NetSense Ego Network Degree Report

David Hachen and Omar Lizardo

January 27, 2013

1 Introduction

This report focuses on the correlated of the size of the networks of our 190 or so NetSense participants after approx. 60 weeks. Their network size is defined by the number of different people with who them communicated by text (or voice) during the first 60 weeks of the study. Degree is determined by any communicative interaction irrespective of number of times and direction of interaction (i.e. who initiated it)

Five variables are used:

1. *D_PostND* Degree as determined from all the different ways we used to identify whether alter is a ND student. See earlier report on this.
2. *D_PreND* Count of number of alters ego interacted with who are identifies as people ego knew prior to ND (and not family)
3. *D_Family* Could of number of alters interacted with who are identified by ego as family members
4. *D_unk* Count of number of alters ego communicated with that we can not identify (unknown)
5. *D_Full*: The full degree, sum of the 4 above variables.

2 Data

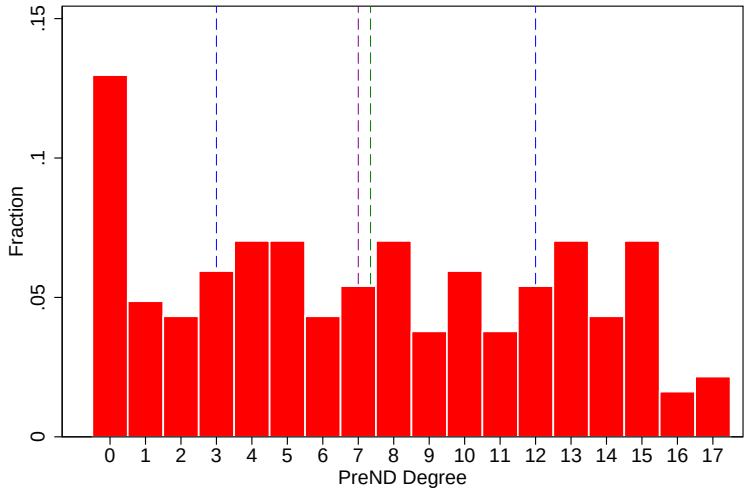
I merged survey data from Wave 1 (SurveyWave1Only.dta) with node level data I created through NodeXL and merging files to get an index of the

size of each student's network as note above (file NodeDegreeAfter60.dta). The creation of these two files and their merging are described in MergeNodeSurvey.do. All files are in by Box directory I am sharing with you – Box Documents/NetSenseFolder/NetSenseLOCAL/EgoNetDegreeAnalysis

2.1 Univariate Degree Distributions

Graphing univariate degree distributions is the most informative way to gain initial understanding of a given variable. Here we graph the univariate distribution of all five degree variables.

In Stata, the primary graphing command for displaying univariate distributions (as a histogram) is `hist`. The best way to construct an informative histogram is to graph the variable according to its type. The `hist` command treats all variables as continuous by default. But degree is a discrete (count) variable, so we must specify that option. In addition, and informative univariate display should contain information about the variable's main location (mean, median) and scale (range, standard deviation) statistics. We can add that to a Stata graph to make it more informative, by first summarizing the variable (using the `details` option) and then using the information stored as macros in `r()`. Here's the univariate distribution of Pre-ND degree:



