Regression II
Vanderbilt University
Prof. Sean P. Corcoran

Problem Set 6 *Solutions*
56 total points

Problem Set 6 *Solutions*

Question 1. This problem will replicate some of the results in Lee (2008). Each observation is a Congressional district election between 1948 and 1998. The running variable is *difdemshare*, the difference between the Democratic candidate's vote share and the largest vote share of the other parties. If the Democrat won, *difdemshare* will be greater than zero.

(a) Conduct a regression discontinuity analysis that includes the following elements. Before doing so, write down the assumptions that need to hold for an RD to produce the causal effect of incumbency. (**40 points**)

There are four key assumptions. First, there is a discontinuous jump in "treatment" at the cutpoint. Define "treatment" as a Democrat winning the election. The running variable in this case is the net Democratic vote share. When this value is below zero, the non-Democratic candidate won. When it exceeds zero, the Democrat won. The discontinuity is sharp. Second, the relationship between the outcome and the running variable is continuous in the neighborhood of the cutpoint, in the absence of treatment. Consider the outcome *difdemsharenext*, which is the net Democratic vote share in election *t+1*. In the absence of a treatment effect, there is no reason to believe that Democratic support in election *t+1* would change discontinuously with the net vote share in election *t*. Later we perform one test of this assumption. Third, the forcing variable has not been manipulated to affect who receives treatment. In the U.S., elections are generally conducted with integrity, so manipulation seems unlikely. More importantly, even if maleficent persons were working to influence the outcome of an election, it would be hard for them to do so precisely enough to have an impact right at the margin of victory (i.e., near the cutpoint). We can at least conduct a test to look for irregularities in the density of the running variable. Fourth, there are no other "treatments" with the same eligibility rule, and thus no confounders.

At the end of this document is a complete RD analysis that includes the elements below, along with the do-file.

(a) Two main outcome variables: *difdemsharenext*, the difference between the Democratic vote share and the largest vote share of the other parties in the next election, and *demwinnext*, an indicator variable equal to one if a Democrat won the next election.

(b) Scatterplot and `binscatter` showing the relationship between *demsharenext* and the running variable. Hint: it may help visually to focus on observations in which $abs(difdemshare) < 0.25$, and to increase the number of bins in `binscatter`.

(c) Parametric RD models assuming a linear relationship with the running variable, then a quadratic, then a quartic (i.e., up to the fourth power). In all three cases allow the relationship to differ on each side of the cutoff, and allow for clustering in the standard error calculation (using the Congressional district id as the clustering variable). Repeat the same models but include covariate controls: *demofficeexp* and *othofficeexp* (measures of the Democrat's and opposition's experience in office).

(d) Non-parametric RD estimates using `rd`, using the default bandwidth. Again use the clustered standard errors.

(e) For your quartic model, create a scatterplot that includes the fitted model on each side of the cutofff.

(f) A histogram for the running variable and a McCrary test to look for manipulation at the cutpoint (Russian hackers?)

(g) A validity check in which you use *demshareprev* and *demwinprev* as the outcome variables. What does this accomplish?

Write up your findings, interpreting and comparing your point estimates across the different models.

A scatterplot and binned scatterplot provide suggestive evidence of an effect of incumbency on the net Democratic vote share in election *t+1*. The discontinuity in the net vote share is more apparent in the binned scatterplot, which focuses on a narrow band around the threshold for a Democratic win.

Estimates from parametric RD models are reported in Tables 1-2. In Table 1, columns (1)-(3) report results from linear, quadratic, and quartic models in which the outcome is the net Democratic share in election *t+1*. Columns (4)-(6) do the same, but for the binary outcome of a Democratic victory in election *t+1*. Table 2 repeats the analysis in Table 1, but includes controls for the Democrat's and opponent's experience in political office. The models without controls indicate that—at the margin of victory—a Democratic win in election *t* increases the likelihood of a Democratic win in election *t+1* by 14.3 to 22.9 percentage points (columns 4-6). The impact on the net Democratic vote share is 5.2 to 8.1 percentage points (columns 1-3). The point estimates from the quartic model stand out as the largest of these

(columns 3 and 6), and the inclusion of controls for prior experience in office generally produce larger point estimates (Table 2).

