I benefited from discussions with Thomas Bearpark and Eric Qian. All errors are my own.

Question 1

Following Meng, Qian and Yared (2015) we estimate the regression of

$$y_{it} = X_{it} \times D_{it}\beta_1 + X_{it} \times (1 - D_{it})\beta_2 + \mathbf{W}_{it}'\gamma + \alpha_t + \varepsilon_{it}, \tag{1}$$

where y_{it} is the log-mortality rate on log-grain production in famine and non-famine periods, year fixed effects, and a vector of covariates containing log of total and urban population. In the regression above β_1 and β_2 can be interpreted as the elasticities of the mortality rate to grain production in famine and non-famine periods, respectively. Table 1 reports the point estimates together with confidence intervals computed using various techniques.

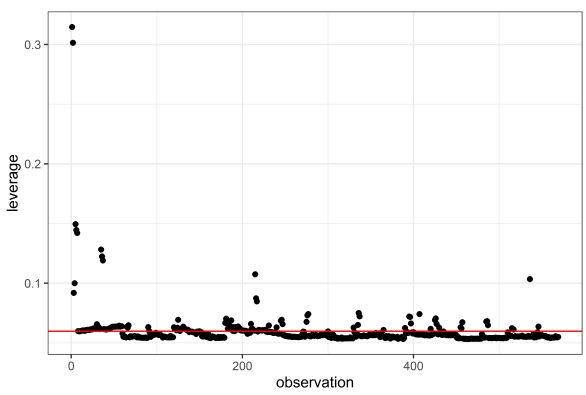
Table 1: *Point estimates of β_1 and β_2 with 95% confidence intervals.*

		Food Production famine		*Food Production non-famine*	
Point Estimate:		0.141		-0.007	
95% Confidence Interval:					
Heteroskedasticity-Robust					
	HC1	0.057	0.225	-0.032	0.019
	HC2	0.056	0.227	-0.034	0.021
	Satt	0.052	0.231	-0.040	0.026
	bootstrap (np)	0.051	0.232	-0.039	0.025
	bootstrap (pct)	0.062	0.241	-0.031	0.024
Cluster-Robust					
	CR1	0.024	0.258	-0.059	0.046
	CR2	0.023	0.260	-0.066	0.052
	Satt	-0.006	0.288	-0.113	0.099
	bootstrap (np)	-0.004	0.286	-0.100	0.086
	bootstrap (pct)	0.026	0.231	-0.056	0.055

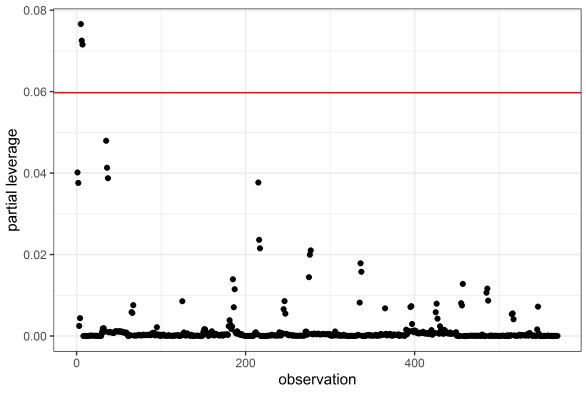
Notes: HC1 uses the Eicker-Huber-White (EHW) standard errors, *HC2* uses the EHW standard errors with the MacKinnon and White (1985) correction for high-leverage observations; *Satt* uses the HC2 standard errors with the Satterthwaite (1946) degrees of freedom approximation; *bootstrap (np)* uses the standard errors obtained estimating $\widehat{\beta}_j, j = 1, 2$ in 50.000 draws of a non-parametric bootstrap; *bootstrap (pct)* constructs confidence interval by computing the robust-t statistic (with HC1 standard errors) in 50.000 draws of a percentile bootstrap; *CR1* uses the Liang and Zeger (1986) cluster-robust standard errors; *CR2* uses the Bell and McCaffrey (2002) cluster-robust standard errors with the correction for high-leverage clusters. The last two rows rely on a block bootstrap where provinces have been sampled instead of sampling the single unit of observation.

Overall, clustering increases the length of the estimated confidence intervals, whereas correcting for the presence of outliers does not affect the results. Indeed, leverage and partial leverage are low for almost all units across various specifications (Figure 3). This does not change our conclusion regarding the effect of food production on mortality in non-famine years, as we always fail to reject

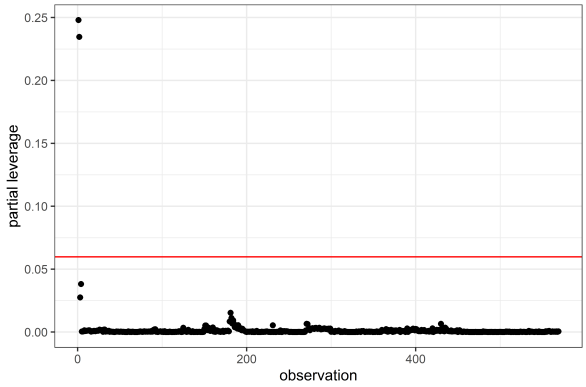
Figure 1: *Various measures of leverage.*



