Kernel Density and Regression Estimation

by

A. Ronald Gallant
Department of Economics
University of North Carolina
Chapel Hill NC 27599-3305 USA

References

Silverman, B. W. (1986), *Density Estimation for Statistics and Data Analysis*, London: Chapman and Hall.

Wand, M. P. and M. C Jones (1995), *Kernel Smoothing*, London: Chapman and Hall.

Park, Byeong U., and J. S. Marron (1990), "Comparison of Data-Driven Bandwidth Selectors," *Journal of the American Statistical Association* 85, pp. 66–72.

Sheather, S. J., and M. C. Jones (1991). "A Reliable Data-Based Bandwidth Selection Method for Kernel Density Estimation,." *Journal of the Royal Statistical Society* B 53, 683-690.

Robinson, Peter M., (1983), "Nonparametric Estimators for Time Series," *Journal of Time Series Analysis* 4, 185–207.

Topics

- Kernel density estimation
- Bandwidth selection
- Multivariate density estimation
- Kernel regression estimation

Empirical distribution

$$\hat{F}_n(x) = \frac{1}{n}(\text{no of } x_1..x_n \leq x)$$

An unbiased estimator of F(x).

Asymptotically normal

$$\mathcal{E}(\hat{F}_n - F)^2 \to 0 \text{ at the rate } \frac{1}{n}$$

Density

$$f(x) = \lim_{h \to 0} \frac{1}{2h}[F(x+h) - F(x-h)]$$

Density estimate?

$$\hat{f}_n(x) = \lim_{h \to 0} \frac{1}{2h}[\hat{F}_n(x+h) - \hat{F}_n(x-h)]$$

for small h.

Density estimate?

$$\begin{aligned}\hat{f}_n(x) &= \frac{1}{2h}[\hat{F}_n(x+h) - \hat{F}_n(x-h)] \\ &= \frac{1}{2nh}[\#x_1, \cdots, x_n \text{ in } (x-h, x+h)] \\ &= \frac{1}{n}\sum_{t=1}^{n}\frac{1}{h}K\Big(\frac{x-x_t}{h}\Big)\end{aligned}$$

where

$$\begin{aligned}K(u) &= \begin{cases} 1/2 & \text{if } |u| < 1 \\ 0 & \text{otherwise} \end{cases} \\ &= \text{uniform density on } (-1, 1)\end{aligned}$$

Table 2.2 Eruption lengths (in minutes) of 107 eruptions of Old Faithful geyser.

4.37	3.87	4.00	4.03	3.50	4.08	2.25
4.70	1.73	4.93	1.73	4.62	3.43	4.25
1.68	3.92	3.68	3.10	4.03	1.77	4.08
1.75	3.20	1.85	4.62	1.97	4.50	3.92
4.35	2.33	3.83	1.88	4.60	1.80	4.73
1.77	4.57	1.85	3.52	4.00	3.70	3.72
4.25	3.58	3.80	3.77	3.75	2.50	4.50
4.10	3.70	3.80	3.43	4.00	2.27	4.40
4.05	4.25	3.33	2.00	4.33	2.93	4.58
1.90	3.58	3.73	3.73	1.82	4.63	3.50
4.00	3.67	1.67	4.60	1.67	4.00	1.80
4.42	1.90	4.63	2.93	3.50	1.97	4.28
1.83	4.13	1.83	4.65	4.20	3.93	4.33
1.83	4.53	2.03	4.18	4.43	4.07	4.13
3.95	4.10	2.72	4.58	1.90	4.50	1.95
4.83	4.12					

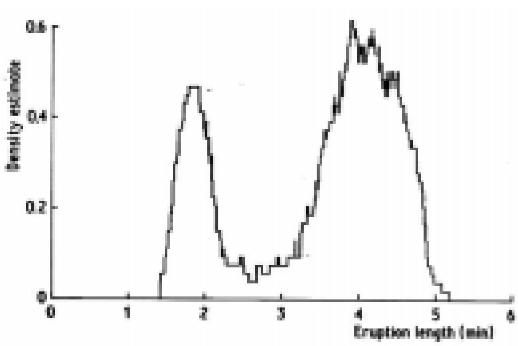


Fig. 2.3 Naive estimate constructed from Old Faithful geyser data, $h = 0.25$.

Round the corners

The roughness can be removed by using a s-mooth density such as the normal

$$K(x) = (2\pi)^{-1/2}\,\mathrm{exp}[(1/2)x^2]$$

in the estimator

$$\hat{f}_n(x) = \frac{1}{n}\sum_{t=1}^{n}\frac{1}{h}K\Big(\frac{x-x_t}{h}\Big)$$