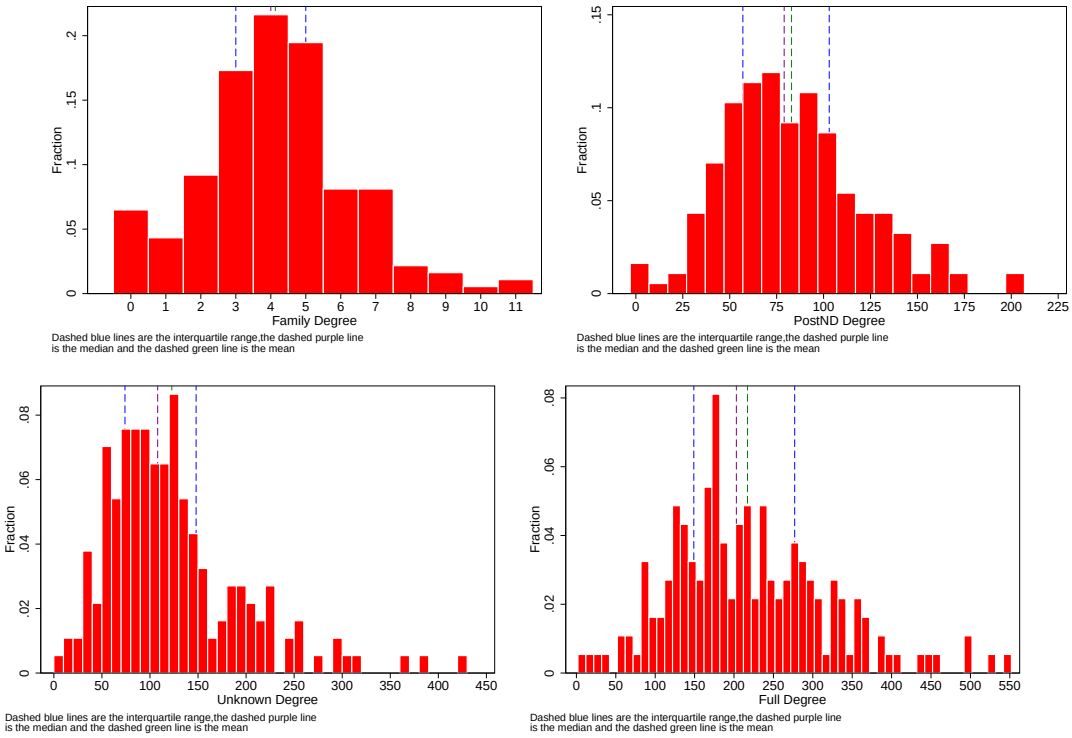
Dashed blue lines are the interquartile range,the dashed purple line is the median and the dashed green line is the mean

And the code to generate this graph is:

```
1   qui sum D_Pre,det
```

```
2   hist D_Pre,disc frac ///
3   bc(black) bfc(red) ///
4   xlab(`r(min)'(1)`r(max)') ///
5   xline(`r(p25)' `r(p75)', lw(vthin) lp(dash) lc(blue)) ///
6   xline(`r(p50)', lw(vthin) lp(dash) lc(purple)) ///
7   xline(`r(mean)', lw(vthin) lp(dash) lc(green)) ///
8   note("Dashed blue lines are the interquartile range,the dashed purpl
9   "is the median and the dashed green line is the mean", span)
```

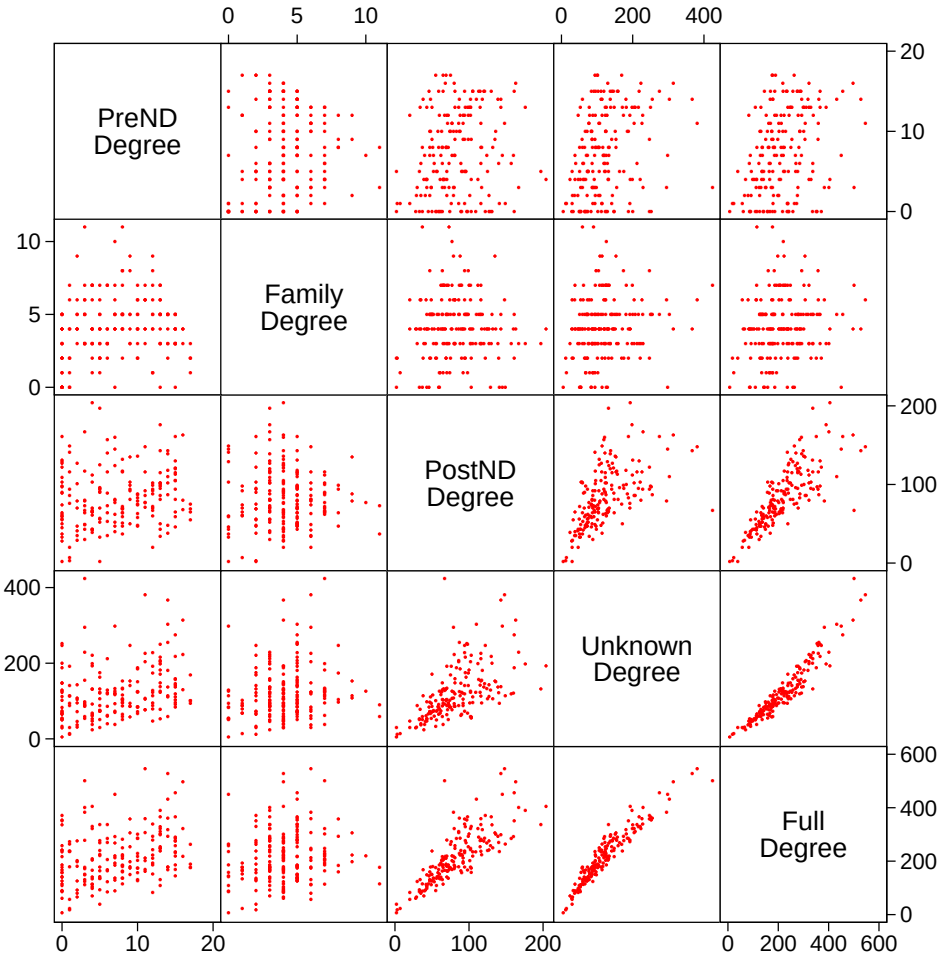
We can then generate similar graphs for the other four degree variables, simply by plugging in that variable where we have "D_Pre" above:



