# Bias

Drew Dimmery drewd@nyu.edu

February 26, 2015

## Measurement Error Sim

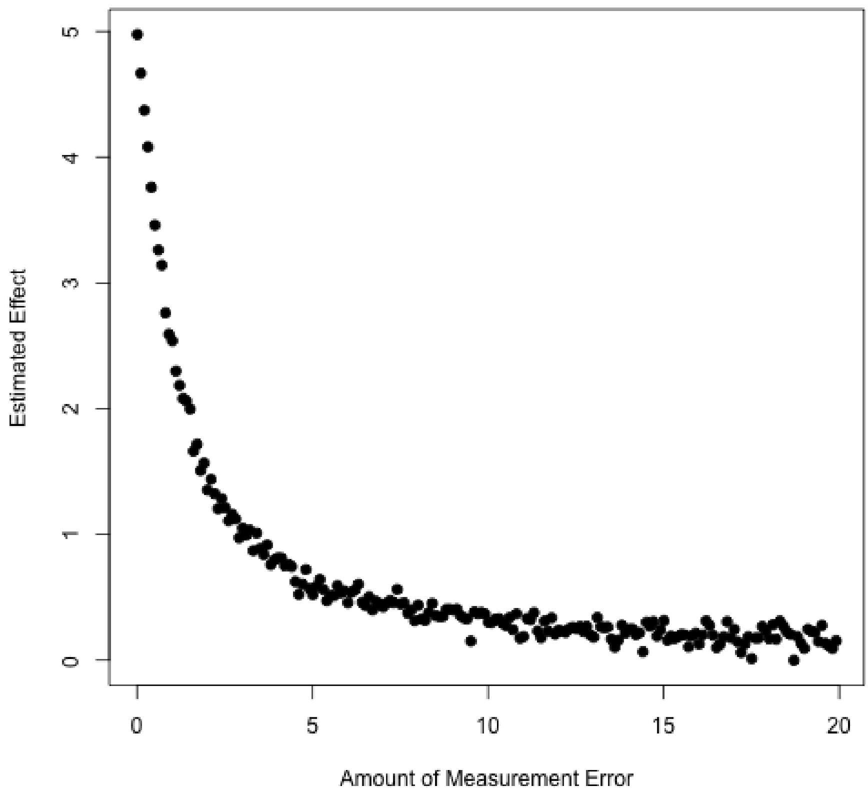
```
Dstar <- rnorm(10000)
D <- ifelse(Dstar > 0, 1 , 0)
Y <- 10 + 5*D + rnorm(10000,0,2)
simME <- function(stdev) {
  Dnew <- ifelse(Dstar + rnorm(10000,0,stdev) > 0,1,0)
  mean(Y[Dnew==1]) - mean(Y[Dnew==0])
}
sdevs <- seq(0.01,20,.1)
eff <- sapply(sdevs,simME)
```

## Plot it

```
plot(sdevs,eff,xlab='Amount of Measurement Error',ylab='Estimated Effect',pch=19)
```

## Sensitivity Analysis

- I'm going to walk you through how to do a generalized version of the Imbens (2003) method.
- It may be easier to use one of the canned routines for your homework, though.
- We're going to keep working with Pat's data, since we already have it handy.
- Imbens process:
  - Simulate (or imagine simulating) an unobserved confounder like the following:
    $Y_d|X, U \sim \mathcal{N}(\tau d + \boldsymbol{\beta}' X + \delta U, \sigma^2)$
    $D|X, U \sim f(\boldsymbol{\gamma}' X + \alpha U)$ (with $f$ known)



Estimated Effect
0
1
2
3
4
5
0
5
10
15
20
Amount of Measurement Error


Figure 1:

- That is, $Y_1, Y_0 \perp D|X, U$
- So we want to generate an additively linear confounder with both $D$ and $Y$.

## Example

```
require(foreign)
d <- read.dta("gwdataset.dta")
```

```
## Warning in read.dta("gwdataset.dta"): value labels ('q2') for 'jan07_q2'
## are missing
```

```
zips <- read.dta("zipcodetostate.dta")
zips<-unique(zips[,c("statenum","statefromzipfile")])
pops <- read.csv("population_ests_2013.csv")
pops$state <- tolower(pops$NAME)
d$getwarmord <- as.double(d$getwarmord)
# And estimate primary model of interest:
out<-lm(getwarmord~ddt_week+educ_hsless+educ_coll+educ_postgrad+educ_dk+party_rep+party_lear
```

## Generate a confounder

- For our analysis, $Y$ is belief in global warming and $D$ is local variation in temperature.
- We want to standardize these variables first.

