

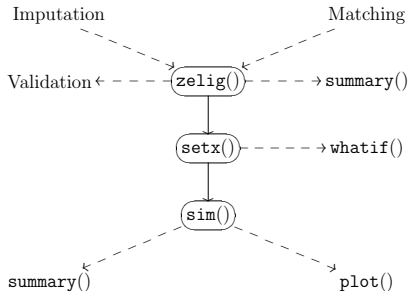
# Zelig and Matching in R with an Application to Conflict and Leader Tenure

Andrew Little  
PhD Candidate  
Department of Politics  
New York University  
andrew.little@nyu.edu

August 6, 2009

# Graphical Summary of Zelig

Figure 4.1: Main Zelig commands (solid arrows) and some options (dashed arrows)



# Zelig Syntax

```
zelig(formula, model, data, by, save.data, cite, ...)
```

# Zelig Syntax

`zelig(formula, model, data, by, save.data, cite, ...)`

- ▶ formula: normal R syntax

# Zelig Syntax

`zelig(formula, model, data, by, save.data, cite, ...)`

- ▶ formula: normal R syntax
- ▶ model: choose from endless list (`help.zelig("models")`)

# Zelig Syntax

`zelig(formula, model, data, by, save.data, cite, ...)`

- ▶ formula: normal R syntax
- ▶ model: choose from endless list (`help.zelig("models")`)
- ▶ data: can be from `amelia/matchit/both`

# Zelig Syntax

`zelig(formula, model, data, by, save.data, cite, ...)`

- ▶ formula: normal R syntax
- ▶ model: choose from endless list (`help.zelig("models")`)
- ▶ data: can be from `amelia/matchit/both`
- ▶ by: estimate the model for each value of a factor

# Zelig Syntax

`zelig(formula, model, data, by, save.data, cite, ...)`

- ▶ formula: normal R syntax
- ▶ model: choose from endless list (`help.zelig("models")`)
- ▶ data: can be from `amelia/matchit/both`
- ▶ by: estimate the model for each value of a factor
- ▶ additional parameters vary by model

# An Example - Ordered Probit Regression

```

> setwd("~/Documents/data/exercices")
> nes<-read.dta(file="nes92nomissclb.dta")
> names(nes)<-c("vote","b.approve","libcon","b.libcon","c.libcon","p.libcon","b.dist","c.dist","p.dist","
+ "mil.force","gulf","pid","school","gov.emp","union","faminc")
> m1<-zelig(as.factor(b.approve)~b.dist+econ.worse+gulf+faminc,data=nes,model="oprobit")
> x.gulf0<-setx(m1,gulf=0)
> x.gulf1<-setx(m1,gulf=1)
> sgulf<-sim(m1,x=x.gulf0,x1=x.gulf1)
> names(m1)
[1] "coefficients" "zeta"          "deviance"          "fitted.values" "lev"          "terms"
"df.residual"   "edf"
[9] "n"            "nobs"           "call"             "method"        "convergence"  "niter"
"Hessian"       "model"
[17] "xlevels"      "inv.link"
> names(sgulf)
[1] "x"          "x1"         "call"       "zelig.call" "par"        "qi$ev"      "qi$pr"      "qi$fd"
"qi$rr"

```

# An Example - Ordered Probit Regression pt 2

```
> summary(sgulf)
Model: oprobit
Number of simulations: 1000

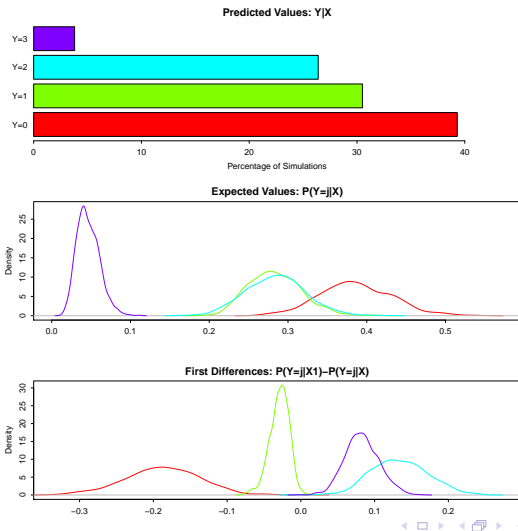
Values of X
(Intercept) b.dist econ.worse gulf faminc
1           1  2.081          3.99    0 47.31

Values of X1
(Intercept) b.dist econ.worse gulf faminc
1           1  2.081          3.99    1 47.31

...
First Differences: P(Y=j|X1)-P(Y=j|X)
      mean      sd  2.5%  97.5%
0 -0.18417 0.05338 -0.28716 -0.08492
1 -0.02900 0.01289 -0.05735 -0.00707
2  0.13230 0.03995  0.05800  0.21114
3  0.08087 0.02395  0.03659  0.12772

Risk Ratio: P(Y=j|X1)-P(Y=j|X)
      mean      sd  2.5%  97.5%
0  0.5259 0.10112 0.3548 0.7465
1  0.8979 0.04289 0.8071 0.9737
2  1.4784 0.18780 1.1823 1.9107
3  2.9651 0.96304 1.5967 5.1953
```