2.2 Matrix Graph

We may also be interested not just in the univariate distribution of the various degree variables, but/ in their *intercorrelation* with one another. The standard way of doing this is simply to look at a correlation matrix, which in Stata is just the command `corr` followed by a list of variables (if you

want the *rank order* correlation then the command is `spearman`). However, we can present the same information as is conveyed in a correlation matrix in graphical form, using the `graph matrix` command. This is a rather under-utilized graphing strategy, but it conveys more information than the simple correlation matrix and it does in powerful form. Here's the matrix graph of the degree variables:

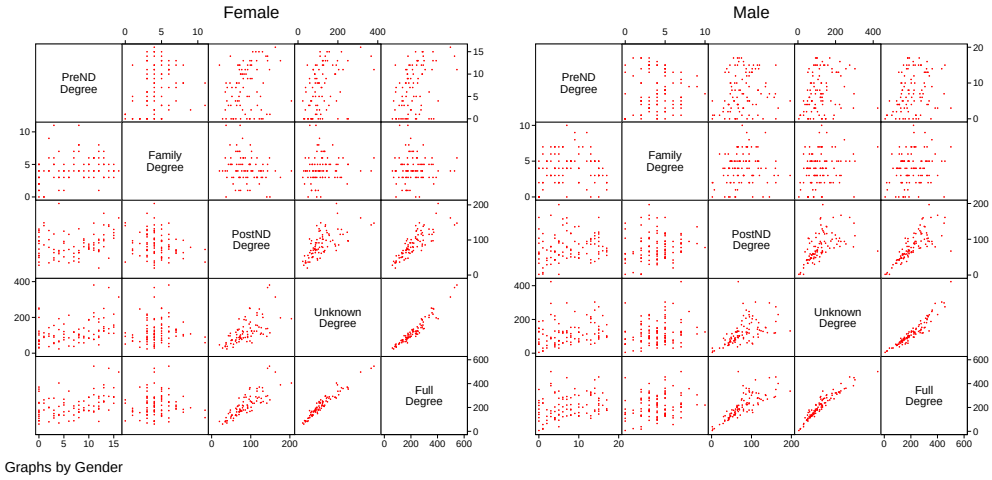


The Stata code for this graph is just two lines:

```
1  graph matrix D_Pre D_Family D_Post D_unk D_Full, ///
2  msym(plus) msize(vsmall) mc(red) ysize(6) xsize(6)
```

Adding the `by(varlist)` option to the graph matrix command allows

us to generate matrix graphs for subgroup of respondents in the data (split by `varlist`). This is the matrix graph for boys and girls:



And the code:

```
1  graph matrix D_Pre D_Family D_Post D_unk D_Full, ///
2  msym(plus) msize(vsmall) mc(red) ysize(6) xsize(6) ///
3  by(gender_1)
```

3 Mean differences Across Groups

3.1 Race

After examining the univariate distribution of the degree, we may be interested in average differences subgroups (race, gender, etc.) on these variables. The bivariate regression of a given degree variable (d_i^k) against a series of $(N-1)$ dummy variables (where N is the number of groups) gives us the ANOVA decomposition of means. Let us say that we are interested in race differences in average (post-Notre Dame) degree and that the race variable is called `race_1`. One way to do this in Stata is simply to type: `regress D_Post i.race_1`.