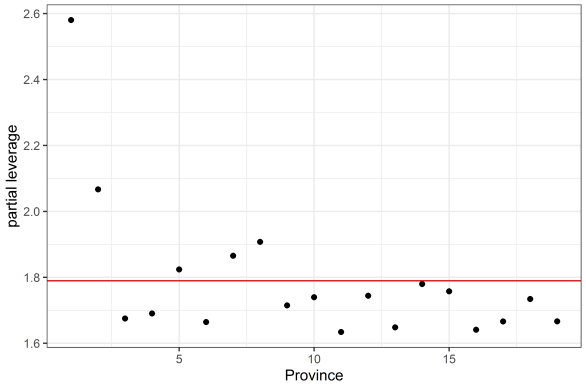
(i) *Unit leverage* H.



(ii) *Partial leverage* $\widetilde{H}_{X\times D}$.



(iii) *Partial leverage* $\widetilde{H}_{X\times(1-D)}$.



(iv) *Partial Cluster-leverage* $\widetilde{H}_{g,X\times D}$.

Notes: Leverage is computed as $H = X\left(X'X\right)^{-1}X'$, where X is the design matrix used in the regression . The partial leverage of the variable Z is defined as $\widetilde{H}_Z = \widetilde{Z}\left(\widetilde{Z}'\widetilde{Z}\right)^{-1}\widetilde{Z}'$, where $\widetilde{Z}$ is the residual of the projection of Z on all the other columns of X. We follow MacKinnon et al. (2022) and compute the cluster leverage as the nuclear norm of H_g. The horizontal solid red line indicates the ideal benchmark of balanced leverage.

the null of a non-zero effect. On the contrary, when it comes to assess whether food production had a statistically significant effect on mortality in famine years our answer is less clear cut. Controlling for correlated shocks at the province level makes the estimate of β_1 not statistically significant with 2 of the 5 different standard errors we use. It also seems that clustering standard errors is the right thing to do, as provinces are likely to be serially correlated and we only observe a subset of the superpopulation of Chinese provinces.

Question 2 - Critical Reading, Kean and Neal (2021)

Summary. The paper discusses about the finite sample properties of the two-stage least squares estimator (TSLS) and addresses the issue of inference in such context.

First, the authors survey the most popular results and suggestions in the literature on weak instrumental variables and then shows how conventional t-tests lead to misleading inference even when instruments are deemed as "strong" according to previous research. Indeed, correct nominal size of 5% is restored at values higher than 100 (shown in Lee, McCrary, Moreira and Porter 2021). This is mainly attributed to the well known fact that the finite sample distribution of TSLS is asymmetric and with fat tails.

Second, the authors focus on the fact that the asymmetry in the distribution has important consequences when it comes to address different null hypotheses (e.g. $H_0 : \beta < 0$ or $H_0 : \beta > 0$). Towards this goal, it is shown that the TSLS estimator has artificially low standard errors precisely when its point estimates are biased towards OLS. This implies that, depending on the direction of the bias, conventional t-tests are extremely low-powered against either positive or negative alternatives.

Third, it is shown that one-tailed tests have much greater size distortion than two-tailed tests. This happens because of the asymmetry of the TSLS distribution. To restore the traditional symmetry between two-tailed and one-tailed tests, independently of the degree of endogeneity, the population F-statistic must be in the thousands. The classical Anderson-Rubin (AR) test achieves such balance for a vastly smaller F.

Finally, the conditional t-test approach is shown to have both correct overall 5% rejection rate (same as AR) and maintain this property also when it comes to one-sided hypothesis tests. Indeed, such approach adjusts the critical values to take the non-normality and asymmetry of the TSLS distribution into account. However, since the AR is the UMP test in the class of two-tailed tests with symmetric critical values, there must be a trade-off. In particular, the authors suggest to use the AR when the true effect goes in the opposite direction to the OLS bias, and the conditional approach in the other case.

"First-stage F in the thousands". The critique that the authors make is relevant only if the degree of endogeneity is extremely high and a researcher is interested in a one-sided hypothesis test. In a sense, it embraces the attitude of creating prescriptions to practitioners that are valid even in the *worst case scenario* (Staiger and Stock 1997, Stock and Yogo 2005, Lee, McCrary, Moreira and Porter 2021). However, despite being an interesting theoretical point, this critique has little practical relevance in economic applications or whenever a researcher has knowledge about the magnitude/sign of β. Furthermore, in the just-identified IV case there exists a one-to-one mapping between β and the degree of endogeneity ρ. Hence, by restricting the value of β on a specific interval, it is also possible to bound the amount of possible endogeneity. Taking these facts into consideration sensibly reduces the range of applications to which the critique can be applied.