# Pretty Graphs from Zelig



# For Those Who Became Bayesian's Last Month

```
> m1.b<-zelig(b.approve~b.dist+econ.worse+gulfg+faminc,data=nes,model="oprobit.bayes")
```

```
> summary(m1)
```

Coefficients:

	Value	Std. Error	t value
b.dist	-0.447114	0.051053	-8.758
econ.worse	-0.348235	0.079531	-4.379
gulfg	0.558955	0.151616	3.687
faminc	0.002855	0.001990	1.435

Intercepts:

	Value	Std. Error	t value
0 1	-2.478	0.384	-6.457
1 2	-1.754	0.374	-4.693
2 3	-0.495	0.364	-1.361

```
> summary(m1.b)
```

Iterations = 1001:11000

Thinning interval = 1

Number of chains = 1

Sample size per chain = 10000

Mean, standard deviation, and quantiles for marginal posterior distributions.

	Mean	SD	2.5%	50%	97.5%
(Intercept)	2.473	0.388	1.732	2.469	3.241
b.dist	-0.446	0.052	-0.548	-0.446	-0.345
econ.worse	-0.349	0.080	-0.507	-0.349	-0.195
gulfg	0.560	0.150	0.266	0.560	0.853
faminc	0.003	0.002	-0.001	0.003	0.007
gamma2	0.708	0.092	0.530	0.706	0.900
gamma3	1.981	0.140	1.741	1.969	2.251

# Background

- ▶ Common goal in (social) sciences: determine causal effect of some  $x$  on outcome  $y$

# Background

- ▶ Common goal in (social) sciences: determine causal effect of some  $x$  on outcome  $y$
- ▶ Ideal(?) solution: randomized control trial (RCT): units sampled randomly from population, randomly treated.

# Background

- ▶ Common goal in (social) sciences: determine causal effect of some  $x$  on outcome  $y$
- ▶ Ideal(?) solution: randomized control trial (RCT): units sampled randomly from population, randomly treated.
- ▶ When RCT is not practical/ethical/feasible, what to do?  
Regression?

# Background

- ▶ Common goal in (social) sciences: determine causal effect of some  $x$  on outcome  $y$
- ▶ Ideal(?) solution: randomized control trial (RCT): units sampled randomly from population, randomly treated.
- ▶ When RCT is not practical/ethical/feasible, what to do?  
Regression?
- ▶ Big problem: model dependence.

## A Little Math (Notation from King et al 2007)

- ▶ Say we are interested in outcome  $Y_i$ ,  $i = 1, \dots, n$ .

## A Little Math (Notation from King et al 2007)

- ▶ Say we are interested in outcome  $Y_i$ ,  $i = 1, \dots, n$ .
- ▶ For each  $i$ ,  $X_i$  is an indicator for whether or not unit  $i$  is “treated.”

## A Little Math (Notation from King et al 2007)

- ▶ Say we are interested in outcome  $Y_i$ ,  $i = 1, \dots, n$ .
- ▶ For each  $i$ ,  $X_i$  is an indicator for whether or not unit  $i$  is “treated.”
- ▶ Each  $i$  also has some set of other covariates  $Z_i$ .

## A Little Math (Notation from King et al 2007)

- ▶ Say we are interested in outcome  $Y_i$ ,  $i = 1, \dots, n$ .
- ▶ For each  $i$ ,  $X_i$  is an indicator for whether or not unit  $i$  is “treated.”
- ▶ Each  $i$  also has some set of other covariates  $Z_i$ .
- ▶ Let  $Y_i(1)$  the observed outcome if unit  $i$  treated ( $X_i = 1$ ),  $Y_i(0)$  the outcome if not treated.

## A Little Math (Notation from King et al 2007)

- ▶ Say we are interested in outcome  $Y_i$ ,  $i = 1, \dots, n$ .
- ▶ For each  $i$ ,  $X_i$  is an indicator for whether or not unit  $i$  is “treated.”
- ▶ Each  $i$  also has some set of other covariates  $Z_i$ .
- ▶ Let  $Y_i(1)$  the observed outcome if unit  $i$  treated ( $X_i = 1$ ),  $Y_i(0)$  the outcome if not treated.
- ▶ So causal effect for unit  $i$  is  $Y_i(1) - Y_i(0)$ . Average Treatment Effect (ATE) is  $E[Y_i(1)|X_i = 1] - E[Y_i(0)|X_i = 0]$

## A Little Math (Notation from King et al 2007)

- ▶ Say we are interested in outcome  $Y_i$ ,  $i = 1, \dots, n$ .
- ▶ For each  $i$ ,  $X_i$  is an indicator for whether or not unit  $i$  is “treated.”
- ▶ Each  $i$  also has some set of other covariates  $Z_i$ .
- ▶ Let  $Y_i(1)$  the observed outcome if unit  $i$  treated ( $X_i = 1$ ),  $Y_i(0)$  the outcome if not treated.
- ▶ So causal effect for unit  $i$  is  $Y_i(1) - Y_i(0)$ . Average Treatment Effect (ATE) is  $E[Y_i(1)|X_i = 1] - E[Y_i(0)|X_i = 0]$
- ▶ Problem: for each unit, we only observe  $Y_i(1)$  OR  $Y_i(0)$ , not both.

