ARE213 Problem Set #2A

Peter Alstone & Frank Proulx

November 6, 2013

1 Problem #1 and #2

These are hand written and attached separately.

2 Problem #3

2.1 Part A

Running pooled bivariate OLS, adding a quadratic time trend, and adding the covariates that we expect to belong produces the models shown in Table 1. The pooled OLS is an ill-informed baseline model but nonetheless tells us that there is a statistically significant negative correlation between the states with primary seatbelt laws and those without. In particular, the mean of the log fatalities per capita is reduced by 0.14 for states with primary seatbelt laws. This does not, however, account for the year to year trends that are captured by including a set of quadratic year terms in the regression as shown in the "quadratic time" regression. This basic trend is that deaths have gone down over time. Correcting for these trends may allow our OLS estimates approach the true ATE, and indeed the apparent effect of primary seatbelt laws is reduced. When further covariates are added that are relevant (see Table 1) further reductions in the apparent effect occur and the effect is no longer statistically significant.

2.2 Part B

No, the standard errors are most likely not correct (are they ever really, truly correct outside a purely theoretical framework?). In this case the error

Table 1: Pooled Models of Fatalities Per Capita

	logfatalpc		
	bivariate	quadratic time	covariates
	(1)	(2)	(3)
primary	-0.144^{***}	-0.075^{***}	0.015
	(0.026)	(0.026)	(0.027)
year		-0.025^{***}	-0.042^{***}
		(0.006)	(0.005)
sqr.year		0.0005^{*}	0.001^{***}
		(0.0002)	(0.0002)
secondary			-0.001
			(0.022)
college			-2.902^{***}
			(0.176)
beer			0.268^{***}
			(0.031)
totalvmt			-0.00000^{***}
			(0.00000)
precip			-0.015^{**}
			(0.006)
snow32			-0.297^{***}
			(0.017)
rural_speed			0.016^{***}
			(0.002)
urban_speed			0.005^{***}
			(0.002)
Constant	-1.703^{***}	-1.506^{***}	-2.153^{***}
	(0.011)	(0.031)	(0.143)
N	1,127	1,127	1,127
R^2	0.027	0.094	0.596
Adjusted R^2	0.027	0.094	0.590
F Statistic	31.007^{***} (df = 1; 1125)	39.030^{***} (df = 3; 1123)	149.510^{***} (df = 11; 1115)

Notes: ***Significant at the 1 percent level.
**Significant at the 5 percent level.
*Significant at the 10 percent level.

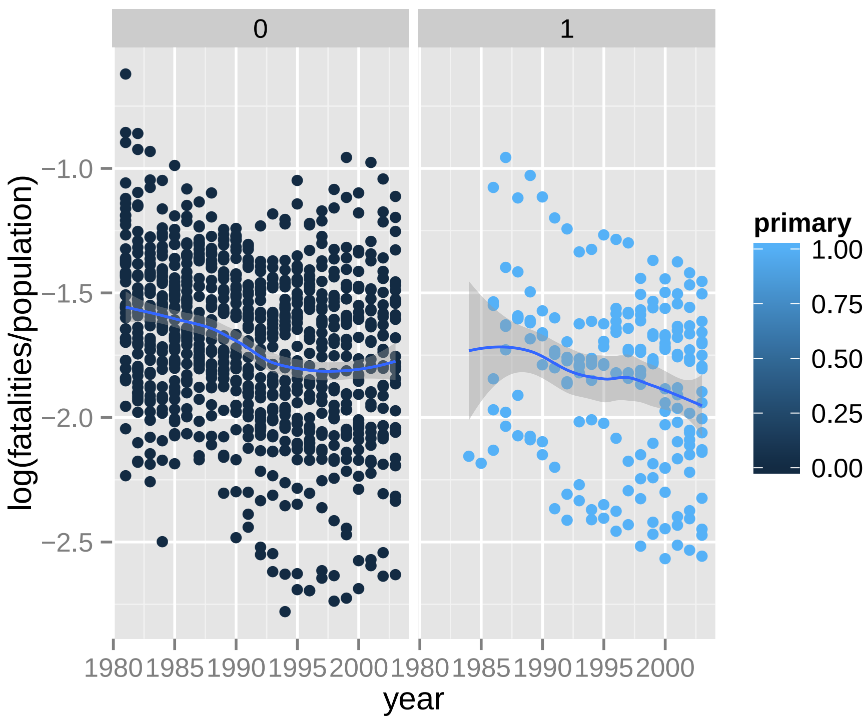

Figure 1: Year to year trends in the log of traffic fatalities per capita, divided by primary seatbelt law presence. A LOESS fit to each dataset is included for reference but is not necessarily indicative of the true underlying function.

in the SE estimates comes from heteroscedasticity in the error terms from the regression. Table 2 shows that introducing robust standard errors (using the HC1 estimator for the OLS covariance matrix). Clustering on the state grouping only increases the SE slightly over the robust pooled version, indicating most of the heteroscedasticity is in the full sample and not within-group. Table 3 shows the results of calculating the standard errors by hand, using the code shown in the code listings section at the end (with a call out to that section of code).