And we get output that looks like this:

```
      Source |       SS       df       MS              Number of obs =     183
-------------+------------------------------           F(  4,   178) =    2.65
       Model |  14440.0545     4  3610.01362            Prob > F      =  0.0351
```

```
    Residual |  242842.252   178  1364.28231           R-squared     =  0.0561
-------------+----------------------------           Adj R-squared =  0.0349
       Total |  257282.306   182  1413.63904           Root MSE      =  36.936

------------------------------------------------------------------------------
      D_Post |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      race_1 |
          2  |   29.01117   10.76787     2.69   0.008     7.762053    50.26028
          3  |   13.29457   9.099814     1.46   0.146    -4.662831    31.25197
          4  |   15.78739   8.544888     1.85   0.066    -1.074928    32.64971
          5  |   3.441935   16.84811     0.20   0.838     -29.8058    36.68967
             |
       _cons |   77.75806   3.316968    23.44   0.000     71.21242    84.30371
------------------------------------------------------------------------------
```

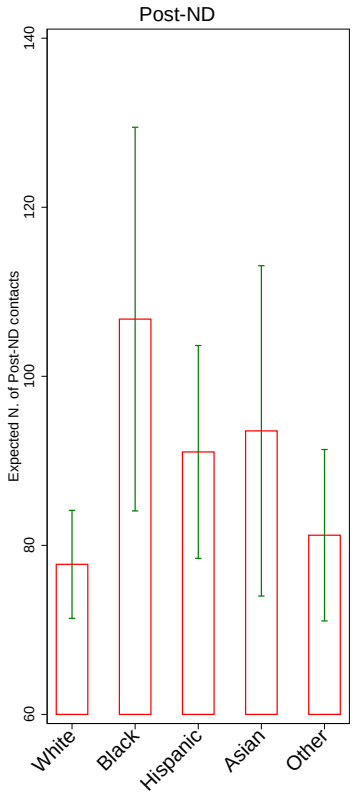
The coefficient estimate for each dummy variable is the difference between the average number of post-ND friends for the "reference" (omitted) category (represented by the constant) and that group (in this case White = 1 is the omitted category, and Black = 2, Hispanic = 3, Asian = 4 and "Other" = 5). So the output is telling us that Whites have on average 78 post-ND friends, but that on average Blacks have 29 more friends than that (Black average is $78 + 29 = 107$). And so on.

All of the information is in the table, but tables are cumbersome to read and we have to add and subtract. Ideally we would like to present all of this information in a graph not a table. We an do this with the Stata `margins` command, followed by `marginsplot`.

The relevant code looks like this:

