Assignment 4

Diana Perez

June 14, 2020

Github repo and summary

1. Download Hansen_dwi.dta from github at the following address.

   use https://github.com/scunning1975/causal-inference-class/raw/master/ hansen_dwi, clear

   Create a new github repo named "RDD". Inside the RDD directory, put all the subdirectories we've discussed in class. Post the link to the repo so I can see it's done as discussed in your assignment. Save the Hansen_dwi.dta file into your new /data subdirectory. Note: The outcome variable is "recidivism" or "recid" which is measuring whether the person showed back up in the data within 4 months.

   Repo's link: https://github.com/dianakperezlp/RDD

   The upload and saving on the data available on the do-file (at the end of this document).

2. In the writing subdirectory, place your assignment. For the first part of this assignment, read Hansen's paper in the /articles directory of the main class github entitled "Hansen AER". Briefly summarize this paper. What is his research question? What data does he use? What is his research design, or "identification strategy"? What are his conclusions?

   Hansen (2015) evaluates the effect of punishments for driving under the influence (DUI) of alcohol on the probability of recidivism. In order to identify this effect, the author take advantage of "administrative records on 512,965 DUI strops from the state of Washington (WA)" for individuals above legal drinking age from 1995 to 2011.

   Measuring the blood alcohol content (BAC) is a low cost's noninvasive procedure that defines DUI (0.08) and aggravated DUI (0.15). Also, it's refusal lead into similar punishments as found guilty of drunk driving. Then, it is plausible that an important amount of the population do not refuse to take the BAC. Hansen (2015) take advantages of this discrete thresholds that defines DUI and aggravated DUI as quasi-random variation of DUI. The author implement a RDD, where he compares individuals just above the threshold with those just below in order to obtain the DUI's effect for the marginal drivers.

The penalties associated to DUI could include a penalty fee ($865.6-$5,000 USD), jail time, home release and license suspension. Also, there are considerable more severe for previous offenders. Then, it is plausible that being encountered DUI effectively reduce the recidivism.

To conclude, the author finds that DUI and aggravated DUI effectively reduce recidivism in 2 percentage points (17%) and 1 percentage points (9%) for the marginal drivers. Also, he finds that thins effect mainly operate through deterrence. Even though, he could not rule out incapacitating and rehabilitation as possible mediators.

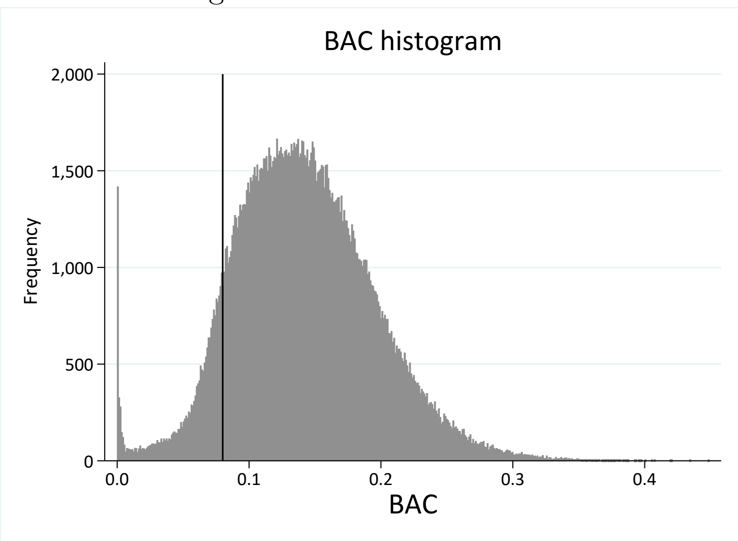
Replication

3. In the United States, an officer can arrest a driver if after giving them a blood alcohol content (BAC) test they learn the driver had a BAC of 0.08 or higher. We will only focus on the 0.08 BAC cutoff. We will be ignoring the 0.15 cutoff for all this analysis. Create a dummy equaling 1 if bac1>= 0.08 and 0 otherwise in your do file or R file.

