Quant I: Fall 2013
Prof. Patrick Egan

Tabular and Graphical Methods for Displaying and Summarizing Relationships Between Two Variables (along with some other related remarks)

I. DISPLAYING the Relationship

a. Crosstab(ulation)s:

- **Y is typically displayed by row; X by column**

```
. tab faminc educyears
```

Household Income, 2004	Years of Education								Total
	0	2.5	5.5	7.5	9	10	11	11.5	
2500	2	0	4	7	4	12	11	2	134
6250	2	6	5	6	7	13	9	0	112
8250	0	3	4	16	5	8	9	2	120
11500	0	2	5	10	11	9	16	5	165
13750	0	1	3	10	13	7	8	0	140
17500	1	5	6	14	17	21	14	6	232
22500	1	2	7	15	21	14	19	5	326
27500	1	1	4	16	14	16	21	3	330
32500	4	2	8	9	19	20	11	3	340
37500	2	3	4	9	12	12	17	3	324
45000	2	1	6	12	20	16	22	14	551
55000	3	4	2	9	12	20	17	8	501
67500	0	0	0	8	12	14	11	13	621
87500	0	0	1	8	16	19	21	2	634
125000	0	0	1	6	10	13	16	2	470
Total	18	30	60	155	193	214	222	68	5,000

Household Income, 2004	Years of Education						Total
	12	13	14	16	18	22	
2500	40	27	7	13	5	0	134
6250	31	27	1	4	1	0	112
8250	48	9	6	9	1	0	120
11500	65	22	9	9	2	0	165
13750	62	24	7	5	0	0	140
17500	105	20	6	16	1	0	232
22500	120	54	22	36	9	1	326
27500	124	63	28	34	5	0	330
32500	115	62	28	46	12	1	340
37500	131	56	25	36	11	3	324
45000	185	96	52	81	40	4	551
55000	150	110	47	84	32	3	501
67500	185	128	65	131	50	4	621
87500	140	116	53	155	93	10	634
125000	67	85	47	138	68	17	470
Total	1,568	899	403	797	330	43	5,000

• Often more informative to provide column percents. Note that convention is typically to provide *column*, not *row*, percents. Include 100s at the bottom so reader can quickly see that columns add up to 100:

```
. tab faminc educyears, col nofr
```

Household Income, 2004	Years of Education								Total
	0	2.5	5.5	7.5	9	10	11	11.5	
2500	11.11	0.00	6.67	4.52	2.07	5.61	4.95	2.94	2.68
6250	11.11	20.00	8.33	3.87	3.63	6.07	4.05	0.00	2.24
8250	0.00	10.00	6.67	10.32	2.59	3.74	4.05	2.94	2.40
11500	0.00	6.67	8.33	6.45	5.70	4.21	7.21	7.35	3.30
13750	0.00	3.33	5.00	6.45	6.74	3.27	3.60	0.00	2.80
17500	5.56	16.67	10.00	9.03	8.81	9.81	6.31	8.82	4.64
22500	5.56	6.67	11.67	9.68	10.88	6.54	8.56	7.35	6.52
27500	5.56	3.33	6.67	10.32	7.25	7.48	9.46	4.41	6.60
32500	22.22	6.67	13.33	5.81	9.84	9.35	4.95	4.41	6.80
37500	11.11	10.00	6.67	5.81	6.22	5.61	7.66	4.41	6.48
45000	11.11	3.33	10.00	7.74	10.36	7.48	9.91	20.59	11.02
55000	16.67	13.33	3.33	5.81	6.22	9.35	7.66	11.76	10.02
67500	0.00	0.00	0.00	5.16	6.22	6.54	4.95	19.12	12.42
87500	0.00	0.00	1.67	5.16	8.29	8.88	9.46	2.94	12.68
125000	0.00	0.00	1.67	3.87	5.18	6.07	7.21	2.94	9.40
Total	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00

Household Income, 2004	Years of Education						Total
	12	13	14	16	18	22	
2500	2.55	3.00	1.74	1.63	1.52	0.00	2.68
6250	1.98	3.00	0.25	0.50	0.30	0.00	2.24
8250	3.06	1.00	1.49	1.13	0.30	0.00	2.40
11500	4.15	2.45	2.23	1.13	0.61	0.00	3.30
13750	3.95	2.67	1.74	0.63	0.00	0.00	2.80
17500	6.70	2.22	1.49	2.01	0.30	0.00	4.64
22500	7.65	6.01	5.46	4.52	2.73	2.33	6.52
27500	7.91	7.01	6.95	4.27	1.52	0.00	6.60
32500	7.33	6.90	6.95	5.77	3.64	2.33	6.80
37500	8.35	6.23	6.20	4.52	3.33	6.98	6.48
45000	11.80	10.68	12.90	10.16	12.12	9.30	11.02
55000	9.57	12.24	11.66	10.54	9.70	6.98	10.02
67500	11.80	14.24	16.13	16.44	15.15	9.30	12.42
87500	8.93	12.90	13.15	19.45	28.18	23.26	12.68
125000	4.27	9.45	11.66	17.31	20.61	39.53	9.40
Total	100.00	100.00	100.00	100.00	100.00	100.00	100.00