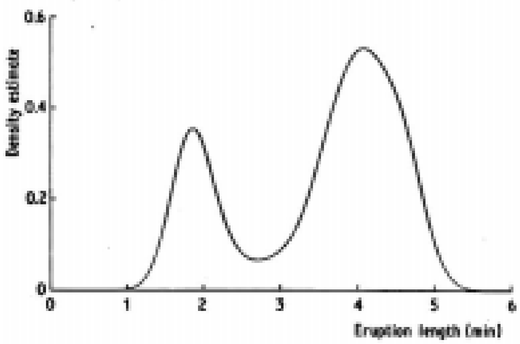


Fig. 2.8 *Kernel estimate for Old Faithful geyser data, window width 0.25.*

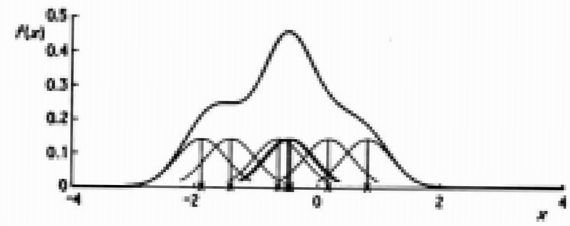


Fig. 2.4 *Kernel estimate showing individual kernels. Window width 0.4.*

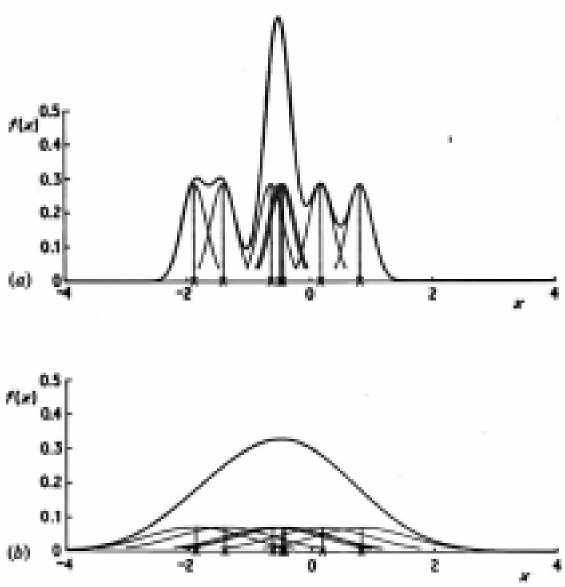


Fig. 2.5 *Kernel estimates showing individual kernels. Window widths: (a) 0.2; (b) 0.8.*

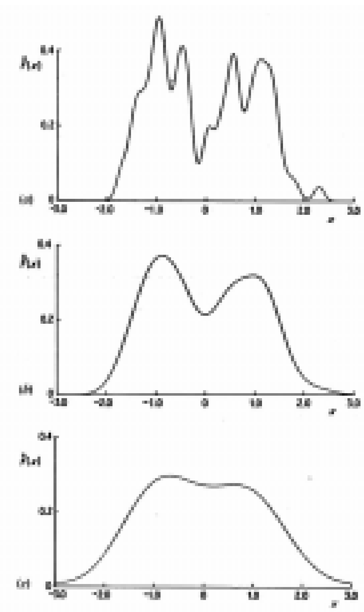


Fig. 2.6 *Kernel estimates for 200 simulated data points drawn from a bimodal density. Window widths: (a) 0.1; (b) 0.3; (c) 0.6.*

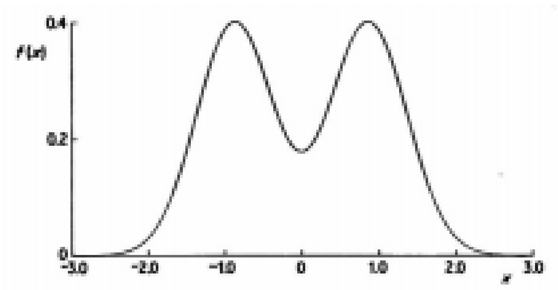


Fig. 2.7 *True bimodal density underlying data used in Fig. 2.6.*

Measures of quality

Mean Square Error (local)

$$\begin{aligned} \text{MSE}_x(\hat{f}) &= \mathcal{E}[\hat{f}(x) - f(x)]^2 \\ &= \mathcal{E}[\hat{f}(x) - \mathcal{E}\hat{f}(x)]^2 + \mathcal{E}[\mathcal{E}\hat{f}(x) - f(x)]^2 \\ &= \text{Var}[\hat{f}(x)] + \{\text{bias}[\hat{f}(x)]\}^2 \end{aligned}$$

Mean Integrated Square Error (global)

$$\begin{aligned} \text{MISE}(\hat{f}) &= \mathcal{E}\left\{\int [\hat{f}(x) - f(x)]^2\,dx\right\} \\ &= \int \left\{\mathcal{E}[\hat{f}(x) - f(x)]^2\right\}\,dx \\ &= \int \text{MSE}_x(\hat{f})\,dx \\ &= \text{IMSE}(\hat{f}) \end{aligned}$$

Approximation to MISE

If

- $K(t)$ is symmetric, defined on $[-1,1]$
- $\int K(t)\,dt = 1$ (K is a density)
- $\int tK(t)\,dt = 0$ (K has mean zero)
- $\int t^2K(t)\,dt = \sigma_K^2 > 0$ (K has positive variance)
- $n \to \infty,\ h \to 0,\ nh \to \infty$

then

$$\text{MISE}(\hat{f}) \doteq \frac{1}{4}\sigma_K^4\, h^4 \int [f``(x)]^2\,dx + \frac{1}{nh}\int [K(t)]^2\,dt$$

Approximation to MISE

$$\text{MISE}(\hat{f}) \doteq \tfrac{1}{4}\sigma_K^4\, h^4 \textstyle\int [f``(x)]^2\,dx + \tfrac{1}{nh}\textstyle\int [K(t)]^2\,dt$$

$$h_{opt} \doteq \left\{\frac{\int [K(t)]^2\,dt}{\sigma_K^4 \int [f``(x)]^2\,dx}\right\}^{1/5} n^{-1/5}$$

$\Rightarrow$ Rougher functions require smaller h.

$$\text{MISE}(f_{opt}) \doteq \tfrac{5}{4}C(K)\{\textstyle\int [f``(x)]^2 dx\}^{1/5} n^{-4/5}$$

$$C(K) = \frac{\{\int [t^2K(t)]dt\}^{2/5}}{\{\int [K(t)]^2 dt\}^{4/5}}$$

$\Rightarrow$ C(K) measures efficiency of $K(t)$.