## A Little Math (Notation from King et al 2007)

- ▶ Say we are interested in outcome  $Y_i$ ,  $i = 1, \dots, n$ .
- ▶ For each  $i$ ,  $X_i$  is an indicator for whether or not unit  $i$  is “treated.”
- ▶ Each  $i$  also has some set of other covariates  $Z_i$ .
- ▶ Let  $Y_i(1)$  the observed outcome if unit  $i$  treated ( $X_i = 1$ ),  $Y_i(0)$  the outcome if not treated.
- ▶ So causal effect for unit  $i$  is  $Y_i(1) - Y_i(0)$ . Average Treatment Effect (ATE) is  $E[Y_i(1)|X_i = 1] - E[Y_i(0)|X_i = 0]$
- ▶ Problem: for each unit, we only observe  $Y_i(1)$  OR  $Y_i(0)$ , not both.
- ▶  $Y_i = Y_i(1)X_i + Y_i(0)(1 - X_i)$ .

## A Little More Math

- ▶ If treatment is random,  $E[Y_i(1)|X_i = 1] - E[Y_i(0)|X_i = 0] = E[Y_i|X_i = 1] - E[Y_i|X_i = 0]$ . We can observe RHS, but want LHS.

## A Little More Math

- ▶ If treatment is random,  $E[Y_i(1)|X_i = 1] - E[Y_i(0)|X_i = 0] = E[Y_i|X_i = 1] - E[Y_i|X_i = 0]$ . We can observe RHS, but want LHS.
- ▶ However, if both treatment and outcome are related to covariates  $X_i$ , the above equation does not hold.

## A Little More Math

- ▶ If treatment is random,  $E[Y_i(1)|X_i = 1] - E[Y_i(0)|X_i = 0] = E[Y_i|X_i = 1] - E[Y_i|X_i = 0]$ . We can observe RHS, but want LHS.
- ▶ However, if both treatment and outcome are related to covariates  $X_i$ , the above equation does not hold.
- ▶ Most basic solution: only keep control observations that exactly match a treated unit on the covariates, weight accordingly.

## A Little More Math

- ▶ If treatment is random,  $E[Y_i(1)|X_i = 1] - E[Y_i(0)|X_i = 0] = E[Y_i|X_i = 1] - E[Y_i|X_i = 0]$ . We can observe RHS, but want LHS.
- ▶ However, if both treatment and outcome are related to covariates  $X_i$ , the above equation does not hold.
- ▶ Most basic solution: only keep control observations that exactly match a treated unit on the covariates, weight accordingly.
- ▶ Another common solution: for each treated observation, select another one (or more) that is “close” on each of the covariates (nearest neighbor matching).

## A Little More Math

- ▶ If treatment is random,  $E[Y_i(1)|X_i = 1] - E[Y_i(0)|X_i = 0] = E[Y_i|X_i = 1] - E[Y_i|X_i = 0]$ . We can observe RHS, but want LHS.
- ▶ However, if both treatment and outcome are related to covariates  $X_i$ , the above equation does not hold.
- ▶ Most basic solution: only keep control observations that exactly match a treated unit on the covariates, weight accordingly.
- ▶ Another common solution: for each treated observation, select another one (or more) that is “close” on each of the covariates (nearest neighbor matching).
- ▶ Often combined with a model for treatment and matching also done on “propensity score”

## A Little More Math

- ▶ If treatment is random,  $E[Y_i(1)|X_i = 1] - E[Y_i(0)|X_i = 0] = E[Y_i|X_i = 1] - E[Y_i|X_i = 0]$ . We can observe RHS, but want LHS.
- ▶ However, if both treatment and outcome are related to covariates  $X_i$ , the above equation does not hold.
- ▶ Most basic solution: only keep control observations that exactly match a treated unit on the covariates, weight accordingly.
- ▶ Another common solution: for each treated observation, select another one (or more) that is “close” on each of the covariates (nearest neighbor matching).
- ▶ Often combined with a model for treatment and matching also done on “propensity score”
- ▶ Relative new method: “coarsen” variables into categories and then perform exact matching

# A Simulation - Setup

- ▶ Saw we want to estimate effect of  $x$  on  $y$ .  $z$  is a confounding variable.

# A Simulation - Setup

- ▶ Saw we want to estimate effect of  $x$  on  $y$ .  $z$  is a confounding variable.
- ▶ Simulate with true DGP:

$$x = \begin{cases} 1 & \text{if } -(z - .4)^2 + \epsilon_1 > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$y = 0.1x + 5((z - .4)^2) + \epsilon_2$$

# A Simulation - Setup

- ▶ Saw we want to estimate effect of  $x$  on  $y$ .  $z$  is a confounding variable.
- ▶ Simulate with true DGP:

$$x = \begin{cases} 1 & \text{if } -(z - .4)^2 + \epsilon_1 > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$y = 0.1x + 5((z - .4)^2) + \epsilon_2$$

- ▶ Can we recover  $\beta_x = 0.1$  without knowing the functional form of  $x$  and  $y$ ?

