Models for Panel Data

Simon Jackman

June 2, 2010

Contents

1 What is Panel Data?

The variables in the analysis span multiple dimensions: e.g., y_{it} where $i = 1, \ldots, n$ indexes units/individuals and $t = 1, \ldots, T$ indexes time. Same for X variables.

- We say a panel data set is "balanced" if we have the same number of observations in every unit; $T_i = T \; \forall \; i = 1, \ldots, n$.
- For either y_{it} or x_{it} (but usually y_{it} is the more interesting variable to consider) we have the following identity:

$$\text{Total Variance} = \text{Between-Unit Variance} + \text{Within-Unit Variance}$$

- Between-Unit Variance: how much variation is cross-sectional (across units)?
- Within-Unit Variance: how much variation is longitudinal?

- Many social science panel data sets from comparative/IR are characterized by the total variance being largely cross-sectional variation; aggregate data from the U.S. states often looks this way too. The things that make units (countries/states) different from one another on *y* are more or less time-invariant; e.g., institutional or geographical characteristics of units.
- Graphical inspection a useful first step: boxplot the dependent variable by cross-sectional units (if they are not too many of them); similarly, by time.

2 Looking at Panel Data

In my `R` package **pscl** the data frame `presidentialElections` contains data on Democratic vote share in presidential elections, by state, by election, since 1932.

Observe immediately that these data are "unbalanced", with an uneven number of observations per election:

R Code

```
> library(pscl)
```

```
Classes and Methods for R developed in the
Political Science Computational Laboratory
Department of Political Science
Stanford University
Simon Jackman
hurdle and zeroinfl functions by Achim Zeileis
```

R Code

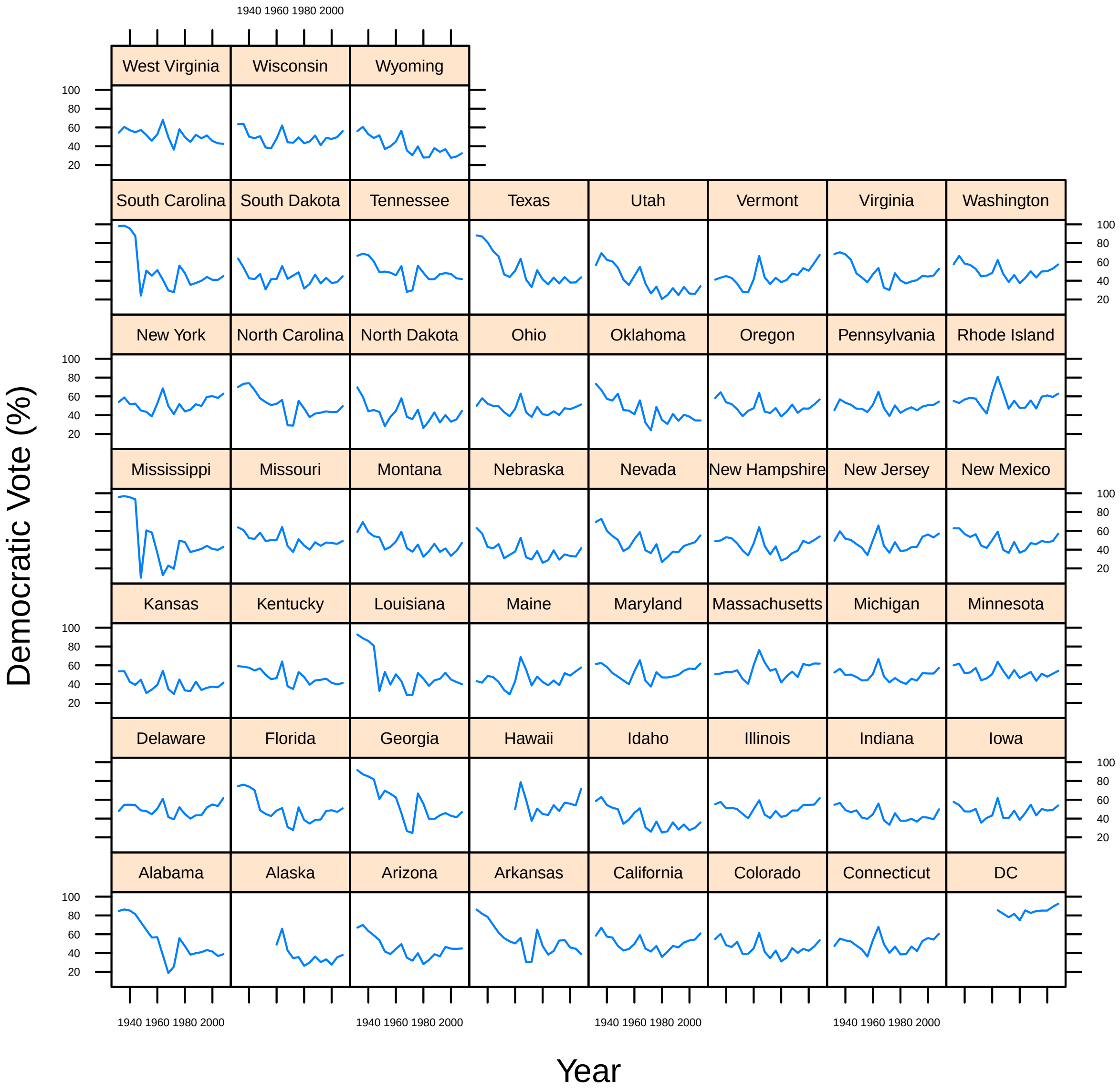
```
> data(presidentialElections)
> attach(presidentialElections)
> table(presidentialElections$year)
```

```
1932 1936 1940 1944 1948 1952 1956 1960 1964 1968 1972 1976 1980 1984 1988 1992
  48   48   48   48   47   48   48   50   50   51   51   51   51   51   51   51
1996 2000 2004 2008
  51   51   51   51
```

We inspect variation in these data by state and by election, using graphs. The 2nd form of the graph overlays the time trends from each state.

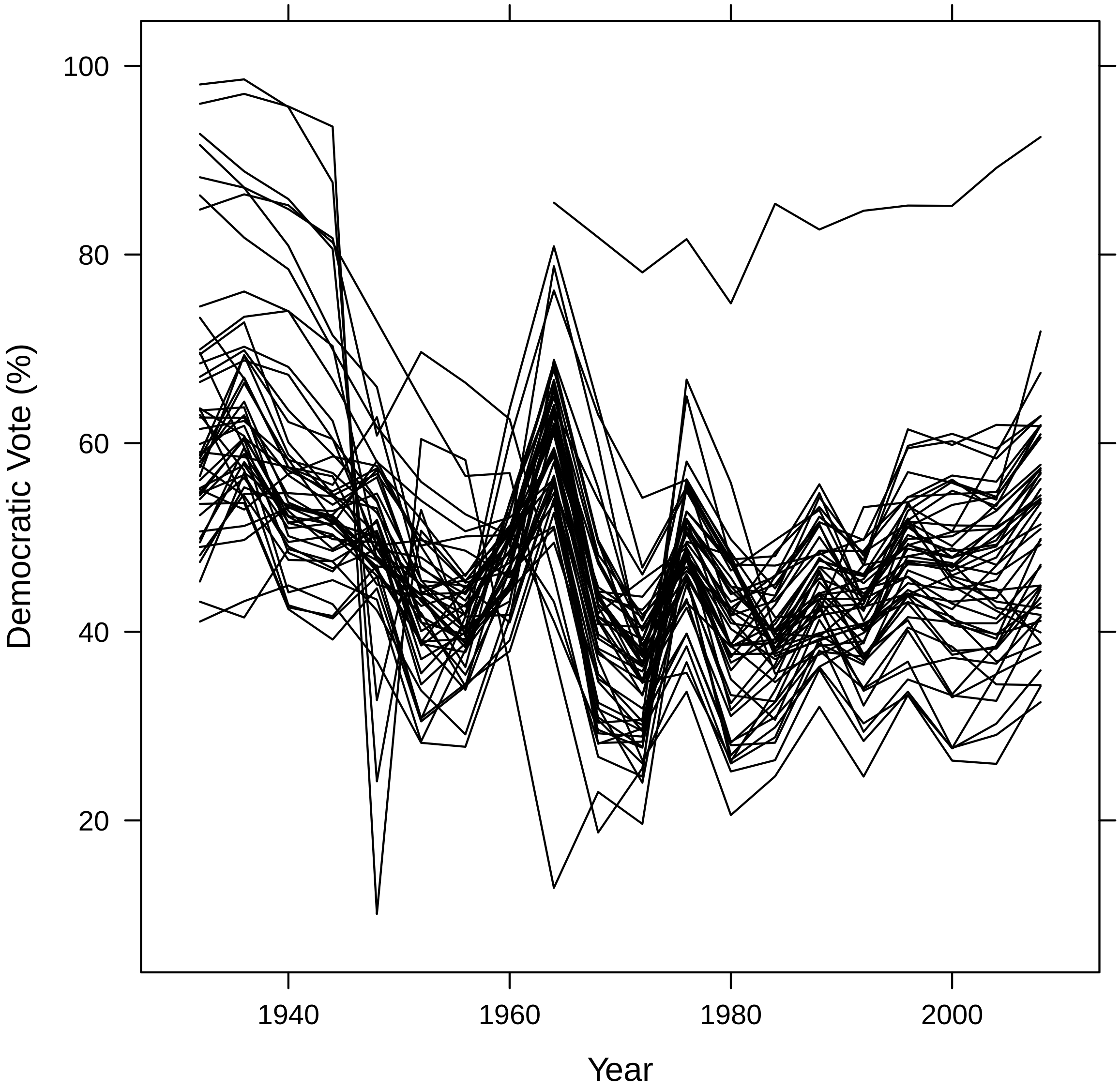
R Code

```
> library(lattice)
> print(xyplot(demVote ~ year | state,
+               data=presidentialElections,
+               scales=list(y=list(cex=.35),x=list(cex=.35)),
+               strip=strip.custom(par.strip.text=list(cex=.5)),
+               xlab="Year",
+               ylab="Democratic Vote (%)",
+               type="l")
+        )
```