Table 3.1 *Some kernels and their efficiencies*

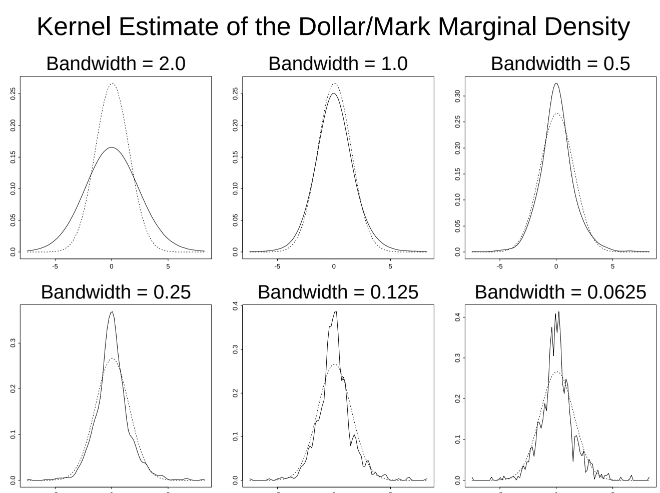
Kernel	$K(t)$	*Efficiency (exact and to 4 d.p.)*
Epanechnikov	$\frac{3}{4}(1-\frac{1}{5}t^2)/\sqrt{5}$ for $\lvert t\rvert < \sqrt{5}$, 0 otherwise	1
Biweight	$\frac{15}{16}(1-t^2)^2$ for $\lvert t\rvert<1$, 0 otherwise	$\left(\frac{3087}{3125}\right)^{1/2} \approx 0.9939$
Triangular	$1-\lvert t\rvert$ for $\lvert t\rvert<1$, 0 otherwise	$\left(\frac{243}{250}\right)^{1/2} \approx 0.9859$
Gaussian	$\frac{1}{\sqrt{2\pi}}e^{-(1/2)t^2}$	$\left(\frac{36\pi}{125}\right)^{1/2} \approx 0.9512$
Rectangular	$\frac{1}{2}$ for $\lvert t\rvert<1$, 0 otherwise	$\left(\frac{108}{125}\right)^{1/2} \approx 0.9295$

Bandwidth selection

Subjective choice

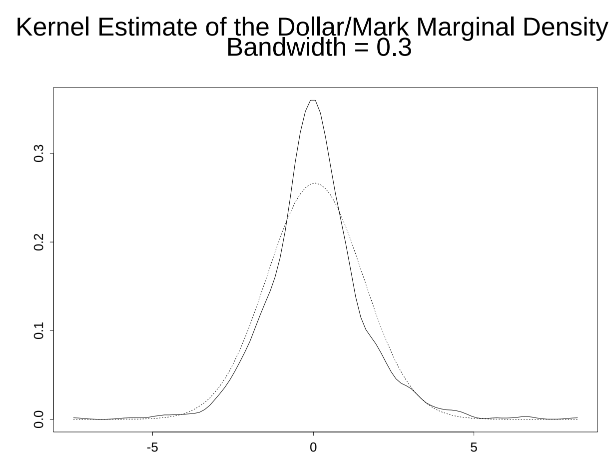
Plot out several curves and choose the estimate that is most in accordance with one's prior ideas about the density. For many applications this approach will be perfectly satisfactory. Indeed the process of examining several plots of the data, all smoothed by different amounts, may well give more insight into the data than merely considering a single automatically produced curve.

Silverman



Solid line is the kernel estimate; dashed line is a reference normal.

K is Gaussian. What is your visual pick?



Solid line is the kernel estimate; dashed line is a reference normal.

Here is my visual pick.

Plug-in rules

$$h_{opt} \doteq \left\{ \frac{\int [K(t)]^2 \, dt}{\sigma_K^4 \int [f``(x)]^2 \, dx} \right\}^{1/5} n^{-1/5}$$

If f and K were both Gaussian

$$\int [f``(x)]^2 dx = (3/8)\pi^{-1/2}\sigma^{-5}$$

$$\sigma_K^2 = \int t^2 K(t) \, dt = 1$$

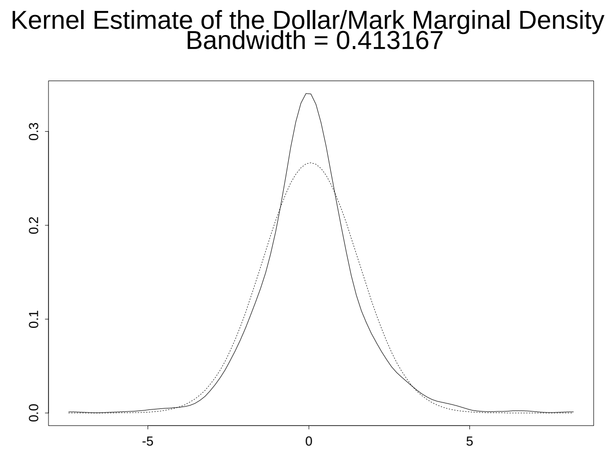
$$\int [K(t)]^2 \, dt = (4\pi)^{-1/2}$$

then

$$h_{opt} = 1.06 \; \sigma \; n^{-1/5}$$

$$\sigma^2 = (1/n) \sum_{t=1}^{n} (x_t - \bar{x})^2$$

$$= (\text{interquartile range})/1.34$$



Solid line is the kernel estimate; dashed line is a reference normal.

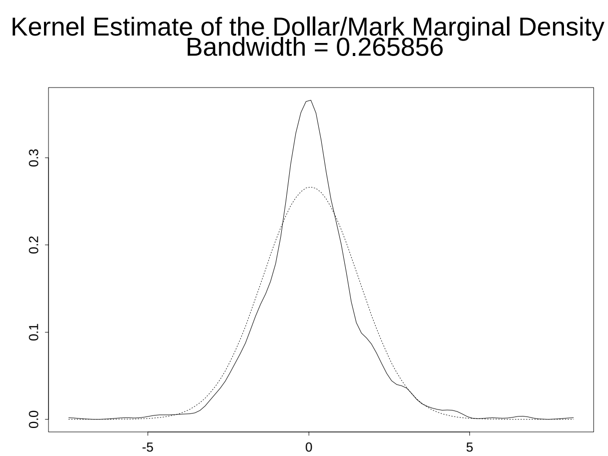
K is Gaussian, normal f plug-in rule.

