Replication Project Draft

Andrew Ballard
PhD Student in Political Science
Duke University

February 27, 2013

1 Summary of Gelman, Shor, Bafumi, and Park (2007). Rich State, Poor State, Red State, Blue State: What's the Matter with Connecticut?

Gelman et al. investigate the recent journalistic trend viewing the Democrats being the party of rich elites instead of the party of the masses, which was the dominant view throughout the 20th century. The authors look to To accomplish this task, they fit National Election Study (NES) and National Annenberg Election Survey (NAES) data from a number of years on income, presidential vote, and demographic characteristics to a number of different logistic regression models. Their main findings are that richer states tend to support Democrats, but within each state, income increases the probability to vote Republican. However, the within-state effect of income is lower in traditionally "blue" states.

Below I present a discussion of the methodology used by Gelman et al., a replication of their main findings, and detailed plans to extend the project.

2 Methodology and Replication

The data presentation in this paper consists of 8 figures, 5 of which I have replicated below. The code used in replicating the figures, as well as all data used in the paper, can be found at my dataverse: **http://dvn.iq.harvard.edu/dvn/dv/ballardao**. I have had the data for this paper less than a week, so much of my time has been spent familiarizing myself with the data. Thus, my progress on this project at this point only includes a replication, and not an extension.[1]

[1] I had originally planned to replicate Jerit, J. and Barabas, J. (2012), Partisan Perceptual Bias and The Information Environment. However, I ran into a number of road blocks. At first, I tried to run the data on my laptop, which was fruitless since the datasets were so large. Next, I wrestled with getting the authors' code to run on the SSRI computer cluster, with no luck. Finally, I enlisted the help of persons much more knowledgeable than myself about Stata and data analysis, to no avail. This was the last straw, and I desperately embarked on a journey to find a suitable replacement study. I was assigned to read Gelman et al. for the February 25th meeting of the Behavior Core course, and Dr. Andrew Gelman was kind enough to send me the code and data for this project. He answered my e-mail within hours, and on a Sunday. I owe him a debt of gratitude.

Particularly, I have spent a great deal of time trying to understand the authors' code. They employ methods of doing logistic regression that are wholly strange and new to me, and I am still working on grasping every detail. Currently, all graphs presented below were created using code from the authors, nearly in its original form. I did have to make a few changes in order to get R to recognize some of the objects from Stata, but this was minor and only took a few hours of debugging. One of my plans for moving forward is to put as much of the code in my own words as possible, although I suspect there will still be a great deal that I use from the authors.

2.1 Figures 1 and 2

To create Figure 1, the authors first fit the data to a logistic regression model predicting Republican vote share by state as a function of income (in tens of thousands of 1996 US dollars). Given the continuous nature of vote share, this model is logical to use instead of something like a probit or ordered probit. The results are presented as income regression coefficients and standard errors from 1952 to 2004 for all states, southern states, and non-southern states. The downward trend shows that recently, Republicans have received a higher share of the vote in poor states than rich states. This effect is more pronounced for southern states than for non-southern states, which fits the widely observed disappearance of the Southern Democrat in the 1950s and 60s.

Figure 1: Replication of regression coefficients and standard errors for Republican vote share as a function of average state income

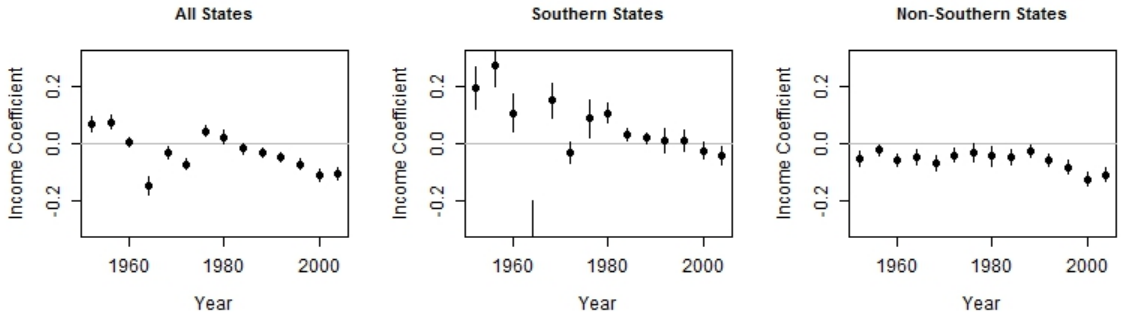
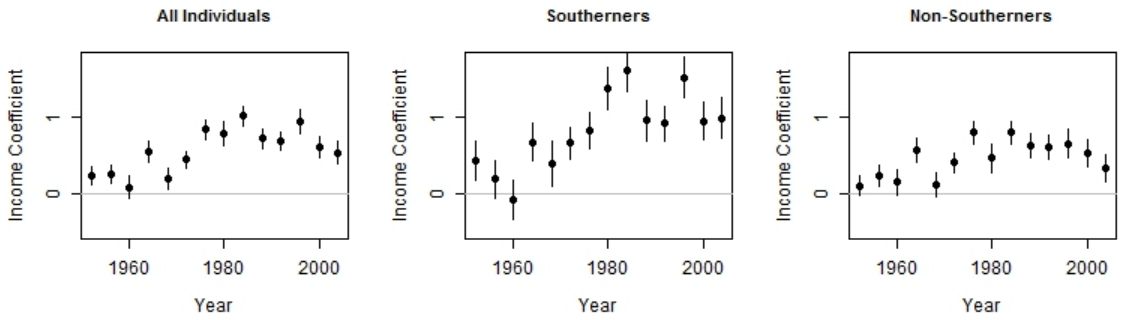


Figure 2 is similar in methodology to Figure 1, except that instead of by state they use individual votes and income as their variables for logistic regression. Since the dependent variable (Vote Republican; Yes = 1, No = 0) is binary, a logistic regression is a fine choice of method. The pattern in figure 2 clearly shows that although rich states tend to have a lower Republican vote share, that overall, wealthier individuals are more likely to vote Republican.