Takeaway. Robust estimators and testing procedures are usually appealing because they save the researcher from justifying why an empirical question has been answered in a certain manner. However, robustness comes at the expenses of efficiency. Hence, if a researcher possesses relevant information about the problem at hand, this should be used to alleviate such trade-off. In the last twenty years, the weak-IV literature has taken a conservative approach and suggested testing procedures that controlled size distortion even under the worst-case scenario. Even though these suggestions have wide applicability, they should not be naively implemented when additional structure can be imposed on the problem at hand, making the worst-case scenario less worse. In

this spirit, Angrist and Kolesár (2022) show that in the economic applications that they consider, $|\rho| < 0.5$ (Table 1). They also show that when $|| < 0.565$ the nominal 5% t-test under-rejects for any population value of F (see also Figure 2). At least in the applications considered in Angrist and Kolesár (2022), the advice given in Kean and Neal (2021) has a little role to play.

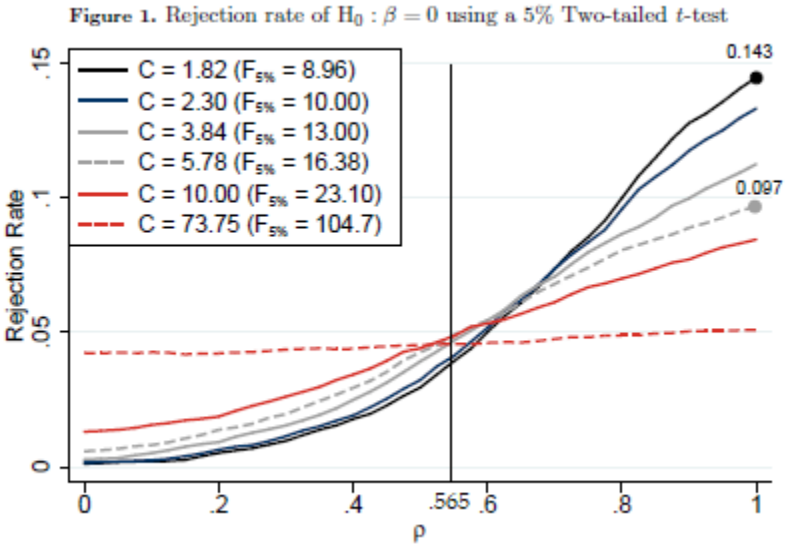


Figure 2: *Source: Kean and Neal (2021).*

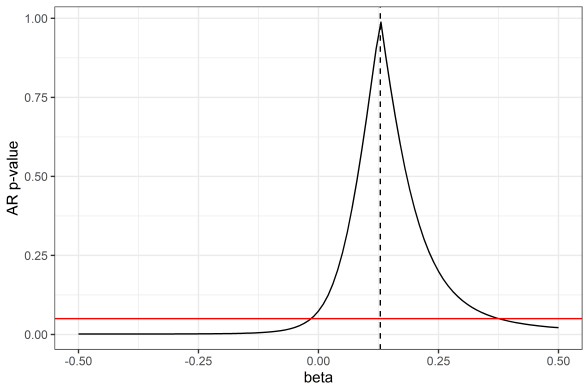
Question 3

Table 2: *Estimated returns to schooling in different sub-samples.*

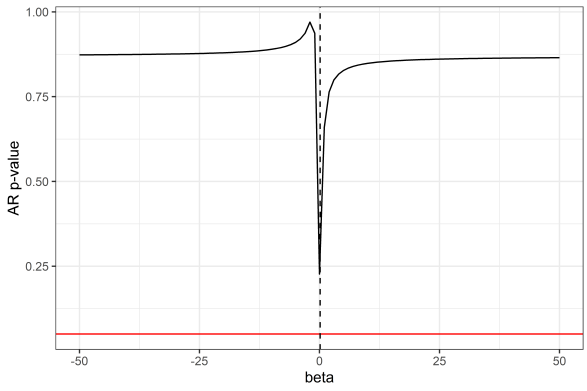
	Pacific	*mid-Atlantic*
Point Estimate:		
	0.13	-1.52
Confidence Interval:		
Wald	$[-0.02; 0.27]$	$[-20.67; 17.62]$
AR	$[-0.02; 0.38]$	$[-\infty; \infty]$
tF	$[-0.14; 0.39]$	$[-\infty; \infty]$
First-stage F	9.29	0.03