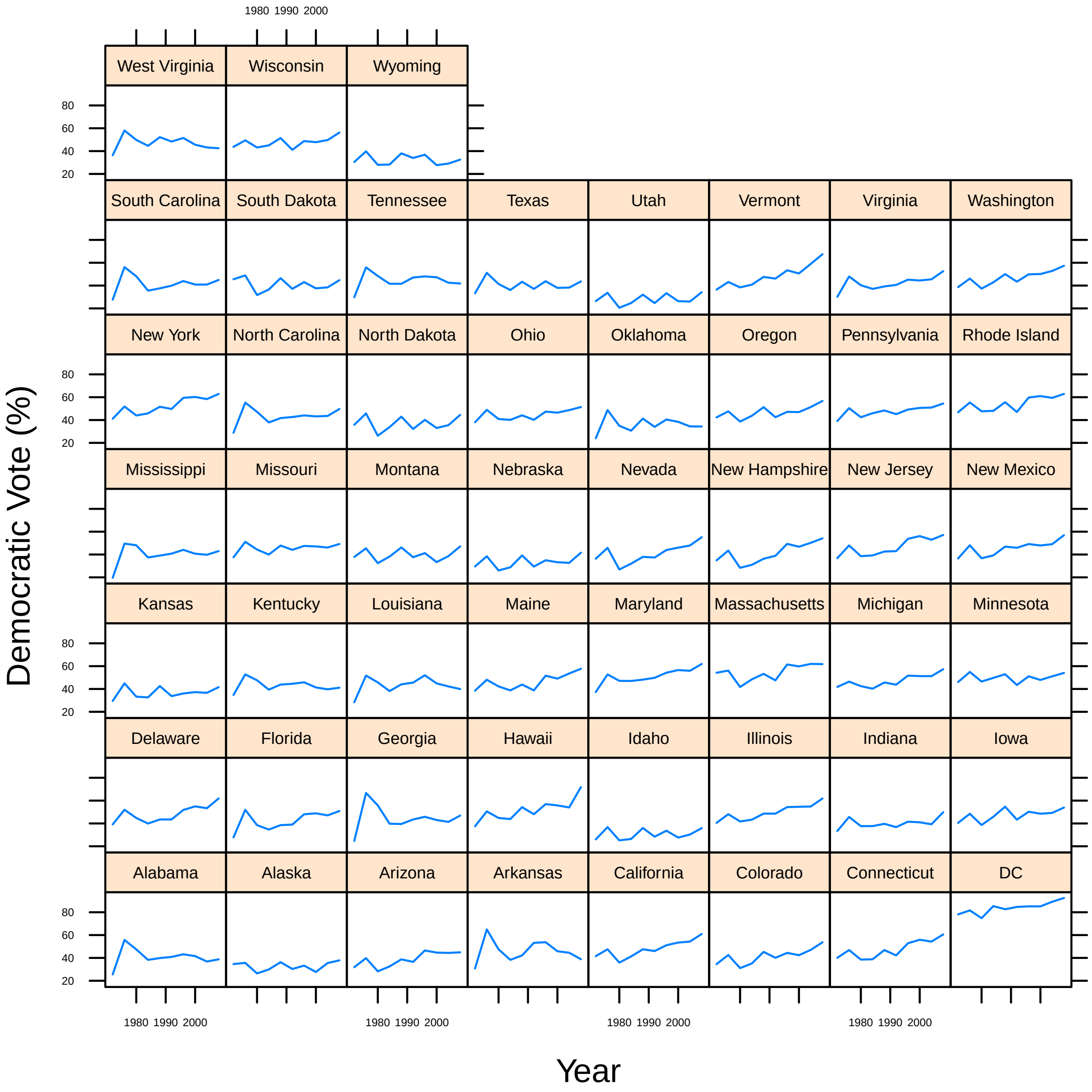
R Code

```
> print(xyplot(demVote ~ year,
+              group=state,
+              data=presidentialElections,
+              col="black",
+              xlab="Year",
+              ylab="Democratic Vote (%)",
+              type="l")
+       )
```



We subset the analysis to elections since 1972:

```
R Code
> print(xyplot(demVote ~ year | state,
+               subset=year>1968,
+               scales=list(y=list(cex=.35),x=list(cex=.35)),
+               strip=strip.custom(par.strip.text=list(cex=.5)),
+               xlab="Year",
+               ylab="Democratic Vote (%)",
+               type="l")
+        )
```



2.1 "Between" and "Within" Variance via Regression Analysis

We can estimate some extremely simple regressions, to give us estimates of the "within" and "between" variation. Consider the regression

$$E(y_{it}) = \mu + \alpha_i$$

where μ is a grand mean and α_i is a unit-specific offset. We can operationalize this regression by creating a set of *mutually exclusive and exhaustive* dummy variables spanning the units in the data set. That is, let

$$D_{it} = \left\{ \begin{array}{ll} 1 & \text{observation } it \text{ comes from unit } i \\ 0 & \text{otherwise} \end{array} \right\}, \quad i = 1, \ldots, n; t = 1, \ldots, T$$

R can easily create these dummy variables for us if the unit label is a `factor`.

Note that this regression is simply a fancy way of estimating the means within each group: i.e.,

$$\bar{y}_i = \frac{1}{T_i}\sum_{t=1}^{T_i} y_{it} = \hat{\mu} + \hat{\alpha}_i$$

where μ is estimated with the grand mean

$$\bar{y} = \hat{\mu} = \frac{\sum_{i=1}^{n}\sum_{t=1}^{T} y_{it}}{\sum_{i=1}^{n} T_i}$$

and so α_i is estimated with $\hat{\alpha}_i = \bar{y}_i - \bar{y}$.

Note that the parameters $\boldsymbol{\theta} = (\mu, \alpha_1, \ldots, \alpha_n)'$ are not jointly identified and so can not be estimated simultaneously. Typically a regression package will set one of the α_i to zero.

The key thing is that r^2 from this regression --- however we parameterize it --- gives us an estimate of how much variation in the y_{it} is due to the unit-specific means α_i:

R Code

```
> ## within regression
> w <- lm(demVote ~ as.factor(state),
+         subset=year>1968)
> summary(w)$r.squared
```

```
[1] 0.6229344
```

R Code

```
> anova(w)
```

```
Analysis of Variance Table

Response: demVote
                  Df Sum Sq Mean Sq F value    Pr(>F)
as.factor(state)  50  33223  664.45  15.166 < 2.2e-16 ***
Residuals        459  20110   43.81
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The r^2 for this regression is not small, and the F-statistic is extremely significant, indicating that the grouping by state is an important component of these data.

Intra-class correlation coefficient. We can also compute the *intra-class correlation coefficient*, the ratio of the variance in y due to the fixed effects to the total variance in y,

$$ICC = \omega^2/(\omega^2 + \sigma^2)$$

where ω^2 is the "between variance" (the cross-unit variance of the unit-specific averages) and σ^2 is the "within variance" (the variance of the y_{it} around the unit-specific averages). Equivalently (assuming a balanced panel),

$$ICC = \frac{MSB - MSW}{MSB + (\frac{nT}{n+1} - 1)MSW}$$

where

$$MSB = n^{-1}T\sum_{i=1}^{n}(\bar{y}_i - \bar{y})^2$$

is the “between” unit mean-corrected sum-of-squares and

$$MSW = \frac{1}{nT - n - 1} \sum_{i=1}^{n} \sum_{t=1}^{T} (y_{it} - \bar{y}_i)^2$$

is the “within” mean-corrected sum-of-squares.

There are some easy-to-use tools for performing this calculation in `R`, including the `ICC` functions in the **psychometric** package:

R Code

