Nonparametric Function Estimation
Chapter 6 Kernel Estimators Stat 550[1]

David W Scott[2]

Rice University

October 17
Fall 2023
Rice University

[1] A course based upon the 2nd edition of *Multivariate Density Estimation; Theory, Practice, and Visualization*, *John Wiley & Sons, 2015*

[2] www.stat.rice.edu/∼scottdw/

Chapter VI: Kernel Density Estimators (KDE)

- The basic kernel estimator may be written compactly as

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) = \frac{1}{n} \sum_{i=1}^{n} K_h(x - x_i)$$

  where $K_h(t) = K(t/h)/h$, which is a simple rescaling (think of h as the standard deviation)
- The second form makes it clear that the KDE is an equal mixture of "kernels" **centered** on each data point (not a parsimonious mixture!)
- The kernel function is usually centered at 0. The Gaussian kernel is popular.

Alternative Motivations for Kernel Density Estimators

- From the vantage point of a statistician or instructor, the averaging of shifted histograms seems a natural motivation for kernel estimators.
- However, scientists from other disciplines may find it more natural to think in terms of
  - numerical analysis (finite differences)
  - convolution smoothing (high pass/low pass filters signal processing)
  - approximating functions by orthogonal series
- we will briefly examine each of these

Numerical Analysis and Finite Differences

- Rosenblatt derived the kernel estimator as a one-sided finite difference approximation to the derivative of $F_n(\cdot)$:

$$\hat{f}(x) = \frac{F_n(x) - F_n(x-h)}{h} = \frac{1}{nh}\sum_{i=1}^{n} \boldsymbol{I}_{[x-h,x)}(x_i)$$

$$= \frac{1}{nh}\sum_{i=1}^{n} \boldsymbol{I}_{(0,1]}\left(\frac{x - x_i}{h}\right),$$

  which a kernel estimator with $K = U(0,1]$.

- As $\mathrm{E}[F_n(x)] = F(x)$ for all x, then with the Taylor's series

$$F(x-h) = F(x) - hf(x) + \frac{1}{2}h^2 f'(x) - \frac{1}{6}h^3 f''(x) + \cdots$$

- Gives $Bias\{\hat{f}(x)\} = \mathrm{E}[\hat{f}(x)] - f(x) = -\frac{1}{2}hf'(x) + O(h^2)$
- Giving the integrated squared bias is $h^2 R(f')/4$, like a histogram (but 3 times larger!!)

Numerical Analysis and Finite Differences (cont'd)

- Rosenblatt next considered the two-sided finite difference

$$\hat{f}(x) = \frac{F_n\left(x + \frac{h}{2}\right) - F_n\left(x - \frac{h}{2}\right)}{h}$$

  for which the bias turns out to be $h^2 f''(x)/24$; cf FP.

- The kernel here is $U(-0.5, 0.5)$, which is centered at 0.

