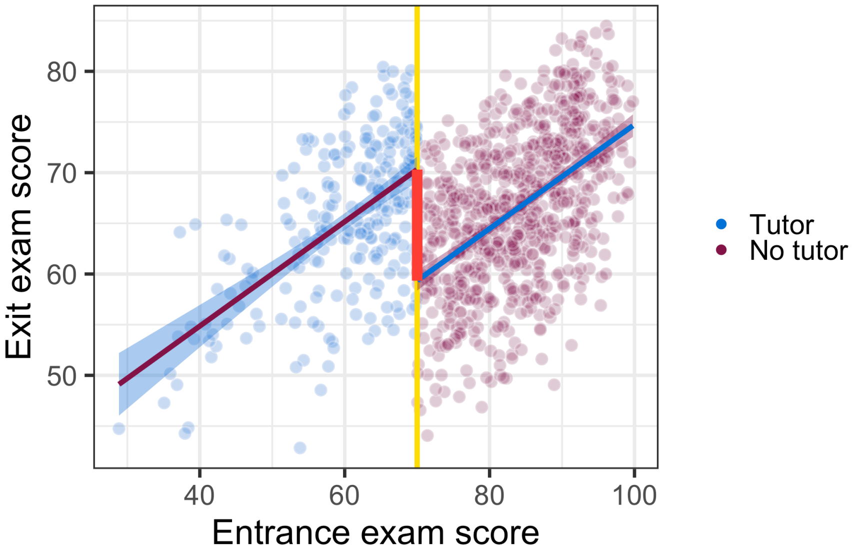
β_0 This is the intercept. It shows the predicted exit exam score when entrance_centered is 0 (i.e. 70) and when tutoring is FALSE.)

β_1 This is the coefficient for entrance_centered. For every point above 70 that people score on the entrance exam, they score 0.55 points higher on the exit exam. We don't really care that much about this number.

β_2 This is the coefficient for the tutoring program, and this is the one we care about the most. This is the shift in intercept when tutoring is true, or the difference between scores at the threshold. Participating in the tutoring program increases exit exam scores by 10.2 points.

To whom does the LATE apply?

- This is a **local** ATE. Do we care about the effect only at the cutoff?
- If we assume constant treatment effects, it would apply to everyone
- Problem arises if treatment effect is heterogeneous and dependent on Z
- Nevertheless, often the local effect is relevant to policy makers (should they change the threshold?)



Fuzzy RD

- What if there is imperfect compliance? (not all students eligible receive tutoring)
- We use the same model as for the sharp RD:
  $\text{Exit exam} = \beta_0 + \beta_1 \text{Entrance exam score}_{\text{centered}} + \beta_2 \text{Tutoring program} + \epsilon$
- But this time, this gives us the effect of **treatment eligibility (the intention-to-treat, ITT)** not **treatment**

Fuzzy RD (continued)

- Four possible types of students: always takers (get tutored regardless of eligibility), never takers (would never be tutored regardless of eligibility), compliers (get tutored if eligible) and deniers (only get tutored if ineligible)
- What if we want to know the effect of the **treatment**?
- We focus on the compliers
- Assume monotonicity (no deniers)
- Divide the effect of ITT by the probability of treatment at the cutoff to get the **complier average treatment effect (CACE)**
  - $CACE = \frac{\lim_{z \uparrow c} E[Y_i(1)|Z_i=z] - \lim_{z \downarrow c} E[Y_i(0)|Z_i=z]}{\lim_{z \uparrow c} E[X_i=1|Z_i=z] - \lim_{z \downarrow c} E[X_i=1|Z_i=z]}$
  - This is akin to an instrumental variable approach, where $1[Z_i < c \mid Z_i \to c]$ is the instrument

Fuzzy RD (continued)

- Like with an IV, must assume Z_i being just above or just below c only effects Y_i through X_i **(the exclusion restriction)**
- Can use **two-stage least squares (2SLS) estimation** to estimate the CACE

```
library(sem)
model_fuzzy <- tsls(exit_exam ~  tutoring_fuzzy + entrance_centered, ~tutoring + entrance_cen
                    data = filter(tutoring_centered, entrance_centered >= -5 & entrance_cent
```

```r
summary(model_fuzzy)
```

```
##
##  2SLS Estimates
##
## Model Formula: exit_exam ~ tutoring_fuzzy + entrance_centered
##
## Instruments: ~tutoring + entrance_centered
##
## Residuals:
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -18.948   -5.064    1.071    0.000    5.081   20.951
##
##                      Estimate Std. Error  t value   Pr(>|t|)
## (Intercept)        58.8895756  1.4549821 40.47443 < 2.22e-16 ***
## tutoring_fuzzyTRUE 13.0350207  2.7591769  4.72424 4.5108e-06 ***
## entrance_centered   0.7912343  0.4200305  1.88375   0.061142 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.7005034 on 188 degrees of freedom
```

```r
confint(model_fuzzy)
```

```
##                          2.5 %    97.5 %
## (Intercept)        56.03786310 61.741288
## tutoring_fuzzyTRUE  7.62713338 18.442908
## entrance_centered  -0.03201037  1.614479
```

Conclusions

- **Benefits:**
  - Exchangeability (can balance measured and unmeasured baseline covariates)
  - Analysis is fairly simple
  - Causal effect estimation requires few assumptions
  - When done well, offers strong causal estimate
- **Concerns:**
  - Statistical power (it's greedy, you need lots of data)
  - Bias can be introduced by **manipulation of the running variable** and **discontinuous potential outcomes** at the cutoff
  - Consider the relevance of the estimand (LATE for sharp, CATE for fuzzy)