```
> library(psychometric)
> tmp <- aov(demVote ~ state,
+            data=presidentialElections,
+            subset=year>1968)
> summary(tmp)
```

```
             Df Sum Sq Mean Sq F value    Pr(>F)    
state        50  33223  664.45  15.166 < 2.2e-16 ***
Residuals   459  20110   43.81                      
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

R Code

```
> ICC1(tmp)
```

```
[1] 0.5861937
```

R Code

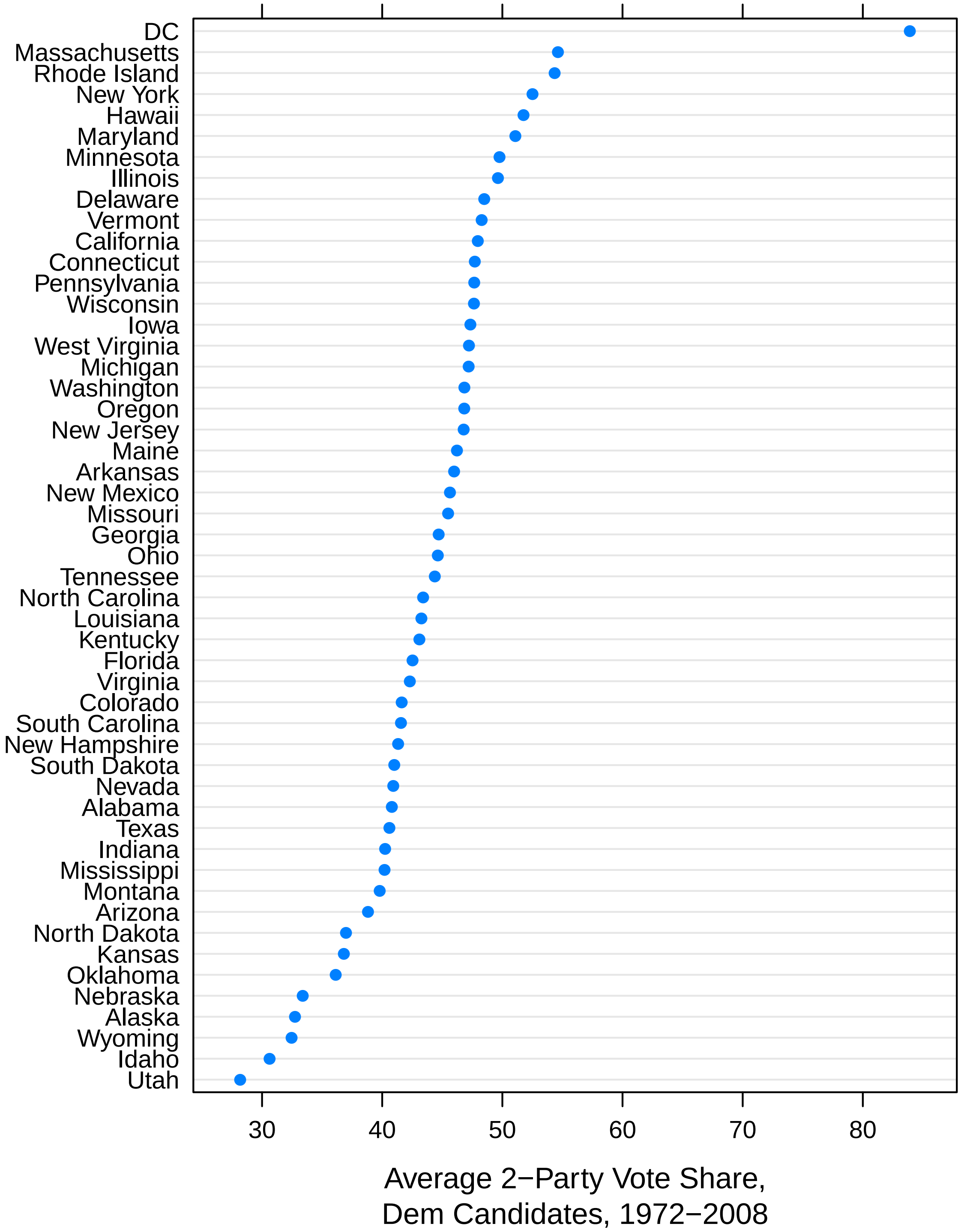
```
> ICC1.lme(demVote,state,presidentialElections[year>1968,])
```

```
The following object(s) are masked from 'presidentialElections':

    demVote, south, state, year
[1] 0.5861937
```

If the goal was simply to compute averages of y_{it} by state (i.e., $\bar{y}_i$), then we can do this rather simply, revealing Ohio to be “the median state”, say, ranking the state averages:

R Code

```
> ## state averages "manually"
> ok <- year>1968
> alpha <- tapply(demVote[ok],state[ok],mean)
> print(dotplot(sort(alpha),
+                xlab="Average 2-Party Vote Share,\nDem Candidates, 1972-2008"))
```


DC
Massachusetts
Rhode Island
New York
Hawaii
Maryland
Minnesota
Illinois
Delaware
Vermont
California
Connecticut
Pennsylvania
Wisconsin
Iowa
West Virginia
Michigan
Washington
Oregon
New Jersey
Maine
Arkansas
New Mexico
Missouri
Georgia
Ohio
Tennessee
North Carolina
Louisiana
Kentucky
Florida
Virginia
Colorado
South Carolina
New Hampshire
South Dakota
Nevada
Alabama
Texas
Indiana
Mississippi
Montana
Arizona
North Dakota
Kansas
Oklahoma
Nebraska
Alaska
Wyoming
Idaho
Utah
30
40
50
60
70
80
Average 2−Party Vote Share,
Dem Candidates, 1972−2008

3 Simpson's Paradox: Traffic Fatalities and the Beer Tax

We examine another panel data set, examining the link between traffic fatalities and beer taxes, in the 48 contiguous U.S. states, annually from 1982 to 1988, due to Ruhm (1996). The dependent variable is the vehicle fatality rate (annual, per 10,000 people). We begin (as usual) with some graphics:

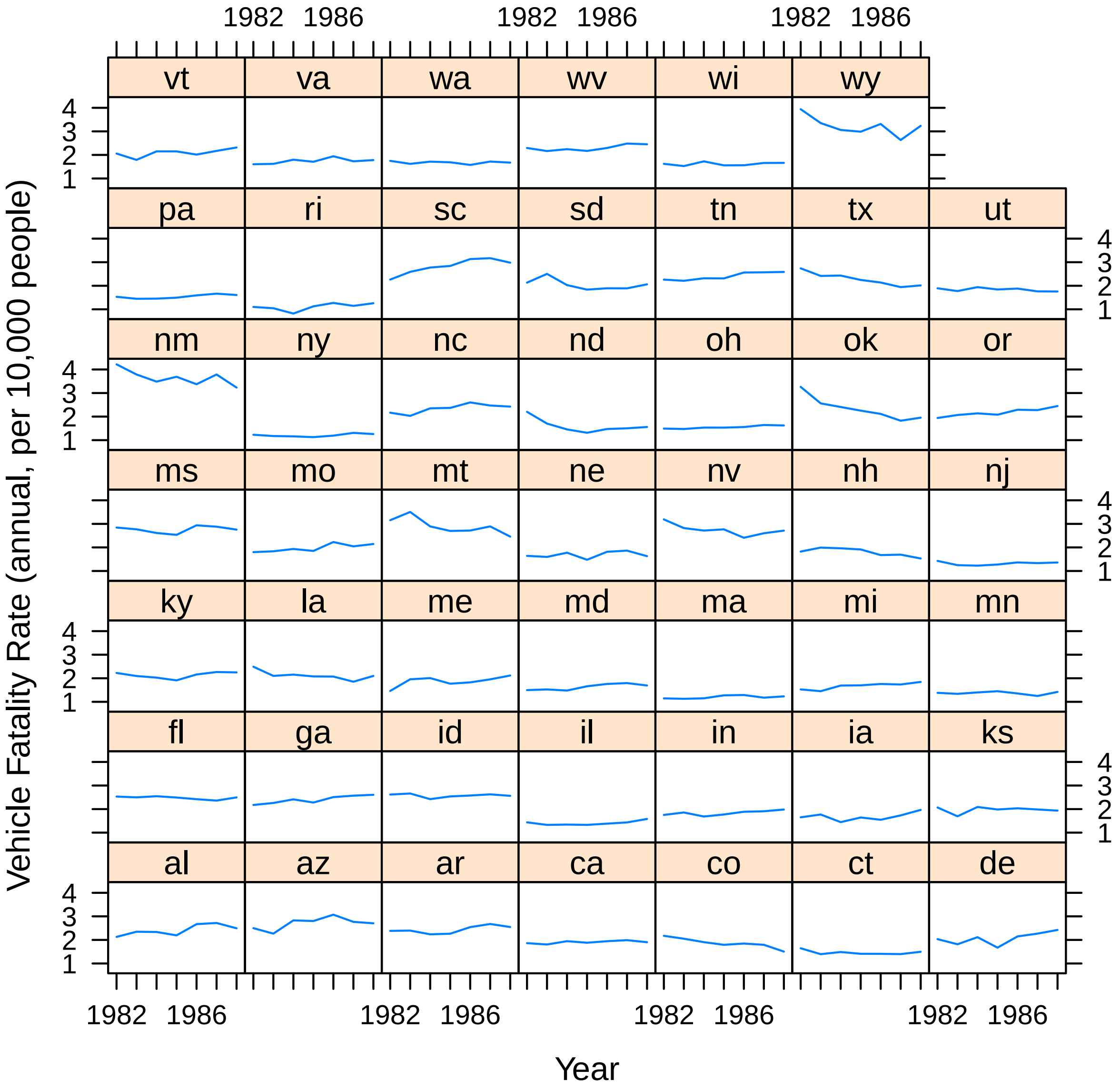
R Code

```
> library(foreign)
> fatality <- read.dta(file="fatality.dta")
> fatality$y <- fatality$mrall*10000
> summary(fatality$y)
```

```
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
0.8212  1.6240  1.9560  2.0400  2.4180  4.2180
```