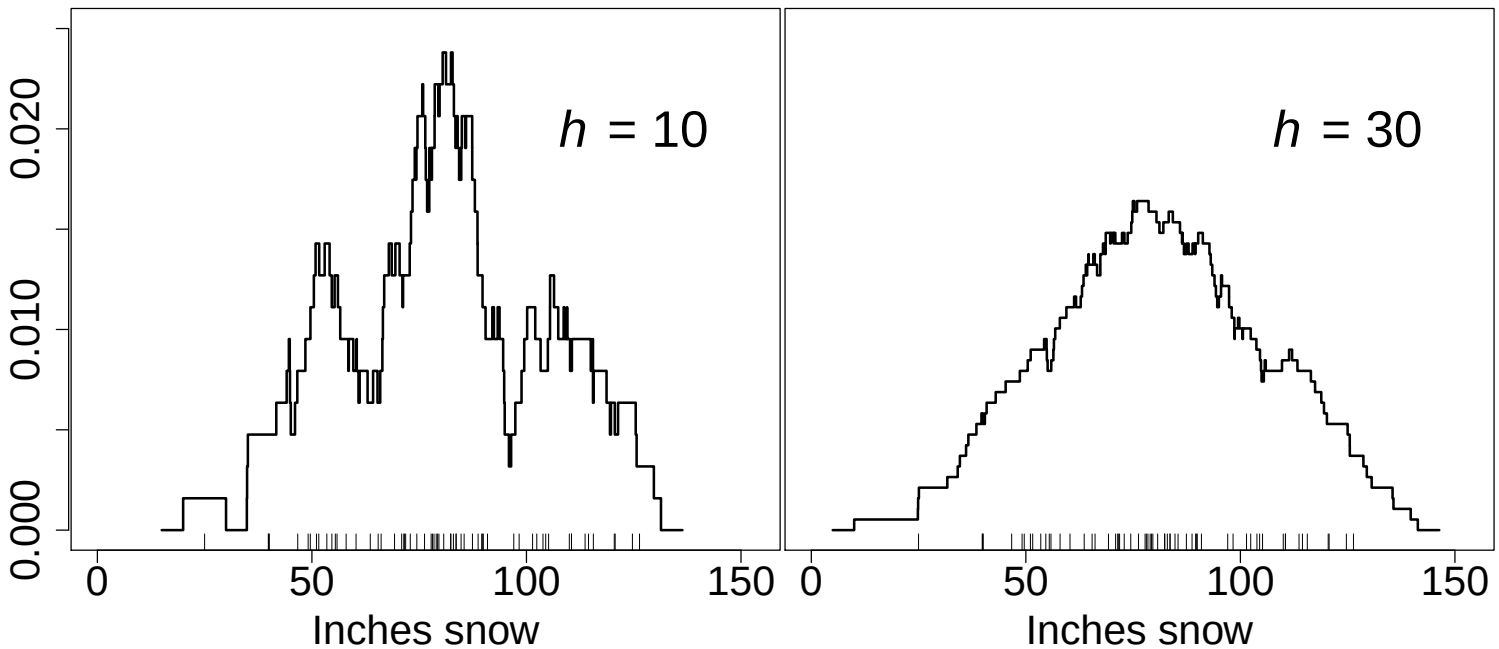

Figure: Central difference estimates of the Buffalo snowfall data.

Smoothing by Convolution

- An engineer would smooth a noisy function, f, by convolving with a smooth filter,

$$(f * w)(x) = \int_{-\infty}^{\infty} f(t)\, w(x-t)\, dt$$

- Recall f_n is the generalized derivativer of the ecdf, F_n:

$$\begin{aligned}\left[\frac{dF_n}{dx}\right] * w &= \int_{-\infty}^{\infty} \left[\frac{1}{n}\sum_{i=1}^{n} \delta(t-x_i)\right] w(x-t)\, dt \\ &= \frac{1}{n}\sum_{i=1}^{n} \left[\int_{-\infty}^{\infty} \delta(t-x_i)\, w(x-t)\, dt\right] \\ &= \frac{1}{n}\sum_{i=1}^{n} w(x-x_i)\end{aligned}$$

  which is precisely a KDE with kernel $K_h(\cdot) = w(\cdot)$

- Engineers speak of low-pass filters and half-power points etc

Orthogonal Series Approximations

- Suppose the true density is supported on $(0, 1)$ and is periodic
- So a Fourier series basis, $\phi_\nu(t) = \exp(2\pi i \nu t)$ is appropriate
- We write the density as the Fourier series

$$f(x) = \sum_{\nu=-\infty}^{\infty} f_\nu \, \phi_\nu(x)$$

where

$$\begin{aligned} f_\nu &= < f, \phi_\nu > = \int_0^1 f(x) \, \phi_\nu^*(x) \, dx \\ &= E \, \phi_\nu^*(X) \qquad \text{in statistical terms} \end{aligned}$$

$$\text{hence} \quad \hat{f}_\nu = \frac{1}{n} \sum_{\ell=1}^{n} \phi_\nu^*(x_\ell) \qquad \text{is an unbiased estimate}$$

Orthogonal Series Approximations (continued)

- Note that the Fourier cofficients for the epdf are

$$f_\nu = \int_0^1 \left[\frac{1}{n} \sum_{\ell=1}^{n} \delta(x - x_\ell) \right] \phi_\nu^*(x)\, dx = \frac{1}{n} \sum_{\ell=1}^{n} \phi_\nu^*(x_\ell) = \hat{f}_\nu$$

so that this orthogonal series estimator just reproduces the epdf!!! Tarter proposed truncating the number of Fourier coefficients included, while Wahba suggested applying a smooth window to accomplish the same task.

  - Tarter's boxcar filter

$$w_\nu(k) = \begin{cases} 1 & |\nu| \le k \\ 0 & \text{otherwise} \end{cases}$$

  - Wahba's tapering window filter

$$w_\nu(\lambda, p) = \frac{1}{1 + \lambda(2\pi\nu)^{2p}} \qquad \text{for } |\nu| \le n/2$$

Orthogonal Series Approximations (continued)

- Both forms of the weighted/truncated Fourier estimate may be written explicitly as

$$\begin{aligned}\hat{f}(x) &= \sum_{\nu} w_\nu \left[\frac{1}{n}\sum_{\ell=1}^{n} \phi_\nu^*(x_\ell)\right] \phi_\nu(x) \\ &= \frac{1}{n}\sum_{\ell=1}^{n}\left[\sum_{\nu} w_\nu\, \phi_\nu^*(x_\ell)\, \phi_\nu(x)\right] \\ &= \frac{1}{n}\sum_{\ell=1}^{n}\left[\sum_{\nu} w_\nu\, e^{2\pi i \nu (x - x_\ell)}\right]\end{aligned}$$

where the bracketed quantity is the equivalent kernel.

