Quant II

Lab 9: Front-door criterion, distributional effects, multiple endpoints

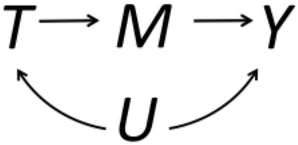
Giacomo Lemoli

April 18, 2022

Today's plan

- Front-door criterion review
- Distributional effects in practice: quantile regression, distribution regression
- Multiple endpoints: indexes, p-value adjustment

Front-door criterion



Setting: we want to estimate the effect of (binary) T on Y, but it is confounded in the data. We have the (binary) mediator M which *fully* mediates the effect of T.

Basic idea: under full mediation and unconfoundedness, the effect of T on Y is identified.

Front-door criterion

Assumptions:

- No direct effect of T on Y (exclusion restriction)
  - $Y_i(t, m) = Y_i(1 - t, m) \equiv Y_i(m)$
- Unconfoundedness
  - $M_i(y) \perp T_i$ for $t \in \{0, 1\}$
  - $Y_i(m) \perp M_i(t) | T_i$ for $t \in \{0, 1\}, m \in \{0, 1\}$

These assumptions imply that the effect of T on M and of M on Y are identified. We can separately estimate them and multiply them to obtain the effect of T on Y.

Front-door criterion

In the lecture slides we have derived that

$$ATT = \underbrace{(Pr[M = 1|T = 1] - Pr[M = 1|T = 0])}_{\text{Effect of T on M}} \times \underbrace{(E[Y_i|M_i = 1, T_i = 1] - E[Y_i|M_i = 0, T_i = 1])}_{\text{Effect of M on Y among the treated}}$$

and

$$ATC = \underbrace{(Pr[M = 1|T = 1] - Pr[M = 1|T = 0])}_{\text{Effect of T on M}} \times \underbrace{(E[Y_i|M_i = 1, T_i = 0] - E[Y_i|M_i = 0, T_i = 0])}_{\text{Effect of M on Y among the controls}}$$

Front-door criterion

We can recover the ATE as the product of identified quantities.

$$ATE = ATT \times Pr(T_i = 1) + ATC \times Pr(T_i = 0)$$

As the component quantities rely on selection on observables, we can estimate them using regression, matching or weighting.

Front-door criterion: application

The Paper of How: Estimating Treatment Effects Using the Front-Door Criterion*

Marc F. Bellemare† Jeffrey R. Bloem‡ Noah Wexler§

June 18, 2020

Abstract

We present the first application of Pearl's (1995, 2000) front-door criterion to observational data in which the required assumptions plausibly hold. For identification, the front-door criterion relies on the presence of a single, strictly exogenous mediator variable on the causal path between the treatment and outcome variables. After first explaining how to use the front-door criterion in practice, we present empirical illustrations. Our core application uses data on over 890,000 Uber and Lyft rides in Chicago to estimate the average treatment effect of the authorization of ride sharing—that is, the decision to authorize the app to overlap one's ride with a stranger's ride—on tipping behavior. We exploit as mediator the (conditionally) exogenous variation in whether one actually gets to share a ride, since authorizing a shared ride does not necessarily result in sharing a ride. Comparing our front-door criterion results to those of naïve regressions of tipping on the decision to authorize ride sharing, we find that almost all of the naïve negative relationship between authorizing a shared ride and tipping is due to selection effects. Finally, we explore the consequences for applied work of violating some of the assumptions underpinning the front-door criterion approach

Quantile regression

Rather than estimating the treatment effect on the conditional mean, estimates the treatment effect on conditional quantiles of the outcome variable.

Answers the question "how do different parts of the distribution change due to treatment?".

Note: results to be interpreted in terms of distribution moments, not of individual outcomes.

Quantile regression in practice

To implement quantile regression:

- Stata: `qreg` and sister commands
    - `sqreg` and `bsqreg` for bootstrapped standard errors
- R: `quantreg`
    - Several methods available in the arguments, different SEs implemented through `summary`

Working example

Our working example is the effect of military repression on support for democracy in the 1988 Chilean plebiscite.