Plug-in rules

Empirically determined improvement

$$h_{opt} = (0.9) A n^{-1/5}$$

$A = \min($standard deviation, interquartile range/1.34$)$



Solid line is the kernel estimate; dashed line is a reference normal.

K is Gaussian, Silverman empirically determined plug-in rule.

Least squares cross validation

The idea is to use the data to estimate

$$\int \hat{f}^2 - 2\int \hat{f} f,$$

which is the part of the integrated square error

$$\int (\hat{f} - f)^2 = \int \hat{f}^2 - 2\int \hat{f} f + \int f^2$$

that depends on the bandwidth h, plot

the estimate against h, and choose the

minimizing value to compute $\hat{f}$.

Least squares cross validation

An estimate of

$$\int \hat{f}^2 - 2\int \hat{f} f$$

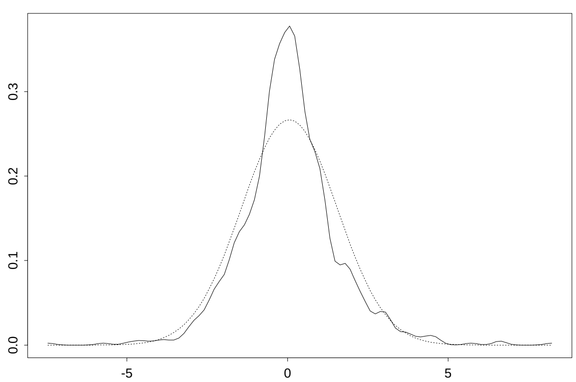
is

$$\begin{aligned} M_1(h) &= n^{-2}h^{-1}\sum_i \sum_j K^*[h^{-1}(x_i - x_j)] \\ &+ 2n^{-1}h^{-1}K(0) \end{aligned}$$

where

$$\begin{aligned} K^*(u) &= K^{(2)}(u) - 2K(u) \\ K^{(2)}(u) &= \int K(u-t)K(t)dt \end{aligned}$$

Kernel Estimate of the Dollar/Mark Marginal Density
Bandwidth = 0.2



Solid line is the kernel estimate; dashed line is a reference normal.

K is Gaussian, least-squares cross-validated bandwidth.

Least squares cross validation

Stone:

The difference between least-squares cross-validation estimate of h and the bandwidth that minimizes the integrated square error

$$\int [\hat{f}(u) - f(u)]^2 \, du$$

converges to zero as sample increases.

Marron:

The variability in the least-squares cross-validation estimate of h is so large that the result is useless.

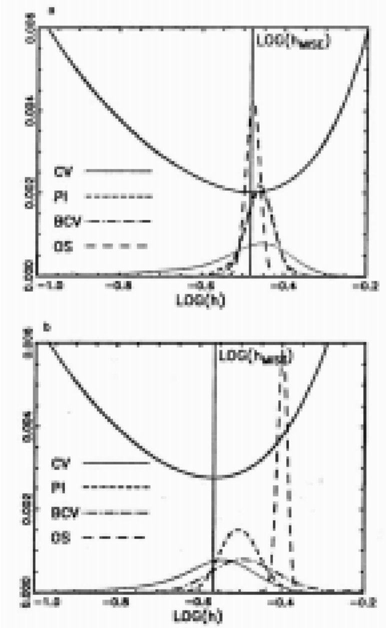


Figure 3. MISE(h) and Kernel Density Estimates of the Distributions of the Automatically Selected Bandwidths (on $\log_{10}$ scale), Taken From 500 Monte Carlo Replications of Samples of Size 400, From (a) N(0, 1) and (b) .5N(−1, 4/9) + .5N(1, 4/9).

Adaptive plug-in rules

Park & Marron:

$h_{PI} \doteq \sigma_K^{-4/5}[R(K)/R(\hat{f}''_a)]^{1/5} n^{-1/5}$

$R(g) = \int g^2(u)$ du

$\sigma_g^2 = \int t^2 g(t) dt$

$\hat{f}_a$ is a kernel estimate with bandwidth a

Adaptive plug-in rules

Park and Marron find the optimal a in terms of f. They combine this with the formula for h_{opt} to find a in terms of h_{opt} : $a = A(h_{opt})$. They find the root of

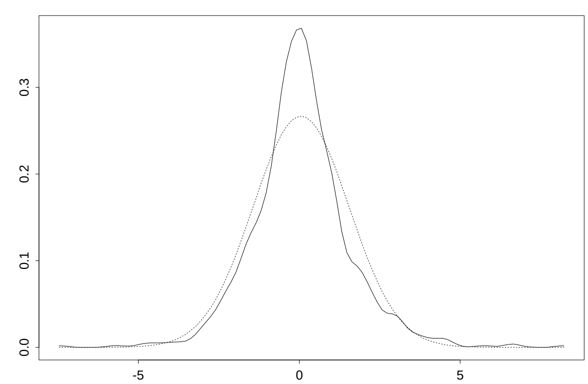
$$h \doteq \sigma_K^{-4/5}[R(K)/R(\hat{f}''_{A(h)})]^{1/5} n^{-1/5}$$

to get h_{PI}.

Currently recommended variant:

Sheather, S. J., and M. C. Jones (1991). "A Reliable Data-Based Bandwidth Selection Method for Kernel Density Estimation,." *Journal of the Royal Statistical Society* B 53, 683-690.