Table 2 reports the point estimate for returns of schooling in different sub-samples of the original Angrist and Krueger (1991) dataset. We can see that in the Pacific sub-sample, the first-stage F is around 9, suggesting that the traditional Wald confidence interval might be undercovering. Both the Anderson-Rubin and the tF procedure yield larger confidence interval. In the mid-Atlantic case, the instrument we are using is extremely weak and has a first-stage F that is almost 0. This induces the IV estimator to be extremely noisy (returns to schooling are indeed very negative) and the estimand of interest to be ultimately unidentifiable. Conventional Wald confidence interval ignore that in a region of the parameter space the model is weakly identified, thus they continue to be a bounded interval. However, the possibility of obtaining an infinite confidence set is a necessary condition for having a procedure robust to weak instruments (Dufour 1997). If instruments are weak, then the data contain little information about the coefficient of interest, resulting in infinite confidence sets. Both the Anderson-Rubin and the tF procedure have correct nominal coverage independently of the first-stage F, thus their expected length is infinite. In particular, this happens because a necessary and sufficient condition for these intervals to be bounded is the population F being such that $F < q_{1-\alpha}$.

Figure 3: *Anderson-Rubin confidence intervals in different sub-samples.*



(i) *Pacific division sub-sample.*



(ii) *mid-Atlantic division sub-sample.*

References

Angrist, J. D. and Krueger, A. B. (1991), 'Does compulsory school attendance affect schooling and earnings?', *The Quarterly Journal of Economics* **106**(4), 979–1014.

Angrist, J. and Kolesár, M. (2022), 'One instrument to rule them all: The bias and coverage of just-id iv'.

Bell, R. M. and McCaffrey, D. F. (2002), 'Bias reduction in standard errors for linear regression with multi-stage samples', *Survey Methodology* **28**(2), 169–182.

Dufour, J.-M. (1997), 'Some impossibility theorems in econometrics with applications to structural and dynamic models', *Econometrica: Journal of the Econometric Society* pp. 1365–1387.

Kean, M. and Neal, T. (2021), A practical guide to weak instruments, Working paper, SSRN.
URL: https://doi.org/10.2139/ssrn.3846841

Lee, D. S., McCrary, J., Moreira, M. J. and Porter, J. R. (2021), Valid t-ratio inference for iv, Technical report, National Bureau of Economic Research.

Liang, K.-Y. and Zeger, S. L. (1986), 'Longitudinal data analysis using generalized linear models', *Biometrika* **73**(1), 13–22.

MacKinnon, J. G., Nielsen, M. Ã., Webb, M. D. et al. (2022), 'Leverage, influence, and the jackknife in clustered regression models: Reliable inference using summclust'.

MacKinnon, J. G. and White, H. (1985), 'Some heteroskedasticity-consistent covariance matrix estimators with improved finite sample properties', *Journal of econometrics* **29**(3), 305–325.

Meng, X., Qian, N. and Yared, P. (2015), 'The institutional causes of china's great famine, 1959–1961', *The Review of Economic Studies* **82**(4), 1568–1611.

Satterthwaite, F. E. (1946), 'An approximate distribution of estimates of variance components', *Biometrics bulletin* **2**(6), 110–114.

Staiger, D. and Stock, J. (1997), 'Instrumental variables regression with weak instruments', *Econometrica* **65**, 557–586.

Stock, J. and Yogo, M. (2005), *Asymptotic distributions of instrumental variables statistics with many instruments*, Vol. 6, Chapter.

Question 1 Code

```
setwd("D:/Dropbox/Universit/PhD/II Year/Spring/ECO519 - Non-linear Econometrics/psets/ps3")

# Load stuff
library(haven)
library(sandwich)
library(dfadjust)
library(fastDummies)
library(xtable)
library(rsample)
library(tidyverse)

theme_set(theme_bw())

data <- haven::read_dta("famine.dta")

# Generate interactions
data$x1 <- data$lgrain_pred*data$famine
data$x2 <- data$lgrain_pred*(1-data$famine)

# Run regression
reg.out <- lm(ldeaths ~ x1 + x2 + ltotpop + lurbpop + as.factor(year) - 1,
              data = data)
beta.hat <- reg.out$coefficients[1:2]

iters <- 50000  # bootstrap draws

################################################################################
## Exercise 1a - Various standard errors allowing for heteroskedasticity
################################################################################

# 1) Robust EHW variance covariance
hc1.vcov <- sandwich::vcovHC(reg.out, type = 'HC1')

# 2) Robust HC2 variance covariance
hc2.vcov <- sandwich::vcovHC(reg.out, type = 'HC2')

# 3) Satterthwaite (1946) adjustment for degrees of freedom
sw.ses <- dfadjustSE(reg.out)$coefficients[1:2,4, drop = F]

# 4-5) non-parametric and percentile bootstrap
out.npboot <- matrix(NA, nrow=iters, 2)
out.pcboot <- matrix(NA, nrow=iters, 2)

# Bootstrap units
set.seed(8894)
bs <- rsample::bootstraps(data, times=iters)

for (i in seq_len(iters)) {

  df.boot <- tibble::as_tibble(bs$splits[[i]])

  reg.boot <- lm(ldeaths ~ x1 + x2 + ltotpop + lurbpop + as.factor(year) - 1,
                 data = df.boot)
  vcov <- sandwich::vcovHC(reg.boot, type = 'HC1')[1:2,1:2]