The non-parametric RD estimates using the optimal bandwidth are reported in the first row of Table 3. A Democratic win in election *t* increases the likelihood of a Democratic win in election *t+1* by 21.0 to 22.4 percentage points (columns 3-4). These are close to the point estimates from the quartic model. This result is rather robust to the choice of bandwidths, as the rows *lwald50* and *lwald200* show. The estimated impact on the net Democratic share in election *t+1* is 7.4 to 7.8 percentage points (columns 1-2). These numbers are actually quite close to those in Table 2 of Lee (2008).

The histogram and the McCrary test (both included in the figures) show no evidence of manipulation at the cutpoint. The test statistic for the McCrary test is .107, with a standard error of 0.080. A rejection of the null hypothesis would suggest a discontinuity at the cutpoint, but in this case we cannot reject the null.

Finally, Table 4 shows the non-parametric RD estimates in which the outcomes are the Democratic vote share in the *previous* election and a binary Democratic win in the previous election. While a small change in the vote share in the current election can produce a large change in the identity of the victor (and, as we have shown, the probability of winning in the following year) there is no reason to think a small change in this year's election would have an effect on the *prior* outcome. Indeed that is what we see here, at least using the optimal bandwidth, where there are no statistically significant effects.

(b) Test for continuity in the relationship between *difdemsharenext* and the running variable by creating 9 dummy variables equal to one if x (the running variable) is greater than the 1st decile of x, greater than the 2nd decile of x, and so on. Then, estimate an OLS regression of *difdemshare* with a quartic in x and these nine dummy variables. (Also include the original indicator of a Democratic win, since we know there is a discontinuity there). Conduct a joint F-test for the significance of these nine dummies, and interpret. (**5 points**)

The code is provided in the attached do-file, and the results are shown below. Both a linear and a quartic model are shown. The *abovej* indicators are equal to one for values of *difdemshare* above specific decile thresholds. These deciles have no practical meaning, so one would not expect to see discontinuous jumps at these points. (Note I have also included the *demwin* threshold, since this correlates with *above40*). A joint *F*-test of