. . .

```
d$getwarmord <- scale(d$getwarmord)
d$ddt_week <- scale(d$ddt_week)
genConfound<-function(alpha,delta) {
  e <- rnorm(nrow(d),0,1)
  U <- alpha * d$ddt_week + delta * d$getwarmord + e
  return(U)
}
```

. . .

- So we can vary partial correlations with $D$ and $Y$ by varying `alpha` and `delta`.

. . .

```r
U1<-genConfound(0,2)
U2<-genConfound(10,10)
c(D=cor(U1,d$ddt_week),Y=cor(U1,d$getwarmord))
```

```
##          D          Y 
## 0.03851302 0.89405287
```

```r
c(D=cor(U2,d$ddt_week),Y=cor(U2,d$getwarmord))
```

```
##         D         Y 
## 0.7200823 0.7198210
```

```r
c(D=coef(lm(paste0("ddt_week~U1+",X),d))["U1"],Y=coef(lm(paste0("getwarmord~U1+",X),d))["U1'
```

```
##        D.U1        Y.U1 
## 0.006572182 0.387065938
```

```r
c(D=coef(lm(paste0("ddt_week~U2+",X),d))["U2"],Y=coef(lm(paste0("getwarmord~U2+",X),d))["U2'
```

```
##       D.U2       Y.U2 
## 0.03236123 0.06691239
```

## Continued

- More importantly, we can see how this changes our estimate of the treatment effect:

. . .

```r
out <- lm(paste0("getwarmord~ddt_week+",X),d)
coef(out)["ddt_week"]
```

```
##   ddt_week 
## 0.03618393
```

```r
coef(lm(paste0("getwarmord~ddt_week+U1+",X),d))["ddt_week"]
```

```
##    ddt_week 
## 0.008236237
```

```r
coef(lm(paste0("getwarmord~ddt_week+U2+",X),d))["ddt_week"]
```

```
##   ddt_week
## -0.9904723
```

- Now we want to do this over a larger number of values of `alpha` and `delta`

. . .

```r
alphas<-rnorm(100,0,.5)
deltas<-rnorm(100,0,.5)
results<-NULL
for(i in seq_len(length(alphas))) {
  U<-genConfound(alphas[i],deltas[i])
  corD<-cor(U,d$ddt_week)
  corY<-cor(U,d$getwarmord)
  estTE<-coef(lm(paste0("getwarmord~ddt_week+U+",X),d))["ddt_week"]
  names(estTE)<-NULL
  res<-c(estTE=estTE,corD=corD,corY=corY)
  results<-rbind(results,res)
}
results<-cbind(results,TEchange=(results[,"estTE"]-coef(out)["ddt_week"]))
```

## More

```r
resultsSens<-NULL
for(i in seq_len(length(alphas))) {
  U<-genConfound(alphas[i],deltas[i])
  corD<-cor(U,d$ddt_week)
  corY<-cor(U,d$getwarmord)
  estTE<-coef(lm(paste0("getwarmord~ddt_week+U+",Xsens),d))["ddt_week"]
  names(estTE)<-NULL
  res<-c(estTE=estTE,corD=corD,corY=corY)
  resultsSens<-rbind(resultsSens,res)
}
resultsSens<-cbind(resultsSens,TEchange=(resultsSens[,"estTE"]-coef(out)["ddt_week"]))
```

## Plot Simulation Code

```r
color<-ifelse(results[,"estTE"]<=.5*coef(out)["ddt_week"],"red",NA)
color<-ifelse(is.na(color) & results[,"estTE"]>=1.5*coef(out)["ddt_week"],"blue",color)
```

```
color<-ifelse(is.na(color),"green",color)
plot(results[,"corD"],results[,"corY"],col=color,xlab="correlation with D",ylab="correlation
vars<-strsplit(X,"[+]",perl=TRUE)[[1]]
vars<-vars[grep("factor",vars,invert=TRUE)]
for(v in vars) {
  corD<-with(d,cor(get(v),d$ddt_week))
  corY<-with(d,cor(get(v),d$getwarmord))
  points(corD,corY,pch="+",col="black")
}
abline(v=0,col="grey",lty=3)
abline(h=0,col="grey",lty=3)
```