```

```r
  out.npboot[i, ] <- reg.boot$coefficients[1:2]
  out.pcboot[i, ] <- (reg.boot$coefficients[1:2] - beta.hat)/sqrt(diag(vcov))
}

# compute standard deviation across draws (non-parametric bootstrap)
npboot.ses <- apply(out.npboot, 2, sd, na.rm=T)

# compute quantiles of robust t-statistic (percentile bootstrap)
pcboot.qtles <- apply(out.pcboot, 2,
                  function(x) quantile(x, probs = c(0.025,0.975), na.rm=T))

##############################################################################
## Construct all confidence intervals
##############################################################################
# first variable
# HC1
CI1.hc1 <- c(beta.hat[1] - qnorm(0.975)*sqrt(hc1.vcov[1,1]),
             beta.hat[1] - qnorm(0.025)*sqrt(hc1.vcov[1,1]))

# HC2
CI1.hc2 <- c(beta.hat[1] - qnorm(0.975)*sqrt(hc2.vcov[1,1]),
             beta.hat[1] - qnorm(0.025)*sqrt(hc2.vcov[1,1]))

# Satterthwaite adjustment
CI1.sw <- c(beta.hat[1] - qnorm(0.975)*sw.ses[1],
             beta.hat[1] - qnorm(0.025)*sw.ses[1])

# Non-parametric bootstrap
CI1.np <- c(beta.hat[1] - qnorm(0.975)*npboot.ses[1],
            beta.hat[1] - qnorm(0.025)*npboot.ses[1])

# Percentile bootstrap on robust t-statistic
CI1.pct <- c(beta.hat[1] - pcboot.qtles[2,1]*sqrt(hc1.vcov[1,1]),
              beta.hat[1] - pcboot.qtles[1,1]*sqrt(hc1.vcov[1,1]))

# second variable
# HC1
CI2.hc1 <- c(beta.hat[2] - qnorm(0.975)*sqrt(hc1.vcov[2,2]),
             beta.hat[2] - qnorm(0.025)*sqrt(hc1.vcov[2,2]))

# HC2
CI2.hc2 <- c(beta.hat[2] - qnorm(0.975)*sqrt(hc2.vcov[2,2]),
             beta.hat[2] - qnorm(0.025)*sqrt(hc2.vcov[2,2]))

# Satterthwaite adjustment
CI2.sw <- c(beta.hat[2] - qnorm(0.975)*sw.ses[2],
            beta.hat[2] - qnorm(0.025)*sw.ses[2])

# Non-parametric bootstrap
CI2.np <- c(beta.hat[2] - qnorm(0.975)*npboot.ses[2],
            beta.hat[2] - qnorm(0.025)*npboot.ses[2])

# Percentile bootstrap on robust t-statistic
CI2.pct <- c(beta.hat[2] - pcboot.qtles[2,1]*sqrt(hc1.vcov[2,2]),
              beta.hat[2] - pcboot.qtles[1,1]*sqrt(hc1.vcov[2,2]))

methods <- c("Coefficient","HC1", "HC2", "Satterthwaite", "non-parametric bootstrap",
             "percentile bootstrap")
cols <- c("lb", "ub","lb", "ub")
```

```r

tab1a <- rbind(CI1.hc1,CI1.hc2,CI1.sw,CI1.np,CI1.pct)
tab1b <- rbind(CI2.hc1,CI2.hc2,CI2.sw,CI2.np,CI2.pct)

tab1 <- rbind(c(rep(beta.hat[1],2),rep(beta.hat[2],2)),
              cbind(tab1a, tab1b))
colnames(tab1) <- cols
rownames(tab1) <- methods

################################################################################
## Exercise 1b - Various standard errors allowing for province clusters
################################################################################

# 1) Robust LR variance covariance
cr1.vcov <- sandwich::vcovCL(reg.out, cluster = data$prov, type = 'HC1')

# 2) Robust CR2 (Bell and McAffrey) variance covariance
cr2.ses <- dfadjustSE(reg.out, clustervar = as.factor(data$prov),
                      IK = FALSE)$coefficients[1:2, 3, drop = F]