2.3 Part C

The between estimator will give an unbiased estimate of the effect of primary seat belt laws insofar as variation within states (across time) is uncorrelated with the observables.

We don't think that this criterion is met here. For example, within a given state, the total vmt per year probably tracks very closely with fatalities, as the higher VMT within a given year, the more likely there are to be fatal crashes (ceteris paribus).

The standard errors are sufficiently large in this model that we cannot rule out the null hypothesis (no effect of primary seatbelt laws) in either the simple case or the case with covariates.

2.4 Part D

The RE estimator will give an unbiased estimate so long as the within states variation is uncorrelated with observables. Again, this assumption is probably not met here. In the RE model we find (for the first time) that seat belt laws appear to be correlated with statistical significance with reduced fatalities....but do we trust these? Table **??** summarizes the results for the RE model both with and without covariates.

The Random Effects estimator has the advantage over pooled OLS that it allows for (and assumes) unobserved heterogeneity. OLS has the advantage that it is more efficient than the Random Effects estimator when said heterogeneity does not exist.

Table 2: Comparison of Standard Error HC Methods for Full Pooled Model

	Conventional	HC1 Robust	HC1 Robust + Cluster
	(1)	(2)	(3)
primary	0.015	0.015	0.015
	(0.027)	(0.054)	(0.054)
year	-0.042^{***}	-0.042^{***}	-0.042^{***}
	(0.005)	(0.008)	(0.008)
sqr.year	0.001^{***}	0.001^{***}	0.001^{***}
	(0.0002)	(0.0002)	(0.0002)
secondary	-0.001	-0.001	-0.001
	(0.022)	(0.036)	(0.036)
college	-2.902^{***}	-2.902^{***}	-2.902^{***}
	(0.176)	(0.446)	(0.448)
beer	0.268^{***}	0.268^{***}	0.268^{***}
	(0.031)	(0.082)	(0.082)
totalvmt	-0.00000^{***}	-0.00000^{***}	-0.00000^{***}
	(0.00000)	(0.00000)	(0.00000)
precip	-0.015^{**}	-0.015	-0.015
	(0.006)	(0.020)	(0.020)
snow32	-0.297^{***}	-0.297^{***}	-0.297^{***}
	(0.017)	(0.052)	(0.053)
rural_speed	0.016^{***}	0.016^{***}	0.016^{***}
	(0.002)	(0.004)	(0.004)
urban_speed	0.005^{***}	0.005	0.005
	(0.002)	(0.003)	(0.003)
Constant	-2.153^{***}	-2.153^{***}	-2.153^{***}
	(0.143)	(0.277)	(0.278)

Notes: ***Significant at the 1 percent level.
**Significant at the 5 percent level.
*Significant at the 10 percent level.

Table 3: Comparison of Standard Error HC Methods for Full Pooled Model, as calculated by hand

Estimand	Conventional	Robust	Clustered
(Intercept)	0.143092450051489	0.126275629333705	0.281056533719271
primary	0.0265477092390901	0.0242822007396392	0.0546894368905662
year	0.00545503338475101	0.00598682459203165	0.008116358583959
sqr.year	0.000190755819683876	0.000207479039052758	0.000246751758184774
secondary	0.0217508276594709	0.0209373279397145	0.0366521092326508
college	0.176245827507398	0.160526783925961	0.452606339396885
beer	0.0308798425250644	0.0281117000066882	0.0832087347114592
totalvmt	1.49368657811726e-07	1.27908631373882e-07	4.18638264362754e-07
precip	0.00596563120372605	0.00600959436593229	0.0203958411221209
snow32	0.0174085751012179	0.0181916999791943	0.0531907707309941
rural_speed	0.0021413549507371	0.00209161539190941	0.00376049997220133
urban_speed	0.00176934150104297	0.00163910252777972	0.00316920102721627

2.5 Part E

The conventional standard errors for the RE model are likely incorrect because of a built-in assumption that the errors within each individual are correlated equally over time. Because this is a long-term dataset we would prefer to relax this assumption by introducing robust clustered SE estimates. Table 6 shows that this increases the SE of the coefficient on primary seatbelt use, but only by about double. The SE is still small compared to the estimand.

