STAT 400 Data Analysis September 15, 2003

Today: Bisquare – a way to make a fit robust to outliers (pp 110-119)
Slicing – a way to examine a bivariate function (pp 128-135)
Density Estimation – nonparametric fits to a histogram (1D & 2D)
Homework- A3

Bisquare: (Robust Fitting pp 110-119)

The bisquare procedure is illustrated in the text using the polarization data:

Here is a bit of it – I'll email the whole file to you:

Row	conc	babinet
1	9.0	25.9
2	9.0	24.6
3	10.0	20.7
4	12.0	22.5
5	12.0	25.3
6	13.5	23.9
7	14.0	25.5
8	15.0	20.6
9	15.0	20.9
10	15.0	24.2
11	15.0	24.3
12	15.0	24.6
13	15.5	24.1
14	15.5	24.2
……..		
346	100.5	15.0
347	106.0	15.7
348	107.0	15.5
349	108.5	15.9
350	112.0	17.9
351	116.0	16.1
352	118.5	15.2
353	118.5	15.1
354	121.5	16.4
355	122.5	15.6

The idea behind bisquare is this: You want to use some fitting technique but don't want to worry about the effect that outliers might have. So you do the following iterative procedure:

1. Fit the data with your fitting technique (for example, loess, or linear least squares).
2. Compute the residuals (resid = obs – fit)
3. Redo the fit using a weighted version of the fitting technique, where the weights are designed to reflect the residuals (small weights for big residuals)
4. Re-calculate the residuals (obs-new fit)
5. Repeat 3. and 4. until convergence (hopefully convergence occurs!)

Non-robust fitting methods use weight 1 for all points. p 114 shows how least squares is made into weighted least squares.

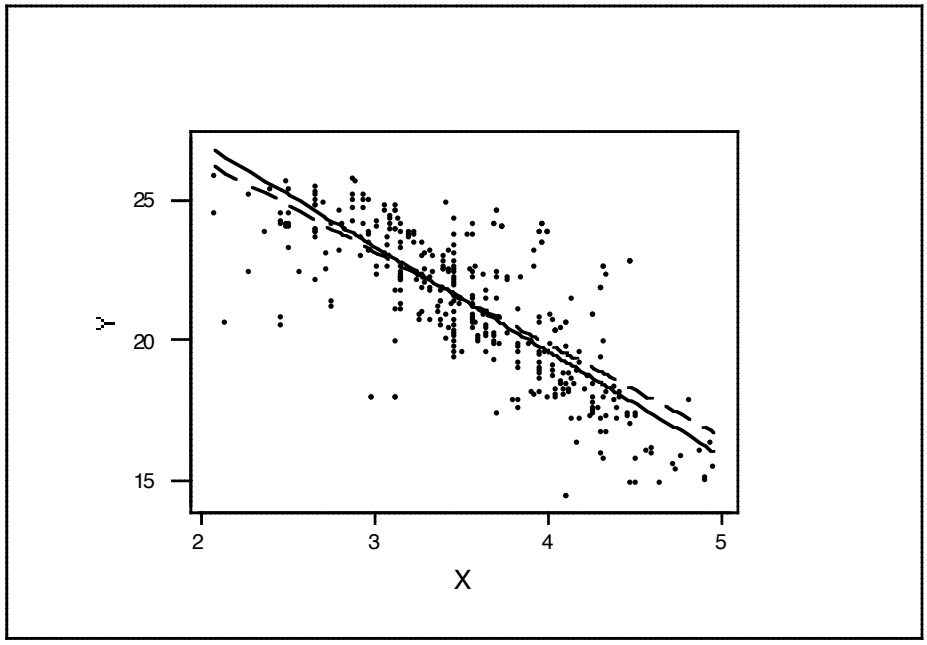
The MINITAB program for the bisquare procedure is given below, assuming the data are such that X=c1 and Y=c2 and the fitting method is OLS.

```
Gmacro
bisquare.mac
# This program uses "bisquare" as described in
Cleveland's book Visualizing Data
# to fit a  regression model to data.
Let k1 =1         # the column number of the
independent variable
Let k2 =2    # the column number of the dependent
variable
let k3 =3    # the column of the residual
let k4= 4      #the column of the weight function
(initially filled with 1s)
let k5=25     #number of data rows
let k12=5     #column of predicted values of Y
let k8 =100  # index of a group of scratch columns
let k9=k8+1
set ck4
25(1)
end
regr ck2 1 ck1 ck8 ck9     #save ordinary
regression for comparison
    #ck8 holds standardized resids and
                      #ck9 hold the predicted values
let k14=k12+1         # designate a scratch column
for OLS result
let ck14=ck9          #save it for plot later (ck9
gets used again)
Do k6=1:10                      #iterate 10 times
regress ck2 1 ck1;   #do the weighted regression
resid ck3;              #save the raw residuals
weights ck4.
Let ck12=ck2-ck3         #Y-(Y-Yhat) = Yhat the
predicted value from wtd regr
let ck8=abs(ck3)                #absolute residuals
median ck8 k7              #median abs residual is
robust estimate of scale
```

```
if k6 eq 1                  #save median absolute
residual first time through
    let k13=k7
endif
let k9=k8+1                 #designate a scratch
column for res/6s
let ck9=ck3/(6*k13)
let k10=k9+1
let ck10=ck9 lt 1           # this makes ck10=1 if
ck9 is <1 and = 0 otherwise
let ck4=ck10*(1-ck9**2)**2   #computes the new
weight function based on resids
enddo                        # ends the loop.  pass
through here when k6=10
name ck2 'Y' ck1 'X'
plot ck2*ck1 ck12*ck1 ck14*ck1;   #plot data,
bisquare line, and OLS line
symbol;
type 5 0 0;                          # only use
symbol for the first plot
connect;
type 0 1 2;                          # only
connect the second and third plot
overlay.
endmacro
```