- **Lots of categories of both X and Y, so maybe you recode:**

```
. gen educ_rc = educyears

. recode educ_rc (0/7.5 =1) (9/11.5=2) (12=3) (13/14=4) (16=5) (18/22=6)
(educ_rc: 5000 changes made)

. label def educ_rc  1 "< 8th grade" 2 "< H.S" 3 "HS Diploma" 4 "Some college" 5 "B.A."
6 "Post-grad"

. label values educ_rc educ_rc

. tab educ_rc

     educ_rc |      Freq.     Percent        Cum.
-------------+-----------------------------------
 < 8th grade |        263        5.26        5.26
       < H.S |        697       13.94       19.20
  HS Diploma |      1,568       31.36       50.56
Some college |      1,302       26.04       76.60
        B.A. |        797       15.94       92.54
   Post-grad |        373        7.46      100.00
-------------+-----------------------------------
       Total |      5,000      100.00


. gen faminc_rc = famincome

. recode faminc_rc (0/20000 = 1) (21000/40000=2) (41000/70000=3) (71000/max=4)
(faminc_rc: 5000 changes made)

. label def faminc_rc 1 "<$20K" 2 "$21-$40K" 3 "$41-$70K" 4 ">$70K"

. label values faminc_rc faminc_rc

. tab faminc_rc

  faminc_rc |      Freq.     Percent        Cum.
------------+-----------------------------------
      <$20K |        903       18.06       18.06
   $21-$40K |      1,320       26.40       44.46
   $41-$70K |      1,673       33.46       77.92
      >$70K |      1,104       22.08      100.00
------------+-----------------------------------
      Total |      5,000      100.00
```

• This crosstab is now easier to read (though of course we've lost some information).

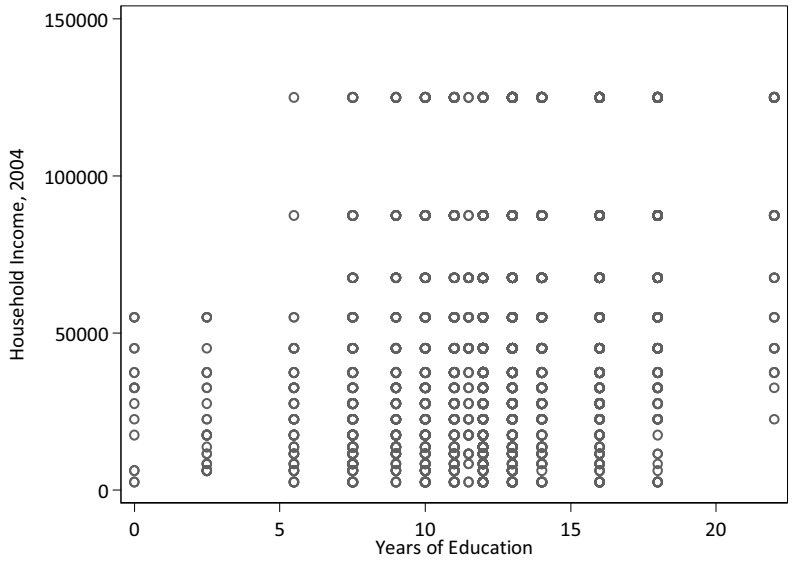
• To detect a relationship between X and Y, see how the column percents change within rows (note how it's now easier to do this than it was on the bigger crosstab earlier):

```
. tab faminc_rc educ_rc, col nofr
```

faminc_rc	educ_rc						
	< 8th gra	< H.S	HS Diplom	Some coll	B.A.	Post-grad	Total
<$20K	42.59	29.99	22.39	12.67	7.03	2.68	18.06
$21-$40K	33.46	30.13	31.25	25.96	19.07	11.26	26.40
$41-$70K	17.87	25.68	33.16	38.25	37.14	35.66	33.46
>$70K	6.08	14.20	13.20	23.12	36.76	50.40	22.08
Total	100.00	100.00	100.00	100.00	100.00	100.00	100.00

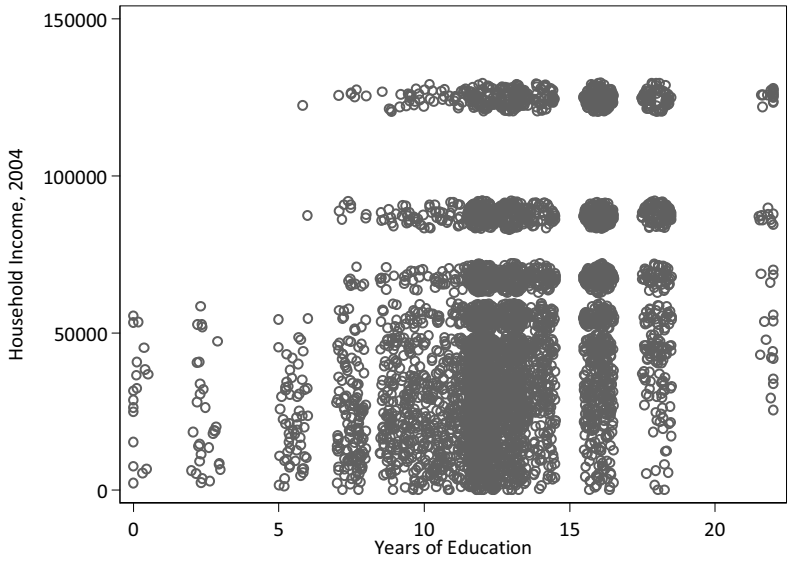
b. <u>Scatterplots</u> (both variables at interval-level or higher):

```
. twoway (scatter famincome educyears)
```