the significance of these thresholds cannot reject the null hypothesis of a zero effect.

```
. reg difdemsharenext difdemshare above* demwin

      Source |       SS           df       MS      Number of obs   =     6,559
-------------+----------------------------------   F(11, 6547)     =     10.15
       Model |  3.03305267        11  .275732061   Prob > F        =    0.0000
    Residual |  177.773476     6,547  .027153425   R-squared       =    0.0168
-------------+----------------------------------   Adj R-squared   =    0.0151
       Total |  180.806529     6,558  .027570376   Root MSE        =    .16478

------------------------------------------------------------------------------
difdemshar~t |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
 difdemshare |   -.027245   .0259887    -1.05   0.295    -.0781913    .0237013
     above10 |   .0129858   .0117359     1.11   0.269    -.0100204     .035992
     above20 |   .0066519   .0097351     0.68   0.494     -.012432    .0257359
     above30 |  -.0048204    .009614    -0.50   0.616     -.023667    .0140263
     above40 |   .0029194   .0163976     0.18   0.859    -.0292252    .0350639
     above50 |  -.0194496    .009988    -1.95   0.052    -.0390294    .0001302
     above60 |  -.0051901   .0096556    -0.54   0.591    -.0241181     .013738
     above70 |   .0046187    .009777     0.47   0.637    -.0145474    .0237848
     above80 |   .0130572    .010726     1.22   0.224    -.0079692    .0340837
     above90 |  -.0122603   .0129665    -0.95   0.344    -.0376788    .0131582
      demwin |   .0678978   .0166981     4.07   0.000      .035164    .1006316
       _cons |  -.0411057   .0172003    -2.39   0.017     -.074824   -.0073874
------------------------------------------------------------------------------

. test above10 above20 above30 above40 above50 above60 above70 above80 above90

 ( 1)  above10 = 0
 ( 2)  above20 = 0
 ( 3)  above30 = 0
 ( 4)  above40 = 0
 ( 5)  above50 = 0
 ( 6)  above60 = 0
 ( 7)  above70 = 0
 ( 8)  above80 = 0
 ( 9)  above90 = 0

       F(  9,  6547) =    1.62
            Prob > F =    0.1045

. reg difdemsharenext ///
>         c.difdemshare##c.difdemshare##c.difdemshare##c.difdemshare##i.demwin ///
>         above*

      Source |       SS           df       MS      Number of obs   =     6,559
-------------+----------------------------------   F(18, 6540)     =      6.63
       Model |  3.24061118        18  .180033954   Prob > F        =    0.0000
    Residual |  177.565918     6,540  .027150752   R-squared       =    0.0179
-------------+----------------------------------   Adj R-squared   =    0.0152
       Total |  180.806529     6,558  .027570376   Root MSE        =    .16477

----------------------------------------------------------------------------------------------
             difdemsharenext |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-----------------------------+----------------------------------------------------------------
                 difdemshare |  -.1611464   .3427799    -0.47   0.638    -.8331069    .5108142

 c.difdemshare#c.difdemshare |  -.1537496   1.476585    -0.10   0.917    -3.048339     2.74084

[Higher order terms dropped for legibility]
```

```
                 1.demwin |   .0648211    .020304     3.19   0.001     .0250185    .1046237

       demwin#c.difdemshare |
                        1  |   .2503272   .4718965     0.53   0.596    -.6747441    1.175399

[Higher order terms dropped for legibility]

                  above10 |   -.010661   .0184461    -0.58   0.563    -.0468215    .0254995
                  above20 |   .0034798   .0173102     0.20   0.841    -.0304537    .0374134
                  above30 |   .0026576   .0171931     0.15   0.877    -.0310465    .0363616
                  above40 |   .0102668   .0220208     0.47   0.641     -.032901    .0534347
                  above50 |  -.0290993   .0190673    -1.53   0.127    -.0664774    .0082787
                  above60 |  -.0121689   .0163824    -0.74   0.458    -.0442837     .019946
                  above70 |   .0055458   .0170026     0.33   0.744    -.0277848    .0388764
                  above80 |   .0296708   .0166851     1.78   0.075    -.0030374    .0623791
                  above90 |   .0127944   .0332424     0.38   0.700    -.0523715    .0779604
                    _cons |  -.0306526   .0436966    -0.70   0.483    -.1163121    .0550069
------------------------------------------------------------------------------------------------

. test above10 above20 above30 above40 above50 above60 above70 above80 above90

 ( 1)  above10 = 0
 ( 2)  above20 = 0
 ( 3)  above30 = 0
 ( 4)  above40 = 0
 ( 5)  above50 = 0
 ( 6)  above60 = 0
 ( 7)  above70 = 0
 ( 8)  above80 = 0
 ( 9)  above90 = 0

       F(  9,  6540) =    0.73
            Prob > F =    0.6803
```

(c) Let's demonstrate to ourselves what `rd` is doing behind the scenes. First, use `rd` to get a non-parametric estimate of the effect of incumbency on *difdemsharenext*. Specifically set the bandwidth to be 0.275. Note the point estimate. Then try the following syntax. (**5 points**)

```
lpoly difdemsharenext difdemshare if difdemshare < 0, deg(1) ker(tri) bwidth(0.275) ///
     gen(L) at(difdemshare) graph
lpoly difdemsharenext difdemshare if difdemshare >= 0, deg(1) ker(tri) bwidth(0.275) ///
     gen(R) at(difdemshare) graph
gen diff = R - L
sum diff if difdemshare==0
drop R L diff
```

Finally, try the syntax below. How does the OLS point estimate below compare to what you found using `rd` and `lpoly`?

```
gen kwt=max(0,0.275-abs(difdemshare))
gen win=difdemshare>0
/* see the triangle kernel */
scatter kwt difdemshare if abs(difdemshare)<=0.275
reg difdemsharenext difdemsahre win [pw=kwt]
```