2.2 Figures 3, 4, and 5

This may be seemingly paradoxical, that rich states show more support for Democrats, but rich people tend to vote Republican. To untangle this puzzle, we turn to Figures 3 and 4, which fit a multilevel model using the lmer() function in R to Annenberg pol data from 2000 and 2004. In order to create the plots themselves, the authors wrote a 189-line function called "Superplot", which I do not yet fully understand. Figure 3 shows a varying intercept (but not varying slope)

Figure 2: Replication of regression coefficients and standard errors for Republican vote as a function of individual income



multilevel model. The three lines represent Mississippi (the poorest state, historically dominated by the Republicans), Ohio (a "middle of the road" income state, usually a swing state), and Connecticut (the richest state, perennially "blue"). The x-axis values are a recoded version of a five-point quantile-based NES scale.[2] The y-axis shows the probability of voting Republican, or for President GW Bush. Within each state, wealthier voters tend to vote Republican, but richer states are more likely to support Democrats.

Figure 3: Replication of the probability of voting Republican as a function of individual income in 2000 and 2004 – Varying intercepts

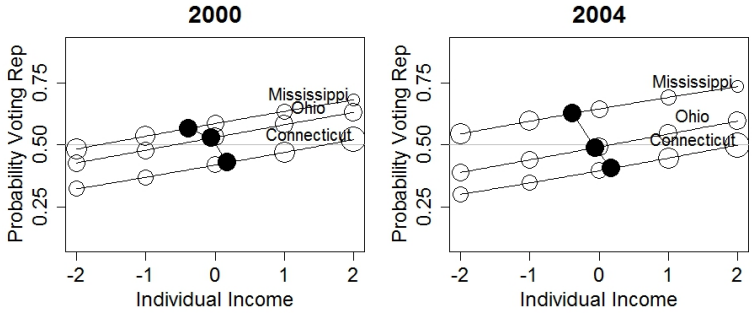


Figure 4 uses a similar concept as Figure 3, except now both the intercept and slope of the probability to vote Republican as a function of individual income for each state can vary. This analysis uncovers the trend that income matters more in poorer states and less in richer states, which is evaluated further in Figure 5.

The three states chosen in Figures 3 and 4 are good choices, but they are not representative of all 50 states. Thus, we turn to Figure 5 to look at estimated slopes for the multilevel model of all 50 states. There is a clear negative relation between a state's slope (as it pertains to the model presented in Figures 3 and 4) and the state's median income. This means that income matters less in determining the probability to vote Republican for rich states than for poor states. However, this effect appears in 2000 with much more clarity than in 2004.

[2]The x-axis values are as follows: -2 = 0-16 percentile, -1 = 17-33 percentile, 0 = 34-67 percentile, 1 = 68-95 percentile, and 2 = 96-100 percentile.

Figure 4: Replication of the probability of voting Republican as a function of individual income in 2000 and 2004 – Varying slopes and intercepts

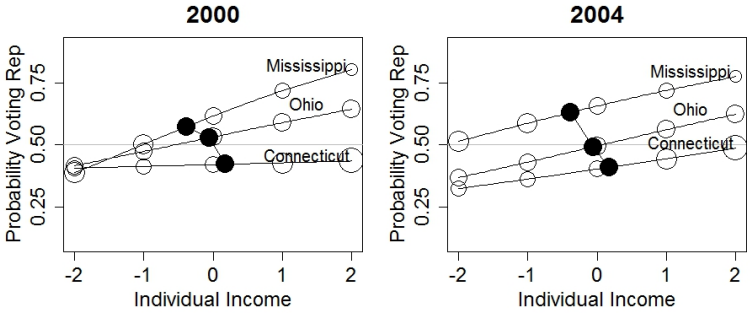
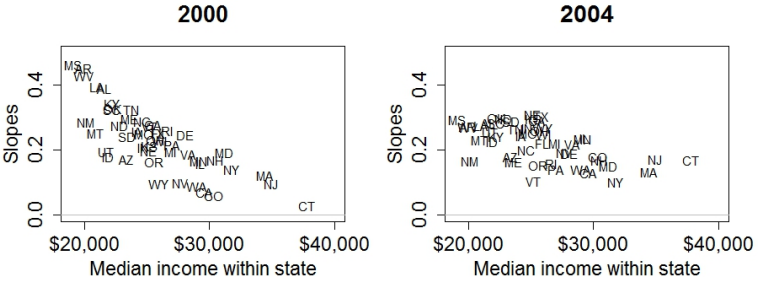


Figure 5: Replication of slope estimates for all 50 states as a function of income



3 Extension Ideas

Gelman et al. provide some valuable ideas in this paper, but I believe there are a number of ways to improve their findings.

3.1 Is it Really Income? If So, How?

I am unconvinced that income is the driving force behind vote choice. Personally I tend to advocate an approach that values explaining complex decisions like voting with a number of variables, including income but also including race, gender, education, geography, and religion. The authors partially account for some of these, such as race and education, in their model. However, I believe additional analyses can be performed in order to improve the results. Specifically, I would like to see what effect religion has on these results.