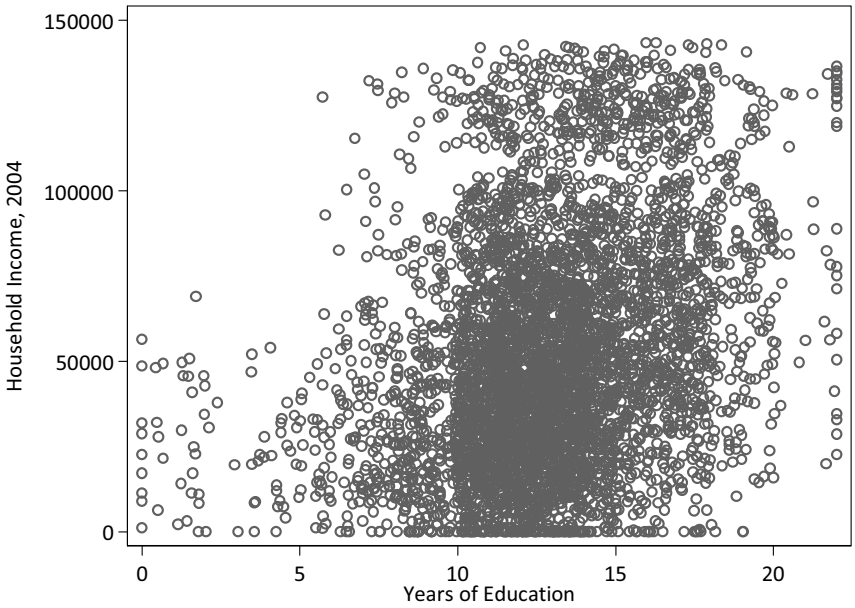
Not very informative, so let's jitter, adding noise to each marker that is equivalent to 5% of area of graph:

```
twoway (scatter famincome educyears, jitter(5))
```