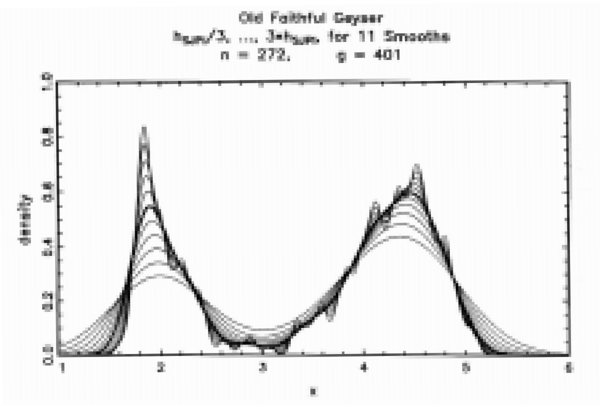
Kernel Estimate of the Dollar/Mark Marginal Density
Sheather-Jones bandwidth = .255366



Solid line is the kernel estimate; dashed line is a reference normal.

K is Gaussian, Sheather-Jones bandwidth.

Visual methods



Overplot densities for a large number of bandwidths with the data determined bandwidth shown with a wider line width than the others.

A Good Way to Display Panel Data

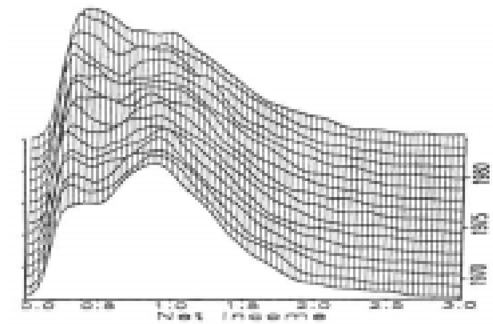


Figure 2. Expanded Representation of the Density Estimates in Figure 1b.

Adaptive methods

Variable bandwidth kernels

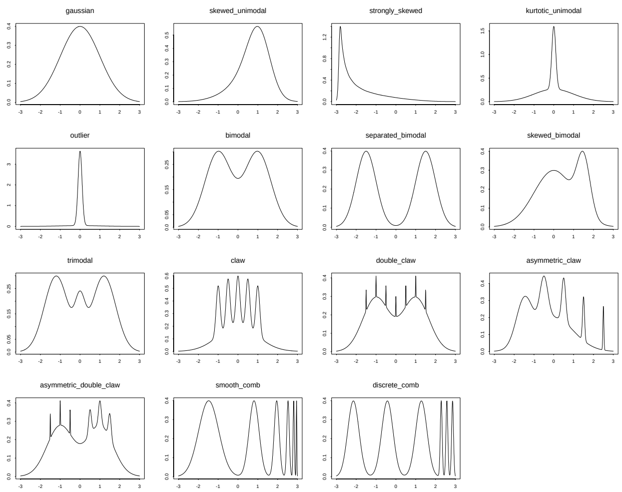
Silverman, B. W. (1986), *Density Estimation for Statistics and Data Analysis*, London: Chapman and Hall.

Transformation methods

Wand, M. P., J. S. Marron, and D. Rupert (1991), "Transformations in Density Estimation," *Journal of the American Statistical Association* 86, 343–352.

Mixture: generates density plots and data

```
usage: mixture -n value -d name [options]

flags: -n value number of observations or number of plot points
                the flag -m is a synonym
       -d name  density to sample or plot, choices: gaussian,
                skewed_unimodal, strongly_skewed, kurtotic_unimodal,
                outlier, bimodal, separated_bimodal, skewed_bimodal,
                trimodal, claw, double_claw, asymmetric_claw,
                asymmetric_double_claw, smooth_comb, discrete_comb

options: -s value supply an integer seed rather than accept default
         -p       generate plot points
         -e       output observations in e format with full precision

examples: mixture -n 200 -d claw > out.dat
            The output file contains y values, one per line.
          mixture -n 200 -d claw -p > out.dat
            The output file contains y x pairs, one pair per line,
            y first.

reference: Marron, J. S., and M. P. Wand (1992) Exact Mean Integrated,
           Squared Error, The Annals of Statistics 20, 712--736.
```



The various densities generated by mixture.

Uniden: kernel density estimation

```
usage: uniden [options] filename

options: -p            predict at data points only
         -c            cross validate instead of estimate
         -q            quadratic kernel, normal is default
         -o            override Sheather-Jones bandwidth selection
         -d            do not bin the data
         -f            acts as a filter, takes standard input
         -n            normal density points instead of kernel
         -m     value  maximum sample size
         -b     value  bandwidth
         -g     value  number of plot points sent to cout
         -xmin  value  minimum plot point
         -xmax  value  maximum plot point

example: uniden -b 6.0 -q -g 200 in.dat >out.dat

The input file contains x values, one per line.  The output
file contains y x pairs, one pair per line, y first.
```

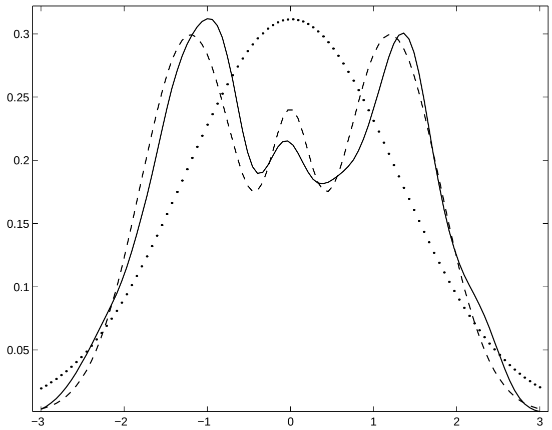

Trimodal example: kernel density estimation

```
mixture -n 500 -d trimodal > mix.dat

mixture: 500 observations from density trimodal

mixture -n 100 -d trimodal -p > mix.tru

mixture: 100 plot points from density trimodal

uniden -xmin -3.0 -xmax 3.0 -g 100 -n mix.dat > mix.nor

uniden -xmin -3.0 -xmax 3.0 -g 100 -o mix.dat > mix.ker

Trials:
cv -0.217074  bw 0.498839
cv -0.217959  bw 0.465583
cv -0.218791  bw 0.432327
cv -0.219566  bw 0.399071
cv -0.220277  bw 0.365815
cv -0.220917  bw 0.332559 = 0.9*min(sd,iqr/1.34)/pow(n,0.2)
cv -0.221475  bw 0.299303
cv -0.221931  bw 0.266048
cv -0.222241  bw 0.232792
cv -0.222305  bw 0.199536
cv -0.221927  bw 0.16628
cv -0.220751  bw 0.133024
cv -0.21832  bw 0.0997678
cv -0.213631  bw 0.0665119

Selection:
bandwidth = 0.199536
```