2.6 Part F

The conventional and clustered standard errors differ (as they have for many other approaches) by about a factor of two in the FE method. These are compared in Table 7. The baseline standard errors (or roughly the coefficient of variation) is small for the fixed effects model because all of the unobserved individual state characteristics are wrapped up in the FE model without any between-ness muddying their influence. However, there remain heteroscedastic errors within each state from idiosyncrasies between time periods when states did and did not have seat belt laws (national-level trends, automobile manufacturing standards, etc.) that are exogenous to the states and link

Table 4: Between models of effects of primary seatbelt use laws

	Dependent variable: logfatalpc (1)	Dependent variable: logfatalpc (2)
primary	−0.071	0.114
	(0.155)	(0.165)
secondary		−0.025
		(0.159)
college		-2.603^{***}
		(0.636)
totalvmt		-0.00000^{**}
		(0.00000)
snow32		-0.266^{***}
		(0.083)
rural_speed		0.063^{***}
		(0.012)
Constant	-1.716^{***}	-4.978^{***}
	(0.052)	(0.789)
Observations	49	49
R^2	0.004	0.758
Adjusted R^2	0.004	0.650
F Statistic	0.212 (df = 1; 47)	21.898^{***} (df = 6; 42)

Note: *p<0.1; **p<0.05; ***p<0.01

Table 5: Random Effects Models

	Dependent variable:	
	logfatalpc	
	(1)	(2)
primary	-0.230^{***}	-0.138^{***}
	(0.016)	(0.015)
secondary		-0.065^{***}
		(0.010)
college		-1.420^{***}
		(0.169)
unemploy		-0.024^{***}
		(0.002)
beer		0.757^{***}
		(0.038)
totalvmt		-0.00000^{***}
		(0.00000)
precip		-0.024^{***}
		(0.006)
snow32		-0.018
		(0.014)
rural_speed		-0.006^{***}
		(0.001)
urban_speed		0.003^{***}
		(0.001)
Constant	-1.688^{***}	-1.902^{***}
	(0.044)	(0.092)
Observations	1,127	1,127
R^2	0.153	0.604
Adjusted R^2	0.153	0.598
F Statistic	203.173^{***} (df = 1; 1125)	170.214^{***} (df = 10; 1116)
Note:		*p<0.1; **p<0.05; ***p<0.01

Table 6: Comparison of Standard Error HC Methods for RE model with covariates

	Conventional SE	HC1 Robust + Cluster
	(1)	(2)
primary	-0.138^{***}	-0.138^{***}
	(0.015)	(0.029)
secondary	-0.065^{***}	-0.065^{***}
	(0.010)	(0.018)
college	-1.420^{***}	-1.420^{***}
	(0.169)	(0.296)
unemploy	-0.024^{***}	-0.024^{***}
	(0.002)	(0.003)
beer	0.757^{***}	0.757^{***}
	(0.038)	(0.068)
totalvmt	-0.00000^{***}	-0.00000^{**}
	(0.00000)	(0.00000)
precip	-0.024^{***}	-0.024^{***}
	(0.006)	(0.007)
snow32	-0.018	-0.018
	(0.014)	(0.020)
rural_speed	-0.006^{***}	-0.006^{***}
	(0.001)	(0.002)
urban_speed	0.003^{***}	0.003^{**}
	(0.001)	(0.001)
Constant	-1.902^{***}	-1.902^{***}
	(0.092)	(0.149)

Notes: ***Significant at the 1 percent level.
**Significant at the 5 percent level.
*Significant at the 10 percent level.

them. Figure 2 shows how the adoption of state-level primary seatbelt laws occurred non-uniformly through the sample.

Table 7: Comparison of Standard Error HC Methods for basic FE model

	Conventional SE	HC1 Robust + Cluster
	(1)	(2)
primary	-0.080^{***}	-0.080^{**}
	(0.015)	(0.033)
year	-0.025^{***}	-0.025^{***}
	(0.002)	(0.004)
sqr.year	0.0005^{***}	0.0005^{***}
	(0.0001)	(0.0001)

Notes: ***Significant at the 1 percent level.
**Significant at the 5 percent level.
*Significant at the 10 percent level.

2.7 Part G

The addition of covariate terms to the FE model reduces the estimate of the ATE for enacting primary seatbelt laws to about a 6% reduction in annual fatalities. With robust / clustered standard errors this is estimate maintains its significance at a 5% confidence level.

3 Code Listings

We used R to complete this assignment...which turned out to be quite challenging in this case, since R has no equivalently easy version of "robust" or "cluster(group)," but it was a learning experience. The code is below:

```
## Frank's wd
setwd("/media/frank/Data/documents/school/berkeley/fall13/are213/are213/ps2")
## Peter's wd
setwd("~/Google Drive/ERG/Classes/ARE213/are213/ps2")
```