Orthogonal Series Approximations (concluded)

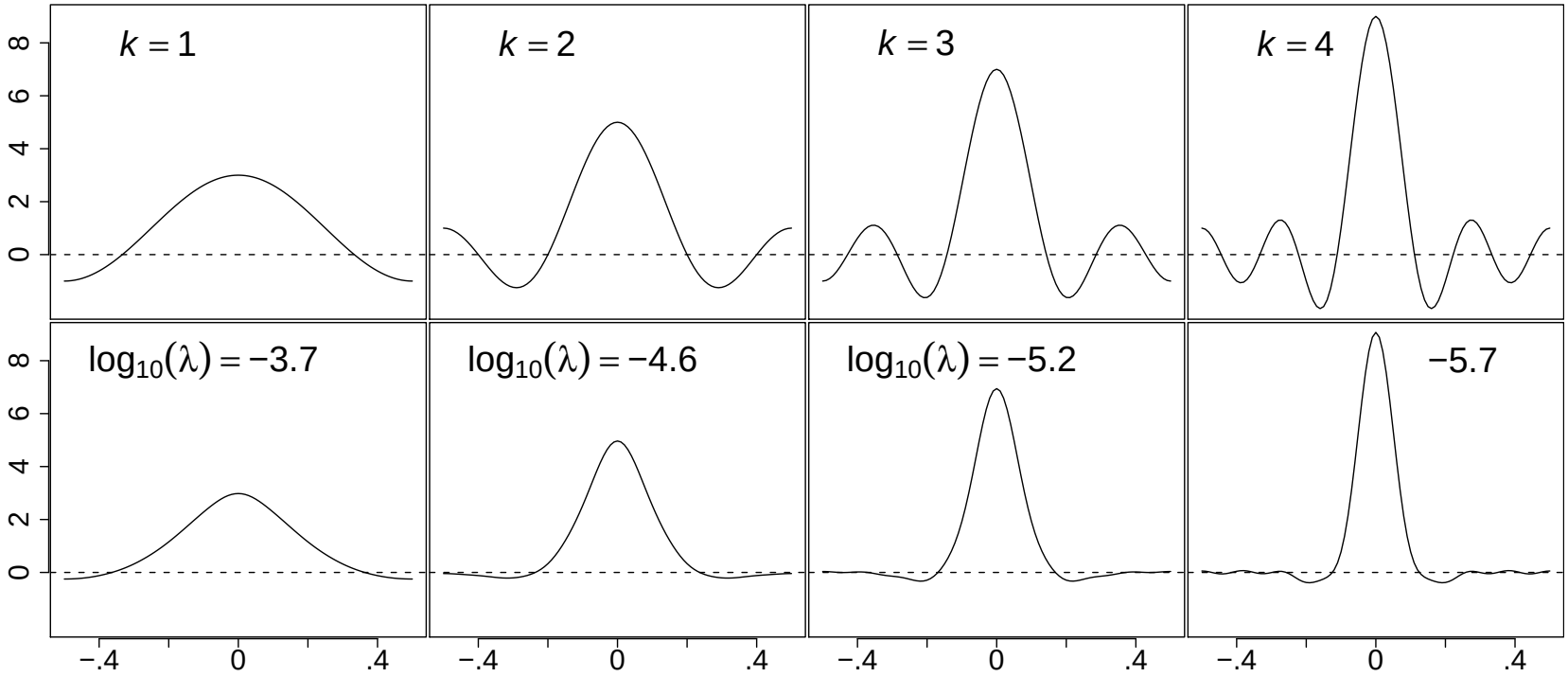

Figure: Examples of equivalent kernels for orthogonal series estimators. The 4 Wahba kernels (bottom row) have been selected to match the peak height of the corresponding Kronmal–Tarter–Watson kernels (top row). The Kronmal–Tarter–Watson kernels are independent of sample size; the Wahba examples are for $n = 16$.

Theoretical Properties: Univariate Case

- The kernel estimator is much easier to analyze than the binned estimators, since it is the arithemtic mean of n iid random variables

$$\frac{1}{h}K\left(\frac{x - X_i}{h}\right) \equiv K_h(x, X_i)$$

- Therefore,

$$E\{\hat{f}(x)\} = \mathrm{E}K_h(x, X) \quad \text{and}$$
$$var\ \{\hat{f}(x)\} = \frac{1}{n}\mathrm{var}\ K_h(x, X)$$

Theoretical Properties: Univariate Case (cont'd)

- Now

$$\mathrm{E}K_h(x,X) = \int \frac{1}{h}K\left(\frac{x-t}{h}\right) f(t)\,dt = \int K(w)f(x-hw)\,dw$$
$$= f(x)\int K(w) - hf'(x)\int wK(w) + \frac{1}{2}h^2 f''(x)\int w^2K(w) + \cdots$$