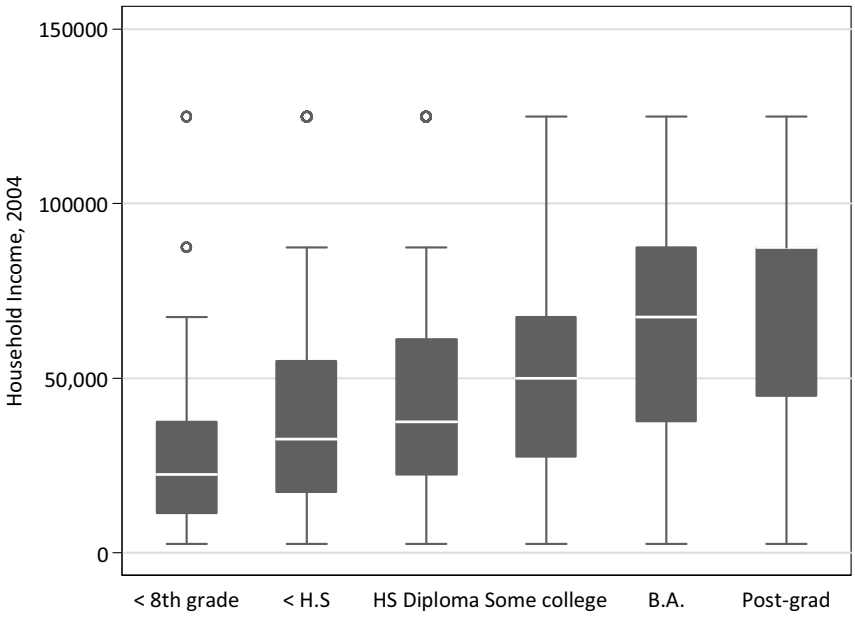


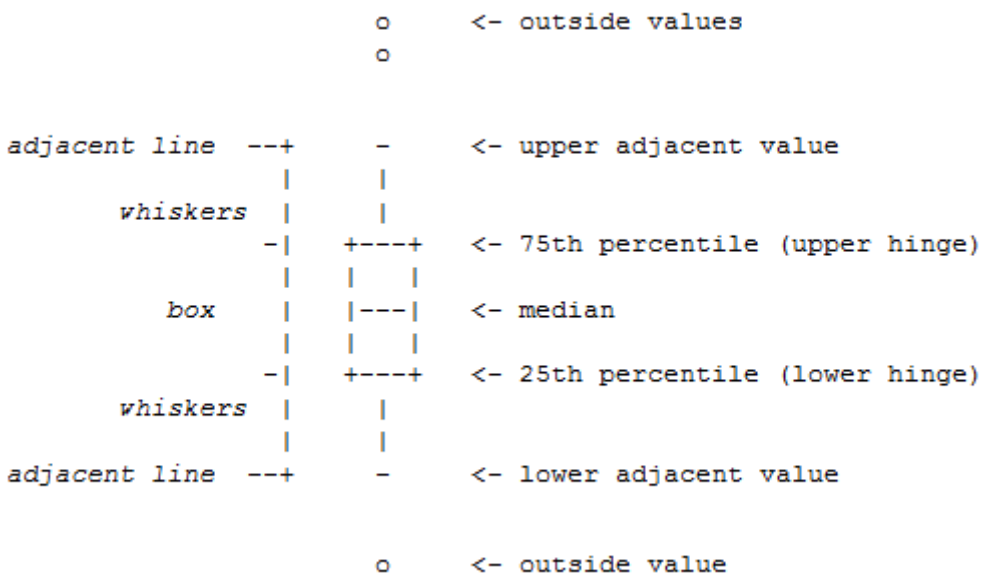
Still not so good, let's try 20%:

```
twoway (scatter famincome educyears, jitter(20))
```



c. <u>Boxplots</u> (appropriate when Y is at interval-level, and X has few categories and/or is less than interval level):





OK, but what's an "adjacent value"? From Stata manual:

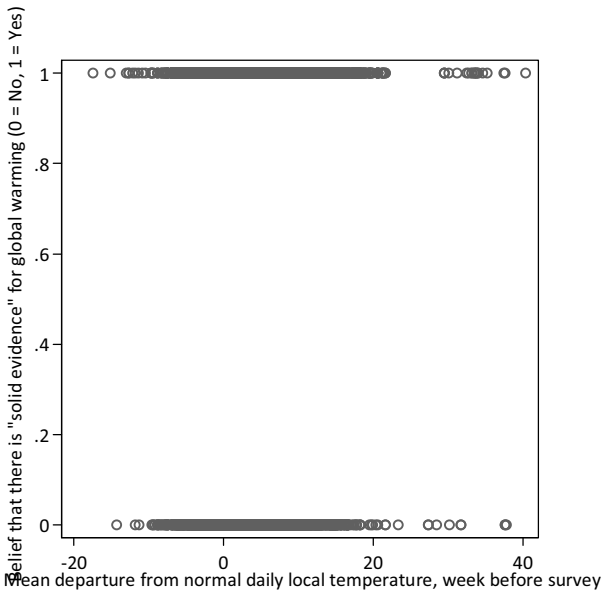
> The upper and lower adjacent values are as defined by Tukey (1977):
>
> Let x represent a variable for which adjacent values are being calculated. Define $x_{(i)}$ as the ith ordered value of x, and define $x_{[25]}$ and $x_{[75]}$ as the 25th and 75th percentiles.
>
> Define U as $x_{[75]} + \frac{3}{2}(x_{[75]} - x_{[25]})$. The upper adjacent value is defined as x_i, such that $x_{(i)} \leq U$ and $x_{(i+1)} > U$.
>
> Define L as $x_{[25]} - \frac{3}{2}(x_{[75]} - x_{[25]})$. The lower adjacent value is defined as x_i, such that $x_{(i)} \geq L$ and $x_{(i-1)} < L$.

d. "Binning out" X and then displaying the mean Y in each "bin" of X. Good when X is continuous/takes on many values but Y is dichotomous. From Egan & Mullin (2012):

```
. sum getwarm01, d

      Belief that there is "solid evidence" for global
                  warming (0 = No, 1 = Yes)
-------------------------------------------------------------
      Percentiles      Smallest
 1%           0              0
 5%           0              0
10%           0              0       Obs                7224
25%           0              0       Sum of Wgt.        7224

50%           1                      Mean            .732835
                        Largest       Std. Dev.      .4425099
75%           1              1
90%           1              1       Variance        .195815
95%           1              1       Skewness       -1.052411
99%           1              1       Kurtosis        2.107569

. sum temp, d

          Mean departure from normal daily local
              temperature, week before survey
-------------------------------------------------------------
      Percentiles      Smallest
 1%          -7      -17.42857
 5%    -4.285714      -15.14286
10%    -2.714286      -14.28571       Obs                7971
25%    -.1428571            -13       Sum of Wgt.        7971

50%     2.571429                      Mean           3.536448
                        Largest       Std. Dev.      5.728437
75%     6.285714       37.57143
90%     11.57143       37.57143       Variance         32.815
95%     14.28571       37.71429       Skewness       .9993781
99%     18.28572       40.28571       Kurtosis       5.503608
```

Here, a scatterplot is virtually worthless:

```
twoway scatter getwarm01 temp, aspect(1)
```



So we "bin out" our X (temp), creating equal-sized bins of temp. Do-file below does this for any x (called 'x-var' in the do-file)