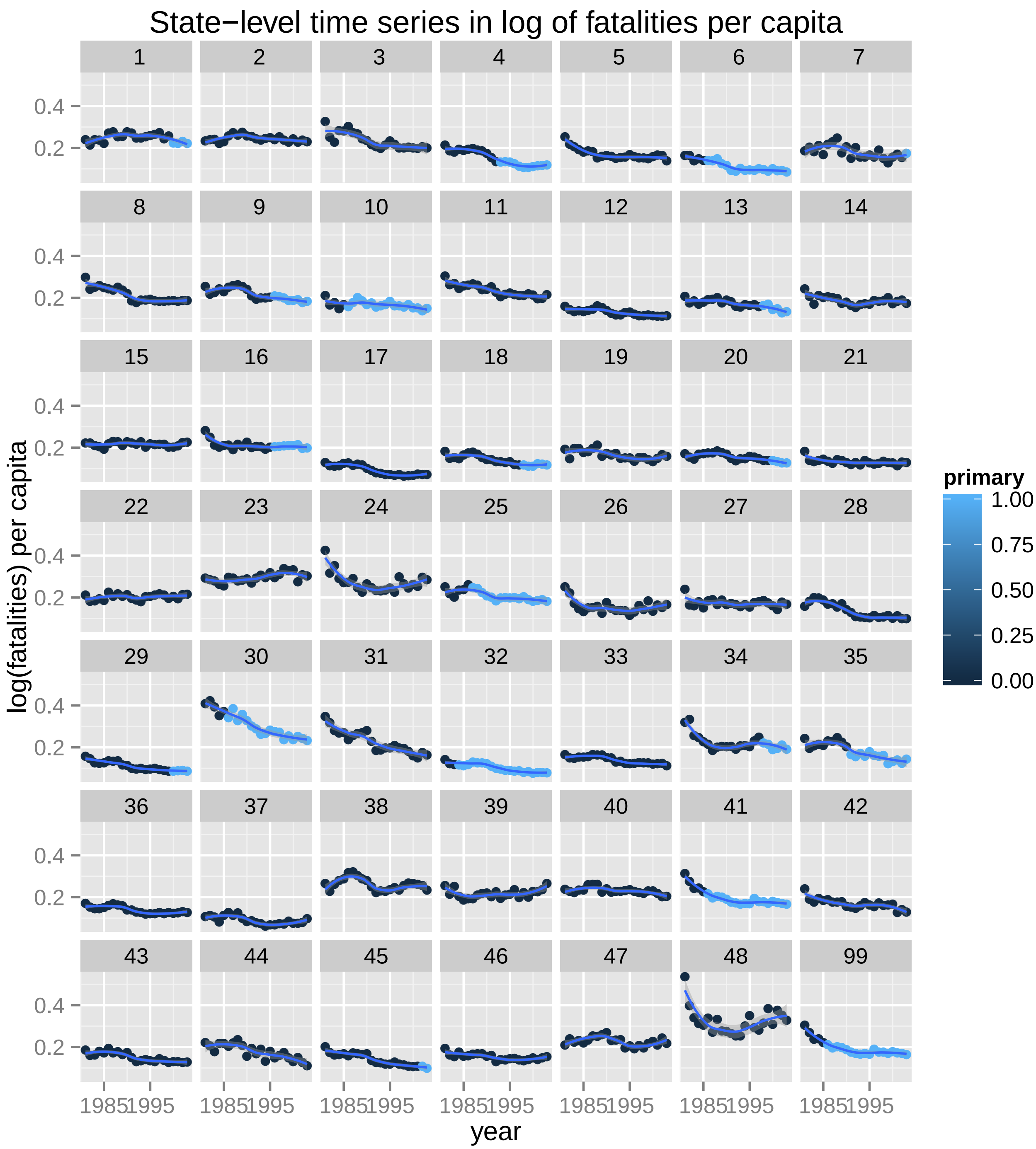

Figure 2: Year to year trends in the log of traffic fatalities per capita, with facets by state ID. A LOESS fit to each dataset is included for reference but is not necessarily indicative of the true underlying function.