   The generation of eligibility variables available on the do-file (at the end of this document).

4. The first thing to do in any RDD is look at the raw data and see if there's any evidence for manipulation ("sorting on the running variable"). If people were capable of manipulating their blood alcohol content (bac1), describe the test we would use to check for this. Now evaluate whether you see this in these data? Either recreate Figure 1 using the bac1 variable as your measure of blood alcohol content or use your own density test from software. Do you find evidence for sorting on the running variable?

Figure 1. BAC Distribution



The Figure 1 presents the BAC density in order to explore possible sorting in the running variable. Near to the DUI cutoff, there is a small bunching to the right sight. But, it is not clear weather it could be associated with sorting and Hansen (2015) does not discuss this. In order to statistically evaluate if there's a hipping at the DUI threshold a manipulation test from Cattaneo et al. (2017) is implemented. The p-value associated is 0.027. Then, at 95% of confidence there is sorting on the BAC at the DUI threshold.

5. The second thing we need to do is check for covariates balance. Recreate Table 2 Panel A but only white male, age and accident (acc) as dependent variables. Use your equation 1) for this. Are the covariates balanced at the cutoff? It's okay if they are not exactly the same as Hansen's.

Table 2. Regression Discontinuity Estimates for the Effect of Exceeding BAC Thresholds on Predetermined Covariates

	(1)	(2)	(3)	(4)
VARIABLES	Male	White	Age	Accident
DUI	0.006	0.006	-0.140	-0.003
	(0.006)	(0.005)	(0.164)	(0.004)
Mean at (0.079)	0.787	0.845	34.90	0.100
Controls	No	No	No	No
Observations	89,967	89,967	89,967	89,967

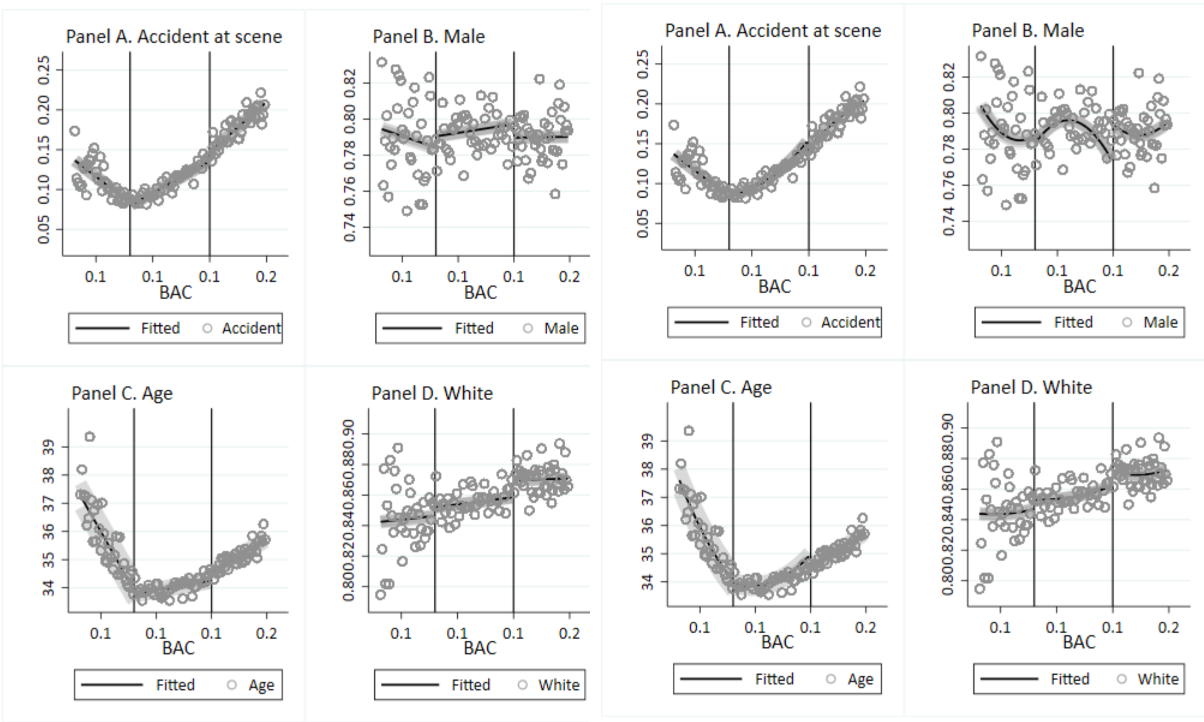
Notes: This table contains regression discontinuity based estimates of the effect of having BAC above the legal thresholds on predetermined characteristics. All regressions have a bandwidth of 0.05 and use a rectangular kernel for weighting. Based on data from the 1999–2007 Washington State Impaired Driver Program. White-Huber standard errors are in parentheses.
*** Significant at the 1 percent level.
** Significant at the 5 percent level.
* Significant at the 10 percent level.