# 3) Satterthwaite (1946) adjustment for degrees of freedom
cr2sw.ses <- dfadjustSE(reg.out, clustervar = as.factor(data$prov),
                      IK = FALSE)$coefficients[1:2, 4, drop = F]

# 4-5) non-parametric and percentile bootstrap
out.npboot.cl <- matrix(NA, nrow=iters, 2)
out.pcboot.cl <- matrix(NA, nrow=iters, 2)

# Extract id for each province
ids <- data %>% nest('ID' = -prov)

# Bootstrap provinces (not observations!!)
set.seed(8894)
bs.cl <- rsample::bootstraps(ids, times=iters)

for (i in seq_len(iters)) {

  df.boot <- as_tibble(bs.cl$splits[[i]]) %>% unnest(cols = c(ID))

  reg.boot <- lm(ldeaths ~ x1 + x2 + ltotpop + lurbpop + as.factor(year) - 1,
                 data = df.boot)

  vcov <- sandwich::vcovCL(reg.boot, cluster = df.boot$prov,
                           type = 'HC1')[1:2,1:2]

  out.npboot.cl[i, ] <- reg.boot$coefficients[1:2]
  out.pcboot.cl[i, ] <- (reg.boot$coefficients[1:2] - beta.hat)/sqrt(diag(vcov))
}

# compute standard deviation across draws (non-parametric bootstrap)
npbootcl.ses <- apply(out.npboot.cl, 2, sd, na.rm=T)

# compute quantiles of robust t-statistic (percentile bootstrap)
pcbootcl.qtles <- apply(out.pcboot.cl, 2,
                      function(x) quantile(x, probs = c(0.025,0.975), na.rm=T))

################################################################################
## Construct all confidence intervals (cluster robust)
################################################################################
```

```r
# first variable
# CR1
CI1.cr1 <- c(beta.hat[1] - qnorm(0.975)*sqrt(cr1.vcov[1,1]),
             beta.hat[1] - qnorm(0.025)*sqrt(cr1.vcov[1,1]))

# CR2 (Bell and McCaffrey)
CI1.cr2 <- c(beta.hat[1] - qnorm(0.975)*cr2.ses[1],
             beta.hat[1] - qnorm(0.025)*cr2.ses[1])

# Satterthwaite adjustment
CI1.cr2sw <- c(beta.hat[1] - qnorm(0.975)*cr2sw.ses[1],
               beta.hat[1] - qnorm(0.025)*cr2sw.ses[1])

# Non-parametric bootstrap
CI1.npcl <- c(beta.hat[1] - qnorm(0.975)*npbootcl.ses[1],
              beta.hat[1] - qnorm(0.025)*npbootcl.ses[1])

# Percentile bootstrap on robust t-statistic
CI1.pctcl <- c(beta.hat[1] - pcbootcl.qtles[2,1]*sqrt(cr1.vcov[1,1]),
               beta.hat[1] - pcbootcl.qtles[1,1]*sqrt(cr1.vcov[1,1]))

# second variable
# CR1
CI2.cr1 <- c(beta.hat[2] - qnorm(0.975)*sqrt(cr1.vcov[2,2]),
             beta.hat[2] - qnorm(0.025)*sqrt(cr1.vcov[2,2]))

# CR2
CI2.cr2 <- c(beta.hat[2] - qnorm(0.975)*cr2.ses[2],
             beta.hat[2] - qnorm(0.025)*cr2.ses[2])

# Satterthwaite adjustment
CI2.cr2sw <- c(beta.hat[2] - qnorm(0.975)*cr2sw.ses[2],
               beta.hat[2] - qnorm(0.025)*cr2sw.ses[2])

# Non-parametric bootstrap
CI2.npcl <- c(beta.hat[2] - qnorm(0.975)*npbootcl.ses[2],
              beta.hat[2] - qnorm(0.025)*npbootcl.ses[2])

# Percentile bootstrap on robust t-statistic
CI2.pctcl <- c(beta.hat[2] - pcboot.qtles[2,1]*sqrt(cr1.vcov[2,2]),
               beta.hat[2] - pcboot.qtles[1,1]*sqrt(cr1.vcov[2,2]))

methods <- c("CR1", "CR2", "Satterthwaite", "non-parametric bootstrap",
             "percentile bootstrap")
cols <- c("lb", "ub","lb", "ub")