Table 8: Fixed Effects Models

	Dependent variable:		
	logfatalpc		
	panel linear		*coefficient test*
	Simple FE	FE with Cov.	Robust/Clustered SE
	(1)	(2)	(3)
primary	-0.080^{***}	-0.063^{***}	-0.063^{**}
	(0.015)	(0.016)	(0.032)
year	-0.025^{***}	-0.017^{***}	-0.017^{***}
	(0.002)	(0.003)	(0.004)
sqr.year	0.0005^{***}	0.0002^{*}	0.0002
	(0.0001)	(0.0001)	(0.0002)
secondary		0.006	0.006
		(0.011)	(0.021)
college		0.377	0.377
		(0.236)	(0.462)
beer		0.788^{***}	0.788^{***}
		(0.041)	(0.079)
totalvmt		-0.00000	-0.00000
		(0.00000)	(0.00000)
precip		-0.032^{***}	-0.032^{***}
		(0.006)	(0.005)
snow32		-0.007	-0.007
		(0.014)	(0.017)
rural_speed		0.004^{***}	0.004^{***}
		(0.001)	(0.001)
urban_speed		0.001	0.001
		(0.001)	(0.001)
Observations	1,127	1,127	
R^2	0.459	0.625	
Adjusted R^2	0.438	0.591	
F Statistic	304.360^{***} (df = 3; 1075)	161.455^{***} (df = 11; 1067)	
Note:			*p<0.1; **p<0.05; ***p<0.01

```
library(foreign) #this is to read in Stata data
library(Hmisc)
library(psych)
library(stargazer)
library(ggplot2) # for neato plotting tools
library(plyr) # for nice data tools like ddply
library(car) # "companion for applied regression" - recode fxn, etc.
library(gmodels) #for Crosstabs
library(plm) # for panel data
require(lmtest)
# require(sandwich)

source("../util/are213-func.R")
#source("../util/watercolor.R") # for watercolor plots

ps2a.data <- read.dta('traffic_safety2.dta')
ps2a.datakey <- data.frame(var.name=names(ps2a.data), var.labels = attr(ps2a.data, "var.labels"))
ps2a.data$logfatalpc <- with(ps2a.data, log(fatalities/population))
ps2a.data$sqyears <- with(ps2a.data, year^2)

# Graphical exploration of data
pdf(file="allstates.pdf", width = 7, height = 7.5)
ggplot(ps2a.data, aes(year, fatalities/population)) +
  geom_point(aes(color=primary)) +
  scale_x_continuous(breaks = c(1985, 1995)) +
  scale_y_continuous(breaks = c(0.2, 0.4)) +
  facet_wrap("state") +
  ylab("log(fatalities) per capita") +
  ggtitle("State-level time series in log of fatalities per capita") +
  stat_smooth()
dev.off()

#plot 3a
makeplot3a <- FALSE
if(makeplot3a){
pdf("plot3a.pdf", width=5, height=4)
ggplot(ps2a.data, aes(year, log(fatalities/population))) + geom_point(aes(color=primary)) + facet_grid(.~primary) + stat_smooth()
dev.off()
}


#Puts data into pdata.frame for use with the plm package.
ps2a.pdata <- pdata.frame(ps2a.data, index = c("state", "year"))

## Problem 3

# Issues with non-conforming matrix seem to trace to bad factor class for year....fixed here.
ps2a.pdata$year <- as.numeric(ps2a.pdata$year)

# ADDDING square year term HERE WORKS FOR WHATEVER REASON
ps2a.pdata$sqr.year <- with(ps2a.pdata, year^2)