Let us assume for a moment that after performing the additional analyses mentioned above, the results do not change. What then? Gelman et al. standardize income to 1996 US dollars, but they do not normalize it. A dollar is worth much more in Arkansas than it is in Connecticut, and adjusting for cost-of-living could have striking implications for the results of this paper. Adjusting for cost-of-living is straightforward, and I plan to include it in my extension.

3.2 Presentation, Presentation, Presentation

Sometimes of the Figures are straightforward, but others are somewhat confusing. Particularly, I would like to do a better job presenting the findings of Figure 5. I find it difficult to make the transformation from "Slope" on the y-axis to the slope of a line in space in my head. I think it could be valuable to present all 50 states in a similar fashion to Figures 3 and 4, but to color code them along a continuum to show the relative wealth of different states. It would also be interesting to try and present the data in different ways. I think that separation plots would be a worthwhile investment for showing how the model predicts Republican votes.

3.3 Validity and Simulated Scenarios

Lastly, I plan to include in my extension a discussion of the validity of the model in predicting presidential votes in the form of a cross validation, and use that to run simulations on scenarios for a number of different types of states. The simulations will include more than one independent variable, such as those discussed in improving the model in subsection 3.1 above.

Appendix - Gelman et al. Replication R Code

Figures 1 and 2

```
# QJPS Replication
# Figures 1 and 2
# Updated by David K. Park

# Set Working Directory HERE
# setwd("C:/Research/Blue Red States/QJPS/Data/Replication/Replication1-2")
setwd("C:/Andy/Dropbox/ReplicationProject/Gelman") #desktop
setwd("C:/Duke/Dropbox/ReplicationProject/Gelman") #laptop
setwd("C:/Users/aob5/Dropbox/ReplicationProject/Gelman") #Bunche
# load foreign package to read in STATA file
library(foreign)

fips.cbs <- read.dta("fips.icpsr.cbs.naes.dta")

# Figure 1 Regressions
# state-level
st <- read.table("st.dat")

st.all <- merge(st, fips.cbs, by.x="stateabb", by.y="st")
st.s <- subset(st.all, st.all$regioncbs==3)
st.ns <- subset(st.all, st.all$regioncbs!=3)

p.st.yr <- seq(min(st$year), max(st$year),4)
n.st.yr <- length(p.st.yr)

col.st.names <- c("st.int", "z.st.inc.pop")
  n.st.coef <- length(col.st.names)

B.st.lm <- array(NA, c(n.st.yr, n.st.coef))
  colnames(B.st.lm) <- col.st.names
SE.st.lm <- array(NA, c(n.st.yr, n.st.coef))
  colnames(SE.st.lm) <- col.st.names

B.st.s.lm <- array(NA, c(n.st.yr, n.st.coef))
  colnames(B.st.s.lm) <- col.st.names
SE.st.s.lm <- array(NA, c(n.st.yr, n.st.coef))
  colnames(SE.st.s.lm) <- col.st.names

B.st.ns.lm <- array(NA, c(n.st.yr, n.st.coef))
  colnames(B.st.ns.lm) <- col.st.names
SE.st.ns.lm <- array(NA, c(n.st.yr, n.st.coef))
  colnames(SE.st.ns.lm) <- col.st.names

for (i in 1:n.st.yr){
    st.temp <- subset(st, st$year==p.st.yr[i])
    st.reg <- summary(lm(st.temp$st.repshare ~ st.temp$z.st.inc.pop))
    B.st.lm[i,] <- st.reg$coef[,1]
    SE.st.lm[i,] <- st.reg$coef[,2]
}

for (i in 1:n.st.yr){
    st.s.temp <- subset(st.s, st.s$year==p.st.yr[i])
    st.s.reg <- summary(lm(st.s.temp$st.repshare ~ st.s.temp$z.st.inc.pop))
    B.st.s.lm[i,] <- st.s.reg$coef[,1]
    SE.st.s.lm[i,] <- st.s.reg$coef[,2]
}

for (i in 1:n.st.yr){
    st.ns.temp <- subset(st.ns, st.ns$year==p.st.yr[i])
    st.ns.reg <- summary(lm(st.ns.temp$st.repshare ~ st.ns.temp$z.st.inc.pop))
    B.st.ns.lm[i,] <- st.ns.reg$coef[,1]
```

```
62      SE.st.ns.lm[i,] <- st.ns.reg$coef[,2]
63  }
64
65  B.st.lm <- data.frame(B.st.lm)
66  SE.st.lm <- data.frame(SE.st.lm)
67
68  B.st.s.lm <- data.frame(B.st.s.lm)
69  SE.st.s.lm <- data.frame(SE.st.s.lm)
70
71  B.st.ns.lm <- data.frame(B.st.ns.lm)
72  SE.st.ns.lm <- data.frame(SE.st.ns.lm)
73
74  #  Figure 1 Plot
75  windows(width=8, height=2.5)
76  par(mfrow=c(1,3))
77  n.st <- NROW(B.st.lm)
78
79  plot(0, 0, type='n', ylim=c(-.3, .3), xlim=c(min(st$year), max(st$year)),
80    cex.lab=1.15, xaxt="n", yaxt="n",  main="All States", cex.main=1, xlab="Year", ylab="Income
        Coefficient")
81    axis(side=1, at=c(1960, 1980, 2000), cex.axis=1.1)
82    axis(side=2, at=c(-.2, 0, .2), cex.axis=1.1)
83    abline(h=0, col="gray")
84      for (i in 1:n.st) {
85      points(p.st.yr[i], B.st.lm[i,2], type='p', pch=19)
86      segments(p.st.yr[i], (B.st.lm[i,2]-SE.st.lm[i,2]), p.st.yr[i], (B.st.lm[i,2]+SE.st.lm[i,2]))
87    }
88
89  plot(0, 0, type='n', ylim=c(-.3, .3), xlim=c(min(st$year), max(st$year)),