- And

$$\text{var } K_h(x,X) = \mathrm{E}\left[\frac{1}{h}K\left(\frac{x-X}{h}\right)\right]^2 - \left[\mathrm{E}\frac{1}{h}K\left(\frac{x-X}{h}\right)\right]^2$$

- We just computed the second term as $[f(x)\int K(w) + \cdots]^2$, while the first term is

$$\int \frac{1}{h^2}K\left(\frac{x-t}{h}\right)^2 f(t)\,dt = \int \frac{1}{h}K(w)^2 f(x-hw)\,dw \approx \frac{f(x)R(K)}{h}$$

Theoretical Properties: Univariate Case (cont'd)

- If the kernel satisfies the three constraints

$$\int K(w) = 1, \quad \int wK(w) = 0, \quad \text{and} \quad \int w^2 K(w) \equiv \sigma_K^2 > 0$$

- then the expectation $EK_h(x, X)$ becomes

$$= f(x) \int K(w) - hf'(x) \int wK(w) + \frac{1}{2} h^2 f''(x) \int w^2 K(w) + \cdots$$

$$= f(x) - 0 + \frac{1}{2} h^2 \sigma_K^2 f''(x) + \cdots$$

so that the bias is $O(h^2)$ like the FP.

- In fact,

$$\text{Bias}(x) = \frac{1}{2} h^2 \sigma_K^2 f''(x) + O(h^4) \;\Rightarrow\; \text{ISB} = \frac{1}{4} h^4 \sigma_K^4 R(f'') + O(h^6)$$

Theoretical Properties: Univariate Case (cont'd)

- Similarly,

$$\text{var}\,(x) = \frac{f(x)R(K)}{nh} - \frac{f(x)^2}{n} + O\left(\frac{h}{n}\right) \Rightarrow$$

$$\text{IV} = \frac{R(K)}{nh} - \frac{R(f)}{n} + \cdots .$$

- Assembling, we have shown

$$AMISE = \frac{R(K)}{nh} + \frac{1}{4}h^4\sigma_K^4 R(f'')$$

$$h^* = \left[\frac{R(K)}{\sigma_K^4 R(f'')}\right]^{1/5} n^{-1/5}$$

$$AMISE^* = \frac{5}{4}[\sigma_K R(K)]^{4/5} R(f'')^{1/5} n^{-4/5}$$

Comments

- It is easy to check that the ratio of AIV to AISB in the AMISE* is 4:1 (versus 2:1 for a histogram)
- Since $R(\phi''(x|0,\sigma^2)) = 3/(8\sqrt{\pi}\sigma^5)$, the normal reference rule bandwidth with a normal kernel is

$$\boxed{\text{normal reference rule}: \quad h = (4/3)^{1/5}\sigma\, n^{-1/5} \approx 1.06\,\hat{\sigma}\, n^{-1/5}}$$

Estimation of Derivatives

- Estimation of the derivative is straight-forward

$$\hat{f}^{(r)}(x) = \frac{d^r}{dx^r}\frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) = \frac{1}{nh^{r+1}}\sum_{i=1}^{n} K^{(r)}\left(\frac{x - x_i}{h}\right).$$

- To see how the variance and bias behave:

$$\text{var}\left\{\hat{f}^{(r)}(x)\right\} \approx \frac{n}{(nh^{r+1})^2}\text{E}\left[K^{(r)}\left(\frac{x - X}{h}\right)^2\right] \approx \frac{f(x)R(K^{(r)})}{nh^{2r+1}};$$

$$\text{E}\hat{f}'(x) = \frac{1}{h}\left[f_x\int K' - hf'_x\int wK' + \frac{h^2}{2}f''_x\int w^2K' - \frac{h^3}{6}f'''_x\int w^3K' + \cdot\right]$$

where $f_x^{(r)} \equiv f^{(r)}(x)$.

- If K is symmetric, $\int w^r K' = 0$ for even r. while $\int wK' = -1$ and $\int w^3K' = -3\sigma_K^2$ integrating by parts.

Estimation of Derivatives: Theoretical Summary

- Combining these results gives us

$$AMISE(\hat{f}^{(r)}) = \frac{R(K^{(r)})}{nh^{2r+1}} + \frac{1}{4}h^4\sigma_K^4 R(f^{(r+2)})$$

$$h_r^* = \left[\frac{(2r+1)R(K^{(r)})}{\sigma_K^4 R(f^{(r+2)})}\right]^{1/(2r+5)} n^{-1/(2r+5)}$$

$$AMISE^*(\hat{f}^{(r)}) = \frac{2r+5}{4} R(K^{(r)})^{\frac{4}{2r+5}} \left[\sigma_K^4 R(f^{(r+2)})/(2r+1)\right]^{\frac{2r+1}{2r+5}} n^{\frac{-4}{2r+5}}$$

- The bias term remains $O(h^4)$, but each additional derivative order introduces 2 extra powers of h in the variance.
- The optimal smoothing parameters h^* for the first and second derivatives are $O(n^{-1/7})$ and $O(n^{-1/9})$, respectively, while the AMISE* is $O(n^{-4/7})$ and $O(n^{-4/9})$.
- Each order of derivative is as hard as adding 2 dimensions to the density!