```

```
## Part A (pooled OLS, quadratic time trend, and all possible covariates) ------
pooled.OLS <- plm(logfatalpc ~ primary, data = ps2a.pdata, model = "pooling")

pooled.quadtime <- plm(logfatalpc ~ primary + year + sqr.year, data = ps2a.pdata, model = "pooling")

pooled.full <- plm(logfatalpc ~ primary + year + sqr.year + secondary + college + beer + totalvmt + precip + snow32 + rural_speed + urban_speed, data = ps2a.pdata, model = "pooling")

stargazer(pooled.OLS, pooled.quadtime, pooled.full,
          title = "Pooled Models of Fatalities Per Capita",
          style="qje",
          out = 'p3a.tex',
          font.size = "footnotesize",
          column.labels = c("bivariate", "quadratic time", "covariates"),
          label="tab:3a")

## Part B (Standard Error Errors) --------

#typical
a.typ.full <- coeftest(pooled.full)
#robust
a.robust.full <- coeftest(pooled.full, vcov = vcovHC)
#clustered
a.clust.full <- coeftest(pooled.full, vcov = vcovHC(pooled.full, type="HC1", cluster = "group"))

stargazer(a.typ.full, a.robust.full, a.clust.full,
          title = "Comparison of Standard Error HC Methods for Full Pooled Model",
          style = "qje",
          out = 'p3b1.tex',
          font.size = "footnotesize",
          column.labels = c("Conventional", "HC1 Robust", "HC1 Robust + Cluster"),
          label="tab:3b1"
          )


## Robust by hand
                                        #This calculates the Huber-White Robust standard errors -- code based on http://thetarzan.wordpress.com/2011/05/28/heteroskedasticity-robust-and-clustered-standard-errors-in-r/
s <- summary(pooled.full)
X <- model.matrix(pooled.full)
u2 <- residuals(pooled.full)^2
XDX <- 0

for(i in 1:nrow(X)) {
    XDX <- XDX +u2[i]*X[i,]%*%t(X[i,])
}
```

```
                                        # inverse(X'X)
XX1 <- solve(t(X)%*%X)

                                        #Compute variance/covariance matrix
varcovar <- XX1 %*% XDX %*% XX1

                                        # Degrees of freedom adjustment
dfc <- sqrt(nrow(X))/sqrt(nrow(X)-ncol(X))

stdh <- dfc*sqrt(diag(varcovar))

t <- pooled.full$coefficients/stdh
p <- 2*pnorm(-abs(t))
results.robust <- cbind(pooled.full$coefficients, stdh, t, p)
dimnames(results.robust) <- dimnames(s$coefficients)
results.robust

## cluster by hand -- using many of the same variables as defined in the robust section (above), with some modifications:
cluster <- "state"
clus <- cbind(X, "state"=ps2a.data[,cluster], "resid" = resid(pooled.full))
                                        #number of clusters
m <- dim(table(clus[,cluster]))
k <- dim(X)[2]

uj <- matrix(NA, nrow=m, ncol = k)
gs <- names(table(ps2a.data[,cluster]))
for (i in 1:m){
    uj[i,] <- t(matrix(clus[clus[,cluster]==gs[i], 'resid'])) %*% clus[clus[,cluster]==gs[i], 1:k]
}


                                        #Compute variance/covariance matrix
varcovar <- XX1 %*% crossprod(uj) %*% XX1

                                        # Degrees of freedom adjustment
dfc <- sqrt((m/(m-1)) * (nrow(X)-1)/(nrow(X)-ncol(X)))

stdh <- dfc*sqrt(diag(varcovar))

t <- pooled.full$coefficients/stdh
p <- 2*pnorm(-abs(t))
results.cluster <- cbind(pooled.full$coefficients, stdh, t, p)
dimnames(results.cluster) <- dimnames(s$coefficients)
results.cluster

hand.comparison <- cbind(rownames(results.cluster), a.typ.full[,2], results.robust[,2], results.cluster[,2])
colnames(hand.comparison) <- c("Estimand","Conventional", "Robust", "Clustered")
stargazer(data.frame(hand.comparison),
          summary = FALSE,
          title = "Comparison of Standard Error HC Methods for Full Pooled Model, as calculated by hand",
          style = "qje",
```

```
          out = 'p3b2.tex',
          font.size = "footnotesize",
          column.labels = c("Conventional", "HC1 Robust", "HC1 Robust + Cluster"),
          label="tab:3b2"
          )


## Part C: compute between estimator w/ and w/o covariates-----
between.nocov <- plm(logfatalpc ~ primary, data = ps2a.pdata, model = "between")
between.cov <- plm(logfatalpc ~ primary + secondary + college + totalvmt + snow32 + rural_speed, data = ps2a.pdata, model = "between")

stargazer(between.nocov, between.cov, title = "Between models of effects of primary seatbelt use laws", out = "p3c.tex", font.size = "footnotesize")

## Part D
RE.nocov <- plm(logfatalpc ~ primary, data = ps2a.pdata, model = "random") # Assumes effects are uncorrelated
RE.cov <- plm(logfatalpc ~ primary + secondary + college + unemploy + beer + totalvmt + precip + snow32 + rural_speed + urban_speed, data = ps2a.pdata, model = "random")

stargazer(RE.nocov, RE.cov, title = "Random Effects Models", out = "p3d.tex", font.size = "footnotesize")

## Part E compute cluster standard errors for RE

e.typ.recov <- coeftest(RE.cov)
e.clust.recov <- coeftest(RE.cov, vcov = vcovHC(RE.cov, type = "HC1", cluster = "group"))

stargazer(e.typ.recov, e.clust.recov,
          title = "Comparison of Standard Error HC Methods for RE model with covariates",
          style = "qje",
          out = 'p3e.tex',
          font.size = "footnotesize",
          column.labels = c("Conventional SE", "HC1 Robust + Cluster"),
          label="tab:p3e"
)



## Part F compute FE estimator
fixed.primary <- plm(logfatalpc ~ primary + year + sqr.year, data = ps2a.pdata, model = "within")

f.typ.fe <- coeftest(fixed.primary)
f.clust.fe <- coeftest(fixed.primary, vcov = vcovHC(fixed.primary, type='HC1', cluster = "group"))

stargazer(f.typ.fe, f.clust.fe,
          title = "Comparison of Standard Error HC Methods for basic FE model",
          style = "qje",
```

```
           out = 'p3f.tex',
           font.size = "footnotesize",
           column.labels = c("Conventional SE", "HC1 Robust + Cluster"),
           label="tab:p3f"
)


## Part G
fixed.cov <- plm(logfatalpc ~ primary + year + sqr.year + secondary + college + beer + totalvmt + precip + snow32 + rural_speed + urban_speed, data = ps2a.pdata, model = "within")

stargazer(fixed.primary, fixed.cov,
           coeftest(fixed.cov, vcov = vcovHC(fixed.cov, type="HC1", cluster = "group")),
           title = "Fixed Effects Models",
           out = "p3g.tex", font.size = "footnotesize",
           label = "tab:p3g",
           column.labels = c("Simple FE", "FE with Cov.", "Robust/Clustered SE")
           )
```