90    cex.lab=1.15, xaxt="n", yaxt="n",  main="Southern States", cex.main=1, xlab="Year", ylab="Income
        Coefficient")
91    axis(side=1, at=c(1960, 1980, 2000), cex.axis=1.1)
92    axis(side=2, at=c(-.2, 0, .2), cex.axis=1.1)
93    abline(h=0, col="gray")
94      for (i in 1:n.st) {
95      points(p.st.yr[i], B.st.s.lm[i,2], type='p', pch=19)
96      segments(p.st.yr[i], (B.st.s.lm[i,2]-SE.st.s.lm[i,2]), p.st.yr[i], (B.st.s.lm[i,2]+SE.st.s.lm[
          i,2]))
97    }
98
99  plot(0, 0, type='n', ylim=c(-.3, .3), xlim=c(min(st$year), max(st$year)),
100   cex.lab=1.15, xaxt="n", yaxt="n",  main="Non-Southern States", cex.main=1, xlab="Year", ylab="
        Income Coefficient")
101   axis(side=1, at=c(1960, 1980, 2000), cex.axis=1.1)
102   axis(side=2, at=c(-.2, 0, .2), cex.axis=1.1)
103   abline(h=0, col="gray")
104     for (i in 1:n.st) {
105     points(p.st.yr[i], B.st.ns.lm[i,2], type='p', pch=19)
106     segments(p.st.yr[i], (B.st.ns.lm[i,2]-SE.st.ns.lm[i,2]), p.st.yr[i], (B.st.ns.lm[i,2]+SE.st.ns
          .lm[i,2]))
107   }
108
109 # Figure 1
110 savePlot(filename="st.yr", type=c("pdf"), device=dev.cur())
111 savePlot(filename="st.yr", type=c("ps"), device=dev.cur())
112
113
114 # individual level
115 anes <- read.table("recode.anes.qjps.txt")
116 anes.s <- subset(anes, anes$regioncbs==3)
117 anes.ns <- subset(anes, anes$regioncbs!=3)
118
119 # table(anes$relatt, anes$yr)
120 # table(anes$relatt2, anes$yr)
121 p.yr <- seq(min(anes$yr), max(anes$yr),4)
122 n.yr <- length(p.yr)
123 col.names <- c("int", "z.inc")
```

```
n.coef <- length(col.names)

B.glm <- array(NA, c(n.yr, n.coef))
  colnames(B.glm) <- col.names
SE.glm <- array(NA, c(n.yr, n.coef))
  colnames(SE.glm) <- col.names

B.s.glm <- array(NA, c(n.yr, n.coef))
  colnames(B.s.glm) <- col.names
SE.s.glm <- array(NA, c(n.yr, n.coef))
  colnames(SE.s.glm) <- col.names

B.ns.glm <- array(NA, c(n.yr, n.coef))
  colnames(B.ns.glm) <- col.names
SE.ns.glm <- array(NA, c(n.yr, n.coef))
  colnames(SE.ns.glm) <- col.names

for (i in 1:n.yr){
    temp <- subset(anes, anes$yr==p.yr[i])
    reg <- summary(glm(temp$vote ~ temp$z.inc, family=binomial(link=logit)))
    B.glm[i,] <- reg$coef[,1]
    SE.glm[i,] <- reg$coef[,2]
}

for (i in 1:n.yr){
    temp.s <- subset(anes.s, anes.s$yr==p.yr[i])
    reg.s <- summary(glm(temp.s$vote ~ temp.s$z.inc, family=binomial(link=logit)))
    B.s.glm[i,] <- reg.s$coef[,1]
    SE.s.glm[i,] <- reg.s$coef[,2]
}

for (i in 1:n.yr){
    temp.ns <- subset(anes.ns, anes.ns$yr==p.yr[i])
    reg.ns <- summary(glm(temp.ns$vote ~ temp.ns$z.inc, family=binomial(link=logit)))
    B.ns.glm[i,] <- reg.ns$coef[,1]
    SE.ns.glm[i,] <- reg.ns$coef[,2]
}

B.glm <- data.frame(B.glm)
SE.glm <- data.frame(SE.glm)

B.s.glm <- data.frame(B.s.glm)
SE.s.glm <- data.frame(SE.s.glm)

B.ns.glm <- data.frame(B.ns.glm)
SE.ns.glm <- data.frame(SE.ns.glm)


# Figure 2 Plot
windows(width=8, height=2.5)
par(mfrow=c(1,3))
n <- NROW(B.glm)

plot(0, 0, type='n', ylim=c(-.5, 1.75), xlim=c(min(anes$yr), max(anes$yr)),
  cex.lab=1.15, xaxt="n", yaxt="n", main="All Individuals", cex.main=1, xlab="Year", ylab="Income
      Coefficient")
  axis(side=1, at=c(1960, 1980, 2000), cex.axis=1.1)
  axis(side=2, at=c(0, 1), cex.axis=1.1 )
  abline(h=0, col="gray")
    for (i in 1:n) {
    points(p.yr[i], B.glm$z.inc[i], type='p', pch=19)
    segments(p.yr[i], (B.glm$z.inc[i]-SE.glm$z.inc[i]), p.yr[i],  (B.glm$z.inc[i]+SE.glm$z.inc[i])
        )
  }

plot(0, 0, type='n', ylim=c(-.5, 1.75), xlim=c(min(anes$yr), max(anes$yr)),
```

```
188   cex.lab=1.15, xaxt="n", yaxt="n", main="Southerners", cex.main=1, xlab="Year", ylab="Income
          Coefficient")
189   axis(side=1, at=c(1960, 1980, 2000), cex.axis=1.1)
190   axis(side=2, at=c(0, 1), cex.axis=1.1 )
191   abline(h=0, col="gray")
192     for (i in 1:n) {
193     points(p.yr[i], B.s.glm$z.inc[i], type='p', pch=19)
194     segments(p.yr[i], (B.s.glm$z.inc[i]-SE.s.glm$z.inc[i]), p.yr[i],  (B.s.glm$z.inc[i]+SE.s.glm$z
            .inc[i]))
195   }
196
197 plot(0, 0, type='n', ylim=c(-.5, 1.75), xlim=c(min(anes$yr), max(anes$yr)),
198   cex.lab=1.15, xaxt="n", yaxt="n", main="Non-Southerners", cex.main=1, xlab="Year", ylab="Income
          Coefficient")
199   axis(side=1, at=c(1960, 1980, 2000), cex.axis=1.1)
200   axis(side=2, at=c(0, 1), cex.axis=1.1 )
201   abline(h=0, col="gray")
202     for (i in 1:n) {
203     points(p.yr[i], B.ns.glm$z.inc[i], type='p', pch=19)
204     segments(p.yr[i], (B.ns.glm$z.inc[i]-SE.ns.glm$z.inc[i]), p.yr[i],  (B.ns.glm$z.inc[i]+SE.ns.
            glm$z.inc[i]))
205   }
206
207 # Figure 2
208 savePlot(filename="nes.yr", type=c("pdf"), device=dev.cur())
209 savePlot(filename="nes.yr", type=c("ps"), device=dev.cur())
```