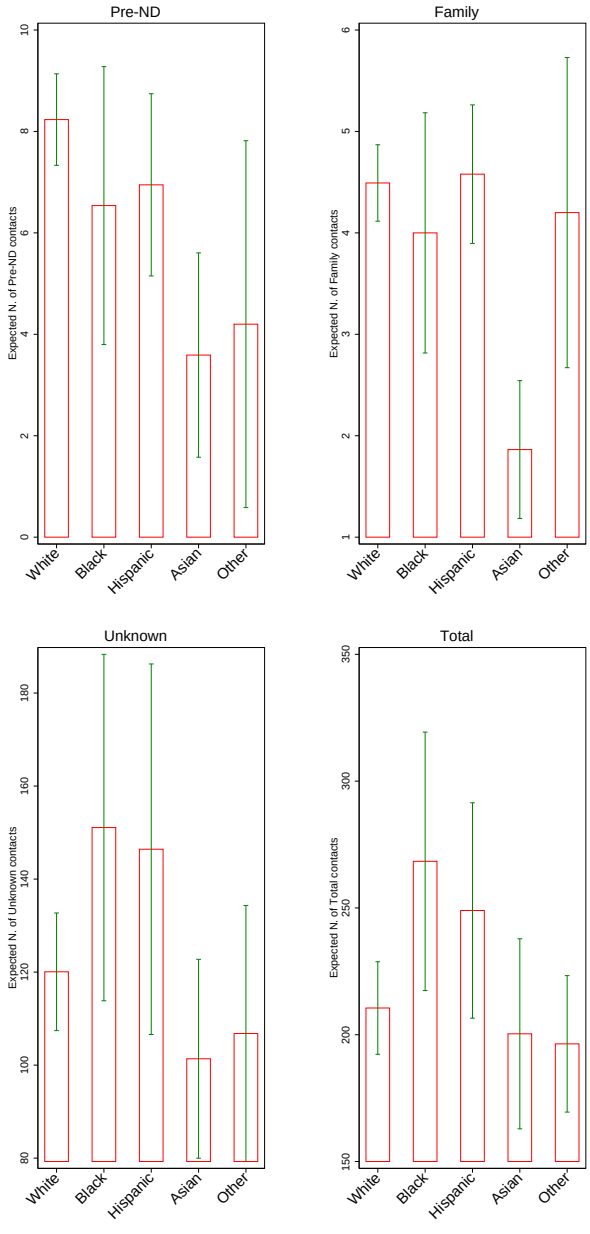
```
1  regress D_Post i.race_1,robust
2  margins race_1
3  marginsplot, ///
4  ciopts(lc(vthin) lc(green)) ///
5  plotopts(recast(bar) fc(none) bc(red) barw(0.5)) ///
6  xlab(,angle(45) labs(large)) ///
7  graphr(margin(l + 10)) ///
8  ysize(6) xsize(3) title("Post--ND") ///
9  ytitle("Expected N. of Post_ND contacts") xtitle("")
```

And the resulting graph that looks like this:



Line 1 runs the regression, line 2 computes the margins (essentially the mean differences in post-ND degree across groups and generates posterior confidence intervals.Lines 3-8 draw the plot. Note that the graph conveys the same information as the (ugly) regression table (Black ND students have on average more friends than White students) and this difference is within conventional levels of statistical significance (the lower bound estimate for Blacks is higher than the upper bound estimate for whites at the 0.05 level).

We can use the same approach to look at race differences across the other degree variables:



To expedite things, we can use the Stata `foreach` command to loop through all of the degree variables and generate the graphs without having to type the code shown above multiple times:

```
1   local titles Pre-ND Family Post-ND Unknown Total
```

```
2  local i = 1
3  foreach y of varlist D_Pre D_Family D_Post D_unk D_Full {
4     local title : word `i' of `titles'
5     qui regress `y' i.race_1,robust
6     margins race_1
7     marginsplot, ///
8     ciopts(lc(vthin) lc(green)) ///
9     plotopts(recast(bar) fc(none) bc(red) barw(0.5)) ///
10    xlab(,angle(45) labs(large)) ///
11    graphr(margin(l + 10)) ///
12    ysize(6) xsize(3) title("`title'",size(large)) ///
13    ytitle("Expected N. of `title' contacts") xtitle("")
14    local ++i
15    cd "M:\OmarWork\degree-report"
16    graph export `y'_by_race.pdf,as(pdf) replace
17    }
```