R Code

```
> print(xyplot(y ~ year | state,
+              data=fatality,
+              panel=panel.lines,
+              ylab="Vehicle Fatality Rate (annual, per 10,000 people)",
+              xlab="Year")
+       )
```



This plot reveals a good deal variation in fatalities by state. Other statistics confirm this:

R Code

```
> ICC1.lme(y,state,fatality)
```

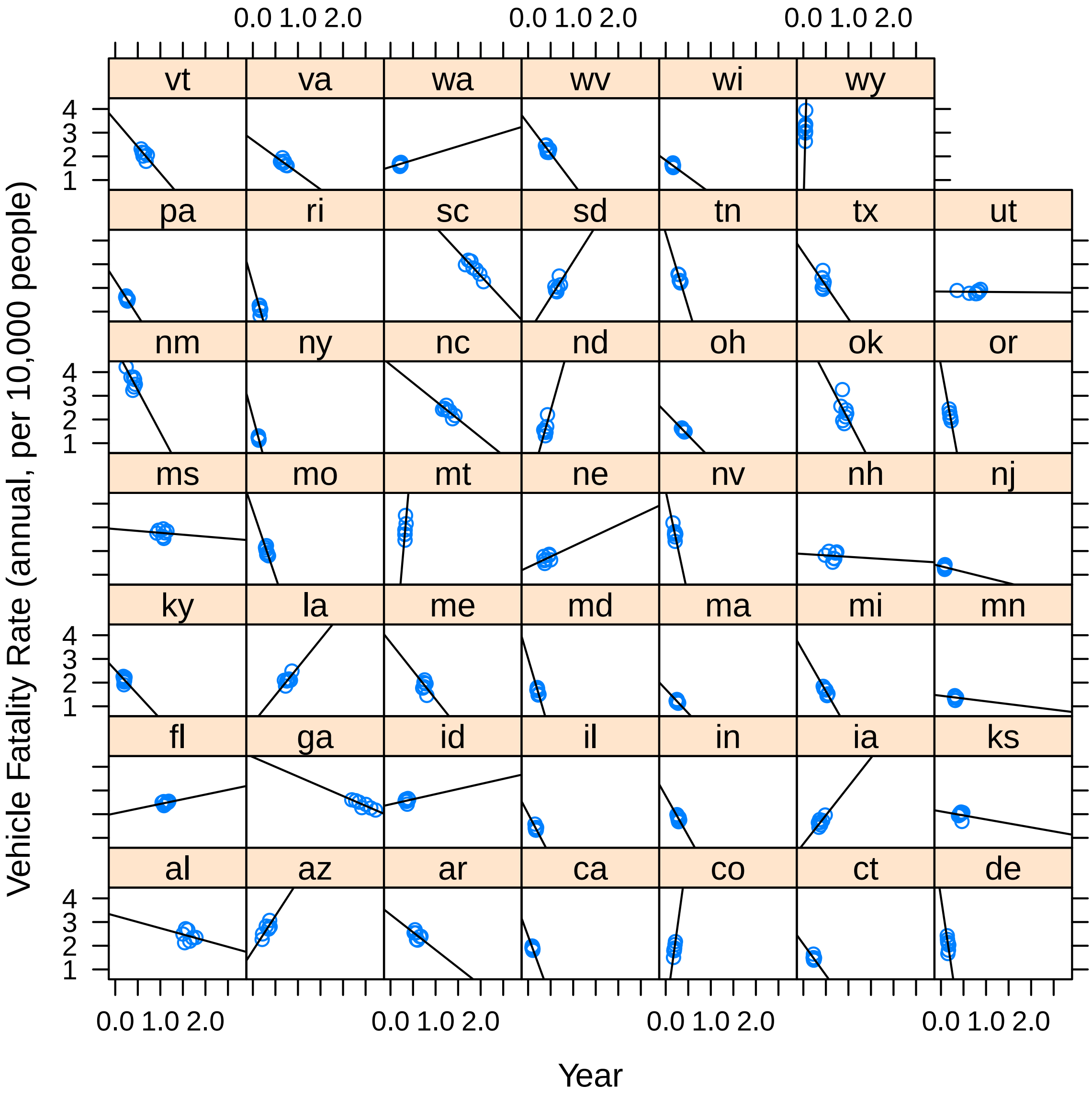
```
The following object(s) are masked from 'presidentialElections':

    state, year
[1] 0.8866498
```

We now plot the relationship between fatality rates and beer tax, by state, again revealing considerable heterogeneity by state, but also extremely little within-state variation in beer taxes:

R Code

```
> print(xyplot(y ~ beertax | state,
+              data=fatality,
+              panel=function(x,y,...){
```

```
+                 panel.xyplot(x,y,...)
+                 panel.lmline(x,y,...)
+               },
+               ylab="Vehicle Fatality Rate (annual, per 10,000 people)",
+               xlab="Year"))
```



We confirm that there is very little within-state variation in beer tax:

R Code

```
> ICC1.lme(beertax,state,fatality)
```

```
The following object(s) are masked from 'presidentialElections':

    state, year
[1] 0.9847402
```

That is, the ICC for beertax is massive: 98% of the variation in beer-taxes is between-state, and so beer-tax is almost a time-invariant fixed attribute of a state.

We begin with a simple model, fitting only a state-specific mean to each state's fatality rate:

R Code

```
> m1 <- lm(y ~ state,
+          data=fatality)
> summary(m1)$r.squared
```

```
[1] 0.9009802
```

3.1 Naive estimate of the effect of the beertax

We also fit the naive regression of fatalities on beer tax, momentarily ignoring the considerable between-state heterogeneity in both y and X, fitting the regression