Superplot Function

```
superplot <- function (results, start, end, rows, columns, indiv = 1, lmer = 0, var.slope = 0,
    fiftydots = 0, rich = 7, med = 35, poor = 24, fiftylines = 0, survey="", predictor="Individual
     Income", lowbound=-2,hibound=2, inc.bins=1, xlabels="all", ylabels="all", sf=4.5)
{
    #i.result=round(br.state[[13]]$summary,2)

    #windows(height=8,width=6.5)
    par (mfrow = c(rows,columns))
    par(mar=c(3.75, 4.25, 3, 1)) # bottom, left, top, right

    for (j in start:end) {
        if (ylabels[1] == "all") {
            if (indiv == 0) { ylabel = "Rep Vote Share" } else { ylabel = "Probability Voting Rep"
                 }
        } else {
            if (j %in% ylabels) { ylabel = "Probability Voting Rep" } else { ylabel = "" }
        }

        if (xlabels[1]=="all") {
            xl = predictor
        } else {
            if (j %in% xlabels) { xl = predictor } else { xl = "" }
        }

        plot(c(lowbound,hibound), c(.1,.9), xaxt="n", yaxt="n", xlab=xl, ylab=ylabel, cex.lab=1.5,
             mgp=c(2.5,.5,0), type="n")

        axis(side=1,seq(lowbound,hibound,(hibound-lowbound)/4),cex.axis=1.5)
        axis(side=2,seq(.25,.75,.25),cex.axis=1.5)
        #axis(side=2,seq(.12,.9,.2),tick=T,labels=F)
        title(paste(survey,main=years[j],sep=""), cex.main=1.75)
        #range(st.repshare[st.year==years[j]])

        superp = matrix(nrow=50,ncol=9)
        rownames(superp)=state.name
        colnames(superp)=c("Intercept","Slope","W Mean","Pred","1","2","3","4","5")

        #attach(cps.data[[j]])
        cps.stats=cps.stats.all[[j]]

        if (lmer == 1 ) {

            constant = 0; slope = 0; # common constant
            #var.int.old = results[[j]][,1]
            #var.slope.old = results[[j]][,2]

            #if (j == 5) {
                statelist = as.integer(rownames(results[[j]]))
                #statelist = seq(1:50)
                #st.income = log(st.inc10k[st.year==years[j]])[statelist]
                #st.income = st.income[as.integer(statelist)]
                #var.int <- results[[j]][,1] + results[[j]][,3]*st.income
                #var.slope <- results[[j]][,2] + results[[j]][,4]*st.income
                var.int <- results[[j]][,1]
                var.slope <- results[[j]][,2]

                #names(var.int)[8:length(var.int)]=as.integer(names(var.int))[8:length(var.int)]-1
                #names(var.slope)[8:length(var.slope)]=as.integer(names(var.slope))[8:length(var.
                    slope)]-1
                #statelist = names(var.int)

            #} else {
                #st.income = cps.stats[,4][1:49] # No Wyoming
                #statelist = as.integer(rownames(results[[j]]))
                #statelist = seq(1:50)
```

```
61                #st.income = log(st.inc10k[st.year==years[j]])[statelist]
62                #var.int <- results[[j]][,1] + results[[j]][,3]*st.income
63                #var.slope <- results[[j]][,2] + results[[j]][,4]*st.income
64                #var.int <- results[[j]][,1]
65                #var.slope <- results[[j]][,2]
66                #names(var.int)=seq(1:50); names(var.slope)=seq(1:50)
67                #statelist = names(var.int)
68
69            #}
70
71
72            for (i in 1:50) {
73                if (sum(statelist==i) > 0) {
74                    superp[i,"Intercept"] = var.int[statelist==i]
75                    superp[i,"Slope"] = var.slope[statelist==i]
76                    superp[i,"W Mean"] = cps.stats[i,"Weighted Count"]
77                    superp[i,"Pred"] = round(invlogit(superp[i,"Slope"]*superp[i,"W Mean"]+superp[
                        i,"Intercept"]),3)
78                    superp[i,5:9] = cps.stats[i,5:9]
79                }
80            }
81            #superp[9:49,1:2]=superp[10:50,1:2]
82
83            # end of lmer
84            } else {
85            if (indiv == 1) {
86                if (is.vector(results[[j]])) { means = results[[j]] } else { means = results[[j
                    ]][,"50%"] }
87                constant = 0; #constant=means["beta.0.adj"]
88                slope = 0; #slope=means["b.inc.0.adj"]
89                offset = 50
90            } else {
91                means = results[[j]][,"mean"]
92                constant=means["constant.adj"]
93                if (var.slope == 0) { slope = results[[j]][52] }
94                offset = 0
95            }
96
97            for (i in 1:50) {
98                if (indiv == 0) { superp[i,"Intercept"] = constant } else { superp[i,"Intercept"]
                    = means[i]+constant }
99                if (var.slope == 0) { superp[i,2] = mean(results[[9]][,"50%"][51:100]) } else {
                    superp[i,2] = means[offset+i]+slope}
100               superp[i,3] = cps.stats[i,"Weighted Count"]
101               if (indiv == 0) {
102                   superp[i,"Pred"] = round(slope*superp[i,"W Mean"]+superp[i,"Intercept"],3)