Trimodal example: Matlab plotting

```
fid = fopen('mix.tru','r');
tru = fscanf(fid,'%f %f',[2,inf])';
fclose(fid);

fid = fopen('mix.nor','r');
nor = fscanf(fid,'%f %f',[2,inf])';
fclose(fid);

fid = fopen('mix.ker','r');
ker = fscanf(fid,'%f %f',[2,inf])';
fclose(fid);

left = min([tru(:,2)',nor(:,2)',ker(:,2)'])-.1;
rite = max([tru(:,2)',nor(:,2)',ker(:,2)'])+.1;

bot = min([tru(:,1)',nor(:,1)',ker(:,1)']);
top = max([tru(:,1)',nor(:,1)',ker(:,1)'])+.01;

figure(1);
plot(ker(:,2),ker(:,1),'-','LineWidth',1.0);
axis([left rite bot top]);
str = ['\fontsize{16} \bf Trimodal Data: ',...
      'Cross Validated Kernel Density Estimate'];
title(str);
str = ['\fontsize{13}Solid line is kernel estimate, ' ...
   'dashed true density, dotted reference normal'];
xlabel(str);
hold on;
plot(tru(:,2),tru(:,1),'--','LineWidth',1.0);
plot(nor(:,2),nor(:,1),'.','LineWidth',1.0);
hold off;

print -r300 -deps2 mix.ps;
```

Trimodal Data: Cross Validated Kernel Density Estimate



Solid line is kernel estimate, dashed true density, dotted reference normal

Multivariate density estimation

dimension	number of feet^d in a yard^d
1	3
2	9
3	27
4	81
5	243
•	•
•	•
•	•
d	3^d

The number of points required to preserve a given grid fineness goes up exponentially with dimension.

Multivariate Normal Kernel

$x \in \Re^d$

$\hat{f}(x) = n^{-1} \sum_{t=1}^{n} h^{-d} K[(x - x_t)/h]$

$K(u) = (2\pi)^{-d/2} [\det(S)]^{-1/2} \exp[-u' S^{-1} u/2]$

$S = n^{-1} \sum_{t=1}^{n} (x_t - \bar{x})(x_t - \bar{x})'$

$h_{opt} = [4/(2d+1)]^{1/(d+4)} n^{-1/(d+4)}$

$M_1(h) = n^{-2} h^{-d} \{\sum_s \sum_t K^*[(x_t - x_s)/h] + 2K(0)\}$

$K^*(u) = K^{(2)}(u) - 2K(u)$

$K^{(2)}(u) =$ density of sum two independent $u \sim K(u)$

$= (2\pi)^{-d/2} [\det(2S)]^{-1/2} \exp[-u'(2S)^{-1} u/2]$ for the normal

Nonparametric Regression

Regression as conditional expectation

$(y, x) \sim f(y, x)$

(y_t, x_t), $t = 1, \ldots, n$, a random sample

$\mathcal{E}(y|x) = \frac{\int y f(y,x)\, dy}{\int f(y,x)\, dy} \equiv$ g(x)

This is definitional, conditional expectation defines the regression

$y_t = g(x_t) + u_t$

u_t uncorrelated with any function of x

Kernel estimate of $f(x)$ and $f(y,x)$

$K(u) = (2\pi)^{-d/2}[\det(S)]^{-1/2}\exp[-u'S^{-1}u/2]$

$x \in \Re^k$

$\hat{f}(x) = n^{-1}\sum_{t=1}^{n} h^{-k}K[(x - x_t)/h]$

$S = n^{-1}\sum_{t=1}^{n}(x_t - \bar{x})(x_t - \bar{x})'$

$\hat{f}(y,x) = n^{-1}\sum_{t=1}^{n} h^{-k+1}K[(y - y_t, x - x_t)/h]$

$S = n^{-1}\sum_{t=1}^{n}(y_t - \bar{y}, x_t - \bar{x})(y_t - \bar{y}, x_t - \bar{x})'$

45

Kernel estimate

If the kernel satisfies

$\int yK(y,x)\,dy = 0$ and $\int K(y,x)\,dy = K(x)$

then

$$\begin{aligned}
\hat{g}(x) &= \frac{\int y\hat{f}(y,x)\,dy}{\int \hat{f}(y,x)\,dy} \\
&= \frac{\sum_{t=1}^{n} y_t K[(x - x_t)/h]}{\sum_{t=1}^{n} K[(x - x_t)/h]} \\
&= \text{a locally weighted average of } y_t
\end{aligned}$$

46

Cross validation

Let $\hat{g}_t$ be the estimator of $g(x)$ with x_t left out of the sample and with bandwidth h. Choose h to minimize

$$\text{CV}(h) = \frac{1}{n}\sum_{t=1}^{n}[y_t - \hat{g}_t(x_t)]^2$$

47

Time series (Robinson)