Choice of Kernel

- Turns out that any probability density with mean 0 that is symmetric and unimodal is good enough to be a kernel.
- Common choices are $N(0,1)$ and $Beta(k+1,k+1)$ shifted to the interval $(-1,1)$:

$$K(t) \propto (1-t^2)_+^k$$

- For small values of k, these are often named
  - $k=0$ Rosenblatt's $U(-1,1)$ kernel
  - $k=1$ Epanechnikov's kernel (oversmoothed)
  - $k=2$ Tukey's biweight kernel
  - $k=3$ Triweight kernel (used by Bill Cleveland in lowess)
- In fact, the limit of these kernels is $N(0,1)$, properly rescaled.

Higher-Order Kernels

- Recall the bias of the KDE is $\frac{1}{2}h^2\sigma_K^2 f''(x)$
- Can you zero this out by choosing a kernel with $\sigma_K^2 = 0$?
- Answer: Not if $K(t) \geq 0$.
- But if allow negative kernels, can choose an order-p kernel satisfying

$$\int K = 1; \quad \int w^i K = 0, \quad i = 1, \cdots, p-1; \quad \text{and} \quad \int w^p K \neq 0$$

- Letting $\mu_i \equiv \int w^i K(w) dw$, the bias becomes

$$Bias\{\hat{f}(x)\} = \frac{1}{p!} h^p \mu_p f^{(p)}(x) + \cdots$$

AMISE with Higher-Order Kernels

- The variance expression is unchanged, so have

$$AMISE(h) = \frac{R(K)}{nh} + \frac{1}{(p!)^2}\mu_p^2 R(f^{(p)})\, h^{2p}$$

$$h^* = \left[\frac{(p!)^2 R(K)}{2p\mu_p^2 R(f^{(p)})}\right]^{1/(2p+1)} n^{-1/(2p+1)}$$

$$AMISE^* = \frac{2p+1}{2p}\left[2p\,\mu_p^2\, R(K)^{2p} R(f^{(p)})/(p!)^2\right]^{1/(2p+1)} n^{-2p/(2p+1)}$$

- Note that as p increases, the rate of convergence approaches the magical $O(n^{-1})$
- In practice, do not see much gain for $p > 4$

Some Examples of Higher-Order Kernels

Table: Some Simple Polynomial Higher-Order Kernels

p	K_p on $(-1,1)$	$N(0,1)$ AMISE*
2	$\frac{3}{4}(1-t^2)$	$0.320n^{-4/5}$
4	$\frac{15}{32}(1-t^2)(3-7t^2)$	$0.482n^{-8/9}$
6	$\frac{105}{256}(1-t^2)(5-30t^2+33t^4)$	$0.581n^{-12/13}$
8	$\frac{315}{4{,}096}(1-t^2)(35-385t^2+1{,}001t^4-715t^6)$	$0.681n^{-16/17}$