tab2a <- rbind(CI1.cr1,CI1.cr2,CI1.cr2sw,CI1.npcl,CI1.pctcl)
tab2b <- rbind(CI2.cr1,CI2.cr2,CI2.cr2sw,CI2.npcl,CI2.pctcl)

tab2 <- cbind(tab2a, tab2b)

tab <- rbind(tab1, tab2)

xtable(as.table(tab), digits = 3)

## Compute leverages
################################################################################

# Store design matrix, outcome, and residuals
```

```r
X <- fastDummies::dummy_cols(as.factor(data$year),
                             remove_first_dummy = FALSE)[,-1]
X <- cbind(data$x1, data$x2, data$ltotpop, data$lurbpop, X)
X <- data.matrix(X)
y <- data$ldeaths
XX <- solve(t(X) %*% X)
beta.ls <- XX %*% t(X) %*% y
res <- y - X %*% beta.ls
N <- length(res)

# Compute leverage
leverage <- data.frame(lvg=stats::hatvalues(reg.out))
leverage$x <- c(1:nrow(leverage))
ggplot(leverage, aes(x=x, y=lvg)) + geom_point() +
  xlab("observation") + ylab("leverage") +
  geom_hline(yintercept = length(reg.out$coefficients)/nrow(leverage),
             color = "red")
ggsave('leverage.png', height = 4, width = 6, dpi = 1000)

# Partial leverage of X1
x1 <- X[,1]
xx1 <- X[,-1]
X1fwl <- x1 - xx1 %*% solve(t(xx1) %*% xx1) %*% t(xx1) %*% x1
lvgx1 <- diag(X1fwl %*% solve(t(X1fwl) %*% X1fwl) %*% t(X1fwl))
leverage <- data.frame(lvg=lvgx1)
leverage$x <- c(1:nrow(leverage))
ggplot(leverage, aes(x=x, y=lvg)) + geom_point() +
  xlab("observation") + ylab("partial leverage") +
  geom_hline(yintercept = length(reg.out$coefficients)/nrow(leverage),
             color = "red")
ggsave('leverage_x1.png', height = 4, width = 6, dpi = 1000)

# Partial leverage of X2
x1 <- X[,2]
xx1 <- X[,-2]
X1fwl <- x1 - xx1 %*% solve(t(xx1) %*% xx1) %*% t(xx1) %*% x1
lvgx1 <- diag(X1fwl %*% solve(t(X1fwl) %*% X1fwl) %*% t(X1fwl))
leverage <- data.frame(lvg=lvgx1)
leverage$x <- c(1:nrow(leverage))
ggplot(leverage, aes(x=x, y=lvg)) + geom_point() +
  xlab("observation") + ylab("partial leverage") +
  geom_hline(yintercept = length(reg.out$coefficients)/nrow(leverage),
             color = "red")
ggsave('leverage_x2.png', height = 4, width = 6, dpi = 1000)


# Compute within group leverage
sobs <- unique(data$prov)
G <- length(sobs)
storelev <- matrix(NA, G, 2)

i <- 1
for (sob in sobs) {
  datacl <- subset(data, data$prov == sob)

  Xg <- fastDummies::dummy_cols(as.factor(datacl$year),
                                remove_first_dummy = FALSE)[,-1]

  if (i==1) Xg<-cbind(0,Xg)
```

```
292
293   Xg <- cbind(datacl$x1, datacl$x2, datacl$ltotpop, datacl$lurbpop, Xg)
294   Xg <- data.matrix(Xg)
295   XXg <- t(Xg)%*%Xg
296
297   storelev[i, 2] <- sum(diag(XXg %*% XX)) # compute nuclear norm
298   storelev[i, 1] <- i
299   i <- i + 1
300 }
301
302 toplot <- as.data.frame(storelev)
303 toplot$V1 <- as.numeric(toplot$V1)
304 toplot$V2 <- as.numeric(toplot$V2)
305 toplot$V4 <- toplot$V2/sum(toplot$V2)
306
307 ggplot(toplot, aes(x=V1, y=V2)) + geom_point() +
308   xlab("Province") + ylab("partial leverage") +
309   geom_hline(yintercept = length(reg.out$coefficients)/G, color = "red")
310 ggsave('leverage_partial_cl.png', height = 4, width = 6, dpi = 1000)
```