# A Simulation - R Code - Naive Models

```

> set.seed(1010101)
> z<-runif(1000,0,1)
> z.t<-(z-.4)^2
> x<-rnorm(1000,0,.2)-z.t>(0)
> y<-.1*x+5*z.t+rnorm(1000,0,.2)
> simdata<-as.data.frame(cbind(z,z.t,x,y))
> summary(zelig(y~x,model="ls",data=simdata,cite=FALSE))$coefficients
      Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.5523    0.01960  28.182 5.009e-129
x            -0.1866    0.03337  -5.593 2.879e-08
> summary(zelig(y~x+z,model="ls",data=simdata,cite=FALSE))$coefficients
      Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.007393    0.03074  -0.2405 8.100e-01
x            -0.058243    0.02826  -2.0608 3.958e-02
z             1.022492    0.04770  21.4380 3.785e-84
> summary(zelig(y~x+z.t,model="ls",data=simdata,cite=FALSE))$coefficients
      Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.005176    0.01105   0.4683 6.397e-01
x             0.094229    0.01409   6.6867 3.795e-11
z.t           4.992997    0.07010  71.2302 0.000e+00

```

# A Simulation - R Code - Matching!

```
> simdata<-as.data.frame(cbind(z,z.t,x,y))
> match1<-matchit(x~z,data=simdata)
> summary(match1)
```

Call:

```
matchit(formula = x ~ z, data = simdata)
```

Summary of balance for all data:

	Means Treated	Means Control	SD Control	Mean Diff	eQQ Med	eQQ Mean	eQQ Max
distance	0.373	0.330	0.105	0.042	0.049	0.043	0.073
z	0.422	0.547	0.297	-0.126	0.144	0.126	0.225

Summary of balance for matched data:

	Means Treated	Means Control	SD Control	Mean Diff	eQQ Med	eQQ Mean	eQQ Max
distance	0.373	0.371	0.085	0.002	0.001	0.002	0.010
z	0.422	0.426	0.229	-0.004	0.002	0.005	0.025

Percent Balance Improvement:

	Mean Diff.	eQQ Med	eQQ Mean	eQQ Max
distance	96.27	98.21	95.82	86.93
z	96.74	98.36	96.30	89.10

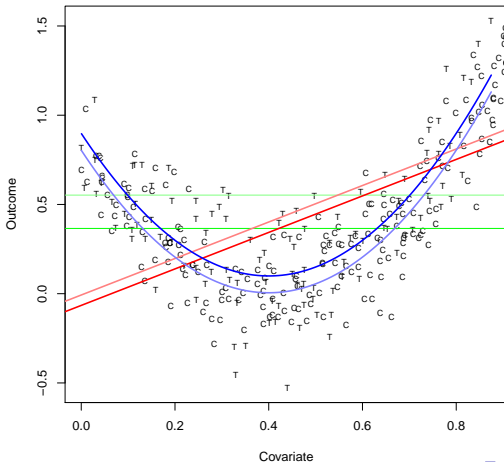
Sample sizes:

	Control	Treated
All	655	345
Matched	345	345
Unmatched	310	0
Discarded	0	0

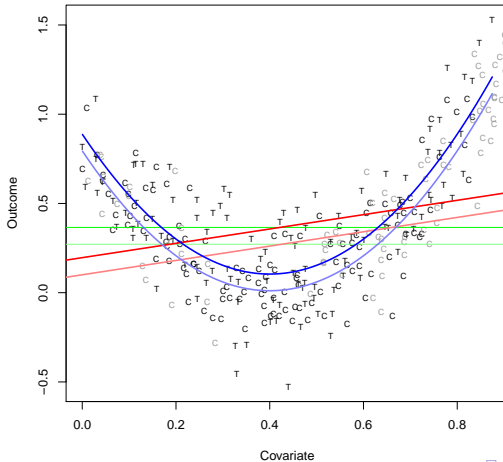
# A Simulation - R Code - Post-Matching Analysis

```
> match1.d<-match.data(match1)
> zelig(y~x,model="ls",data=match1.d,cite=FALSE)$coefficients
(Intercept)      x
  0.27147      0.09424
> zelig(y~x+z,model="ls",data=match1.d,cite=FALSE)$coefficients
(Intercept)      x      z
  0.10038      0.09589      0.40166
> zelig(y~x+z.t,model="ls",data=match1.d,cite=FALSE)$coefficients
(Intercept)      x      z.t
  0.01098      0.09340      4.89970
```

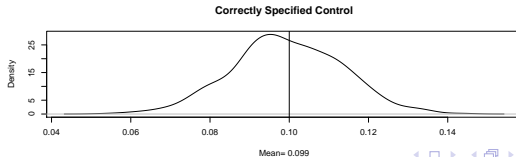
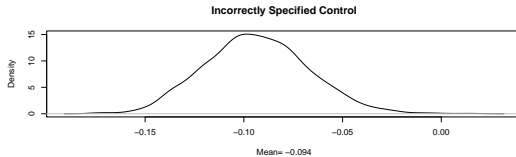
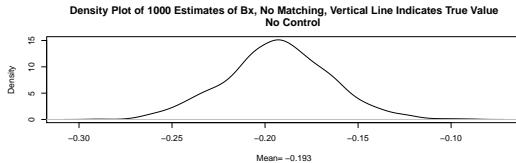
# Visualizing the ATE - No Matching