Some Examples of Higher-Order Kernels

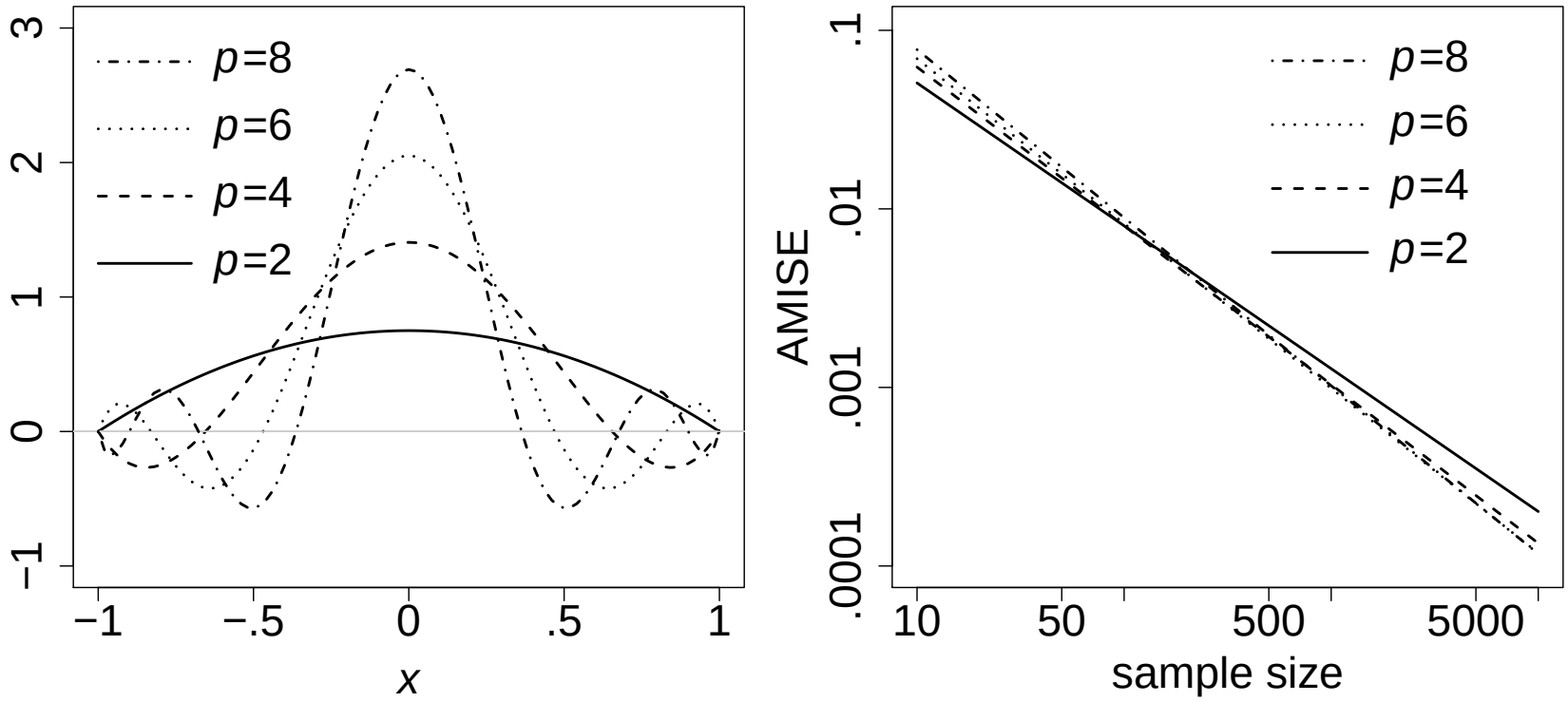

Figure: Examples of higher-order kernels that are low-order polynomials. The right panel shows the corresponding $N(0,1)$ AMISE* curves on a log–log scale.