The program could be easily extended to a multiple regression version.

Here is the result of the above bisquare program applied to the polarization data - the solid line is the bisquare line-fit, and the dotted line is the OLS (ordinary least squares) fit.



Note the difference between the objectives of loess, a nonparametric smoothing technique, and bisquare, a robust fitting technique. In other words, loess is a fitting procedure that results in irregular wavy lines with no particular functional form, determined entirely by the data and the smoothing parameters used; bisquare works with any fitting technique whether parametric or nonparametric - its focus is on down-playing the influence of outliers.

Of course, the two ideas, loess and bisquare, and be combined, and this is discussed in Section 3.6 - about robust-loess-curve fitting. (p122-128)

The next technique discussed in Ch 3 is SLICING. See sec 3.7.(pp 128-135)

Recall that ordinary regression of Y on X tries to fit a function (usually a straight line, but sometimes a polynomial) so that the function represents the conditional mean of the distribution of Y at each X value. ie. We want f(x) = E(Y|X=x), and we usually estimate this by assuming a certain functional form, and finding the parameters of this functional form that will make Y as close to f(x) as possible, in the least squares sense. A regression line or function is a line or function of conditional averages.

Note that the robust loess fit shown in Fig 3.41 plays the same role as the more usual regression function. As such we can examine residuals to study the fit, look for outliers, consider transformations, etc. Fig 3.42 removes the trend to allow concentration on the all the residuals at once, or at least look at wider slices.

The key novel idea in slicing is that the wider intervals we look at can overlap. As long as the trend across the slices reveals the trend in y as we of increase the x-value, there is no harm in letting the intervals overlap. Can you apply this idea to creating a new kind of "histogram"? In fact the idea extends to estimating density functions generally. More about this later – back to slicing….

Details of slicing interval choice are given pp 133-135. Once slices are determined, you can plot the residuals in Fig 3.44 to see if there are any distribution changes or patterns (there are in this case). This is important for data summary since it would be helpful to ignore such differences if they are really negligible, since it would simplify the data summary. (But this is not possible with the polarization data.)

How do we decide how many observations to include in each slice? We start with a target fraction of observations overlapping successive intervals, f, a sample size n., and a number of intervals,k. For example, if n=100, f=.5, and k=5, then the number in each interval would be about 33. Ranks 1-33, 17-49,34-66,51-83,67-100. Easy to check but nice to have the formula p 133 to get the size!