AMERICAN JOURNAL of POLITICAL SCIENCE

The Geography of Repression and Opposition to Autocracy

Maria Angélica Bautista University of Chicago
Felipe González Pontificia Universidad Católica de Chile
Luis R. Martínez University of Chicago
Pablo Muñoz FGV EPGE Brazilian School of Economics and Finance
Mounu Prem Universidad del Rosario

Abstract: *State repression is a prominent feature of nondemocracies, but its effectiveness in quieting dissent and fostering regime survival remains unclear. We exploit the location of military bases before the coup that brought Augusto Pinochet to power in Chile in 1973, which is uncorrelated to precoup electoral outcomes, and show that counties near these bases experienced more killings and forced disappearances at the hands of the government during the dictatorship. Our main result is that residents of counties close to military bases both registered to vote and voted "No" to Pinochet's continuation in power at higher rates in the crucial 1988 plebiscite that bolstered the democratic transition. Potential mechanisms include informational frictions on the intensity of repression in counties far from bases and shifts in preferences caused by increased proximity to the events. Election outcomes after democratization show no lasting change in political preferences.*

Quantile regression in Stata

Estimation with qreg

```
. esttab est_25 est_50 est_75, keep(DMilitaryPresence) se
```

	(1) VoteShareNo	(2) VoteShareNo	(3) VoteShareNo
DMilitaryP~e	2.744	2.097	1.707
	(1.773)	(1.699)	(1.738)
N	276	276	276

Standard errors in parentheses
* p<0.05, ** p<0.01, *** p<0.001

Quantile regression in Stata

Bootstrapping

```
      (1)    (2)    (3)
VARIABLES    q25    q50    q75

DMilitaryPresence 2.74378*     2.09668      1.70705
      (1.589)      (1.519)      (2.040)

Observations        276    276    276
Standard errors in parentheses
*** p<0.01, ** p<0.05, * p<0.1
```

Quantile regression in R

```r
library(quantreg); library(haven); library(dplyr); library(ggplot2)

# Import data
d <- read_dta("FinalDatasetForReplication.dta")

# Prepare
d <- d %>% filter(MainSample == 1)

# Quantile regression
set.seed(10)
mod <- rq(VoteShareNo ~ DMilitaryPresence + share_allende70 + share_alessandri70 +
            lnDistStgo + lnDistRegCapital + Pop70_pthousands + sh_rural_70 + factor(IDProv),
          tau = c(0.25, 0.5, 0.75), data = d)

# Results (bootstrap SE)
est <- summary(mod, se = "boot")
out <- do.call("rbind", lapply(est, function(x) c(x$tau,
                                 x$coefficients["DMilitaryPresence",])))
out
```

```
##              Value Std. Error   t value  Pr(>|t|)
## [1,] 0.25 2.743785   1.664754 1.6481617 0.1006065
## [2,] 0.50 2.096675   1.517925 1.3812773 0.1684573
## [3,] 0.75 1.707053   1.979950 0.8621697 0.3894407
```

Distribution regression

Estimate how the outcome distribution changes by treatment condition.

Differently from the quantile regression, we are estimating changes in density ("y-axis"), not in quantiles ("x-axis").