Some Examples of Higher-Order Kernels

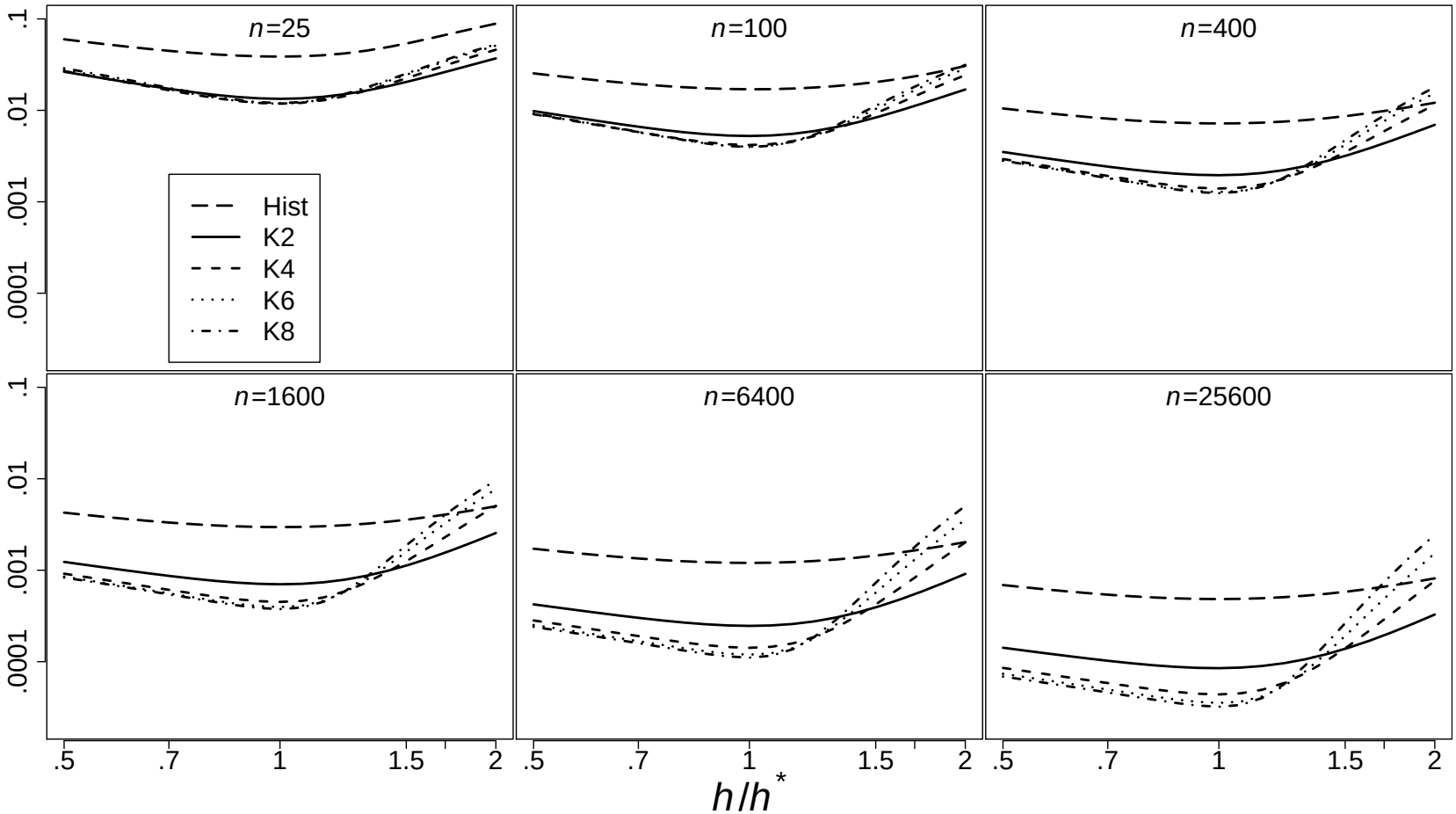

Figure: Exact MISE using higher-order kernels with normal data for several sample sizes. The histogram MISE is included for reference.

Some Examples of Higher-Order Kernels

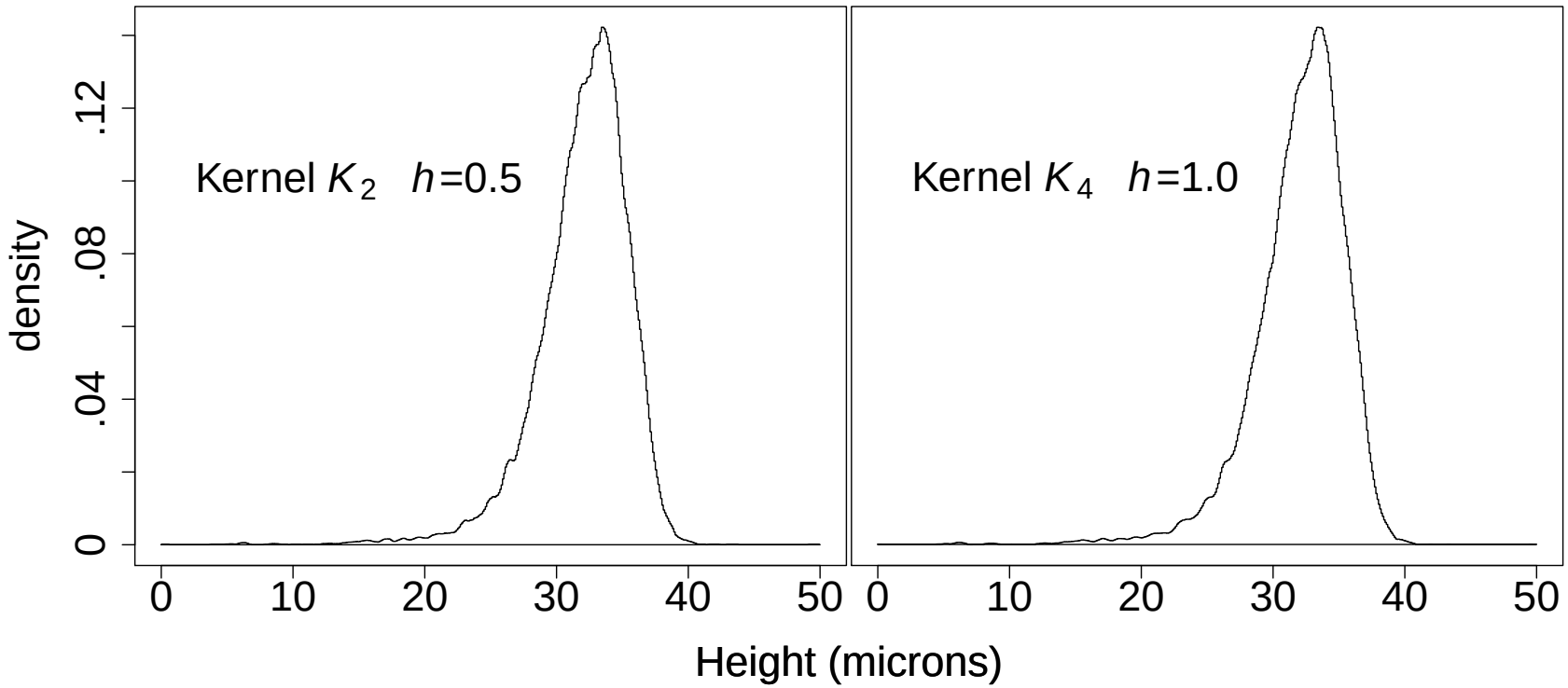

Figure: Positive and negative ASH estimates of the steel surface data. Kernels used were K_2 and K_4.