103               } else {
104                   superp[i,"Pred"] = round(invlogit(superp[i,"Slope"]*superp[i,"W Mean"]+superp[
                        i,"Intercept"]),3)
105               }
106               #superp[i,5:9] = round(count[fips.cps==s2f[i]]/cps.stats[i,"Sum Count"],2)
107               superp[i,5:9] = cps.stats[i,5:9]
108               }
109       # end of non-lmer
110       }
111
112       if (fiftydots==1) {
113           for (i in 1:50) {
114               points(superp[i,3],superp[i,"Pred"],cex=1.25,pch=19,col="gray")
115           }
116           fit <- lm (superp[,"Pred"] ~ superp[,"W Mean"])
117           abline (fit$coeff[1],fit$coeff[2],col="gray")
118       }
119
120       abline (.5,0,col="gray")
121
122
```

```
123         if (fiftylines == 0) {
124             if (indiv == 0) {
125                 abline (superp[rich,1], superp[rich,2], col="blue") # rich
126                 abline (superp[med,1], superp[med,2], col="green") # medium
127                 abline (superp[poor,1], superp[poor,2], col="red") # poor
128             } else {
129                 curve (invlogit (superp[rich,1] + superp[rich,2]*x), lowbound, hibound, add=T) #,
                        col="blue") # lty = 1
130                 curve (invlogit (superp[med,1] + superp[med,2]*x), lowbound, hibound, add=T) #,
                        col="green") # lty = 2
131                 curve (invlogit (superp[poor,1] + superp[poor,2]*x), lowbound, hibound, add=T) #,
                        col="red") # lty = 3
132             }
133         } else {
134         print ("fiftylines!\n")
135         for (i in 1:50) {
136             curve (invlogit (superp[i,1] + superp[i,2]*x), lowbound, hibound, add=T, lwd=0.5, col=
                    "blue")
137             }
138         }
139
140
141         if (inc.bins == 0) {
142             #for (i in c(rich,med,poor)) {
143                 #if (i==rich) { offset = .03 } else { offset = -.01 }
144                 #text(1.45,invlogit(superp[i,2]*2+superp[i,1])+offset, state.name[i],cex=1.35)
145             inc.cat = c(-1,-.6,-.2,.2,.6)
146             st.label = .4
147         } else {
148             inc.cat = c(-2,-1,0,1,2)
149             st.label = 1.35
150         }
151
152             #sf = 4.5
153
154             # draw circles for income quintiles
155             col.inc = "black"
156             if (indiv == 0) {
157                 for (i in c(rich,med,poor)) {
158                     points(superp[i,"W Mean"],superp[i,"Pred"],cex=sf,pch=19)
159                     points(inc.cat[1],slope*inc.cat[1]+superp[i,1],cex=sqrt(superp[i,"1"])*sf,col=
                            col.inc,pch=21)
160                     points(inc.cat[2],slope*inc.cat[2]+superp[i,1],cex=sqrt(superp[i,"2"])*sf,col=
                            col.inc,pch=21)
161                     points(inc.cat[3],slope*inc.cat[3]+superp[i,1],cex=sqrt(superp[i,"3"])*sf,col=
                            col.inc,pch=21)
162                     points(inc.cat[4],slope*inc.cat[4]+superp[i,1],cex=sqrt(superp[i,"4"])*sf,col=
                            col.inc,pch=21)
163                     points(inc.cat[5],slope*inc.cat[5]+superp[i,1],cex=sqrt(superp[i,"5"])*sf,col=
                            col.inc,pch=21)
164                     }
165             } else {
166                 for (i in c(rich,med,poor)) {
167                     points(superp[i,"W Mean"],superp[i,"Pred"],cex=sf,pch=19)
168                     points(inc.cat[1],invlogit(superp[i,2]*inc.cat[1]+superp[i,1]),cex=sqrt(superp
                            [i,"1"])*sf,col=col.inc,pch=21)
169                     points(inc.cat[2],invlogit(superp[i,2]*inc.cat[2]+superp[i,1]),cex=sqrt(superp
                            [i,"2"])*sf,col=col.inc,pch=21)
170                     points(inc.cat[3],invlogit(superp[i,2]*inc.cat[3]+superp[i,1]),cex=sqrt(superp
                            [i,"3"])*sf,col=col.inc,pch=21)
171                     points(inc.cat[4],invlogit(superp[i,2]*inc.cat[4]+superp[i,1]),cex=sqrt(superp
                            [i,"4"])*sf,col=col.inc,pch=21)
172                     points(inc.cat[5],invlogit(superp[i,2]*inc.cat[5]+superp[i,1]),cex=sqrt(superp
                            [i,"5"])*sf,col=col.inc,pch=21)
173                     #if (i==rich) { offset = .1 } else { offset = -.01 }
174                     text(st.label,invlogit(superp[i,2]*2+superp[i,1] + .1), state.name[i],cex
                            =1.25)
```

```
175                     }
176             }
177
178             #legend (1,.7,c(state.abb[rich],state.abb[med],state.abb[poor]),col=c("green","brown
                    ","orange"),lty=1, lwd=3, bg="gray85")
179
180             lines(lowess(c(superp[rich,"W Mean"],superp[med,"W Mean"],superp[poor,"W Mean"]), c(
                    superp[rich,"Pred"],superp[med,"Pred"],superp[poor,"Pred"]) ))
181
182         }
183
184     return (superp)
185 }
```