Table 2 presents a parametric approach to evaluate the local continuity of the covariates. As there is no any DUI coefficient significant, the covariates are balanced and it is plausible that the potential outcomes are smooth.

6. Recreate Figure 2 panel A-D. You can use the -cmogram- command in Stata to do this. Fit both linear and quadratic with confidence intervals. Discuss what you find and compare it with Hansen's paper.

Figure 2. BAC and Characteristics

Panel A. Linear polynomial Panel B. Quadratic polynomial



Notes: Based on administrative records from the Washington State Impaired Driver Testing Program, 1999–2007. Points represent the averages, with fitted values based on local linear models in black lines. The vertical black lines represent the two legal thresholds at 0.08 and 0.15. 95% confidence intervals presented.

As in Hansen's paper, there is no evidence of local discontinuities at the DUI threshold with 95% confidence level for linear and quadratic polynomial.

7. Estimate equation (1) with recidivism (recid) as the outcome. This corresponds to Table 3 column 1, but since I am missing some of his variables, your sample size will be the entire dataset of 214,558. Nevertheless, replicate Table 3, column 1, Panels A and B. Note that these are local linear regressions and Panel A uses as its bandwidth 0.03 to 0.13. But Panel B has a narrower bandwidth of 0.055 to 0.105. Your table should have three columns and two A and B panels associated with the different bandwidths.:

   a. Column 1: control for the bac1 linearly.
   b. Column 2: interact bac1 with cutoff linearly.
   c. Column 3: interact bac1 with cutoff linearly and as a quadratic.
   d. For all analysis, use heteroskedastic robust standard errors.

Table 3. Regression Discontinuity Estimates for the Effect of Exceeding the 0.08 BAC Threshold on Recidivism

	Linear	Linear differentiated	Quadratic differentiated
Panel A. BAC ∈ [0.03,0.13]			
DUI	-0.027***	-0.024***	-0.014**
	(0.004)	(0.004)	(0.006)
Mean	0.107	0.107	0.107
Observations	90,000	90,000	90,000
Panel B. BAC ∈ [0.055,0.105]			
DUI	-0.020***	-0.019***	-0.017**
	(0.005)	(0.005)	(0.008)
Mean	0.105	0.105	0.105
Observations	56,000	56,000	56,000

Notes: This table contains regression discontinuity based estimates of the effect of having BAC above the DUI threshold on recidivism for all drivers, those with no prior test, and drivers with at least one prior test. Panel A contains estimates with a bandwidth of 0.05 while Panel B has a bandwidth of 0.025, with all regressions utilizing a rectangular kernel for weighting. Controls include indicators for year, race, gender, and age of the offender. Based on administrative records from the Washington State Impaired Driver Testing Program, 1999–2007. White-Huber standard errors are in parentheses.
*** Significant at the 1 percent level.
** Significant at the 5 percent level.
* Significant at the 10 percent level.