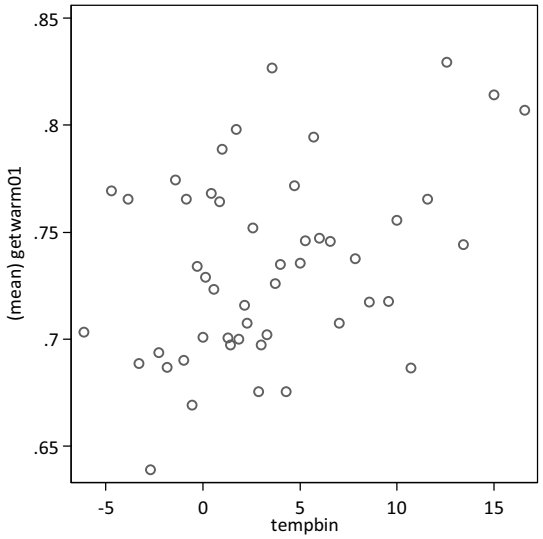
```
*this do-file creates a user-specified total # of "bins" of the variable xvar, each
with an approximately equal number of observations.  the bins are found in a new
variable named the (name of your xvar) + suffix "bin".

*user fills in following three local macros:
local xvar xvargoeshere
local yvar yvargoeshere
local bins numberofbinsdesiredgoeshere

local binsminus1 = `bins'-1
local binsminus2 = `bins'-2


_pctile `xvar', n(`bins')

gen `xvar'bin = .

replace `xvar'bin = 0 if `xvar' <= `r(r1)'

forvalues i = 1(1)`binsminus2' {
       local j = `i'+1
       replace `xvar'bin = `r(r`i')' if `xvar' > `r(r`i')' & `xvar' <= `r(r`j')'
       }

replace `xvar'bin = `r(r`binsminus1')' if `xvar' > `r(r`binsminus1')'

replace `xvar'bin = . if `xvar'==.

save, replace


*now if you wanted to create the dataset you'd use to create the graph (where means of
yvar are plotted against the bin values), you might do the following:

collapse (mean) `yvar', by(`xvar'bin race)

save bindata, replace

twoway (scatter `yvar' `xvar'bin) (lowess `yvar' `xvar'bin)

*etc.
```

Here, I've created 50 bins of temp. Note that because of lumpiness of data they don't all have exactly equal numbers of obs. Not a big deal for our purposes.

tempbin	Freq.	Percent	Cum.
-6.142857	165	2.07	2.07
-4.714286	184	2.31	4.38
-3.857143	90	1.13	5.51
-3.285714	242	3.04	8.54
-2.714286	94	1.18	9.72
-2.285714	153	1.92	11.64
-1.857143	149	1.87	13.51
-1.428571	256	3.21	16.72
-1	98	1.23	17.95
-.8571429	174	2.18	20.14
-.5714286	172	2.16	22.29
-.2857143	205	2.57	24.87
0	277	3.48	28.34
.1428571	197	2.47	30.81
.4285714	106	1.33	32.14
.5714286	232	2.91	35.05
.8571429	126	1.58	36.63
1	108	1.35	37.99
1.271429	206	2.58	40.57
1.428571	176	2.21	42.78
1.714286	121	1.52	44.30
1.857143	217	2.72	47.02
2.142857	99	1.24	48.26
2.285714	97	1.22	49.48
2.571429	286	3.59	53.07
2.857143	91	1.14	54.21
3	164	2.06	56.27
3.285714	112	1.41	57.67
3.571429	93	1.17	58.84
3.714286	254	3.19	62.02
4	182	2.28	64.31
4.285714	181	2.27	66.58
4.714286	140	1.76	68.34
5	133	1.67	70.00
5.285714	198	2.48	72.49
5.714286	125	1.57	74.06
6	189	2.37	76.43
6.571429	126	1.58	78.01
7.033163	168	2.11	80.12
7.857143	157	1.97	82.09
8.571428	187	2.35	84.43
9.571428	134	1.68	86.11
10	134	1.68	87.79
10.71429	188	2.36	90.15
11.57143	180	2.26	92.41
12.57143	130	1.63	94.04
13.42857	176	2.21	96.25
15	156	1.96	98.21
16.57143	143	1.79	100.00
Total	7,971	100.00	

Now generate new file:

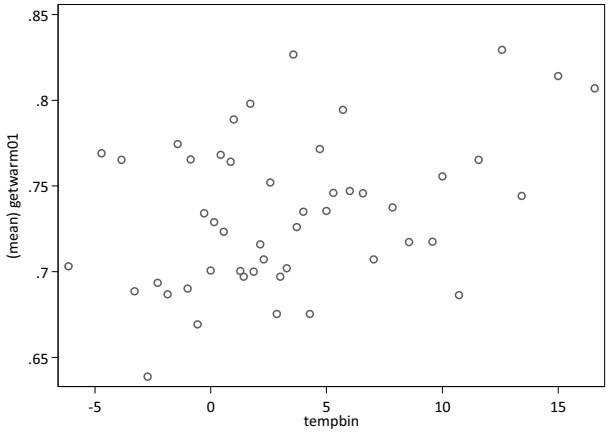
```
collapse (mean) getwarm01, by (tempbin)
twoway scatter  getwarm01 tempbin, aspect(1)
```



So much better, right?