# Visualizing the ATE - No Matching

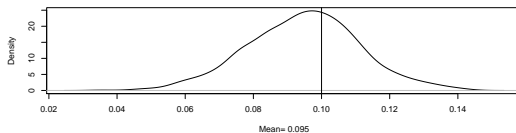


# Results from Monte Carlo Simulation - Naive

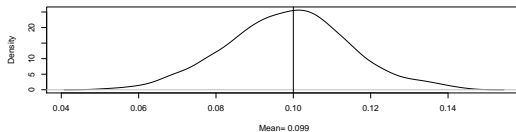


# Results from Monte Carlo Simulation - NN Matching 1

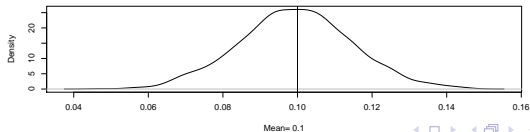
Density Plot of 1000 Estimates of Bx, NN Matching/Wrong Specification, Vertical Line Indicates True Value  
No Control



Incorrectly Specified Control

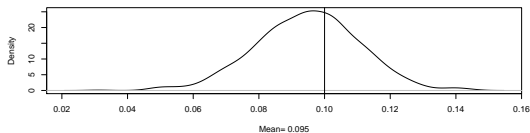


Correctly Specified Control

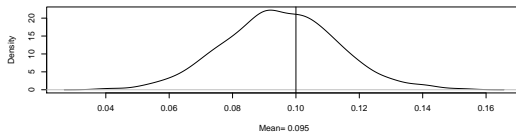


# Results from Monte Carlo Simulation - NN Matching 2

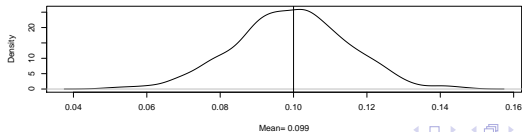
Density Plot of 1000 Estimates of Bx, NN Matching/Right Specification, Vertical Line Indicates True Value  
No Control



Incorrectly Specified Control

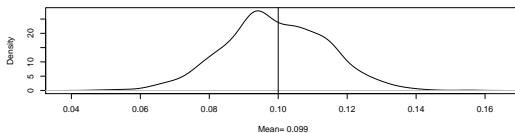


Correctly Specified Control

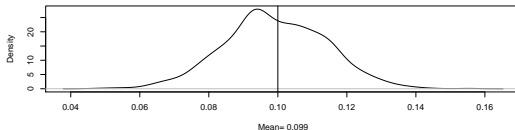


# Results from Monte Carlo Simulation - CEM Matching

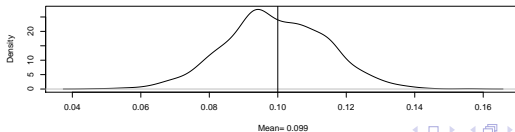
Density Plot of 1000 Estimates of Bx, NN Matching/CEM Specification, Vertical Line Indicates True Value  
No Control



Incorrectly Specified Control



Correctly Specified Control



# General Syntax

```
matchit(formula, data, method = "nearest", discard = "none",  
reestimate = FALSE, ...)
```

# General Syntax

```
matchit(formula, data, method = "nearest", discard = "none",  
reestimate = FALSE, ...)
```

- ▶ formula: standard R format  $y \sim x_1 + x_2$  etc. General standard is to put in all covariates that you use in post-estimation. Make sure they are all pre-treatment!

# General Syntax

```
matchit(formula, data, method = "nearest", discard = "none",  
reestimate = FALSE, ...)
```

- ▶ formula: standard R format  $y \sim x_1 + x_2$  etc. General standard is to put in all covariates that you use in post-estimation. Make sure they are all pre-treatment!
- ▶ method: Lots to choose from! Default is nearest neighbor. More on next slide.

# General Syntax

```
matchit(formula, data, method = "nearest", discard = "none",  
reestimate = FALSE, ...)
```

- ▶ formula: standard R format  $y \sim x_1 + x_2$  etc. General standard is to put in all covariates that you use in post-estimation. Make sure they are all pre-treatment!
- ▶ method: Lots to choose from! Default is nearest neighbor. More on next slide.
- ▶ discard, reestimate: can get rid of matches that don't fit some criteria.

# A Few Methods

- ▶ Nearest neighbor: can specify k:1 matching (*ratio=*), replacement, various distance measures.

# A Few Methods

- ▶ Nearest neighbor: can specify k:1 matching (*ratio=*), replacement, various distance measures.
- ▶ Genetic matching: slow, but finds “best” balance. Specify k:1.

## A Few Methods

- ▶ Nearest neighbor: can specify k:1 matching (*ratio=*), replacement, various distance measures.
- ▶ Genetic matching: slow, but finds “best” balance. Specify k:1.
- ▶ CEM: Can specify cutpoints, force k-to-k matching.

## A Few Methods

- ▶ Nearest neighbor: can specify k:1 matching (*ratio=*), replacement, various distance measures.
- ▶ Genetic matching: slow, but finds “best” balance. Specify k:1.
- ▶ CEM: Can specify cutpoints, force k-to-k matching.
- ▶ Others: Optimal, Full, Exact, Subclass.