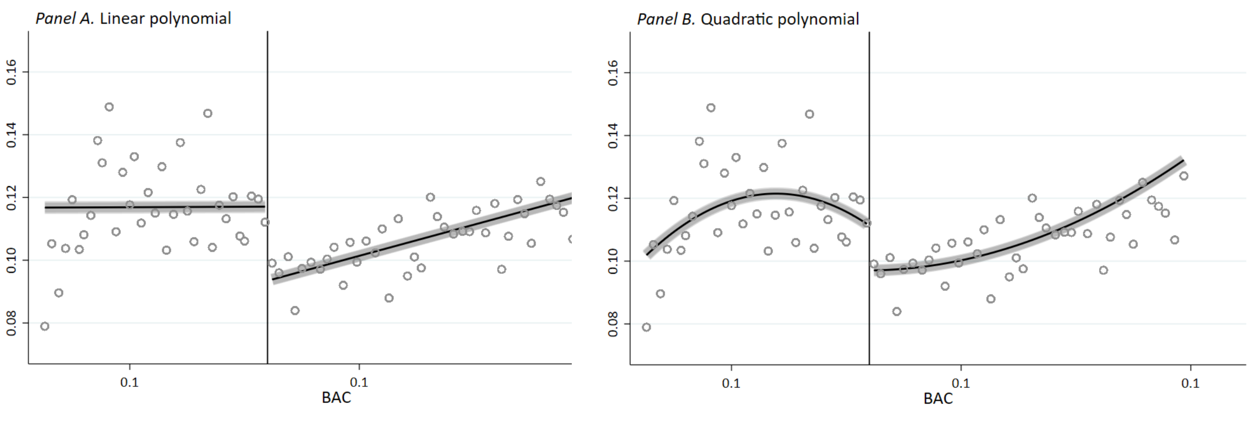
At 99% of confidence, the DUI, on average, reduce the recidivism probability at 2 percentage points foe the marginal drivers. This result is robust to the polynomial specification.

8. Recreate the top panel of Figure 3 according to the following rule:

    a. Fit linear fit using only observations with less than 0.15 bac on the bac1.

    b. Fit quadratic fit using only observations with less than 0.15 bac on the bac1.

Figure 3. BAC and Recidivism



Notes: Based on administrative records from the Washington State Impaired Driver Testing Program, 1999–2007. Points represent the averages, with fitted values based on local linear models in black lines. The vertical black line represent the legal threshold at 0.08. 95% confidence intervals presented.

Reference

Hansen, B. (2015). "Punishment and Deterrence: Evidence from Drunk Driving." American Economic Review. 105(4): 1581–1617