Figures 3, 4, and 5

```
#setwd("Research/BlueRed/Analysis/Replication")

library(MASS)
library(foreign)
library(arm)
source("superplot function.R")
load(file="cps.income.Rdata")
results=list()
years=seq(1968,2004,4)

state.region[8]="Northeast" # put Delaware into the Northeast
state.region[20]="Northeast" # put Maryland into the Northeast


#R
#2000 annenberg with region
annen2000.data <- read.dta("annen061229.dta",convert.factors=F)
annen2000.data <- annen2000.data[is.na(annen2000.data$income)==FALSE&is.na(annen2000.data$fips)==
    FALSE,]
state.income.data<-read.dta("avgincome_orig.dta",convert.factors=T)
income.new <- annen2000.data$income-3
y <- annen2000.data$rep_presvote
state.new <- annen2000.data$fips
region <- annen2000.data$region
#region <- state.region[annen2000.data$state]
data<-cbind(annen2000.data, income.new, y, state.new)
n.state <-length(unique(state.new))
state.income<-as.list(rep(NA,9))
i <- min(state.income.data$st_year)
  while(i <= 2000){
state.income[[((i-min(state.income.data$st_year))/4)+1]] <- state.income.data[state.income.data$st
    _year==i,]
i <- i + 4
}
avg.income<-log(state.income[[9]]$st_inc10k)
n <- length(y)
#income.new:  individual-level income
#state.new:  the state index variable
#avg.income:  state-level income (a vector of length 50)
avg.income.expanded <- avg.income[state.new]

# varying intercept
fit.vi.2000 <- lmer (y ~ income.new + factor(region) + avg.income.expanded +
                      (1 | state.new), family=binomial(link="logit"))
display(fit.vi.2000)
coef(fit.vi.2000)

# varying intercept, varying slopes
fit.2000 <- lmer (y ~ income.new*factor(region) + income.new*avg.income.expanded +
                   (1 + income.new | state.new), family=binomial(link="logit"))
display (fit.2000)
# then, the coefs can be accessed using
coef(fit.2000)

# region.indic<-read.dta("region_dummies_84.dta",convert.factors=F)
region.indic<-read.dta("region_indic_annen2000.dta",convert.factors=F)


#2004 annenberg with region
annen2004.data <- read.dta("annen2004_processed.dta",convert.factors=F)
annen2004.data <- annen2004.data[is.na(annen2004.data$state)==FALSE,]
state.income.data<-read.dta("avgincome_orig.dta",convert.factors=T)
income.new <- annen2004.data$income-3
y <- annen2004.data$votechoice
state.new <- annen2004.data$state
```

```
region <- annen2004.data$region
#region <- state.region[annen2004.data$state]
data2004<-cbind(annen2004.data, income.new, y, state.new)
n.state <-length(unique(state.new))
state.income<-as.list(rep(NA,9))
i <- min(state.income.data$st_year)
  while(i <= 2004){
state.income[[((i-min(state.income.data$st_year))/4)+1]] <- state.income.data[state.income.data$st_year==i,]
i <- i + 4
}
avg.income<-log(state.income[[9]]$st_inc10k)
n <- length(y)
#income.new:  individual-level income
#state.new:  the state index variable
#avg.income:  state-level income (a vector of length 50)
avg.income.expanded <- avg.income[state.new]

fit.vi.2004 <- lmer (y ~ income.new + factor(region) + avg.income.expanded + (1 | state.new), family=binomial(link="logit"))
display(fit.vi.2004)
coef(fit.vi.2004)

fit.2004 <- lmer (y ~ income.new*factor(region) + income.new*avg.income.expanded +
(1 + income.new | state.new), family=binomial(link="logit"))
display (fit.2004)
#then, the coefs can be accessed using
coef(fit.2004)
#region.indic <- read.dta("region_dummies_84.dta",convert.factors=F)
region.indic<-read.dta("region_indic_annen2000.dta",convert.factors=F)

### Plots ###

# Figure 3
## plot varying intercepts ##
# 2000
sl2000 = as.integer(rownames(coef(fit.vi.2000)[[1]]))
income.vi.slope.2000 <- coef(fit.vi.2000)$state.new[,2]
income.vi.intercept.2000 <- coef(fit.vi.2000)$state.new[,1] +
 region.indic[,1]*coef(fit.vi.2000)$state.new[,3] +
 region.indic[,2]*coef(fit.vi.2000)$state.new[,4] +
 region.indic[,3]*coef(fit.vi.2000)$state.new[,5] +