## Drawbacks to Matching

1. Reduces the sample size, may lead to less precise estimates.
2. Leads to even more modeling decisions: to match or not to match, what technique, 1:1 vs. k:1, calipers, cutpoints for CEM, etc.

## Drawbacks to Matching

1. Reduces the sample size, may lead to less precise estimates.
  - ▶ Matching enthusiasts respond that the observations dropped are ones that could lead to false inference.
  
2. Leads to even more modeling decisions: to match or not to match, what technique, 1:1 vs. k:1, calipers, cutpoints for CEM, etc.

# Drawbacks to Matching

1. Reduces the sample size, may lead to less precise estimates.
  - ▶ Matching enthusiasts respond that the observations dropped are ones that could lead to false inference.
  - ▶ Matching may also *reduce* the standard error of estimates by reducing the relationship between the treatment and other covariate(s).
2. Leads to even more modeling decisions: to match or not to match, what technique, 1:1 vs. k:1, calipers, cutpoints for CEM, etc.

# Drawbacks to Matching

1. Reduces the sample size, may lead to less precise estimates.
  - ▶ Matching enthusiasts respond that the observations dropped are ones that could lead to false inference.
  - ▶ Matching may also *reduce* the standard error of estimates by reducing the relationship between the treatment and other covariate(s).
2. Leads to even more modeling decisions: to match or not to match, what technique, 1:1 vs. k:1, calipers, cutpoints for CEM, etc.
  - ▶ But these likely won't matter too much, and should greatly reduce the importance of other modeling decisions

# Background

- ▶ Huge literature on the question of whether or not being involved in a conflict makes leaders more or less secure in office.

# Background

- ▶ Huge literature on the question of whether or not being involved in a conflict makes leaders more or less secure in office.
- ▶ One problem not dealt with: need to be in office to get kicked out. (Can almost solve with matching, but I have a better way now, sadly with no interesting R angle.)

# Background

- ▶ Huge literature on the question of whether or not being involved in a conflict makes leaders more or less secure in office.
- ▶ One problem not dealt with: need to be in office to get kicked out. (Can almost solve with matching, but I have a better way now, sadly with no interesting R angle.)
- ▶ Other problem: non-random selection. In fact, strategic selection. Tons of theory about this too, empirical record shaky.

# Background

- ▶ Huge literature on the question of whether or not being involved in a conflict makes leaders more or less secure in office.
- ▶ One problem not dealt with: need to be in office to get kicked out. (Can almost solve with matching, but I have a better way now, sadly with no interesting R angle.)
- ▶ Other problem: non-random selection. In fact, strategic selection. Tons of theory about this too, empirical record shaky.
- ▶ While we can never fully solve this (measurement error, unknown covariates), matching vastly superior to regression with controls which is vastly superior to doing nothing.

# Background

- ▶ Huge literature on the question of whether or not being involved in a conflict makes leaders more or less secure in office.
- ▶ One problem not dealt with: need to be in office to get kicked out. (Can almost solve with matching, but I have a better way now, sadly with no interesting R angle.)
- ▶ Other problem: non-random selection. In fact, strategic selection. Tons of theory about this too, empirical record shaky.
- ▶ While we can never fully solve this (measurement error, unknown covariates), matching vastly superior to regression with controls which is vastly superior to doing nothing.
- ▶ Requires a little customization of matching (teachable moment?).

## Data/Setup

- ▶ The data: 10,000 leader-year observations (Archigos). Info about leader, economy, regime type, days survived.

## Data/Setup

- ▶ The data: 10,000 leader-year observations (Archigos). Info about leader, economy, regime type, days survived.
- ▶ Augment this data with MID disputes, down to day of start and end. Also care about hostility level

# Data/Setup

- ▶ The data: 10,000 leader-year observations (Archigos). Info about leader, economy, regime type, days survived.
- ▶ Augment this data with MID disputes, down to day of start and end. Also care about hostility level
- ▶ Run a Cox duration model with time-varying covariates.

# Data/Setup

- ▶ The data: 10,000 leader-year observations (Archigos). Info about leader, economy, regime type, days survived.
- ▶ Augment this data with MID disputes, down to day of start and end. Also care about hostility level
- ▶ Run a Cox duration model with time-varying covariates.
- ▶ Very naive estimate: consider entire year of conflict ending treatment, none after that.

## Data/Setup

- ▶ The data: 10,000 leader-year observations (Archigos). Info about leader, economy, regime type, days survived.
- ▶ Augment this data with MID disputes, down to day of start and end. Also care about hostility level
- ▶ Run a Cox duration model with time-varying covariates.
- ▶ Very naive estimate: consider entire year of conflict ending treatment, none after that.
- ▶ Less naive estimate Consider all post-conflict period treatment, control for various things.

# Data/Setup

- ▶ The data: 10,000 leader-year observations (Archigos). Info about leader, economy, regime type, days survived.
- ▶ Augment this data with MID disputes, down to day of start and end. Also care about hostility level
- ▶ Run a Cox duration model with time-varying covariates.
- ▶ Very naive estimate: consider entire year of conflict ending treatment, none after that.
- ▶ Less naive estimate Consider all post-conflict period treatment, control for various things.
- ▶ Hopefully even less naive estimate: at the end of a conflict, match leaders of belligerents to comparable leaders who are not in a post-conflict phase at the time. See who lasts longer.