Cattaneo, M. D., Jansson, M., and Ma, X. (2017) "rddensity: Manipulation Testing Based on Density Discontinuity." The Stata Journal. ii: 1-24.

```
* Causal Inference and Research Design
* Assignment 4
* Autor: Diana Perez

clear all
set more off
cap log close
cls
graph set window fontface "Calibri"


**************************************************************************************
** ASSIGNMENT 3                                                                    **
**************************************************************************************
global path "C:\Users\Diana\Documents\GitHub\RDD"
cd "${path}"

* I. Github repo and summary
* ------------------------------------------------------------------------------------

*) 1. Saving data
use                                                                                  ///
"https://github.com/scunning1975/causal-inference-class/raw/master/hansen_dwi",  ///
clear

compress
save "${path}\Data\Hansen_dwi.dta", replace

* II. Replication
* ------------------------------------------------------------------------------------
use "${path}\Data\Hansen_dwi.dta", clear

*) 3. Eligibility variable
gen eligibility=bac1>=0.08 if !missing(bac1)
gen bac1_ajust=bac1-0.08
gen bac2_ajust=bac1_ajust*bac1_ajust

    * Labels
    label var male "Male"
    label var white "White"
    label var aged "Age"
    label var acc "Accident"
    label var bac1_ajust "DUI"

*) 4. Testing Manipulation on the RV

    * Packages
    net install rddensity,                                                           ///
        from("https://sites.google.com/site/rdpackages/rddensity/stata") replace
    net install lpdensity,                                                           ///
        from("https://sites.google.com/site/nppackages/lpdensity/stata") replace

    * Cattaneo et al.
    rddensity bac1_ajust /*P-value: 0.0276 */

    * Figure 1
    hist bac1, freq bin(450) bc(gs9) lc(gs9) graphregion(fcolor(white))          ///
        ti("BAC histogram", color(black) size(vlarge) lwidth(vvthick))           ///
        xti("BAC", size(vlarge) lwidth(vvthick))                                 ///
        yti("Frequency", size(medium) lwidth(vvthick))                           ///
        addplot(pci  0 0.08 2000 0.08, lc(black))                                ///
        xlabel(0(0.1)0.4) xvarformat(%2.1f) yvarformat(%9.0gc)                   ///
        ylabel(, angle(0)) legend(off)
```

```
    gr export "${path}\Figures\Figure 01.pdf", replace as(pdf)

*) 5. Table 2. Covariance continuity

    * Editing e(N)
    cap program drop changeN
    program define changeN, eclass
        /* This program edits the e(N). It replace it for any scalar named nobs.*/

        ereturn scalar N = nobs
    end

    * Regressions and table
    local bw=0.05
    local n=0

    global covs male white aged acc

    foreach var of varlist $covs {

        local ++n
        local vlab: variable label `var'

        * Estimation
        rdrobust `var' bac1_ajust, kernel(uniform) h(`bw' `bw') p(1) vce(hc0)
        est store rdrob
        * Output
        if "`n'"=="1" local comp="replace"
        else local comp="append"

        if "`n'"=="3" local j=1
        else local j=3

        scalar nobs=e(N_h_l)+e(N_h_r)
        local nobs=string(e(N_h_l)+e(N_h_r),"%9.0gc")
        changeN

        qui sum `var' if bac1_ajust>=-`bw' & bac1_ajust<0
        local mu=string(r(mean), "%5.`j'f")

        outreg2 using "${path}\Tables\Table2.tex", `comp' nocons nor2 decm(.)   ///
            dec(3)                                                              ///
            addstat(Mean at (0.079), `mu')                                      ///
            addtext(Controls,No)                                                ///
            label nonotes                                                       ///
            addn("Standard errors are in parentheses."                          ///
            "*** Significant at the 1 percent level."                           ///
            "** Significant at the 5 percent level."                            ///
            "* Significant at the 10 percent level.")