 coef(fit.vi.2000)$state.new[,6]*avg.income[sl2000]
results[[9]]=cbind(income.vi.intercept.2000, income.vi.slope.2000); rownames(results[[9]])=sl2000

# 2004
sl2004 = as.integer(rownames(coef(fit.vi.2004)[[1]]))
income.vi.slope.2004 <- coef(fit.vi.2004)$state.new[,2]
income.vi.intercept.2004 <- coef(fit.vi.2004)$state.new[,1] +
 region.indic[,1]*coef(fit.vi.2004)$state.new[,3] +
 region.indic[,2]*coef(fit.vi.2004)$state.new[,4] +
 region.indic[,3]*coef(fit.vi.2004)$state.new[,5] +
 coef(fit.vi.2004)$state.new[,6]*avg.income[sl2004]
results[[10]]=cbind(income.vi.intercept.2004, income.vi.slope.2004); rownames(results[[10]])=sl2004

windows(width=9.5,height=4)
superp=superplot(results, start=9, end=10, rows=1, columns=2, indiv=1, lmer=1, var.slope=1, lowbound=-2, hibound=2, sf=3)

savePlot("annen2000-2004_vi.superplot", type=c("pdf"))
savePlot("annen2000-2004_vi.superplot", type=c("ps"))
savePlot("figure3", type=c("eps"))

# Figure 4
## plot varying intercept, varying slopes ##
# 2000
```

```
sl2000 = as.integer(rownames(coef(fit.2000)[[1]]))
##
income.slope.2000 <- coef(fit.2000)$state.new[,2] + (coef(fit.2000)$state.new[,10])*(avg.income[
    sl2000] )+
 region.indic[,1]*coef(fit.2000)$state.new[,7] +
 region.indic[,2]*coef(fit.2000)$state.new[,8] +
 region.indic[,3]*coef(fit.2000)$state.new[,9]
income.intercept.2000 <- coef(fit.2000)$state.new[,1] + (coef(fit.2000)$state.new[,6])*( avg.
    income [sl2000])+
 region.indic[,1]*coef(fit.2000)$state.new[,3] +
 region.indic[,2]*coef(fit.2000)$state.new[,4] +
 region.indic[,3]*coef(fit.2000)$state.new[,5]
results[[9]]=cbind(income.intercept.2000,income.slope.2000); rownames(results[[9]])=sl2000

# 2004
sl2004 = as.integer(rownames(coef(fit.2004)[[1]]))
income.slope.2004 <- coef(fit.2004)$state.new[,2] + (coef(fit.2004)$state.new[,10])*(avg.income[
    sl2004] )+
 region.indic[,1]*coef(fit.2004)$state.new[,7] +
 region.indic[,2]*coef(fit.2004)$state.new[,8] +
 region.indic[,3]*coef(fit.2004)$state.new[,9]
income.intercept.2004 <- coef(fit.2004)$state.new[,1] + (coef(fit.2004)$state.new[,6])*( avg.
    income [sl2004])+
 region.indic[,1]*coef(fit.2004)$state.new[,3] +
 region.indic[,2]*coef(fit.2004)$state.new[,4] +
 region.indic[,3]*coef(fit.2004)$state.new[,5]
results[[10]]=cbind(income.intercept.2004,income.slope.2004); rownames(results[[10]])=sl2004

windows(width=9.5,height=4)
superp=superplot(results, start=9, end=10, rows=1, columns=2, indiv=1, lmer=1, var.slope=1,
    lowbound=-2, hibound=2, sf=3)

savePlot("annen2000-2004_superplot",type=c("pdf"))
savePlot("annen2000-2004_superplot",type=c("ps"))
savePlot("figure4",type=c("eps"))

# Figure 5
avg.income.2004<-(state.income[[9]]$st_income)
avg.income.2000<-(state.income[[9]]$st_income)

windows(width=9.5,height=4)
par(mfrow=c(1,2))

plot(c(19000,40000),c(0,.5), main=2000, cex.lab=1.4, cex.main=1.7, type="n", ylab="Slopes", xlab="
    Median income within state", cex.axis=1.5, yaxt="n", xaxt="n", mgp=c(2.5,.5,0))
axis(side=2,seq(0,.4,.2),cex.axis=1.4)
axis(side=1,c(20000,30000,40000),labels=c("$20,000","$30,000","$40,000"),cex.axis=1.4)
text((avg.income.2000[sl2000]),income.slope.2000, label=state.abb[sl2000],cex=.9)
#lines(lowess((avg.income.2000[sl2000]),income.slope.2000))
abline (0,0,col="gray")

plot(c(19000,40000),c(0,.5), main=2004, cex.lab=1.4, cex.main=1.75, type="n", ylab="Slopes", xlab=
    "Median income within state", cex.axis=1.5, yaxt="n", xaxt="n", mgp=c(2.5,.5,0))
axis(side=2,seq(0,.4,.2),cex.axis=1.4)
axis(side=1,c(20000,30000,40000),labels=c("$20,000","$30,000","$40,000"),cex.axis=1.4)
text((avg.income.2004[sl2004]),income.slope.2004, label=state.abb[sl2004],cex=.9)
#lines(lowess((avg.income.2004[sl2004]),income.slope.2004))
abline (0,0,col="gray")

savePlot("annen2000-2004_slopes_income",type=c("pdf"))
savePlot("annen2000-2004_slopes_income",type=c("ps"))
savePlot("figure5",type=c("eps"))
```