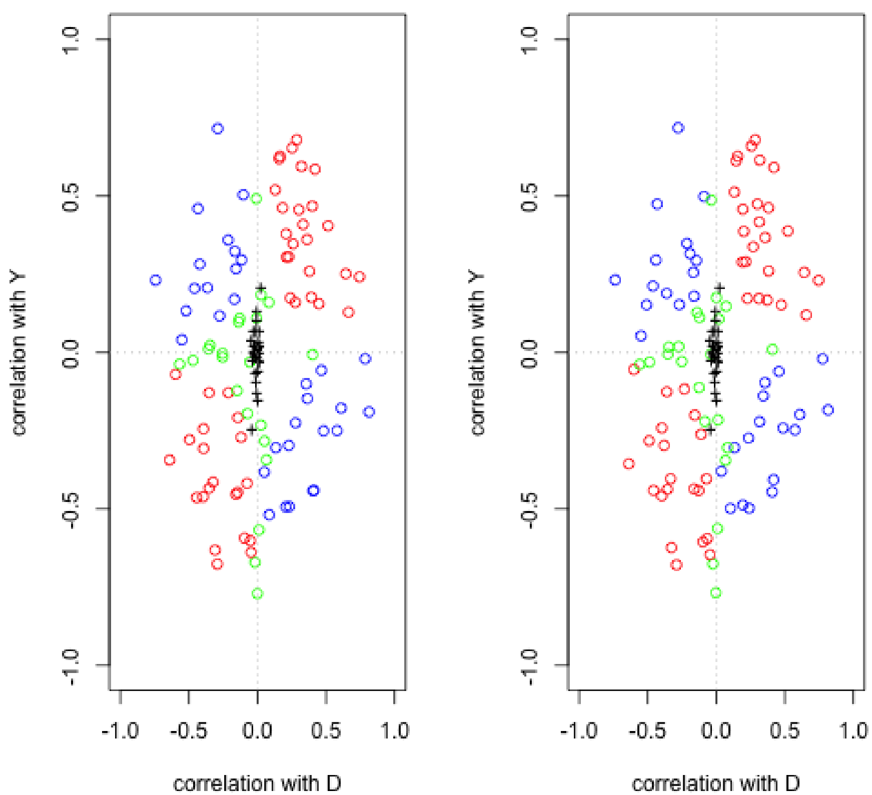
# Plot Sensitive Model

```
colorS<-ifelse(resultsSens[,"estTE"]<=.5*coef(out)["ddt_week"],"red",NA)
colorS<-ifelse(is.na(colorS) & resultsSens[,"estTE"]>=1.5*coef(out)["ddt_week"],"blue",color
colorS<-ifelse(is.na(colorS),"green",colorS)
plot(resultsSens[,"corD"],resultsSens[,"corY"],col=color,xlab="correlation with D",ylab="cor
vars<-strsplit(Xsens,"[+]",perl=TRUE)[[1]]
for(v in vars) {
  corD<-with(d,cor(get(v),d$ddt_week))
  corY<-with(d,cor(get(v),d$getwarmord))
  points(corD,corY,pch="+",col="black")
}
abline(v=0,col="grey",lty=3)
abline(h=0,col="grey",lty=3)
```

# Plot of the Results

# Blackwell (2013)

- Instead, imagine a function which defines the confounding.
- $q(d, x) = E[Y_i(d)|D_i = d, X_i = x] - E[Y_i(d)|D_i = 1 - d, X_i = x]$
- Treated counterfactuals always higher (lower): $q(d, x; \alpha) = \alpha$
- Treated group potential outcomes always higher (lower): $q(d, x; \alpha) = \alpha(2d - 1)$
- Package on CRAN: `causalsens`
- You should probably use this for the homework.



correlation with Y
correlation with D
-1.0
-0.5
0.0
0.5
1.0
correlation with Y
correlation with D
-1.0
-0.5
0.0
0.5
1.0


Figure 2:

## Example

- Remove the fixed effects to make it sensitive:

```
require(causalsens)
```

```
## Loading required package: causalsens
```

```
d$ddt_week<-ifelse(d$ddt_week>0,1,0)
out<-lm(paste0("getwarmord~ddt_week+",paste(vars,collapse="+")),data=d)
coef(out)["ddt_week"]
```

```
##   ddt_week
## 0.04557408
```

```
outD<-glm(paste0("ddt_week~",paste(vars,collapse="+")),data=d,family=binomial())
alpha<-seq(-.1, .1, by = .001)
SensAnalysis<-causalsens(out,outD,as.formula(paste0("~",paste(vars,collapse="+"))),data=d,al
```