Question 3 Code

```
1  setwd("D:/Dropbox/Universit/PhD/II Year/Spring/ECO519 - Non-linear Econometrics/psets/ps3")
2
3  # Load stuff
4  library(haven)
5  library(ivreg)
6
7  library(sandwich)
8  library(dfadjust)
9  library(xtable)
10 library(tidyverse)
11 library(reshape2)
12
13 theme_set(theme_bw())
14
15 CI.store <- matrix(NA, 3, 6)
16
17 data <- haven::read_dta("ak91.dta")
18
19 # Extract subsets of data
20 data.pac <- subset(data, census == 1980 & cohort == 2 & division == 9)
21 data.atl <- subset(data, census == 1980 & cohort == 2 & division == 2)
22
23 # Create date of birth instrument
24 data.pac$birthq1 <- 1*(data.pac$age == floor(data.pac$age))
25 data.atl$birthq1 <- 1*(data.atl$age == floor(data.atl$age))
26
27 # IV regression
28 iv.pac <- ivreg(lwage ~ 1 + educ | birthq1, data = data.pac)
29 se.pac <- sandwich::vcovHC(iv.pac, type = 'HC1')
30 iv.atl <- ivreg(lwage ~ 1 + educ | birthq1, data = data.atl)
31 se.atl <- sandwich::vcovHC(iv.atl, type = 'HC1')
32
33 # Wald Confidence Interval
```

```r
b.iv.pac <- iv.pac$coefficients[2]
b.iv.atl <- iv.atl$coefficients[2]

CI.store[1, ] <- c(b.iv.pac - qnorm(0.975)*sqrt(se.pac[2,2]),
                   b.iv.pac,
                   b.iv.pac + qnorm(0.975)*sqrt(se.pac[2,2]),
                   b.iv.atl - qnorm(0.975)*sqrt(se.atl[2,2]),
                   b.iv.atl,
                   b.iv.atl + qnorm(0.975)*sqrt(se.atl[2,2]))

# Anderson-Rubin Confidence Interval
beta.grid.pac <- seq(from=-0.5, to=0.5, by=0.001)
beta.grid.atl <- seq(from=-50, to=50, by = 1)
ar.pv.pac <- matrix(NA, nrow = length(beta.grid.pac), ncol = 2)
ar.pv.atl <- matrix(NA, nrow = length(beta.grid.atl), ncol = 2)
ar.ts.pac <- ar.pv.pac
ar.ts.atl <- ar.pv.atl

i <- 1

for (b in beta.grid.pac) {
  # residualize outcome
  data.pac$lwage.res <- data.pac$lwage - b*data.pac$educ

  # run residualized reduced form
  ar.pac <- lm(lwage.res ~ 1 + birthq1, data = data.pac)

  # compute robust variance-covariance
  vc.pac <- sandwich::vcovHC(ar.pac, type = 'HC1')

  # test coefficient on instrument
  test.pac <- ar.pac$coefficients[2]/sqrt(vc.pac[2,2])
  ar.ts.pac[i,1] <- test.pac

  # retrieve p-value
  ar.pv.pac[i,1] <- pnorm(abs(test.pac), lower.tail = FALSE)*2
  i <- i + 1
}

ar.pv.pac[,2] <- "Pacific"

i <- 1

for (b in beta.grid.atl) {
  # residualize outcome
  data.atl$lwage.res <- data.atl$lwage - b*data.atl$educ

  # run residualized reduced form
  ar.atl <- lm(lwage.res ~ 1 + birthq1, data = data.atl)

  # compute robust variance-covariance
  vc.atl <- sandwich::vcovHC(ar.atl, type = 'HC1')

  # test coefficient on instrument
  test.atl <- ar.atl$coefficients[2]/sqrt(vc.atl[2,2])
  ar.ts.atl[i,1] <- test.atl

  # retrieve p-value
  ar.pv.atl[i,1] <- pnorm(abs(test.atl), lower.tail = FALSE)*2
```

```
  i <- i + 1
}

ar.pv.atl[,2] <- "mid-Atlantic"


toplot <- data.frame(beta = c(beta.grid.pac, beta.grid.atl),
                     pv = rbind(ar.pv.pac, ar.pv.atl))
colnames(toplot) <- c("beta", "pvalue", "division")
toplot$pvalue <- as.numeric(toplot$pvalue)


ggplot(subset(toplot, division=="Pacific"), aes(x=beta, y=pvalue)) +
  geom_line() + geom_hline(yintercept=0.05, color = "red") +
  geom_vline(xintercept = b.iv.pac, color = "black", linetype="dashed") +
  ylab("AR p-value")
ggsave('AR_pacific.png', height = 4, width = 6, dpi = 1000)

ggplot(subset(toplot, division=="mid-Atlantic"), aes(x=beta, y=pvalue)) +
  geom_line() + geom_hline(yintercept=0.05, color = "red") +
  geom_vline(xintercept = b.iv.pac, color = "black", linetype="dashed") +
  ylab("AR p-value")
ggsave('AR_atlantic.png', height = 4, width = 6, dpi = 1000)

ar.ci.pac <- beta.grid.pac[as.numeric(ar.pv.pac[,1]) >= 0.05]

CI.store[2, ] <- c(ar.ci.pac[1],
                   b.iv.pac,
                   ar.ci.pac[length(ar.ci.pac)],
                   -Inf,
                   b.iv.atl,
                   Inf)

# tF procedure

# First stage
fs.pac <- lm(educ ~ 1 + birthq1, data = data.pac)
fs.atl <- lm(educ ~ 1 + birthq1, data = data.atl)
fs.vc.pac <- sandwich::vcovHC(fs.pac, type ='HC1')
fs.vc.atl <- sandwich::vcovHC(fs.atl, type ='HC1')

t.pac <- fs.pac$coefficients[2]/sqrt(vcovHC(fs.pac, type = 'HC1')[2,2])
f.pac <- t.pac^2
t.atl <- fs.atl$coefficients[2]/sqrt(vcovHC(fs.atl, type = 'HC1')[2,2])
f.atl <- t.atl^2

CI.store[3, ] <- c(-0.13506071, b.iv.pac, 0.39288127,
                   -Inf, b.iv.atl, Inf)

# Store table
tab <- as.table(rbind(CI.store, c(NA,f.pac,NA,NA,f.atl,NA)))
rownames(tab) <- c("Wald CI", "Anderson-Rubin CI", "tF CI", "First-stage F")
xtable(tab)
```