Density Estimation:

A univariate histogram of sample data gives a rough idea of the shape of the density function. However, the need to use rectangular frequency indicators produces graphs that do not look like density functions at all (unless the data set is huge and the number of bins huge). There is a simple way to improve on this. First I want to introduce a new data set:

B-D Rates:

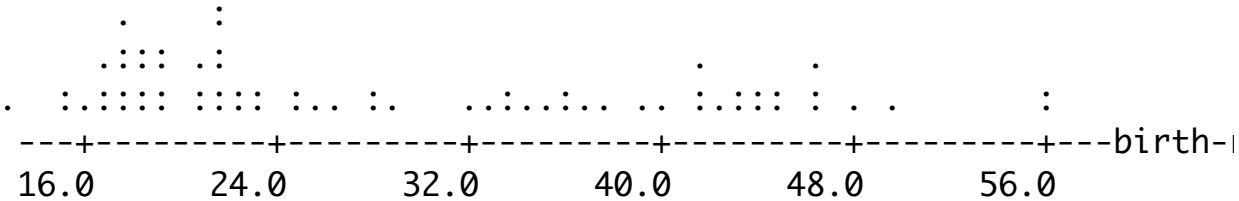
Row	birth-rt	death-rt	country	group

1	36.4	14.6	alg	1
2	37.3	8.0	con	1
3	42.1	15.3	egy	1
4	55.8	25.6	gha	1
5	56.1	33.1	ict	1
6	41.8	15.8	mag	1
7	46.1	18.7	mor	1
8	41.7	10.1	tun	1
9	41.4	19.7	cam	1
10	35.8	8.5	cey	1
11	34.0	11.0	chi	1
12	36.3	6.1	tai	1
13	32.1	5.5	hkg	2
14	20.9	8.8	ind	2
15	27.7	10.2	ids	2
16	20.5	3.9	irq	2
17	25.0	6.2	isr	2
18	17.3	7.0	jap	2
19	46.3	6.4	jor	2
20	14.8	5.7	kor	2
21	33.5	6.4	mal	2
22	39.2	11.2	mog	2
23	28.4	7.1	phl	2
24	26.2	4.3	syr	2
25	34.8	7.9	tha	2
26	23.4	5.1	vit	2
27	24.8	7.8	can	3
28	49.9	8.5	cra	3
29	33.0	8.4	dmr	3
30	47.7	17.3	gut	3
31	46.6	9.7	hon	3
32	45.1	10.5	mex	3
33	42.9	7.1	nic	3
34	40.1	8.0	pan	3
35	21.7	9.6	usa	3
36	21.8	8.1	arg	3
37	17.4	5.8	bol	3
38	45.0	13.5	bra	3
39	33.6	11.8	chl	3
40	44.0	11.7	clo	3

41	44.2	13.5	ecu	3
42	27.7	8.2	per	3
43	22.5	7.8	urg	3
44	42.8	6.7	ven	3
45	18.8	12.8	aus	4
46	17.1	12.7	bel	4
47	18.2	12.2	brt	4
48	16.4	8.2	bul	4
49	16.9	9.5	cze	4
50	17.6	19.8	dem	4
51	18.1	9.2	fin	4
52	18.2	11.7	fra	4
53	18.0	12.5	gmy	4
54	17.4	7.8	gre	4
55	13.1	9.9	hun	4
56	22.3	11.9	irl	4
57	19.0	10.2	ity	4
58	20.9	8.0	net	4
59	17.5	10.0	now	4
60	19.0	7.5	pol	4
61	23.5	10.8	pog	4
62	15.7	8.3	rom	4
63	21.5	9.1	spa	4
64	14.8	10.1	swe	4
65	18.9	9.6	swz	4
66	21.2	7.2	rus	4
67	21.4	8.9	vug	4
68	21.6	8.7	ast	3
69	25.5	8.8	nzl	3

Lets focus on the birth rates:

Distribution of birth rates?