$$y_{it} = \beta_0 + \beta_1 x_{it} + \varepsilon_{it}$$

R Code

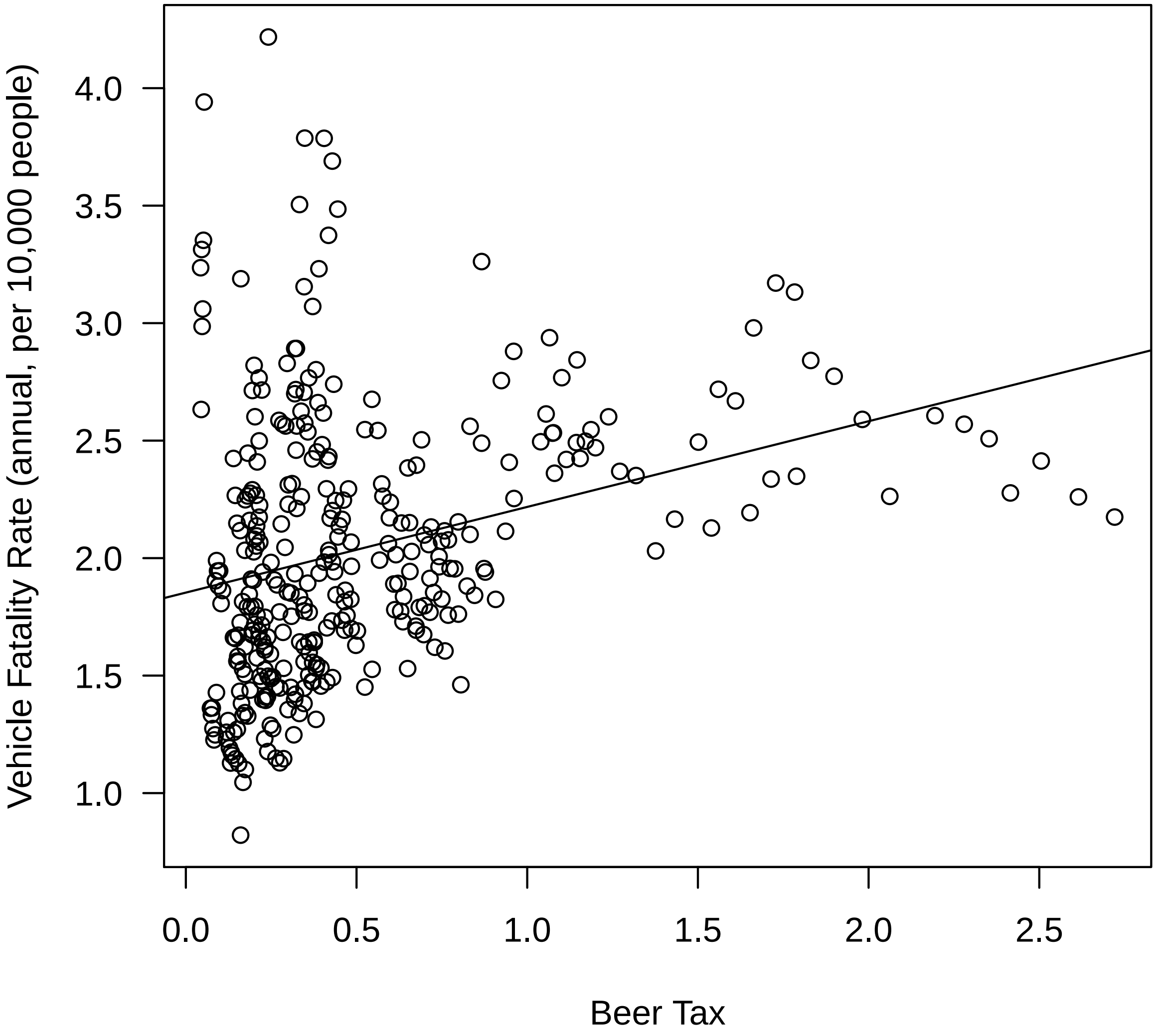
```
> m2 <- lm(y ~ beertax,
+          data=fatality)
> summary(m2)
```

```
Call:
lm(formula = y ~ beertax, data = fatality)

Residuals:
     Min       1Q   Median       3Q      Max
-1.09060 -0.37768 -0.09436  0.28548  2.27643

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.85331    0.04357  42.539  < 2e-16 ***
beertax      0.36461    0.06217   5.865 1.08e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5437 on 334 degrees of freedom
Multiple R-squared: 0.09336,	Adjusted R-squared: 0.09065
F-statistic: 34.39 on 1 and 334 DF,  p-value: 1.082e-08
```

We overlay this regression fit on an unconditional scatterplot of the data, again momentarily ignoring the grouping by state:

R Code

```
> plot(y~beertax,
+      xlab="Beer Tax",
+      ylab="Vehicle Fatality Rate (annual, per 10,000 people)",
+      las=1,
+      data=fatality)
> abline(lm(y~beertax,data=fatality))
```



Note that this regression leads to the seemingly odd result that increases in the beer tax *increase* the fatality rate; each additional dollar-per-case increase in the beer tax leads to additional 36 deaths per million residents ($t = 5.9$). Do note the poor fit of this model with an r^2 of .09 relative to the .90 r^2 we obtained when we use just fixed effects for states as predictors.

3.2 Unit-Specific Omitted Variables?

What is going on here? Consider regression models for panel data, generically,

$$E(y_{it}|\mathbf{x}_{it}, c_i) = \beta_0 + \mathbf{x}_{it}\boldsymbol{\beta} + c_i$$

where $i = 1, \ldots, n$ indexes units; $t = 1, \ldots, T$ indexes time. To keep the exposition simple, we restriction attention to the balanced case. In this setup:

- c_i is an unobserved, time-constant (time-invariant) quantity; a source of unobserved, unit-specific heterogeneity in y_{it}
- c_i might be particularly relevant if the data display a good deal of cross-sectional or *between* variation, *and* the $\mathbf{x}_{it}$ available for analysis do a poor job of soaking up that cross-sectional variation.
- If $\mathbf{x}_{it}$ does a good job of capturing variation in y_{it}, then maybe we can ignore c_i?
- Critical issue: if the c_i are *uncorrelated* with the $\mathbf{x}_{it}$, we can ignore them, wrapping them into a *compound error term*, $v_{it} = c_i + u_{it}$ that has zero mean and is uncorrelated with the regressors $\mathbf{x}_{it}$.

Ignoring the c_i?:

- A situation in which we might ignore the c_i is when x_i is assigned randomly to cases, as in an experiment.
- In such a case the x_i are uncorrelated with the c_i (and anything else for that matter), and so we can recover an unbiased estimate of $\boldsymbol{\beta}$ by ignoring c_i.
- On the other hand, under these conditions, while $\hat{\boldsymbol{\beta}}$ is unbiased/consistent, there are *efficiency gains* to be had from extracting the c_i from the error term.
- This is what the **random effects** (RE) estimator does.
- When c_i is correlated with X_i then the **fixed effects** estimator is an attractive alternative.

4 Fixed Effects Estimator

- Treat the c_i as *parameters* to be estimated; but (for purposes of implementation) then apply transformations to the data that effectively make the c_i disappear.
- Suppose we have at least two time periods, and average the model

  $$y_{it} = \mathbf{x}_{it}\boldsymbol{\beta} + c_i + u_{it}$$

  within a unit to yield

  $$\bar{y}_i = \bar{\mathbf{x}}_i\boldsymbol{\beta} + c_i + \bar{u}_i$$

  where $\bar{y}_i = T^{-1}\sum_{i=1}^{T} y_{it}$ and similarly for $\bar{\mathbf{x}}_i$. Note that the c_i appears in the averaged model, since the average of a constant is the constant. But if we then consider the *differenced* model, the c_i drops out:

  $$y_{it} - \bar{y}_i = (\mathbf{x}_{it} - \bar{x}_i)\boldsymbol{\beta} + u_{it} - \bar{u}_i$$

- Running OLS on this differences model produces the *fixed effects estimator* of $\boldsymbol{\beta}$.
- Equivalent to running the regression and including a series of mutually exclusive and exhaustive dummy variables for the units/individuals (the unique values of the i subscript); but this is a computationally klunky way to do it for large n.

- Obvious that information about $\boldsymbol{\beta}$ comes from the within-unit covariation between $\mathbf{x}_i$ and $\mathbf{y}_i$; hence the fixed effects estimator is sometimes called the *within estimator*, and the differencing above is sometimes called the *within transformation*.
- Any variable that has no "within" variation has to be dropped from the analysis (such a variable is co-linear with the fixed effects for the units).
- If a variable is time-invariant within one unit (but not all), then the fixed effect for that unit isn't identified; `agl` example (the U.S. is time-invariant on the `left` cabinet seats variable in the `agl` data).

4.1 Fixed Effects for the Traffic Fatalities example

We now run the panel estimators: first, we consider a fixed effects or "within" estimator; we use the `plm` package in `R`, which has a lot of functions for **p**anel **l**inear **m**odels.

We first create a `plm.data` object called `plm.fatality`. We then estimate the fixed effects regression by specifying the `method="within"` option to `plm`

R Code

```
> library(plm)
```

```
[1] "kinship is loaded"
```

R Code

```
> plm.fatality <- plm.data(fatality,
+                          index=c("state","year"))
> m3 <- plm(y ~ beertax,
+           data=plm.fatality,
+           method="within")
> summary(m3)
```

```
Oneway (individual) effect Within Model

Call:
plm(formula = y ~ beertax, data = plm.fatality, method = "within")

Balanced Panel: n=48, T=7, N=336

Residuals :
    Min.  1st Qu.   Median  3rd Qu.     Max.
-0.58700 -0.08280 -0.00127  0.07950  0.89800

Coefficients :
        Estimate Std. Error t-value Pr(>|t|)
beertax -0.65587    0.18785 -3.4915 0.000556 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Total Sum of Squares:    10.785
Residual Sum of Squares: 10.345
F-statistic: 12.1904 on 1 and 287 DF, p-value: 0.00055597
```

Note the striking change in the estimated effect of the beer tax; in the presence of fixed effects for state, each dollar-per-case increase in the beer-tax is estimated to reduce traffic fatalities by about 66 people per million residents. The r^2 from this regression isn't reported by the `plm` software, but it is easily computed from the `lm` analog:

R Code

```
> summary(lm(y~beertax+state,data=fatality))$r.squared
```

```
[1] 0.9050147
```

That is, we get a tiny boost in goodness-of-fit from including the beer-tax information versus merely including the state fixed effects, but not much.