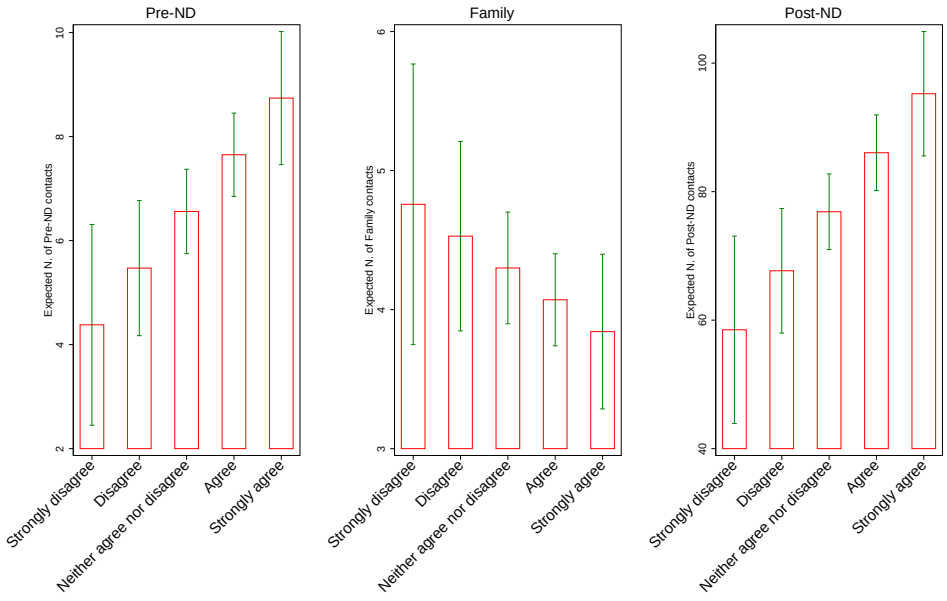
3.2 Extraversion

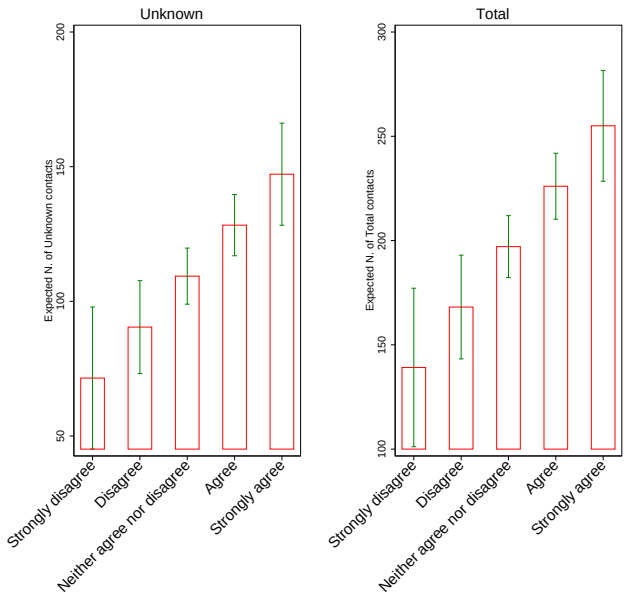
We can use the same approach to look at cross-group differences based on other individual variables. Let us look at "extraversion." Students were asked they Agreed or Disagreed with the statement "I am someone who is outgoing" (five point Likert scale from "Strongly Disagree" to "Agree" to "Neither Agree nor Disagree" to "Agree" to "Strongly Agree"). We use a similar approach except that here we take advantage of the ordered nature of the variable. Instead of running our ANOVA regression by presuming that the factor variable is nominal (indicated by the `i.` prefix), we presume that it is ordered (indicated by the `c.` prefix). This requires a small modification of the margins call, but everything else is the same:

```
1  local titles Pre-ND Family Post-ND Unknown Total
2  local i = 1
3  foreach y of varlist D_Pre D_Family D_Post D_unk D_Full {
4     local title : word `i' of `titles'
5     qui regress `y' c.iamsomeonewho36_1,robust
6     margins,at(iamsomeonewho36_1 = (1(1)5))
7     marginsplot, ///
8     ciopts(lc(vthin) lc(green)) ///
```

```
    plotopts(recast(bar) fc(none) bc(red) barw(0.5)) ///
    xlab(,angle(45) labs(large)) ///
    graphr(margin(l + 10)) ///
    ysize(6) xsize(3) title("`title'",size(large)) ///
    ytitle("Expected N. of `title' contacts") xtitle("")
    local ++i
    cd "M:\OmarWork\degree-report"
    graph export `y'_by_extroversion.pdf,as(pdf) replace
    }
```

Here are the resulting graphs:




Unknown
Expected N. of Unknown contacts
50
100
150
200
Strongly disagree
Disagree
Neither agree nor disagree
Agree
Strongly agree
Total
Expected N. of Total contacts
100
150
200
250
300
Strongly disagree
Disagree
Neither agree nor disagree
Agree
Strongly agree

References