    }

*) 6. Figure 2.
local cut=0.08
local cut2=0.15

local bw=0.05
local bw2=`cut2'-`cut'

local bwl=`cut'-`bw'
local bwu=`cut'+`bw'
local bwl2=`cut2'-`bw'
```

```
local bwu2=`cut2'+`bw'

local nbinl=35
local nbinr=40

global covs acc male aged white

foreach j of numlist 1/2{

    local n=0
    foreach var of  varlist $covs {

        local ++n
        local vlab: variable label `var'

        if "`var'"=="acc" {
            local lti="Panel A. Accident at scene"
            local yax="ylabel(0.05(0.05)0.25) yscale(range(0.03 0.26))"
        }
        else if "`var'"=="male" {
            local lti="Panel B. Male"
            local yax="ylabel(0.74(0.02)0.82) yscale(range(0.73 0.83))"
        }
        else if "`var'"=="aged" {
            local lti="Panel C. Age"
            local yax="ylabel(34(1)39) yscale(range(33.5 40))"
            *local yax=" "
        }
        else if "`var'"=="white" {
            local lti="Panel D. White"
            local yax="ylabel(0.8(0.02)0.9) yscale(range(0.79 0.91))"
        }

        if "`n'"=="3" local l=0
        else local l=2

        * 2nd cutoff and above
        cap drop rdplot_*
        rdplot `var' bac1 if inrange(bac1,`bwl2',`bwu2'), binselect(es)         ///
            c(`cut2') genvars p(`j')  kernel(uniform) h(`bw' `bw2')             ///
            nbins(`nbinr' `nbinr')
        cap drop v2rdplot_mean_y v2rdplot_mean_x v2rdplot_hat_y
        rename (rdplot_mean_y rdplot_mean_x rdplot_hat_y)                       ///
            (v2rdplot_mean_y v2rdplot_mean_x v2rdplot_hat_y)

        * origin to 2nd cutoff
        cap drop rdplot_*
        rdplot `var' bac1 if inrange(bac1,`bwl',`cut2'), binselect(es)          ///
            c(`cut') genvars p(`j')  kernel(uniform) h(`bw' `bw')               ///
            nbins(`nbinl' `nbinr')

        replace rdplot_mean_y=v2rdplot_mean_y if inrange(bac1,`cut2',`bwu2')
        replace rdplot_mean_x=v2rdplot_mean_x if inrange(bac1,`cut2',`bwu2')
        replace rdplot_hat_y=v2rdplot_hat_y if inrange(bac1,`cut2',`bwu2')

        * Confidence intervals
        cap drop ep2
        cap drop sd
        cap drop rdplot_ci_l
        cap drop rdplot_ci_r

        gen ep2=(rdplot_mean_y-rdplot_hat_y)^2
        sum ep2 if inrange(bac1,`bwl',`cut')
```

```
        gen sd=r(mean)*r(N)/(r(N)-2) if inrange(bac1,`bwl',`cut')
        sum ep2 if inrange(bac1,`cut',`cut2')
        replace sd=r(mean)*r(N)/(r(N)-2) if inrange(bac1,`cut',`cut2')
        sum ep2 if inrange(bac1,`cut2',`bwu2')
        replace sd=r(mean)*r(N)/(r(N)-2) if inrange(bac1,`cut2',`bwu2')

        gen rdplot_ci_l=rdplot_hat_y-1.96*sd
        gen rdplot_ci_r=rdplot_hat_y+1.96*sd

        * Graphs
        if `j'==1 local q="l"
        else local q="q"

        tw (`q'fit rdplot_hat_y rdplot_mean_x if inrange(bac1,`bwl',`cut'),      ///
            lcolor(black) lwidth(medthick) lpattern(solid) xvarformat(%2.1f)     ///
            yvarformat(%3.`l'f))                                                 ///
            (`q'fit rdplot_ci_r rdplot_mean_x if inrange(bac1,`bwl',`cut'),      ///
            lcolor(black%15) lwidth(vvthick) lpattern(solid))                    ///
            (`q'fit rdplot_ci_l rdplot_mean_x if inrange(bac1,`bwl',`cut'),      ///
            lcolor(black%15) lwidth(vvthick) lpattern(solid))                    ///
            (`q'fit rdplot_hat_y rdplot_mean_x if inrange(bac1,`cut',`cut2'),    ///
            lcolor(black) lwidth(medthick) lpattern(solid))                      ///
            (`q'fit rdplot_ci_r rdplot_mean_x if inrange(bac1,`cut',`cut2'),     ///
            lcolor(black%15) lwidth(vvthick) lpattern(solid))                    ///
            (`q'fit rdplot_ci_l rdplot_mean_x if inrange(bac1,`cut',`cut2'),     ///
            lcolor(black%15) lwidth(vvthick) lpattern(solid))                    ///
            (`q'fit rdplot_hat_y rdplot_mean_x if inrange(bac1,`cut2',`bwu2'),   ///
            lcolor(black) lwidth(medthick) lpattern(solid))                      ///
            (`q'fit rdplot_ci_r rdplot_mean_x if inrange(bac1,`cut2',`bwu2'),    ///