Results are shown below. Note the point estimate for *difdemsharenext* using the optimal bandwidth (0.275) is 0.074. `lpoly` **fits a local linear regression, here with a triangle kernel. We do this on the left and right-hand side of the cutpoint. L and R are fitted values from this procedure. We then compare the value of L and R at the cutpoint value of 0. The result is the same as the RD point estimate of 0.074. The second set of Stata commands creates a triangle weight *kwt* that gives less weight to values of *difdemshare* away from zero. The OLS regression restricted to the same bandwidth of 0.275 and applying the kernel weight produces a very similar point estimate of 0.0744.**

```
. // ************************************************************************
. // ************************************************************************
. // Checking to see what rd does behind the scenes
.
. rd difdemsharenext difdemshare, z0(0) strineq cluster(statedisdec) bwidth(0.275)
Two variables specified; treatment is
assumed to jump from zero to one at Z=0.

 Assignment variable Z is difdemshare
 Treatment variable X_T unspecified
 Outcome variable y is difdemsharenext

Estimating for bandwidth .275
Estimating for bandwidth .1375
Estimating for bandwidth .55
------------------------------------------------------------------------------
difdemshar~t |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       lwald |   .0741821   .0111595     6.65   0.000       .05231    .0960542
     lwald50 |    .061339    .015595     3.93   0.000     .0307733    .0919047
    lwald200 |   .0631781   .0082062     7.70   0.000     .0470942    .0792621
------------------------------------------------------------------------------

.
. lpoly difdemsharenext difdemshare if difdemshare <= 0, deg(1) ker(tri) ///
>         bwidth(0.275) gen(L) at(difdemshare)
note: label truncated to 80 characters

. lpoly difdemsharenext difdemshare if difdemshare > 0, deg(1) ker(tri) ///
>         bwidth(0.275) gen(R) at(difdemshare)
note: label truncated to 80 characters

. gen diff = R - L
(3,537 missing values generated)

. sum diff if difdemshare==0

    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
        diff |          1    .0741821           .   .0741821   .0741821

. drop R L diff

. gen kwt=max(0,0.275-abs(difdemshare))

. gen win=difdemshare>0

. /* see the triangle kernel */
```

```
. scatter kwt difdemshare if abs(difdemshare)<=0.275

. reg difdemsharenext difdemshare win [pw=kwt]
(sum of wgt is 433.7192835665555)

Linear regression                               Number of obs     =      3,025
                                                F(2, 3022)        =      41.51
                                                Prob > F          =     0.0000
                                                R-squared         =     0.0359
                                                Root MSE          =     .14853

------------------------------------------------------------------------------
             |               Robust
difdemshar~t |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
 difdemshare |  -.1069388   .0426862    -2.51   0.012    -.1906357   -.0232418
         win |   .0744441   .0113069     6.58   0.000     .0522741    .0966141
       _cons |  -.0324489   .0064667    -5.02   0.000    -.0451285   -.0197694
------------------------------------------------------------------------------
```

Question 2. Consider the sharp RD model in which the running variable (x_i) is allowed to have a linear relationship with the outcome (Y_i) that varies on either side of the cutoff (c). Let the treatment status variable $D_i = 1$ whenever $x_i > c$.