## Sensitivity Plots

```
par(mfrow=c(1,2))
plot(SensAnalysis,type="raw",bty="n")
plot(SensAnalysis,type="r.squared",bty="n")
```

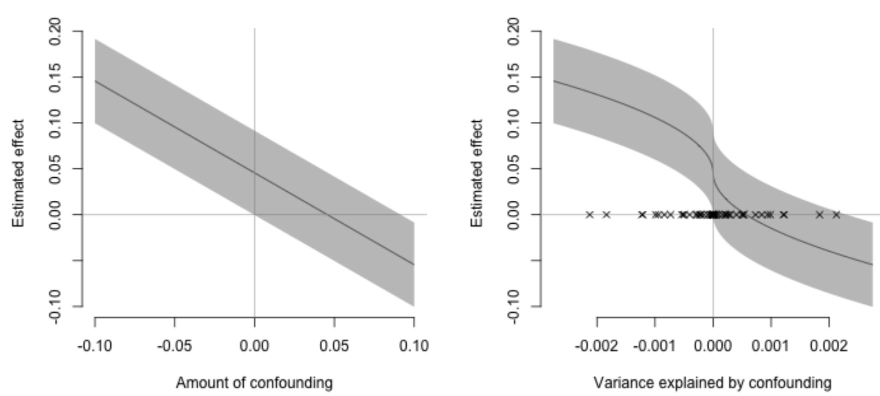



Figure 3:

## Sensitivity Analysis

- We're adding to the discussion on post-treatment bias with a sensitivity analysis.
- This is also in Rosenbaum (1984).
- The variable which one might think could induce post-treatment bias in our example is that of "public acceptance".

## Rosenbaum Bounding

- In general Rosenbaum is a proponent of trying to "bound" biases.
- He does this in his "normal" sensitivity analysis method, and we do the same, here.
- We will assume a "surrogate" for $U$ (necessary for CIA), which is observed post-treatment.
- The surrogate has two potential outcomes: $S_1$ and $S_0$
- It is presumed to have a linear response on the outcome.
- (As are the other observed covariates)
- This gives us the following two regression models: $E[Y_1|S_1 = s, X = x] = \mu_1 + \beta' x + \gamma' s$ and
  $E[Y_0|S_0 = s, X = x] = \mu_0 + \beta' x + \gamma' s$
- This gives us:
  $\tau = E[(\mu_1 + \beta' X + \gamma' S_1) - (\mu_0 + \beta' X + \gamma' S_0)]$
- Which is equal to the following useful expression:
  $\tau = \mu_1 - \mu_0 + \gamma'(E[S_1 - S_0])$
- For us, this means that $\tau = \beta_1 + \beta_2 E[S_1 - S_0]$

## (Re)introduce Example

```
require(foreign,quietly=TRUE)
d <- read.dta("replicationdataIOLGBT.dta")
#Base specification
d$ecthrpos <- as.double(d$ecthrpos)-1
d.lm <- lm(policy~ecthrpos+pubsupport+ecthrcountry+lgbtlaws+cond+eumember0+euemploy+coemembe
d <- d[-d.lm$na.action,]
d$issue <- as.factor(d$issue)
d$ccode <- as.factor(d$ccode)
d.lm <- lm(policy~ecthrpos+pubsupport+ecthrcountry+lgbtlaws+cond+eumember0+euemploy+coemembe
```

## Back to Bounding

- Our surrogate is public acceptance.
- But it can be swayed by court opinions, right? This is at least plausible.
- Let's try and get some reasonable bounds on $\tau$.

. . .

```
sdS <- sd(d$pubsupport)
Ediff <- c(-1.5*sdS,-sdS,-sdS/2,0,sdS/2,sdS,1.5*sdS)
tau <- coef(d.lm)[2] + coef(d.lm)[3]*Ediff
names(tau)<-c("-1.5","-1","-.5","0",".5","1","1.5")
tau
```

```
##       -1.5         -1        -.5          0         .5          1
## 0.06620715 0.06817761 0.07014808 0.07211854 0.07408901 0.07605947
##        1.5
## 0.07802994
```

- But with this method, you don't necessarily have to assume that the regression functions are this rigid.
- Can you think about how one might relax some assumptions?