            lcolor(black%15) lwidth(vvthick) lpattern(solid))                    ///
            (`q'fit rdplot_ci_l rdplot_mean_x if inrange(bac1,`cut2',`bwu2'),    ///
            lcolor(black%15) lwidth(vvthick) lpattern(solid))                    ///
            (scatter rdplot_mean_y rdplot_mean_x, msymbol(circle_hollow)         ///
            mcolor(gs9%80)),                                                     ///
            xline(`cut', lcolor(black) lw(medium))                               ///
            xline(`cut2', lcolor(black) lw(medium))                              ///
            xti("BAC", size(medlarge) lwidth(vvthick))                           ///
            yti(" ", size(medium))                                               ///
            ti("`lti'", color(black) size(medlarge) lwidth(vvthick)              ///
            j(left) placement(nwest))                                            ///
            xlabel(0.05 0.1 0.15 0.2) xscale(range(0.03 0.21)) `yax'             ///
            graphregion(fcolor(white))                                           ///
            legend(order (1 10) label (1 "Fitted") label (10 "`vlab'"))          ///
            name(G`n'_`j', replace) xsize(7) ysize(7)

    }

    gr combine G1_`j' G2_`j' G3_`j' G4_`j', col(2) graphregion(color(white))     ///
        ysize(14) xsize(12)
    gr export "${path}\Figures\Figure 02_P`j'.pdf", replace as(pdf)
}

*) Table 3
mat MAT_tabla=J(8,6,.)
mat MAT_tabla_s=J(8,6,0)

global controls aged white i.year male

local n=1
foreach j of numlist 0.05 0.03{

    * BAC
    qui reg recid eligibility bac1_ajust                                         ///
```

```
        ${controls} if inrange(bac1_ajust,-`j',`j'), r

    mat MAT_tabla[`n'+1,1]=_b[eligibility]
    mat MAT_tabla[`n'+1,2]=_se[eligibility]
    local N=string(e(N),"%5.0gc")
    mat MAT_tabla[`n'+3,1]=`N'

    local p=string(ttail(e(df_r),abs(_b[eligibility]/_se[eligibility]))*2,      ///
        "%6.4f")
    matrix MAT_tabla_s[`n'+1,1] = (`p' <= 0.1) + (`p' <= 0.05) + (`p' <= 0.01)


    qui sum recid if e(sample)
    mat MAT_tabla[`n'+2,1]=r(mean)

    * BAC x Elegibility
    qui reg recid eligibility bac1_ajust eligibility#c.bac1_ajust               ///
        ${controls} if inrange(bac1_ajust,-`j',`j'), r

    mat MAT_tabla[`n'+1,3]=_b[eligibility]
    mat MAT_tabla[`n'+1,4]=_se[eligibility]
    local N=string(e(N),"%5.0gc")
    mat MAT_tabla[`n'+3,3]=`N'

    local p=string(ttail(e(df_r),abs(_b[eligibility]/_se[eligibility]))*2,      ///
        "%6.4f")
    matrix MAT_tabla_s[`n'+1,3] = (`p' <= 0.1) + (`p' <= 0.05) + (`p' <= 0.01)

    qui sum recid if e(sample)
    mat MAT_tabla[`n'+2,3]=r(mean)

    * BAC2 x Eligibility
    qui reg recid eligibility bac1_ajust bac2_ajust eligibility#c.bac1_ajust    ///
        eligibility#c.bac2_ajust                                                ///
        ${controls} if inrange(bac1_ajust,-`j',`j'), r

    mat MAT_tabla[`n'+1,5]=_b[eligibility]
    mat MAT_tabla[`n'+1,6]=_se[eligibility]
    local N=string(e(N),"%5.0gc")
    mat MAT_tabla[`n'+3,5]=`N'

    local p=string(ttail(e(df_r),abs(_b[eligibility]/_se[eligibility]))*2,      ///
        "%6.4f")
    matrix MAT_tabla_s[`n'+1,5] = (`p' <= 0.1) + (`p' <= 0.05) + (`p' <= 0.01)

    qui sum recid if e(sample)
    mat MAT_tabla[`n'+2,5]=r(mean)

    local n=`n'+4

}

frmttable using "${path}\Tables\Table3", replace tex                            ///
    statmat(MAT_tabla) annotate(MAT_tabla_s) asymbol("*","**","***")            ///
    substat(1) noblankrows                                                      ///
    ct("","Linear","Linear diferenciated","Quadratic diferenciated")            ///
    rt("{\i Panel A}. BAC $\in$ [0.03,0.13]"\""\"DUI"\""\"Mean"\""\             ///
    "Observations"\""\                                                          ///
    "{\i Panel B}. BAC $\in$ [0.055,0.105]"\""\"DUI"\""\"Mean"\""\              ///
    "Observations")                                                             ///
    sdec(3,3,3\3,3,3\3,3,3\3,3,3\3,3,3\3,3,3\0,0,0\                             ///
    3,3,3\3,3,3\3,3,3\3,3,3\3,3,3\3,3,3\0,0,0)