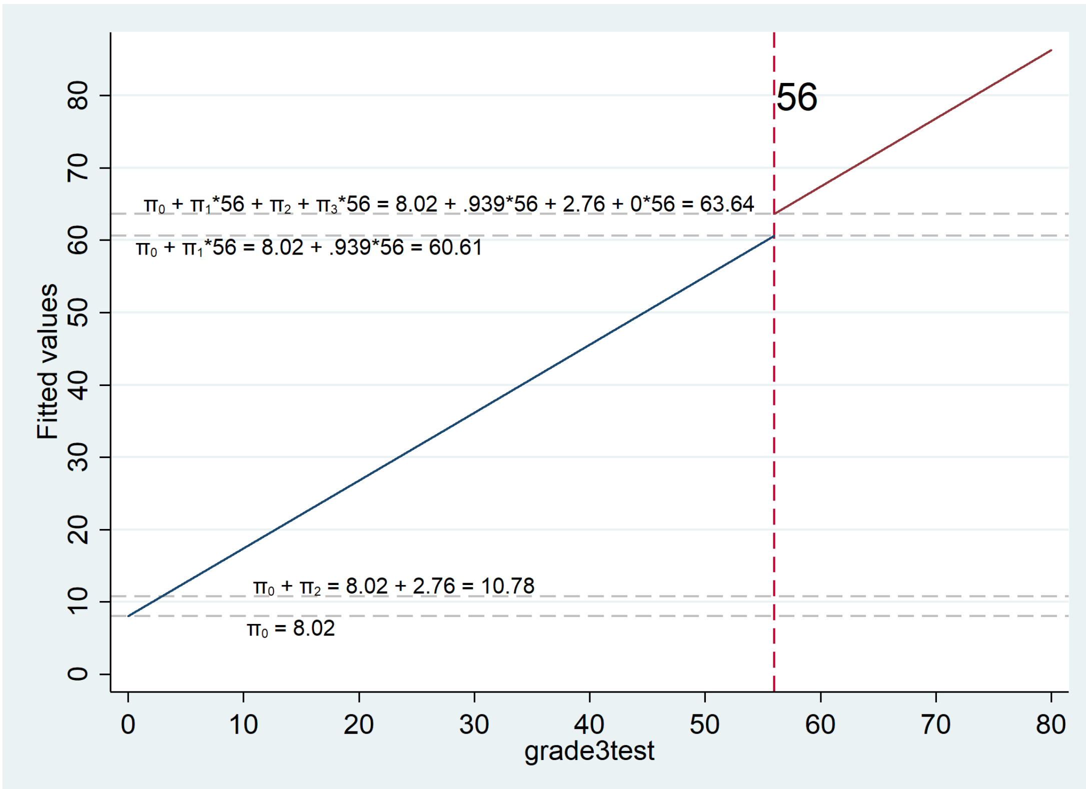
$$Y_i = \pi_0 + \pi_1 x_i + \pi_2 D_i + \pi_3(D_i \times x_i) + v_i$$

Suppose that the running variable x_i is *not* centered at c. (That is, we do not first subtract off c from x_i). Show that π_2 in this case is *not* the impact of the treatment at the threshold c. You can show this however you like: algebraically, using the simulated data from the in-class exercise, or any other valid method. **(6 points)**

Let the cutoff point be c. Intuitively, if $\mathbf{c} \neq 0$ the expected value of $\mathbf{Y_i}$ as we approach c from the left is $\pi_0 + \pi_1 \mathbf{c}$. The expected value of $\mathbf{Y_i}$ as we approach c from the right is $\pi_0 + \pi_1 \mathbf{c} + \pi_2 + \pi_3 \mathbf{c}$. The difference between these two is: $\pi_2 + \pi_3 \mathbf{c}$, not π_2. Put another way, π_2 is the intercept shift when $\mathbf{x} = 0$; $\pi_3 \mathbf{c}$ is the difference in the intercept shift when $\mathbf{x} = \mathbf{c}$. If the cutpoint were 0, the expected value of Y_i as we approach c from the left would be π_0, while the expected value of Y_i as we approach c from the right would be $\pi_0 + \pi_2$.

We can also see this using simulated data from the in-class exercise in which we generated 10,000 student observations with underlying "ability," a grade 3 test score, and a grade 4 test score. Students above the eligibility threshold for the gifted program (56) were assigned to the gifted treatment. When we fit an RD model in which the running variable (grade 3 score) is centered at 0, the coefficient on *inGT* (being at or above the treatment threshold) was 3.02. If we fit the model (using the same data) with a running variable *not* centered at 0, the coefficient on *inGT* is 2.76. To get the actual jump at the cutoff we would