$y_t \in \Re^M$ $x_{t-1} = (y'_{t-L}, \ldots, y'_{t-1})'$

Joint density

$\hat{f}(y,x) = n^{-1}h^{-ML-M}\sum_{t=1}^{n} K[(y - y_t, x - x_{t-1})/h]$

One-step-ahead conditional density

$\hat{f}(y|x) = \hat{f}(y,x)/\int \hat{f}(y,x)dy$

Conditional mean

$\hat{\mathcal{E}}(y|x) = \frac{\sum_{t=1}^{n} y_t K[(x - x_t)/h]}{\sum_{t=1}^{n} K[(x - x_t)/h]}$

Conditional variance

$\hat{V}(y|x) = \frac{\sum_{t=1}^{n}(y_t - \hat{\mathcal{E}}_t)(y_t - \hat{\mathcal{E}}_t)'K[(x - x_t)/h]}{\sum_{t=1}^{n} K[(x - x_t)/h]}$

$\hat{\mathcal{E}}_t = \hat{\mathcal{E}}(y|x_t)$

Works because time series data are correlated in time, not in space.

48

Unireg: kernel regression estimation

```
usage: unireg [options] filename

options: -p             predict at data points only
         -c             cross validate instead of estimate
         -q             quadratic kernel, normal is default
         -d             do not bin the data
         -f             acts as a filter, takes standard input
         -m     value   maximum sample size
         -b     value   bandwidth
         -g     value   number of plot points sent to cout
         -xmin  value   minimum plot point
         -xmax  value   maximum plot point

example: unireg -b 6.0 -q -g 200 in.dat >out.dat

The input file contains y x pairs, one pair per line, y first.
The output file is the same.
```


Exchange rate example: SAS data preparation

```
data work01;
  infile 'dm_fri.dat';
  input
    (yy mm dd day gap spot forwrd tb30);

  s   = 100*(log((1.0/spot)));
  f   = 100*(log((1.0/forwrd)));

  ds = s - lag1(s);
  fs = f - s;
  tb = tb30;

  if _n_ = 1 then delete;

data work02;
  set work01;
  dslag=lag(ds);
  if _n_=1 then delete;

proc sort data=work02 out=work03;
  by dslag;

data work04;
  set work03;
  file "ds_lev01.dat";
  put ds 20.16 dslag 20.16;
```

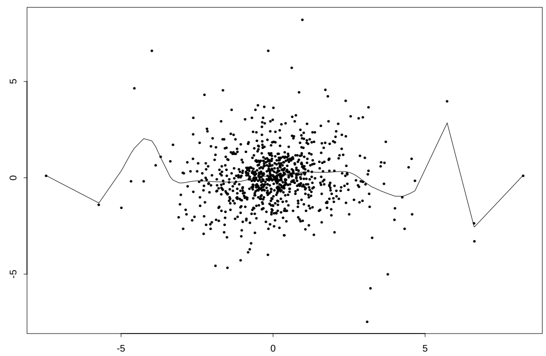

Exchange rate example: kernel regression

```
unireg -p ds_lev01.dat > ds_lev02.a.dat

Trials:
cv 2.26143  bw .398417
cv 2.26179  bw .371856
cv 2.26230  bw .345295
cv 2.26299  bw .318733
cv 2.26387  bw .292172
cv 2.26500  bw .265611 = .9*min(sd,iqr/1.34)/n**.2
cv 2.26640  bw 0.23905
cv 2.26821  bw .212489
cv 2.27078  bw .185928
cv 2.27512  bw .159367
cv 2.28318  bw .132806

Selection:
bandwidth = .398417
```


Exchange rate example: Splus plotting

```
par(mfrow=c(1,1),oma=c(4,3,5,3))

tmp  <- matrix(scan("ds_lev01.dat"),
          ncol=2,byrow=T)
y    <- tmp[,1]
x    <- tmp[,2]

tmp  <- matrix(scan("ds_lev02.a.dat"),
          ncol=2,byrow=T)
yhat <- tmp[,1]
xplt <- tmp[,2]

plot(c(x,xplt),c(y,yhat),type="n",
          ylab="",xlab="")
points(x,y)
lines(xplt,yhat)

mtext(side=1,line=-1,cex=1.3,outer=TRUE,
"Solid line is the kernel estimate; \
points are a scatter plot of the data.")

mtext(side=3,line=1,cex=2,outer=TRUE,
"Kernel Estimate of the Dollar/Mark
Conditional Mean \n
Bandwidth = 0.398417")
```

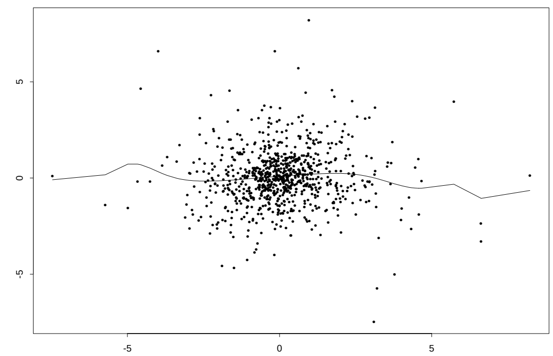
Kernel Estimate of the Dollar/Mark Conditional Mean
Bandwidth = 0.398417



Solid line is the kernel estimate;
points are a scatter plot of the data.

The estimate is ridiculous: The conditional expectation of tomorrow's return given yesterday's return is zero, not the crooked line above.

Kernel Estimate of the Dollar/Mark Conditional Mean
Bandwidth = 0.9



Solid line is the kernel estimate;
points are a scatter plot of the data.

```
unireg -p -b .9 ds_lev01.dat > ds_lev02.b.dat
```

The estimate is better. It suggests momentum in the center. But in subsequent computations it might make more sense to just put $\hat{\mathcal{E}}(y_t|y_{t-1}) = 0$.