One other thing: note that I use the option aspect(1). This forces Stata to produce a graph with aspect ratio (i.e. ratio of height:width) of 1. Otherwise, Stata's default is the "golden ratio," which is approx 1:1.6. This produces a picture that "stretches out" X, leading the eye to underestimate the strength of the relationship between X and Y:

```
twoway scatter  getwarm01 tempbin
```



II. SUMMARIZING the Relationship NON-PARAMETRICALLY

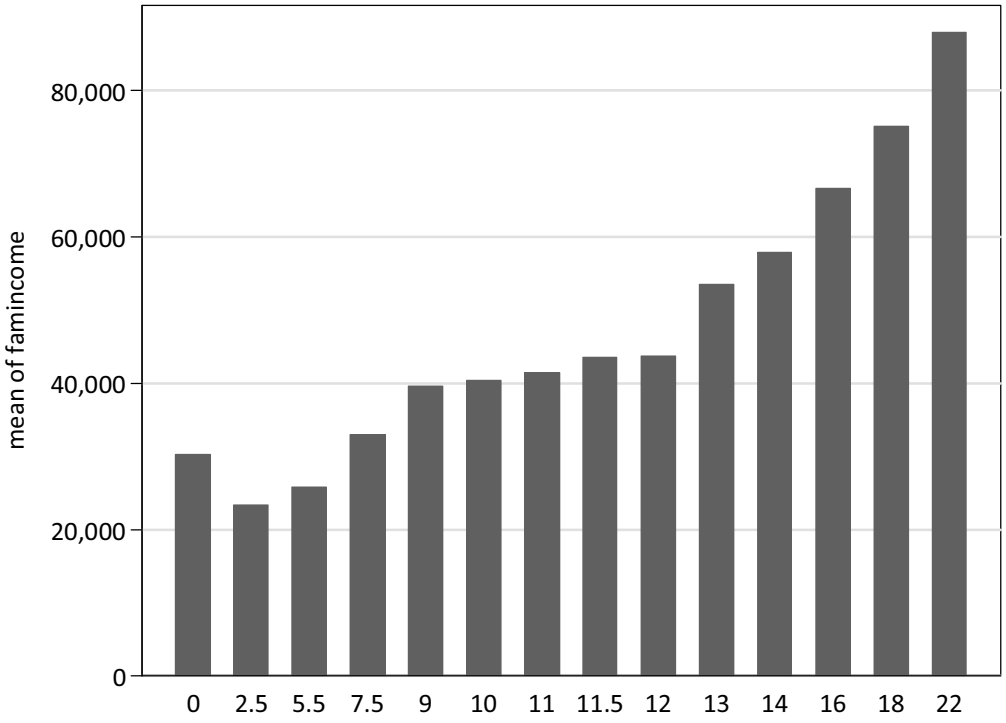
a. Table: Summary Statistics for Y by Values of X

```
. table educyears, c(mean famincome median famincome sd famincom)

------------------------------------------------------------
Years of  |
Education | mean(faminc~e)   med(faminc~e)    sd(faminc~e)
----------+-------------------------------------------------
        0 |       30277.78           32500        17690.95
      2.5 |       23383.33           17500        16861.47
      5.5 |          25800           22500        20889.18
      7.5 |        33012.9           22500        28464.15
        9 |       39599.74           32500        30232.96
       10 |       40429.91           32500        32337.82
       11 |       41450.45           32500        33324.25
     11.5 |       43551.47           45000        25684.93
       12 |        43733.9           37500        28895.93
       13 |       53493.88           45000        33180.55
       14 |       57854.84           55000        32961.58
       16 |       66617.31           67500        34944.65
       18 |       75098.48           67500         32904.3
       22 |       87965.12           87500        34723.64
------------------------------------------------------------
```

b. FIGURES (generally appropriate when X, Y or both are at interval-level or higher)

i. a <u>BAR CHART</u> displaying central tendencies of Y by values of X:

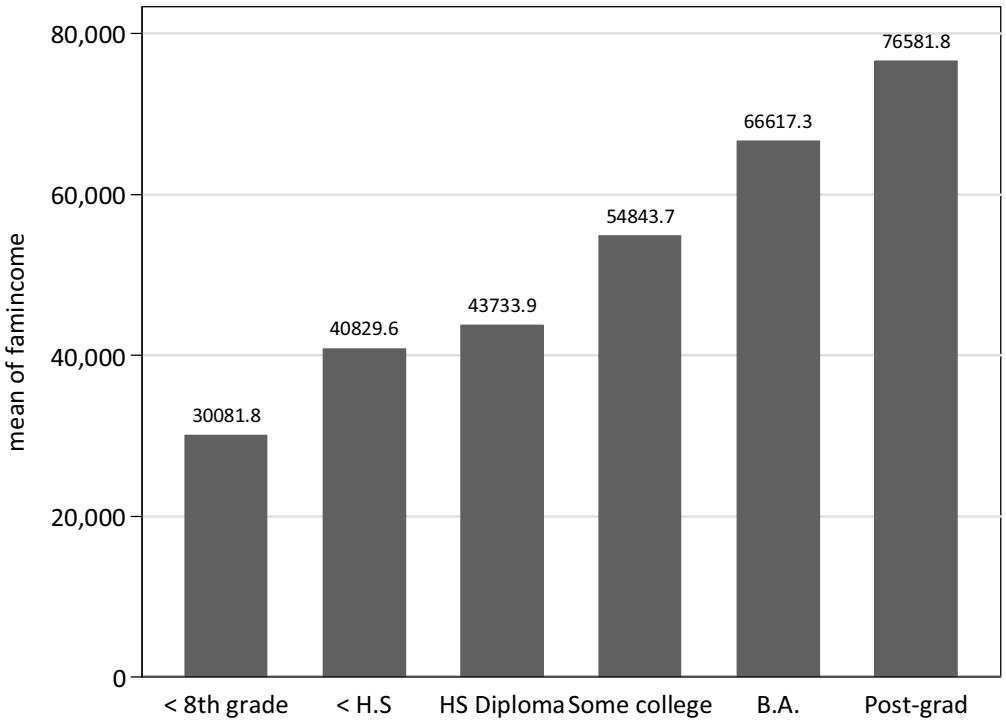
```
. graph bar (mean) famincome, over(educyears)
```



(Note: not ideal because gives false impression that intervals between bars are equally sized.)

Another way, with bar heights labeled:

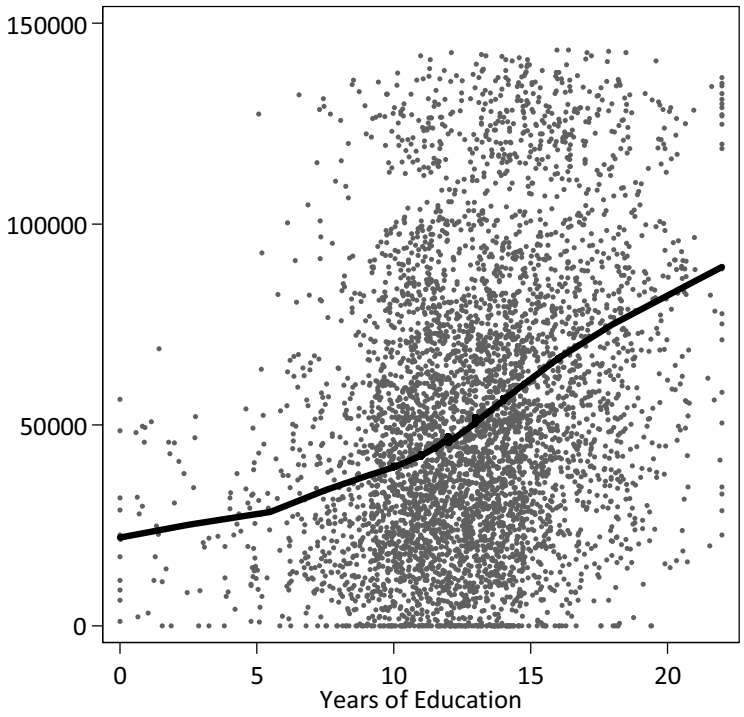
```
. graph bar (mean) famincome, over(educ_rc) blabel(bar)
```



ii. <u>SCATTERPLOT WITH SMOOTHER</u> displaying mean of Y by values of X (both X,Y interval-level or higher):

(We'll hold off on discussing details of how the smoother is constructed. For now, simply note that it requires very little in terms of parametric assumptions.)

```
. twoway (scatter famincome educyears, jitter(20) msize(tiny)) (lowess
famincome educyears, clc(black) clw(thick)), legend(off)
```

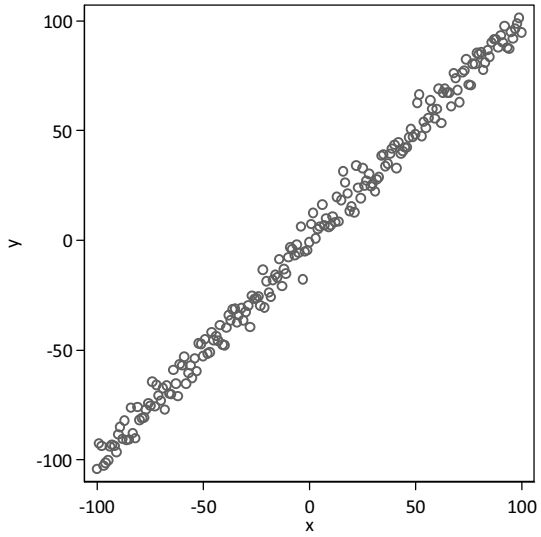


III. SUMMARIZING the relationship PARAMETRICALLY

i. The correlation coefficient

Sometimes it acts like we want it to. Very strong here, as it should be:

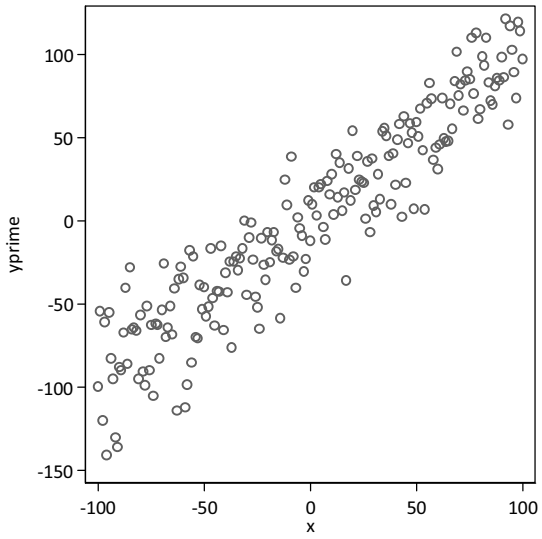
```
*generate 201 x values ranging from -100 to 100 in steps of 1:
. egen x = fill(-100/100)

*generate 201 y values equal to x plus a bit of noise:
. gen y = x +rnormal(0,5)

. scatter y x, aspect(1)
```




```
. corr y x
(obs=201)

             |        y        x
-------------+------------------
           y |   1.0000
           x |   0.9963   1.0000
```

A little weaker here, as we would expect (and want):

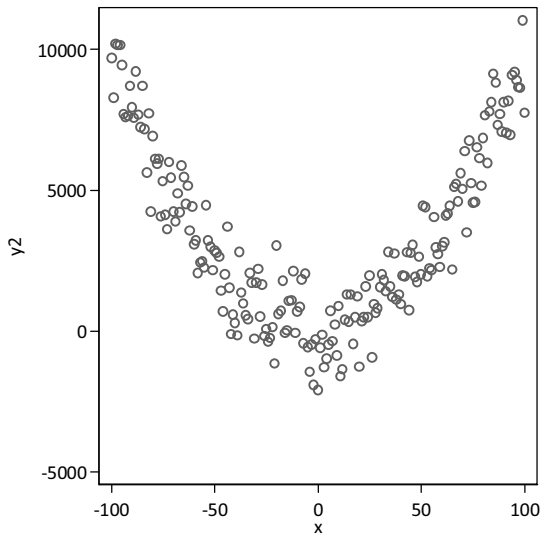
```
*generate 201 y values equal to x plus a lot more noise:
. gen yprime = x +rnormal(0,20)

. scatter yprime x, aspect(1)
```




```
. corr yprime x
(obs=201)

             |   yprime        x
-------------+------------------
      yprime |   1.0000
           x |   0.9348   1.0000
```

But is terrible at detecting non-linear relationships, no matter how strong:

```
*generate 201 y values equal to x-squared plus noise:
. gen y2 = x^2 + rnormal(0,1000)

. scatter y2 x, aspect(1)
```



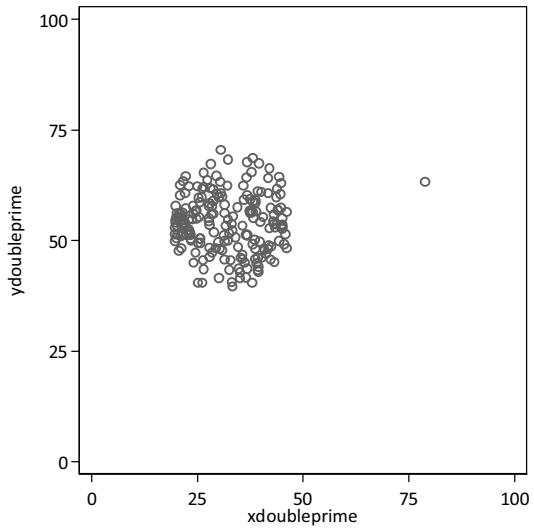

```
. corr y2 x
(obs=201)

             |       y2        x
-------------+------------------
          y2 |   1.0000
           x |   0.0096   1.0000
```

And is very sensitive to outliers. Here there is no relationship between x and y in obs 1 through 99. See how adding just one additional observation changes the correlation coefficient:

```
. set obs 100

. gen xdoubleprime = 33

. egen ydoubleprime = fill(50 52 54 56 58 50 52 54 56 58)

*note no relationship betw x and y among these observations;
*now let's change obs # 100 as follows:

. replace xdoubleprime = 80 in 100
(1 real change made)

. replace ydouble =75 in 100
(1 real change made)

. list xdoubleprime ydoubleprime

...
```

	xdoubleprime	ydoubleprime
91.	33	50
92.	33	52
93.	33	54
94.	33	56
95.	33	58
96.	33	50
97.	33	52
98.	33	54
99.	33	56
100.	**80**	**75**

```
. scatter ydouble xdouble, xsize(3) ysize(3) xlabel(0(25)100)
ylabel(0(25)100) jitter(20)
```




```
. corr y x
(obs=100)

             | ydoubl~e xdoubl~e
-------------+------------------
ydoubleprime |   1.0000
xdoubleprime |   0.5988   1.0000
```

That's a pretty darn high correlation, considering we have nothing but random covariation in obs 1-99!