5 Random Effects

- Model:

$$\begin{aligned} y_{it} &= \mathbf{x}_{it}\boldsymbol{\beta} + c_i + u_{it} \\ V(c_i|\mathbf{x}_i) &= \sigma_c^2 \\ V(u_{it}|\mathbf{x}_i) &= \sigma_u^2 \end{aligned}$$

- That is, we estimate
  1. the regression coefficients $\boldsymbol{\beta}$
  2. the *between-unit* variance of the c_i, σ_c^2
  3. the *within-unit* variance of the c_i, σ_u^2
- We don't estimate the unit-specific terms c_i, we merely assume that they come from a distribution with variance σ_c^2.
- Methods of estimation are beyond our scope here but include FGLS, ML, REML.

5.1 Random Effects for the Traffic Fatalities example

It is interesting to consider how random effects fares with the traffic fatality data. data. We deploy the FGLS-RE estimator in `plm`:

R Code

```
> beertaxRE <- plm(y~beertax,
+                  data=plm.fatality,
+                  model="random")
> summary(beertaxRE)
```

```
Oneway (individual) effect Random Effect Model
   (Swamy-Arora's transformation)

Call:
plm(formula = y ~ beertax, data = plm.fatality, model = "random")

Balanced Panel: n=48, T=7, N=336

Effects:
                  var std.dev share
idiosyncratic 0.03605 0.18986 0.119
individual    0.26604 0.51579 0.881
theta:  0.8622

Residuals :
   Min. 1st Qu.  Median 3rd Qu.    Max.
-0.4710 -0.1200 -0.0215  0.0910  0.9640
```

```
Coefficients :
             Estimate Std. Error t-value Pr(>|t|)
(Intercept)  2.067141   0.099971 20.6773   <2e-16 ***
beertax     -0.052016   0.124176 -0.4189   0.6756
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Total Sum of Squares:    12.648
Residual Sum of Squares: 12.642
F-statistic: 0.175467 on 1 and 334 DF, p-value: 0.67557
```

The estimated effect of the beer tax is indistinguishable from zero in this analysis. The REML and ML estimators in the `lme4` package give us a similar estimate:

R Code

```
> library(lme4)
> r1 <- lmer(y ~ beertax + (1|state),
+            data=fatality)
> r2 <- update(r1,REML=FALSE)   ## ML estimator when REML=FALSE
> cbind(summary(r1)@coefs[,1:2],
+       summary(r2)@coefs[,1:2])
```

```
               Estimate Std. Error   Estimate Std. Error
(Intercept)  2.08420148  0.1045371  2.0790765  0.1028405
beertax     -0.08525512  0.1276586 -0.0752699  0.1262476
```

6 Fixed or Random Effects?

If the c_i are uncorrelated with the regressors $\mathbf{x}_{it}$ then we ought to use the random effects estimator (it is the more efficient estimator). On the other hand, if the c_i and the $\mathbf{x}_{it}$ are correlated, then the random effects estimator of $\boldsymbol{\beta}$ is biased, with the magnitude of the bias a function of the correlation between the $\mathbf{x}_{it}$ and the c_i; the fixed effects estimator is the one to use in the case.

A popular statistical test of whether to use fixed or random effects is the Hausman test. The intuition is to compare the two estimators; if the sample is large, and the two estimators are giving us different results, then it must be the case that the $\mathbf{x}_{it}$ and the c_i are correlated, generating bias in the RE estimator (n.b., FE is an unbiased estimator, and remains so irrespective of the correlation between $\mathbf{x}_{it}$ and the c_i). On the other hand, if the two estimators are giving us (roughly) the same estimate of $\boldsymbol{\beta}$, then we prefer the RE estimator, because it is the more efficient of the two. In bullet-point form:

- Fixed effects estimator: under strict exogeneity and conditional iid assumptions above, $\hat{\boldsymbol{\beta}}_{FE}$ is consistent.

- Random effects estimator: under same assumptions as above, plus $E(c_i|\mathbf{x}_i) = E(c_i)$ (orthogonality of unit-specific effects and predictors), $\hat{\boldsymbol{\beta}}_{RE}$ is also consistent, but more efficient than *FE*.

- Use *RE* if the orthogonality assumption holds; but often it doesn't.

The statistical theory underlying the Hausman test, which turns out to be a χ^2 test:

Proposition 1 (Hsiao (2003, Lemma 3.5.1)) *Based on a sample of n observations, consider two estimates $\hat{\boldsymbol{\beta}}_0$ and $\hat{\boldsymbol{\beta}}_1$ that are both consistent and asymptotically normal, with $\hat{\boldsymbol{\beta}}_0$ attaining the Cramer-Rao bound. Let $\hat{\mathbf{q}} = \hat{\boldsymbol{\beta}}_1 - \hat{\boldsymbol{\beta}}_0$. Then the limiting joint distribution of $\sqrt{n}(\hat{\boldsymbol{\beta}}_0 - \boldsymbol{\beta})$ and $\sqrt{n}\hat{\mathbf{q}}$ has zero covariance between these two terms.*
Proof: Rao (1973, 317).

Corollaries:

1. $V(\hat{\mathbf{q}}) = V(\hat{\boldsymbol{\beta}}_1) - V(\hat{\boldsymbol{\beta}}_0)$
2. (Hausman 1978): Under $H_0 : E(c_i|\mathbf{x}_i) = 0$ (orthogonality), $m = \hat{\mathbf{q}}'V(\hat{\mathbf{q}})^{-1}\hat{\mathbf{q}} \sim \chi^2_k$, where k is the dimension of $\mathbf{q}$.

This suggests an easily-implemented, asymptotically valid test ($T \rightarrow \infty$): simply compare $\hat{\boldsymbol{\beta}}_{FE}$ and $\hat{\boldsymbol{\beta}}_{RE}$ using the testing framework above. If m exceeds a critical quantile of the χ^2_k density, then reject H_0 (orthogonality $\Rightarrow$ *RE*) in favor of H_A (*FE*).

6.1 Hausman Test for Traffic Fatalities Example

A Hausman test leads to a confident rejection that the fixed effects and random effects estimators are giving us similar answers:

R Code

```
> phtest(y~beertax,data=fatality)
```

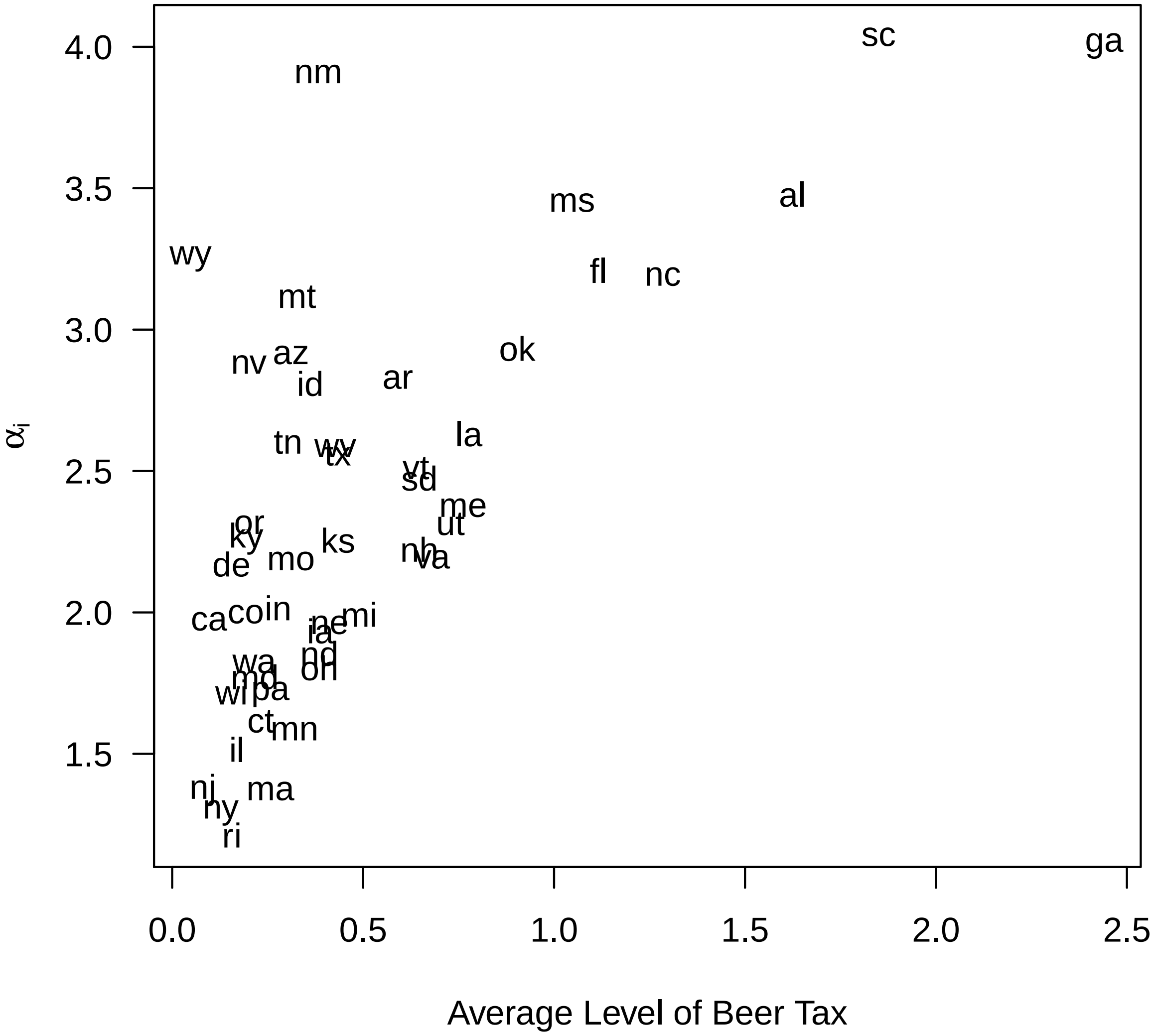
```

        Hausman Test

data:  y ~ beertax
chisq = 18.3534, df = 1, p-value = 1.835e-05
alternative hypothesis: one model is inconsistent
```

On the basis of this evidence we would prefer the fixed effects estimate over the random effects estimate.

The difference between the fixed effects and the random effects estimators is rather obvious: the state-level constants α_i are correlated with the predictor x_{it}. We plot the estimated α_i (from the fixed effects estimators) against the average beer tax in state i, $\bar{x}_i$:

R Code

```
> beerTaxAverage <- tapply(fatality$beertax,
+                          fatality$state,
+                          mean)
> fixed1 <- lm(y ~ -1 + state + beertax,
+             data=fatality)
> stateFE <- coef(fixed1)[1:48]
> plot(y=stateFE,x=beerTaxAverage,
+      las=1,
+      type="n",
+      xlab="Average Level of Beer Tax",
+      ylab=expression(alpha[i]))
> text(beerTaxAverage,stateFE,names(beerTaxAverage))
```



7 Heterogeneity across states in slopes

While it would seem we need state-specific intercepts to model these data, what about the assumption that the effect of the beer tax is constant across states? The graphical exploration of the data earlier suggests that there is considerable heterogeneity in the state-level effects, which we will now attempt to model. That is, we will deploy a model in which intercepts and slopes vary across states: i.e., $y_{it} = \alpha_i + \beta_i x_{it} + \epsilon_{it}$, subject to the constraint that the error variance σ^2 is constant across states. This model is easy to estimate via OLS via `lm`:

R Code

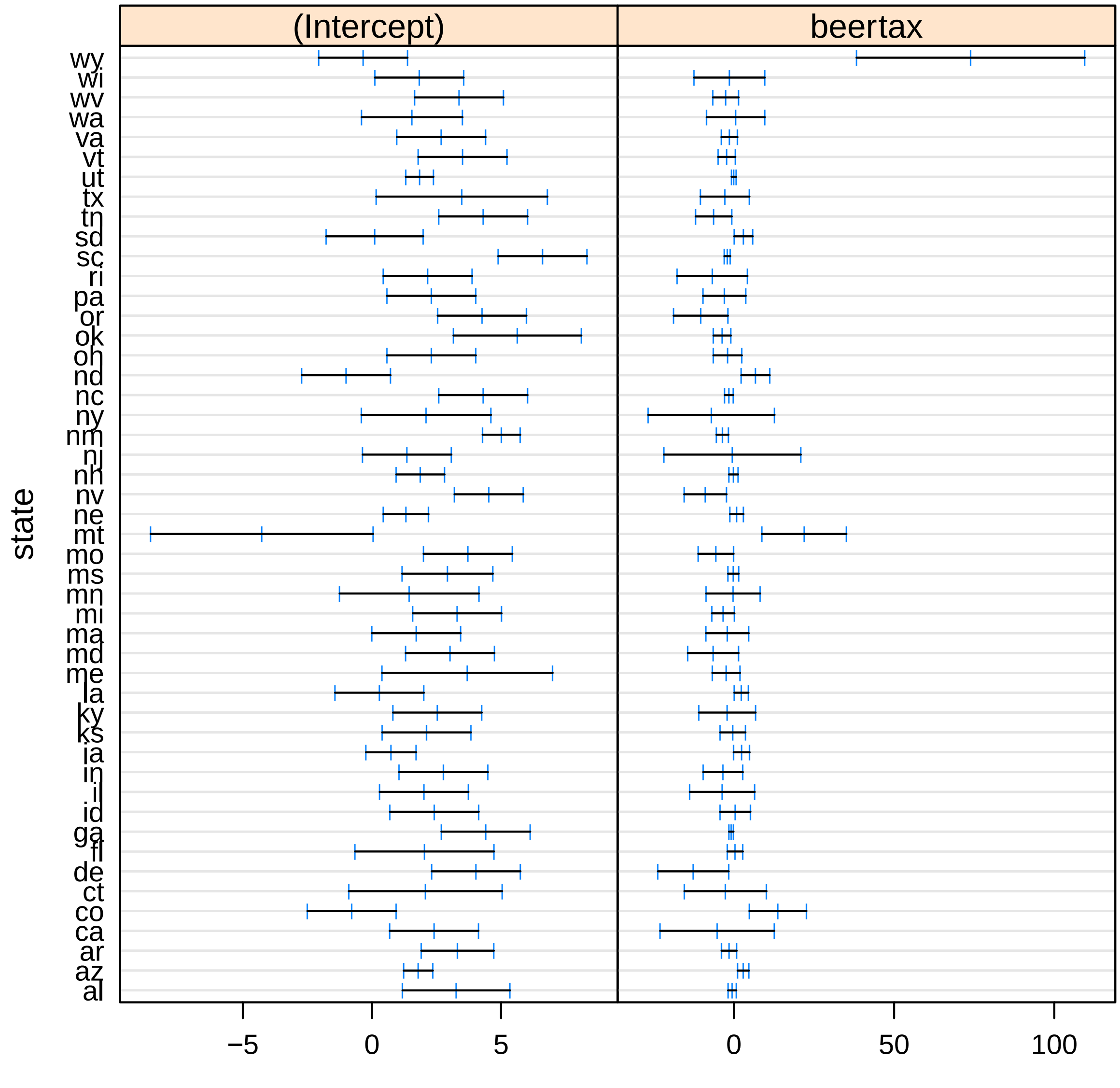
```
> nopool <- lm(y ~ beertax*state,
+              data=fatality)
> summary(nopool)$r.squared
```

```
[1] 0.9416989
```

The model fits well, as it should, fitting a slope and intercept to the data from each state. The model consumes 2 degrees of freedom per state for a total of 2×48 = 96 degrees of freedom. A simple way to recover all the state-specific intercepts and slopes is to use the `lmList` command that is part of the **lme4** and **nlme** packages. We use the **lme4** version, but pull some particular features in **nlme** that let us plot the 96 parameter estimates (and confidence intervals):

R Code

```
> tmp <- lme4::lmList(y ~ beertax | state,
+                       data=fatality)
> tmp.int <- nlme::intervals(tmp)
> attr(tmp.int,"groupsName") <- "state"
> print(nlme:::plot.intervals.lmList(tmp.int))
```



This graphical inspection reveals considerable variation in the state-specific estimates of the beer tax coefficient. Wyoming produces a coefficient of over 70, since there is almost no variation in beer tax within that state:

R Code

```
> summary(tmp[["wy"]])
```

```
Call:
lm(formula = formula, data = data)

Residuals:
    330     331     332     333     334     335     336 
 0.3150 -0.1188 -0.2519 -0.1945  0.2238 -0.3508  0.3772 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)  
(Intercept)  -0.3413     1.7477  -0.195   0.8528  
beertax      73.8848    36.1939   2.041   0.0967 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 

Residual standard error: 0.3256 on 5 degrees of freedom
Multiple R-squared: 0.4546,	Adjusted R-squared: 0.3455 
F-statistic: 4.167 on 1 and 5 DF,  p-value: 0.0967 
```

R Code

```
> fatality$beertax[fatality$state=="wy"]
```

```
[1] 0.05369928 0.05160551 0.04945055 0.04766949 0.04643963 0.04500000 0.04331088
```

Montana and Colorado also have double-digit slope estimates (positive), while Delaware and Oregon have double-digit negative slopes:

R Code

```
> sort(summary(tmp)$coefficients[,"Estimate","beertax"])
```

```
          de           or           nv           ny           ri           md 
-12.65764877 -10.32992038  -8.89801742  -7.02043686  -6.72923297  -6.45344563 
          tn           mo           ca           ok           il           nm 
 -6.28578525  -5.60224348  -5.20930496  -3.64469486  -3.62528814  -3.56052747 
          in           mi           pa           tx           ct           wv 
 -3.38041667  -3.32795766  -2.95837338  -2.78407568  -2.63176945  -2.51856738 
          me           vt           ky           sc           ma           oh 
 -2.39701516  -2.22510792  -2.06377268  -2.04909988  -2.02313650  -1.95730905 
          nc           ar           wi           va           ga           al 
 -1.52858534  -1.48858957  -1.39238846  -1.38338901  -0.82295165  -0.52378082 
          nj           ks           mn           ms           nh           ut 
 -0.47396023  -0.33687100  -0.23488185  -0.15835264  -0.11979406  -0.01475891 
          fl           id           wa           ne           la           ia 
  0.39594660   0.42879208   0.57987869   0.89312087   2.35366477   2.42216109 
          az           sd           nd           co           mt           wy 
  2.93409976   2.99925623   6.75065461  13.74022843  21.94151091  73.88475832 
```

Note also that the state-specific slopes are estimated with considerable imprecision: with just 5 degrees of freedom in the $n = 7$ state-by-state regressions.

Random slope coefficients. A better way to estimate the state-specific slopes would be via an analog with "random-effects". That is, we let the slopes vary across states, but randomly, with the following model:

$$\begin{aligned} y_{it} &\sim N(\alpha_i + \beta_i x_{it}, \sigma^2) \\ \beta_i &\sim N(\beta, \sigma^2_\beta) \end{aligned}$$

and with α_i treated as “fixed effects”. The parameter β is the average effect of the beer-tax, and σ_β^2 is the between-state variation in the state-specific effects β_i. Since we are treating the α_i as fixed effects, we might also consider centering the data by state to yield

$$\begin{aligned} y_{it} - \bar{y}_i &\sim N(\beta_i[x_{it} - \bar{x}_i], \sigma^2) \\ \beta_i &\sim N(\beta, \sigma_\beta^2) \end{aligned}$$

Note that α_i has dropped out of this form of the model. We can estimate this model with `lmer` in **lme4**:

R Code

```
> center <- function(x)x-mean(x)
> tmpData <- data.frame(y=unlist(tapply(fatality$y,fatality$state,center)),
+                       state=fatality$state,
+                       beertax=unlist(tapply(fatality$beertax,
+                         fatality$state,center)))
> randomSlopes <- lmer(y ~ -1 + beertax + (0 + beertax | state),
+                      data=tmpData)
> summary(randomSlopes)
```

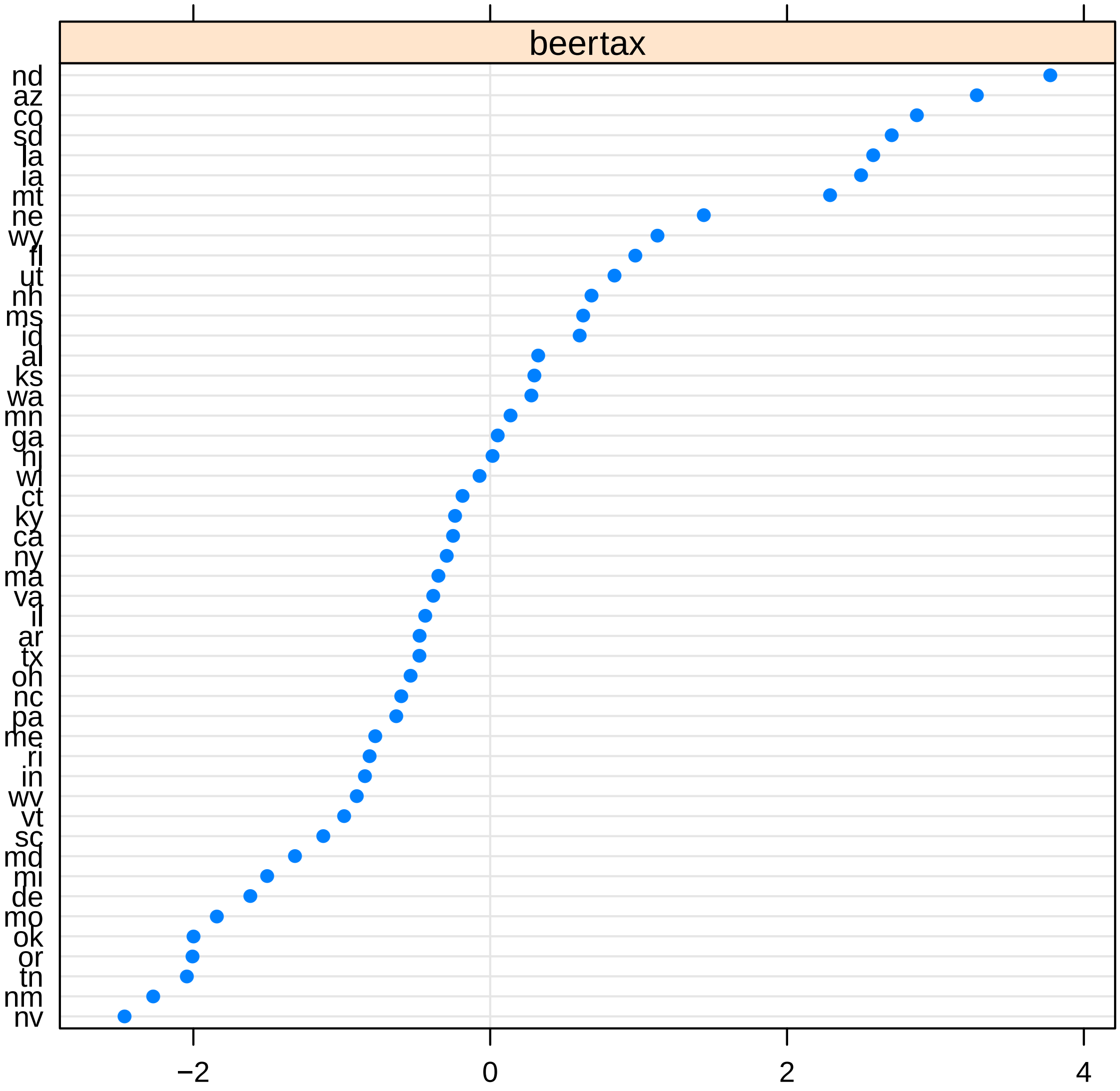
```
Linear mixed model fit by REML
Formula: y ~ -1 + beertax + (0 + beertax | state)
   Data: tmpData
    AIC    BIC logLik deviance REMLdev
 -235.8 -224.4  120.9   -241.6  -241.8
Random effects:
 Groups   Name    Variance Std.Dev.
 state    beertax 4.705741 2.16927
 Residual         0.024873 0.15771
Number of obs: 336, groups: state, 48

Fixed effects:
        Estimate Std. Error t value
beertax  -0.8754     0.4506  -1.943
```

Note that we recover a negative average effect of the beer tax on traffic fatalities, but substantial between-state heterogeneity around the estimated averge effect:

R Code

```
> print(dotplot(ranef(randomSlopes)))
```

```
$state
```



References

Hausman, J. 1978. “Specification Tests in Econometrics.” *Econometrica* 46:1251--1271.

Hsiao, Cheng. 2003. *Analysis of Panel Data*. Second ed. New York: Cambridge University Press.

Rao, C. Radhakrishna. 1973. *Linear Statistical Inference and Its Applications*. Second ed. New York: Wiley.

Ruhm, Christopher J. 1996. “Alcohol policies and highway vehicle fatalities.” *Journal of Health Economics* 15:435--454.