need to calculate $\hat{\pi}_2 + \hat{\pi}_3 * 56 = 2.76 + 0.0046 * 56 = 3.02$. The Stata output and graph is shown below. The syntax for the graph is shown for your reference.




```
// grade3test centered at 0 (gap)
. reg grade4test c.gap##i.inGT

      Source |       SS           df       MS      Number of obs   =    10,000
-------------+----------------------------------   F(3, 9996)      =  32218.65
       Model |  187161.733         3  62387.2445   Prob > F        =    0.0000
    Residual |  19355.9606     9,996   1.9363706   R-squared       =    0.9063
-------------+----------------------------------   Adj R-squared   =    0.9062
       Total |  206517.694     9,999  20.6538348   Root MSE        =    1.3915

------------------------------------------------------------------------------
  grade4test |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         gap |    .939221   .0040802   230.19   0.000     .9312229    .9472191
      1.inGT |    3.02236   .0798731    37.84   0.000     2.865792    3.178927
             |
   inGT#c.gap |
          1  |   .0046223   .0313781     0.15   0.883    -.0568851    .0661296
             |
       _cons |   60.61455   .0306168  1979.78   0.000     60.55453    60.67456
```

```
------------------------------------------------------------------------------

// RD model using non-centered grade 3 test
. reg grade4test c.grade3test##i.inGT

      Source |       SS           df       MS      Number of obs   =    10,000
-------------+----------------------------------   F(3, 9996)      =  32218.65
       Model |  187161.733         3  62387.2445   Prob > F        =    0.0000
    Residual |  19355.9606     9,996   1.9363706   R-squared       =    0.9063
-------------+----------------------------------   Adj R-squared   =    0.9062
       Total |  206517.694     9,999  20.6538348   Root MSE        =    1.3915

-----------------------------------------------------------------------------------
       grade4test |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
------------------+----------------------------------------------------------------
       grade3test |    .939221   .0040802   230.19   0.000     .9312229    .9472191
           1.inGT |   2.763511   1.808055     1.53   0.126    -.7806416    6.307664
                  |
inGT#c.grade3test |
              1   |   .0046223   .0313781     0.15   0.883    -.0568851    .0661296
                  |
            _cons |   8.018173   .2020343    39.69   0.000     7.622145    8.414201
-----------------------------------------------------------------------------------

// Graph
capture drop yhat1

reg grade4test c.grade3test##i.inGT
predict yhat1, xb
matrix temp1=e(b)
local pi0 = temp1[1,6]
local pi1 = temp1[1,1]
local pi2 = temp1[1,3]
local pi3 = temp1[1,5]
// rounded for graph labels
local pi0r = round(‘pi0’,0.01)
local pi1r = 0.939 /* had some formatting issues so directly plugged this in */
local pi2r = round(‘pi2’,0.01)
local pi3r = round(‘pi3’,0.01)
// when x=0 inGT=1
local hat1 = ‘pi0’ + ‘pi2’
local hat1r= round(‘hat1’,0.01)
// when x=56 inGT=0
local hat2 = ‘pi0’ + ‘pi1’*56
local hat2r= round(‘hat2’,0.01)
// when x=56 inGT=1
local hat3 = ‘pi0’ + ‘pi1’*56 + ‘pi2’ + (‘pi3’*56)
local hat3r= round(‘hat3’,0.01)

// added text details
local tsize "small"
```

```
local t0=`pi0'-1.5
local t1=`hat1'+1.5
local t2=`hat2'-1.5
local t3=`hat3'+1.5

twoway (lfit yhat1 grade3test if grade3test<56, range(0 56)) ///
   (lfit yhat1 grade3test if grade3test>=56, range(56 80)), ///
   legend(off) xline(56, lpattern(dash)) ///
   yline(`pi0', lpattern(dash) lcolor(gs12)) ///
     text(`t0' 10 "{&pi}{sub:0} = `pi0r'", placement(right) size(`tsize')) ///
   yline(`hat1', lpattern(dash) lcolor(gs12)) ///
     text(`t1' 10 "{&pi}{sub:0} + {&pi}{sub:2} = `pi0r' + `pi2r' = `hat1r'", ///
     placement(right) size(`tsize')) ///
   yline(`hat2', lpattern(dash) lcolor(gs12)) ///
     text(`t2' -0.3 "{&pi}{sub:0} + {&pi}{sub:1}*56 = `pi0r' + `pi1r'*56 = `hat2r'", ///
     placement(right) size(`tsize')) ///
   yline(`hat3', lpattern(dash) lcolor(gs12)) ///
     text(`t3' -0.3 "{&pi}{sub:0} + {&pi}{sub:1}*56 + {&pi}{sub:2} + {&pi}{sub:3}*56 = ///
     `pi0r' + `pi1r'*56 + `pi2r' + `pi3r'*56 = `hat3r'", ///
     placement(right) size(`tsize')) ///
   ylabel(0(10)80) xlabel(0(10)80) text(80 58 "56", size(large))
```