# Naive Model

```
> nmodel<-zelig(Surv(t0, t, d) ~ incon + postcon + tmixed + tdeparl + tdempres + trans
+
+ civwar + lngdpcapL + growth + txgrowth
+
+ tropen2L + dopen2 + lnpop + age0 +txage0 + entry1 + txentry1
+
+ , na.action=na.exclude, data=lp,model="coxph",cite=FALSE)
```

```
> print(nmodel)
```

Call:

```
zelig(formula = Surv(t0, t, d) ~ incon + postcon + tmixed + tdeparl +
      tdempres + trans + civwar + lngdpcapL + growth + txgrowth +
      tropen2L + dopen2 + lnpop + age0 + txage0 + entry1 + txentry1,
      model = "coxph", data = lp, cite = FALSE, na.action = na.exclude)
```

	coef	exp(coef)	se(coef)	z	p
inconTRUE	-3.32e-01	0.717	1.81e-01	-1.834	6.7e-02
postcon	-3.50e-01	0.705	1.49e-01	-2.349	1.9e-02

(other covariates)

Likelihood ratio test=621 on 17 df, p=0 n= 9593

# Setting Up Matched Data pt 1

```
> tocem<-na.omit(subset(lp,select=c(incon,postcon,tmixed,tdeparl,tdepres,  
+ trans,civwar,lngdpcapL,growth,txgrowth,tropen2L,dopen2,lnpop,age0,txage0,  
+ entry1,txentry1,t0,t,d,ccode,tcount,ecount,yio,incon,leadid)))  
>  
> cut.base<-list(tmixed=c(.001,.999),tdeparl=c(.001,.999),tdepres=c(.001,.999),trans=c(.001,.999),  
+ civwar=.5,lngdpcapL=seq(-1.5,3.5,length.out=10),growth=seq(-1,.6,length.out=10),  
+ tropen2L=seq(0,6,length.out=10),dopen2=seq(-2.3,4.9,length.out=10),  
+ lnpop=seq(-2,7,length.out=10),age0=seq(15,85,length.out=10),entry1=.5)  
> cem.base.init<-matchit(postcon~tmixed + tdeparl + tdepres + trans+ civwar + lngdpcapL + growth  
+ +tropen2L + dopen2 + lnpop + age0 + entry1,data=tocem,  
+ method="cem",cutpoints=cut.base)
```

## Setting Up Matched Data pt 2

```
> tocem2<-tocem
> tocem2$subclass<-cem.base.init$subclass
> tocem2<-subset(tocem2,!is.na(subclass))
> tocem3<-NULL
> for (i in unique(tocem2$subclass)){
+   tocem.temp<-subset(tocem2,subclass==i)
+   treat.leads<-tocem.temp$leadid[tocem.temp$postcon==1]
+   tocem.temp<-subset(tocem.temp,postcon==1 | !(leadid %in% treat.leads))
+   tocem3<-rbind(tocem3,tocem.temp)
+ }
>
> cem.base.final<-matchit(postcon~tmixed + tdemparl + tdempres + trans+ civwar + lngdpcapL + growth
+   +tropen2L + dopen2 + lnpop + age0 + entry1,data=tocem3,
+   method="cem",cutpoints=cut.base)
>
> lp.base.init<-match.data(cem.base.init)
> names(tocem3)[names(tocem3)=="subclass"]<-"sclass"
> lp.base.final<-match.data(cem.base.final)
```

# The Results!

```

> mmodel.base.init<-zelig(Surv(t0, t, d) ~ postcon + tmixed + tdemparl + tdempres + trans
+                               + civwar + lngdpcapL + growth + txgrowth
+                               + tropen2L + dopen2 + lnpop + age0 +txage0 + entry1 + txentry1
+                               , na.action=na.exclude, data=lp.base.init,model="coxph",cite=FALSE)
>
> mmodel.base.final<-zelig(Surv(t0, t, d) ~ postcon + tmixed + tdemparl + tdempres + trans
+                               + civwar + lngdpcapL + growth + txgrowth
+                               + tropen2L + dopen2 + lnpop + age0 +txage0 + entry1 + txentry1
+                               , na.action=na.exclude, data=lp.base.final,model="coxph",cite=FALSE)
> print(mmodel.base.init)
      coef exp(coef) se(coef)      z      p
postcon  6.47e-02   1.067 1.93e-01  0.3344 7.4e-01
(...)
> print(mmodel.base.final)
      coef exp(coef) se(coef)      z      p
postcon -1.48e-01   0.862 2.13e-01 -0.696 4.9e-01
(...)