ps2a–paworking.R

3.1 Hand-coded HC Robust Clustered Standard Errors

```
## Robust by hand
#This calculates the Huber-White Robust standard errors -- code based on http://thetarzan.wordpress.com/2011/05/28/heteroskedasticity-robust-and-clustered-standard-errors-in-r/
s <- summary(pooled.full)
X <- model.matrix(pooled.full)
u2 <- residuals(pooled.full)^2
XDX <- 0

for(i in 1:nrow(X)) {
    XDX <- XDX +u2[i]*X[i,]%*%t(X[i,])
}

# inverse(X'X)
XX1 <- solve(t(X)%*%X)

#Compute variance/covariance matrix
varcovar <- XX1 %*% XDX %*% XX1

# Degrees of freedom adjustment
dfc <- sqrt(nrow(X))/sqrt(nrow(X)-ncol(X))

stdh <- dfc*sqrt(diag(varcovar))

t <- pooled.full$coefficients/stdh
p <- 2*pnorm(-abs(t))
```

```
results.robust <- cbind(pooled.full$coefficients, stdh, t, p)
dimnames(results.robust) <- dimnames(s$coefficients)
results.robust

## cluster by hand -- using many of the same variables as defined in the robust section (above), with some modifications:
cluster <- "state"
clus <- cbind(X, "state"=ps2a.data[,cluster], "resid" = resid(pooled.full))
                                        #number of clusters
m <- dim(table(clus[,cluster]))
k <- dim(X)[2]

uj <- matrix(NA, nrow=m, ncol = k)
gs <- names(table(ps2a.data[,cluster]))
for (i in 1:m){
    uj[i,] <- t(matrix(clus[clus[,cluster]==gs[i], 'resid'])) %*% clus[clus[,cluster]==gs[i], 1:k]
}


#Compute variance/covariance matrix
varcovar <- XX1 %*% crossprod(uj) %*% XX1

# Degrees of freedom adjustment
dfc <- sqrt((m/(m-1)) * (nrow(X)-1)/(nrow(X)-ncol(X)))

stdh <- dfc*sqrt(diag(varcovar))

t <- pooled.full$coefficients/stdh
p <- 2*pnorm(-abs(t))
results.cluster <- cbind(pooled.full$coefficients, stdh, t, p)
dimnames(results.cluster) <- dimnames(s$coefficients)
results.cluster

hand.comparison <- cbind(a.typ.full[,2], results.robust[,2], results.cluster[,2])
colnames(hand.comparison) <- c("Conventional", "Robust", "Clustered")
stargazer(data.frame(hand.comparison),
          summary = FALSE,
          title = "Comparison of Standard Error HC Methods for Full Pooled Model, as calculated by hand",
          style = "qje",
          out = 'p3b2.tex',
          font.size = "footnotesize",
          column.labels = c("Conventional", "HC1 Robust", "HC1 Robust + Cluster"),
          label="tab:3b2"
          )
```

```
# Econometrics helper functions for [R]
#
# Peter Alstone and Frank Proulx
# 2013
# version 1
# contact: peter.alstone AT gmail.com
```

```
# Category: Data Management -------------


# Category: Data Analysis ----------------

# Function: Find adjusted R^2 for subset of data
# This requires a completed linear model...pull out the relevant y-values
    and residuals and feed them to function
# [TODO @Peter] Improve function so it can simply evaluate lm or glm object,
     add error handling, general clean up.
adjr2 <- function(y,resid){
  r2 <- 1-sum(resid^2) / sum((y-mean(y))^2)
  return(r2)
} #end adjr2


# Category: Plots and Graphics -------------

## Function for arranging ggplots. use png(); arrange(p1, p2, ncol=1); dev.
    off() to save.
require(grid)
vp.layout <- function(x, y) viewport(layout.pos.row=x, layout.pos.col=y)
arrange_ggplot2 <- function(..., nrow=NULL, ncol=NULL, as.table=FALSE) {
  dots <- list(...)
  n <- length(dots)
  if(is.null(nrow) & is.null(ncol)) { nrow = floor(n/2) ; ncol = ceiling(n/
      nrow)}
  if(is.null(nrow)) { nrow = ceiling(n/ncol)}
  if(is.null(ncol)) { ncol = ceiling(n/nrow)}
  ## NOTE see n2mfrow in grDevices for possible alternative
  grid.newpage()
  pushViewport(viewport(layout=grid.layout(nrow,ncol) ) )
  ii.p <- 1
  for(ii.row in seq(1, nrow)){
    ii.table.row <- ii.row
    if(as.table) {ii.table.row <- nrow - ii.table.row + 1}
    for(ii.col in seq(1, ncol)){
      ii.table <- ii.p
      if(ii.p > n) break
      print(dots[[ii.table]], vp=vp.layout(ii.table.row, ii.col))
      ii.p <- ii.p + 1
    }
  }
}

robust <- function(model){ #This calculates the Huber-White Robust standard
    errors -- code from http://thetarzan.wordpress.com/2011/05/28/
    heteroskedasticity-robust-and-clustered-standard-errors-in-r/
    s <- summary(model)
    X <- model.matrix(model)
    u2 <- residuals(model)^2