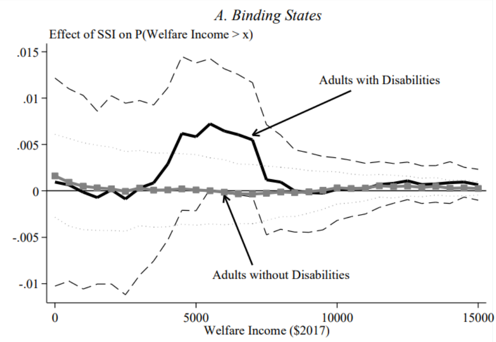
Straightforward idea:

- Create dummies for being below/above a certain value of the outcome distribution: $\mathbb{I}[y_i \geq y]$
- Use them as outcomes in separate regressions
- OLS coefficient interpretation: how much the share of units in that part of the support has been changed by the treatment
- With continuous treatment one can also use logit or probit

Distribution regression

Characterize the changes in the distribution. From Goodman-Bacon and Schmidt (2020):

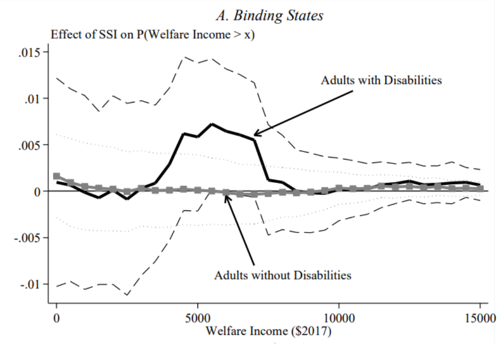
Figure 6. The Effect of SSI on the Distribution of Welfare Income for Adults With and Without Disabilities, 1970 and 1980 Censuses



Distribution regression

Characterize the changes in the distribution. From Goodman-Bacon and Schmidt (2020):

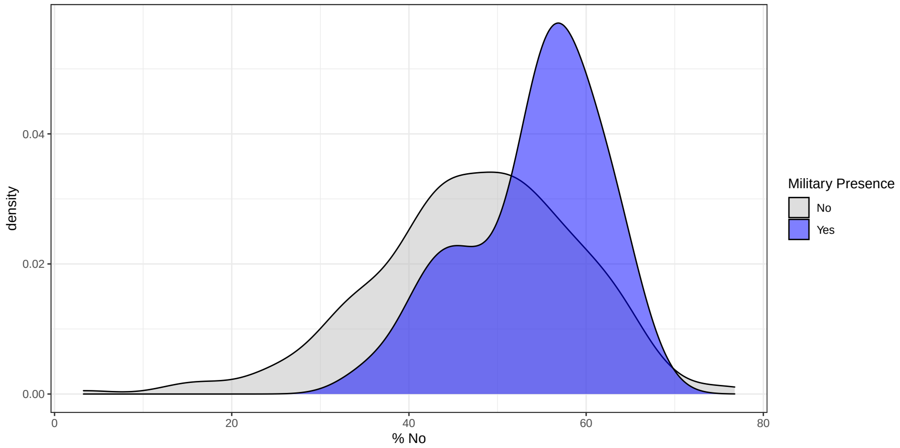
Figure 6. The Effect of SSI on the Distribution of Welfare Income for Adults With and Without Disabilities, 1970 and 1980 Censuses



Let's do a similar exercise for the effect of repression on the Chilean plebiscite

Distribution regression

```r
ggplot(d, aes(x=VoteShareNo,  group=DMilitaryPresence, fill=factor(DMilitaryPresence))) +
  geom_density(alpha=0.5) +
  scale_fill_manual(values = c("grey", "blue"), name = "Military Presence", labels=c("No", "Yes")) +
  labs(x="% No") + theme_bw()
```