```

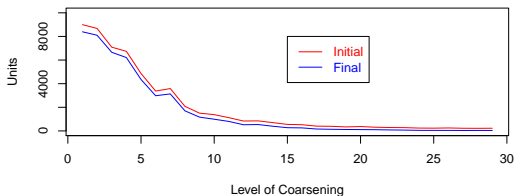
# Now Lets Play with the Level of Coarsening

```
make.cem<-function(len){
  cut<-list(tmixed=c(.001,.999),tdemparl=c(.001,.999),tdempres=c(.001,.999),trans=c(.001,.999),
  civwar=.5,lngdpcapL=seq(-1.5,3.5,length.out=len),growth=seq(-1,.6,length.out=len),
  tropen2L=seq(0,6,length.out=len),dopen2=seq(-2.3,4.9,length.out=len),
  lnpop=seq(-2,7,length.out=len),age0=seq(15,85,length.out=len),entry1=.5)
  ...
  [DO THE SAME PROCESS]
  ...
  return(list(init=lp.base.init,final=lp.base.final))
}

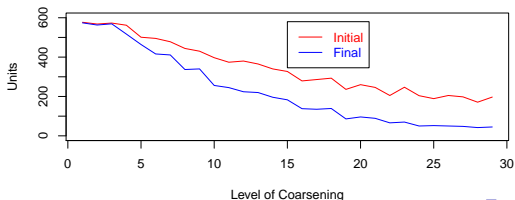
for (i in 2:20){
  mdata<-make.cem(i)
  ...
  [COLLECT NUMBER TREATED/CONTROL, RUN MODEL, EXTRACT COEFFICIENTS]
}
```

# Number of Treated and Control Units

Number of Control Units

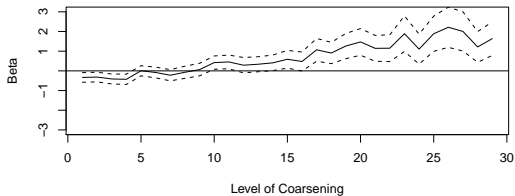


Number of Treated Units

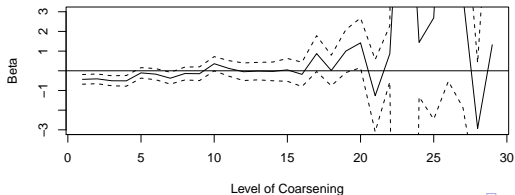


# Estimated Treatment Effect

Estimated Treatment Effect - Initial



Estimated Treatment Effect - Final



## A Few References

- ▶ Daniel Ho, Kosuke Imai, Gary King, and Elizabeth Stuart. Matching as Nonparametric Preprocessing for Reducing Model Dependence in Parametric Causal Inference. *Political Analysis*, Vol. 15 (2007): Pp. 199-236.
- ▶ Kosuke Imai, Gary King, and Olivia Lau. "Toward A Common Framework for Statistical Analysis and Development" *Journal of Computational and Graphical Statistics*, Vol. 17, No. 4 (December), pp. 892-913
- ▶ King, Gary; James Honaker, Anne Joseph, and Kenneth Scheve. Analyzing Incomplete Political Science Data: An Alternative Algorithm for Multiple Imputation, *American Political Science Review*, Vol. 95, No. 1 (March, 2001): Pp. 49-69
- ▶ Introducing Archigos: A Data Set of Political Leaders, 1875–2003. Co-authored with Kristian Skrede Gleditsch and Giacomo Chiozza. *Journal of Peace Research*, Vol. 46, No. 2, (March) 2009: 269-283.
- ▶ Which Way Out? The Manner and Consequences of Losing Office. *Journal of Conflict Resolution*, Vo. 53, No. 6 (December) 2008: 771-794.

# Quick Example of Multiple Imputation with Amelia

```

> library(Amelia)
> library(Zelig)
> lp <- read.dta("WW0-Replication/Poolfail.dta", convert.dates=FALSE)
> toimp<-subset(lp,select=c(t0,t,d,tmixed, tdempar1, tdempres, trans, civwar, lngdpcapL, growth,
+   tropen2L, dopen2, lnpop, age0, entry1, powtimes, initiator2, defender2, inherit,
+   dwinsh, dlosesesh, ddrawsh, dwinwar, dlosewar, ddrawwar, ccode,year,leadid))
>
> imp<-amelia(toimp,
idvars=c("d","dwinsh","dlosesesh","ddrawsh","dwinwar","dlosewar","ddrawwar","t","ccode","leadid"),m=6)
-- Imputation 1 --
  1  2  3  4  5
...
> m.ni<-zelig(Surv(t0,t,d)~lngdpcapL+growth+tropen2L,model="coxph",data=lp,cite=FALSE)
> m.imp<-zelig(Surv(t0,t,d)~lngdpcapL+growth+tropen2L,model="coxph",data=m.imp$imputations,cite=FALSE)
> summary(m.ni)$coefficients
      coef exp(coef) se(coef)      z Pr(>|z|)
lngdpcapL  0.1705   1.18591  0.02772   6.151 7.687e-10
growth    -2.3532   0.09507  0.30490  -7.718 1.188e-14
tropen2L  -1.0271   0.35803  0.14408  -7.129 1.011e-12
> summary(m.imp)$coefficients
      Value Std. Error t-stat  p-value
lngdpcapL  0.1445    0.02703   5.348 1.635e-07
growth    -2.0384    0.29441  -6.923 2.974e-11
tropen2L  -0.7383    0.13436  -5.494 4.598e-06

```