    XDX <- 0

    for(i in 1:nrow(X)) {
        XDX <- XDX +u2[i]*X[i,]%*%t(X[i,])
    }
```

```

# inverse(X'X)
    XX1 <- solve(t(X)%*%X)

#Compute variance/covariance matrix
    varcovar <- XX1 %*% XDX %*% XX1

# Degrees of freedom adjustment
    dfc <- sqrt(nrow(X))/sqrt(nrow(X)-ncol(X))

    stdh <- dfc*sqrt(diag(varcovar))

    t <- model$coefficients/stdh
    p <- 2*pnorm(-abs(t))
    results <- cbind(model$coefficients, stdh, t, p)
    dimnames(results) <- dimnames(s$coefficients)
    results
}

## Two functions for clustered standard errors below from: http://people.su.se/~ma/clustering.pdf -------

clx <-
  function(fm, dfcw, cluster){
    # R-codes (www.r-project.org) for computing
    # clustered-standard errors. Mahmood Arai, Jan 26, 2008.

    # The arguments of the function are:
    # fitted model, cluster1 and cluster2
    # You need to install libraries 'sandwich' and 'lmtest'

    # reweighting the var-cov matrix for the within model
    library(sandwich);library(lmtest)
    M <- length(unique(cluster))
    N <- length(cluster)
    K <- fm$rank
    dfc <- (M/(M-1))*((N-1)/(N-K))
    uj  <- apply(estfun(fm),2, function(x) tapply(x, cluster, sum));
    vcovCL <- dfc*sandwich(fm, meat=crossprod(uj)/N)*dfcw
    coeftest(fm, vcovCL) }

mclx <-
  function(fm, dfcw, cluster1, cluster2){
    # R-codes (www.r-project.org) for computing multi-way
    # clustered-standard errors. Mahmood Arai, Jan 26, 2008.
    # See: Thompson (2006), Cameron, Gelbach and Miller (2006)
    # and Petersen (2006).
    # reweighting the var-cov matrix for the within model

    # The arguments of the function are:
    # fitted model, cluster1 and cluster2
    # You need to install libraries 'sandwich' and 'lmtest'

    library(sandwich);library(lmtest)
    cluster12 = paste(cluster1,cluster2, sep="")
    M1  <- length(unique(cluster1))
    M2  <- length(unique(cluster2))
```

```r
    M12 <- length(unique(cluster12))
    N   <- length(cluster1)
    K   <- fm$rank
    dfc1  <- (M1/(M1-1))*((N-1)/(N-K))
    dfc2  <- (M2/(M2-1))*((N-1)/(N-K))
    dfc12 <- (M12/(M12-1))*((N-1)/(N-K))
    u1j   <- apply(estfun(fm), 2, function(x) tapply(x, cluster1,  sum))
    u2j   <- apply(estfun(fm), 2, function(x) tapply(x, cluster2,  sum))
    u12j  <- apply(estfun(fm), 2, function(x) tapply(x, cluster12, sum))
    vc1   <-  dfc1*sandwich(fm, meat=crossprod(u1j)/N )
    vc2   <-  dfc2*sandwich(fm, meat=crossprod(u2j)/N )
    vc12  <- dfc12*sandwich(fm, meat=crossprod(u12j)/N)
    vcovMCL <- (vc1 + vc2 - vc12)*dfcw
    coeftest(fm, vcovMCL)}

## Function to compute ols standard errors , robust, clustered...
## Based on http://diffuseprior.wordpress.com/2012/06/15/standard-robust-and-clustered-standard-errors-computed-in-r/
ols.hetero <- function(form, data, robust=FALSE, cluster=NULL,digits=3){
  r1 <- lm(form, data)
  if(length(cluster)!=0){
    data <- na.omit(data[,c(colnames(r1$model),cluster)])
    r1 <- lm(form, data)
  }
  X <- model.matrix(r1)
  n <- dim(X)[1]
  k <- dim(X)[2]
  if(robust==FALSE & length(cluster)==0){
    se <- sqrt(diag(solve(crossprod(X)) * as.numeric(crossprod(resid(r1))/(n-k))))
    res <- cbind(coef(r1),se)
  }
  if(robust==TRUE){
    u <- matrix(resid(r1))
    meat1 <- t(X) %*% diag(diag(crossprod(t(u)))) %*% X
    dfc <- n/(n-k)
    se <- sqrt(dfc*diag(solve(crossprod(X)) %*% meat1 %*% solve(crossprod(X))))
    res <- cbind(coef(r1),se)
  }
  if(length(cluster)!=0){
    clus <- cbind(X,data[,cluster],resid(r1))
    colnames(clus)[(dim(clus)[2]-1):dim(clus)[2]] <- c(cluster,"resid")
    m <- dim(table(clus[,cluster]))
    dfc <- (m/(m-1))*((n-1)/(n-k))
    uclust  <- apply(resid(r1)*X,2, function(x) tapply(x, clus[,cluster], sum))
    se <- sqrt(diag(solve(crossprod(X)) %*% (t(uclust) %*% uclust) %*% solve(crossprod(X)))*dfc)
    res <- cbind(coef(r1),se)
  }
  res <- cbind(res,res[,1]/res[,2],(1-pnorm(abs(res[,1]/res[,2])))*2)
  res1 <- matrix(as.numeric(sprintf(paste("%.",paste(digits,"f",sep=""),sep=""),res)),nrow=dim(res)[1])
  rownames(res1) <- rownames(res)
  colnames(res1) <- c("Estimate","Std. Error","t value","Pr(>|t|)")
  return(res1)
```

```
}
```

../util/are213–func.R