Distribution regression

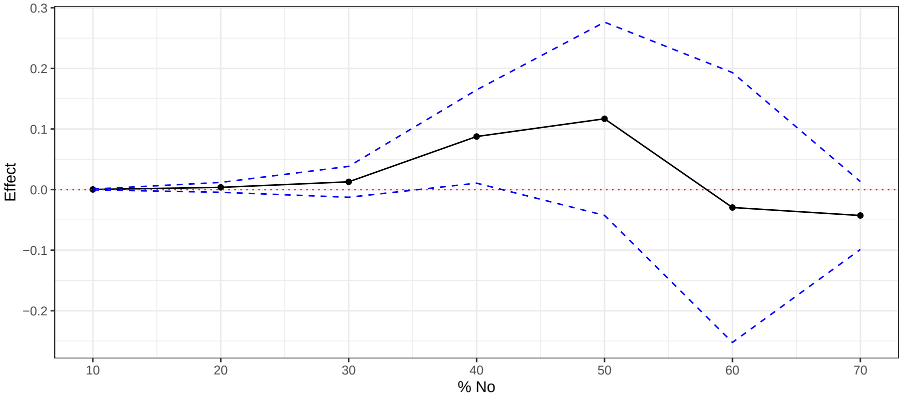
```
distr_reg <- function(i) {
  library(fixest); library(dplyr)
  d <- d %>% mutate(dum = case_when(VoteShareNo >= i ~ 1,
                                    VoteShareNo < i  ~ 0))
  fit <- feols(dum ~ DMilitaryPresence + share_allende70 + share_alessandri70 +
                 lnDistStgo + lnDistRegCapital + Pop70_pthousands +
                 sh_rural_70 | IDProv, data = d, weights = ~Pop70, vcov = "hetero")
  coef <- coefficients(fit)["DMilitaryPresence"]
  ll <- confint(fit)["DMilitaryPresence", "2.5 %"]
  ul <- confint(fit)["DMilitaryPresence", "97.5 %"]

  cbind(i, coef, ll, ul)
}

out <- as.data.frame(do.call("rbind", lapply(as.list(seq(10, 70, 10)),
                                             function(x) distr_reg(x))))
```

Distribution regression

```
ggplot(out, aes(x=i, y = coef)) + geom_point() + geom_line() +
  geom_line(aes(x=i, y=ll), colour = "blue", linetype="dashed") +
  geom_line(aes(x=i, y=ul), colour = "blue", linetype="dashed") +
  geom_hline(yintercept=0, col="red", linetype = "dotted") +
  scale_x_continuous(breaks = seq(10,70,10)) + labs(x="% No", y="Effect") + theme_bw()
```



Multiple endpoints

Testing multiple hypotheses increases the possibility of type I or type II error, i.e. false Positives or false Negatives.

But many times we do want to test multiple hypotheses (several behavioral outcomes in field experiments, votes for different parties in the same family/multiple elections etc)

Some popular strategies:

- Summarize different measures in a single one
- Adjust p-values for multiple comparisons

Principal Component Analysis

Reduces dimensionality of the outcomes by extracting shared variation along different dimensions.

Returns estimates of "principal components" which are orthogonal to each other.

We can use the principal components score (usually the first one) of the outcomes as a single outcome that "summarizes" the shared variation.

- Stata: `pca`
- R: `prcomp`, `princomp`

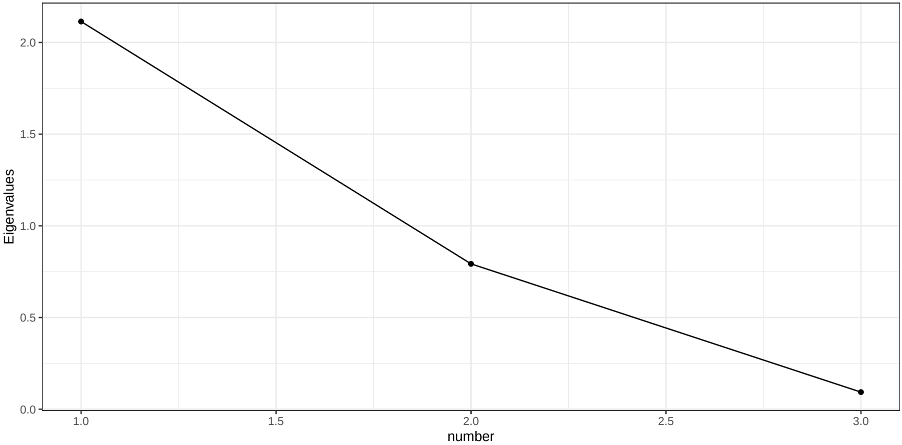
Principal Component Analysis

```r
vars <- c("Share_reg70_w2", "VoteShareNo", "VoteShareNo_pop70")
pca <- prcomp(d[,vars], center=T, scale=T)
pca
```

```
## Standard deviations (1, .., p=3):
## [1] 1.4540286 0.8902977 0.3052389
##
## Rotation (n x k) = (3 x 3):
##                          PC1         PC2        PC3
## Share_reg70_w2    -0.5861655 -0.55108749  0.5938961
## VoteShareNo       -0.4585680  0.82998321  0.3175582
## VoteShareNo_pop70 -0.6679261 -0.08620006 -0.7392187
```