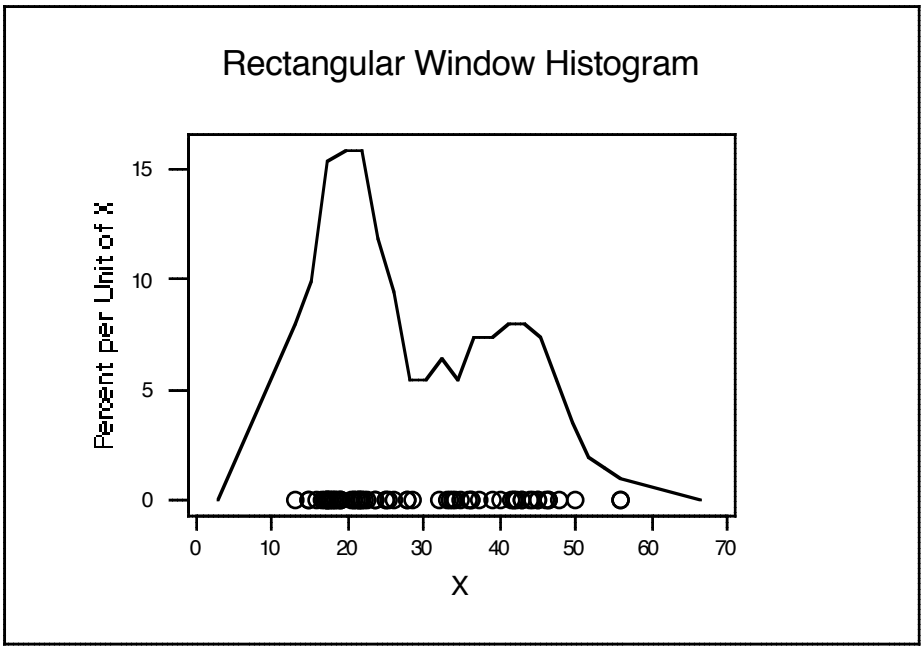


Density? Although the data is not sample data, it is still reasonable to describe the distribution as a continuous curve.

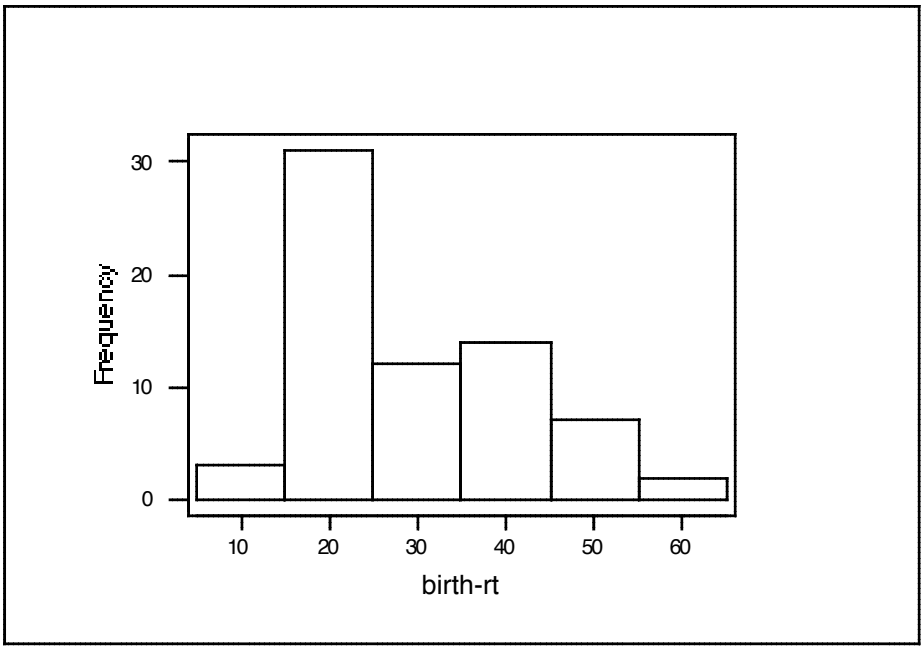
Here is a procedure:

1. choose as grid of m x-values
2. choose a distance d within which you will count frequencies
3. At each grid point, plot as a y value the number of data values within d of the grid point.

This is what you get for m=23, d=5 for the birth rates.



Compare the default:

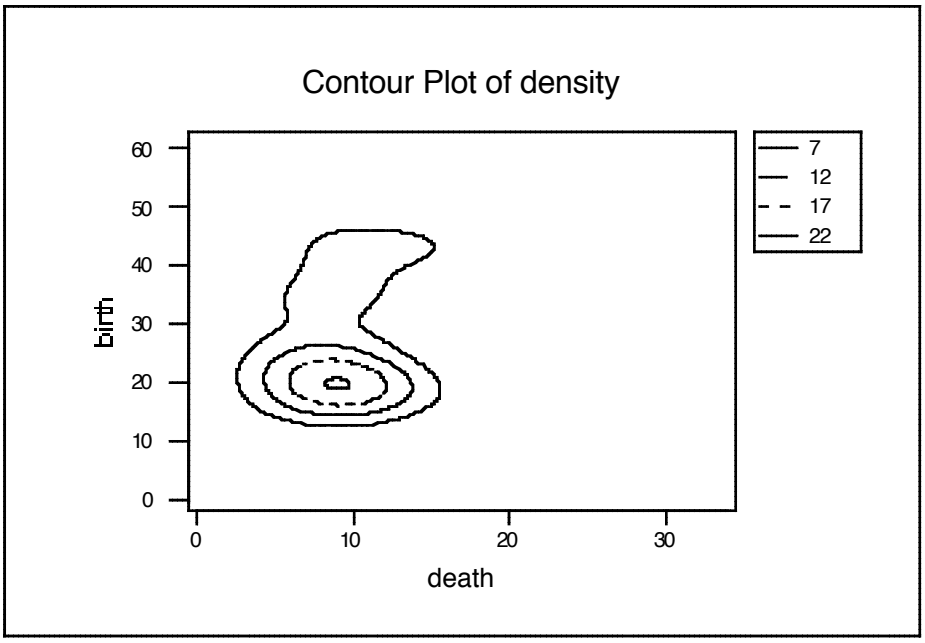


Program for the first one:

```
Gmacro
histo.mac
        # this program is a primitive density estimator
        # it expects the data to be in C1, any length.
erase c2-c4    # it simply uses a rectangular window to accumulate
brief          # the frequency close to each of 23 grid points
mini c1 k1                  # these commands set up the range for the grid
maxi c1 k2
n c1 k10                    # get sample size
let k3=(k2-k1)/k10**.5  # window half-width: use a larger DENOM for
                        #  LESS smoothing
set c2
0:20                        # increase this for more grid points
end
let c2=k1+(c2/20)*(k2-k1)
let k6=k1-2*k3
let k7=k2+2*k3
stack k6 c2 k7 c2
do k4=1:23              # For each grid point, accumuate the frequency
let c3=abs(c1-c2(k4))
let c3=c3 LE k3
```

```
sum c3 k5
let c4(k4)=k5                    # store the accumlated frequency
enddo
brief 2
sum c4 k9       # these commands standardize the pseudofrequencies so
#n c1 k10       # they "integrate" to 1 (this n command already done)
let c4=100*c4*k10/k9*(1/(k2-k1))
name c4 'pctperX' c2 'X'      # these commands set up the graph of the
                                  #resulting density est.
let c5=0*c1
plot c4*c2 c5*c1;
connect;
type 1 0;
symbol;
type 0 1;
overlay;
title 'Rectangular Window Histogram';
Axis 1;
Label "X";
Axis 2;
Label "Percent per Unit of X".
Endmacro
```

How about both birth and death rate distribution?



Program to do this:

```
Gmacro
density.mac
#MINITAB  program  to produce density  plots  with  birth-death data
brief 0      # this  avoids output so speeds up calculation
let k8=5   # This is the smoothing constant - big is smooth
# k1 is birth grid, k2 is deathg rid
let k4=11   #first column for  density estimates
let k5=33    # number of columns
let k6=60    # number of rows
do k2=1:k5
do k1=1:k6
let c6=exp(-(abs((k1-c1)/k8)**2)-(abs((k2-c2)/k8)**2))
sum c6 k3
let ck4(k1)=k3  #record  density  in row k1, column
enddo
let k4=k4+1    # go to next column
```

```
enddo
let k7=k4-1 # remember k4 has been increased by k5
let k4=k4-k5 #retrieve old k4
stack ck4-ck7 c80   # put density in  one column
set c81                  # create  ID for  birth  rate grid
k5(1:k6)
end
set c82              # create ID for death rate grid
(1:k5)k6
end
ContourPlot c80*c81*c82;
Connect.
endmacro
```

Assignment 3 Due Wed Sept 24:

Modify the "histo" program (or write one of your own) to incorporate all of the data in the estimation of the density at each grid-point. Your submission should include:

1. a copy of your program
2. an explanation of your modification
3. an example of its output on the birth-rate distribution

(My objective in this assignment is to ensure that you understand what is called the "kernel" method of density estimation, and also to ensure that you know how to run this kind of program (not merely a procedure from a stat package). Needless to say, I do not want you to use a ready-made procedure from a stat package.)