Exchange rate example: SAS data preparation

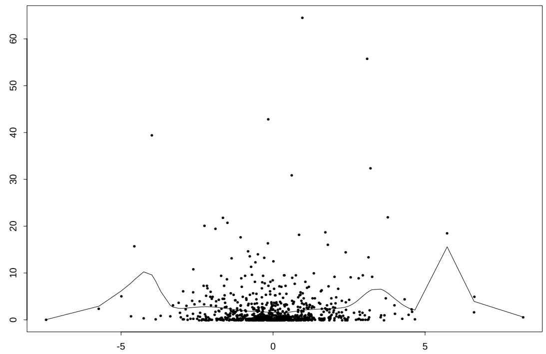
```
data work01;

  infile 'ds_lev01.dat';
  input y x;

  infile 'ds_lev02.b.dat';
  input yhat xplt;

  r2 = (y-yhat)**2;

  file "ds_lev03.dat";
  put r2 20.16 x 20.16;
```


Exchange rate example: kernel regression

```
unireg -p ds_lev03.dat > ds_lev04.a.dat

Trials:
cv 24.4078  bw .398417
cv 24.4928  bw .371856
cv 24.5969  bw .345295
cv 24.7228  bw .318733
cv 24.8726  bw .292172
cv 25.0468  bw .265611 = .9*min(sd,iqr/1.34)/n**.2
cv 25.2450  bw 0.23905
cv 25.4677  bw .212489
cv 25.7219  bw .185928
cv 26.0297  bw .159367
cv 26.4331  bw .132806

Selection:
bandwidth = .398417

unireg -xmin -2. -xmax 2. -b .398417 ds_lev03.dat > ds_lev04.b.dat
```

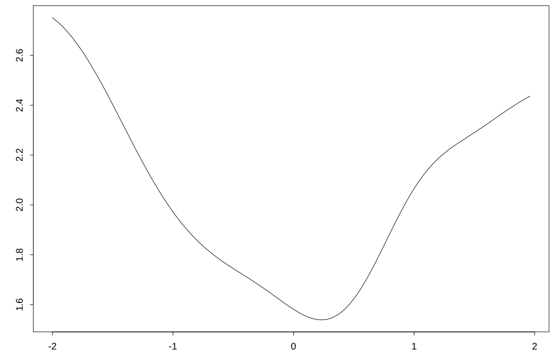
Note: There are two fits with the same bandwidth above. The first has the x-axis ranging from the min and max of the data; the second from -2 to +2.

Kernel Estimate of the Dollar/Mark Conditional Variance
Bandwidth = 0.398417



Solid line is the kernel estimate;
points are a scatter plot of the data.

Kernel Estimate of the Dollar/Mark Conditional Variance
Bandwidth = 0.398417



Localization

A kernel regression estimator can be viewed as a weighted least squares estimate of a mean

$$\begin{aligned} \mu_x &= \frac{\sum_{t=1}^n y_t K[(x-x_t)/h]}{\sum_{t=1}^n K[(x-x_t)/h]} \\ &= (X'WX)^{-1}(X'Wy) \\ \widehat{g}(x) &= \mu_x \end{aligned}$$

where

$$X = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \qquad y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

$$W = \begin{pmatrix} K[(x-x_1)/h] & & & \\ & K[(x-x_2)/h] & & \\ & & \ddots & \\ & & & K[(x-x_2)/h] \end{pmatrix}$$

Note that W depends on both x and h.

Local linear

$$\begin{aligned} \begin{pmatrix} \widehat{\alpha} \\ \widehat{\beta} \end{pmatrix} &= (X'WX)^{-1}(X'Wy) \\ \widehat{g}(x) &= \widehat{\alpha} + \widehat{\beta} x \end{aligned}$$

where

$$X = \begin{pmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{pmatrix} \qquad y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

$$W = \begin{pmatrix} K[(x-x_1)/h] & & & \\ & K[(x-x_2)/h] & & \\ & & \ddots & \\ & & & K[(x-x_2)/h] \end{pmatrix}$$

Note that $\widehat{\alpha}$ and $\widehat{\beta}$ must be recomputed for each x.

Local quadratic

$$\begin{pmatrix} \widehat{\alpha} \\ \widehat{\beta} \\ \widehat{\gamma} \end{pmatrix} = (X'W^{-1}X)^{-1}(X'W^{-1}y)$$

$$\widehat{g}(x) = \widehat{\alpha}$$

where

$$X = \begin{pmatrix} 1 & x_1 - x & (x_1 - x)^2 \\ 1 & x_2 - x & (x_2 - x)^2 \\ & \vdots & \\ 1 & x_n - x & (x_n - x)^2 \end{pmatrix} \qquad y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

$$W^{-1} = \begin{pmatrix} K[(x-x_1)/h] & & & \\ & K[(x-x_2)/h] & & \\ & & \ddots & \\ & & & K[(x-x_2)/h] \end{pmatrix}$$

$\widehat{\alpha}$, $\widehat{\beta}$, and $\widehat{\gamma}$ must be recomputed for each x. Local quadratic is better than local linear because the bias is smaller but the variance is the same. Similarly, local quartic is better that local cubic.

Derivative estimate:

$$\frac{d}{dx}\widehat{g}(x) = \widehat{\beta}$$

Localization

Nearly anything can be localized. Here is local maximum likelihood:

$$\widehat{\theta}_x = \underset{\theta \in \Theta}{\text{argmax}} \sum_{t=1}^{n} K[(x - x_t)/h] \log[p(y_t|x_t, \theta)]$$

$$\widehat{\mathcal{E}}(y|x) = \int y\, p(y|x, \widehat{\theta}_x)\, dy$$

$$\widehat{\text{Var}}(y|x) = \int \left[y - \widehat{\mathcal{E}}(y|x)\right]^2 p(y|x, \widehat{\theta}_x)\, dy$$

θ must be recomputed for each x.