Screeplot

```r
data.frame(number=c(1:3), Eigenvalues = (pca$sdev)^2) %>%
  ggplot(aes(x=number, y=Eigenvalues)) + geom_point() + geom_line() + theme_bw()
```



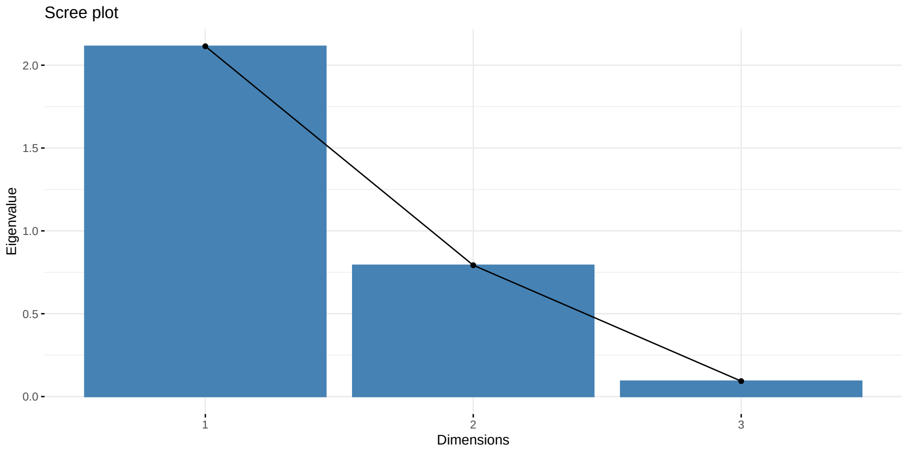
Principal Component Analysis

We can also pass the PCA output through the nice package `factoextra`.

Principal Component Analysis

We can also pass the PCA output through the nice package `factoextra`.

```
library(factoextra)
fviz_eig(pca, choice = "eigenvalue")
```



Principal Component Analysis

The first factor can be used as outcome in the regression.

Note that factors are estimates. To make correct inference we should bootstrap the procedure to incorporate their uncertainty in that of the treatment effect.

Sample code for R and Stata in this week's folder.

Inverse Covariance Weighting

Cyrus provides Stata and R code for computing ICW indexes

p-value adjustment

Most standard p-value adjustment methods implemented in base R through `p.adjust`.

```
# Generate p-values from different tests
set.seed(1)
p <- sort(runif(10, 0.03, 0.07))
d <- data.frame(p = p,
                bonferroni = p.adjust(p, method = "bonferroni"))
d
```

```
##             p bonferroni
## 1  0.03247145  0.3247145
## 2  0.03806728  0.3806728
## 3  0.04062035  0.4062035
## 4  0.04488496  0.4488496
## 5  0.05291413  0.5291413
## 6  0.05516456  0.5516456
## 7  0.05643191  0.5643191
## 8  0.06593559  0.6593559
## 9  0.06632831  0.6632831
## 10 0.06778701  0.6778701
```

Implementing p-value adjustments

Several adjustment options in Stata user-written packages, summarized by David McKenzie

Sample Stata data and code from McKenzie are posted in this week's GitHub folder.

In R there appear to be less resources.