```

```
*) Figure 3
local cut=0.08
local cut2=0.15

local bw=0.05
local bw2=`cut2'-`cut'

local bwl=`cut'-`bw'
local bwu=`cut'+`bw'

local nbinl=35
local nbinr=40

foreach j of numlist 1/2{

        if "`j'"=="1" local lti="{it: Panel A.} Linear polynomial"
        else if "`j'"=="2" local lti="{it: Panel B.} Quadratic polynomial"
        local yax="ylabel(0.08(0.02)0.16) yscale(range(0.07 0.17))"

        * origin to 2nd cutoff
        cap drop rdplot_*
        rdplot recid bac1 if inrange(bac1,`bwl',`cut2'), binselect(es)        ///
            c(`cut') genvars p(`j')  kernel(uniform) h(`bw' `bw')             ///
            nbins(`nbinl' `nbinr')

        * Confidence intervals
        cap drop ep2
        cap drop sd
        cap drop rdplot_ci_l
        cap drop rdplot_ci_r

        gen ep2=(rdplot_mean_y-rdplot_hat_y)^2
        sum ep2 if inrange(bac1,`bwl',`cut')
        gen sd=r(mean)*r(N)/(r(N)-2) if inrange(bac1,`bwl',`cut')
        sum ep2 if inrange(bac1,`cut',`cut2')
        replace sd=r(mean)*r(N)/(r(N)-2) if inrange(bac1,`cut',`cut2')

        gen rdplot_ci_l=rdplot_hat_y-1.96*sd
        gen rdplot_ci_r=rdplot_hat_y+1.96*sd

        * Graphs
        if `j'==1 local q="l"
        else local q="q"

        tw (`q'fit rdplot_hat_y rdplot_mean_x if inrange(bac1,`bwl',`cut'),     ///
            lcolor(black) lwidth(medthick) lpattern(solid) xvarformat(%2.1f)    ///
            yvarformat(%3.2f))                                                ///
            (`q'fit rdplot_ci_r rdplot_mean_x if inrange(bac1,`bwl',`cut'),     ///
            lcolor(black%15) lwidth(vvthick) lpattern(solid))                   ///
            (`q'fit rdplot_ci_l rdplot_mean_x if inrange(bac1,`bwl',`cut'),     ///
            lcolor(black%15) lwidth(vvthick) lpattern(solid))                   ///
            (`q'fit rdplot_hat_y rdplot_mean_x if inrange(bac1,`cut',`cut2'),   ///
            lcolor(black) lwidth(medthick) lpattern(solid))                     ///
            (`q'fit rdplot_ci_r rdplot_mean_x if inrange(bac1,`cut',`cut2'),    ///
            lcolor(black%15) lwidth(vvthick) lpattern(solid))                   ///
            (`q'fit rdplot_ci_l rdplot_mean_x if inrange(bac1,`cut',`cut2'),    ///
            lcolor(black%15) lwidth(vvthick) lpattern(solid))                   ///
            (scatter rdplot_mean_y rdplot_mean_x, msymbol(circle_hollow)        ///
            mcolor(gs9%80)),                                                    ///
            xline(`cut', lcolor(black) lw(medium))                              ///
            xti("BAC", size(medlarge) lwidth(vvthick))                          ///
            yti(" ", size(medium))                                              ///
            ti("`lti'", color(black) size(medlarge) lwidth(vvthick)            ///
```

```
            j(left) placement(nwest))                                           ///
            xlabel(0.05 0.1 0.15) xscale(range(0.03 0.16)) `yax'               ///
            graphregion(fcolor(white))                                          ///
            legend(off)                                                         ///
            name(G_`j', replace) xsize(8) ysize(5)

    gr export "${path}\Figures\Figure 03_P`j'.pdf", replace as(pdf)
}
```