Some Examples of Higher-Order Kernels

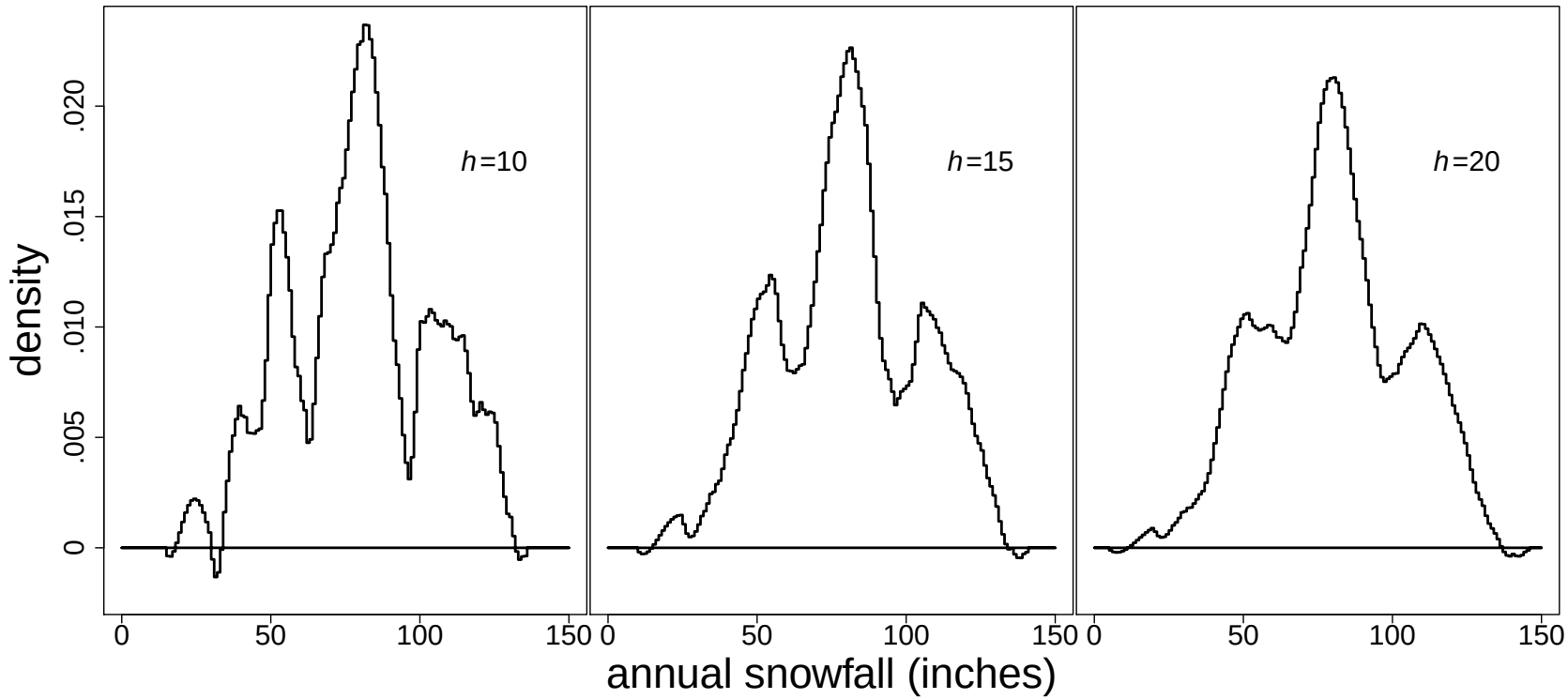

Figure: Kernel K_4 applied to the Buffalo snowfall data with three smoothing parameters. The ASH estimate is depicted in its histogram form. Notice the estimate gets rougher as h increases. Unusual.

Optimal Kernels

- The best kernel minimizes $AMISE^*$

$$AMISE^* \propto [\sigma_K R(K)]^{4/5}$$

- Leading to the optimization problem

$$\min_K R(K) \quad \text{s/t} \quad \sigma_K^2 = \sigma^2$$

- The solution is

$$K_2^*(t) = \frac{3}{4}(1 - t^2) I_{[-1,1]}(t)$$

- Any other kernel requires slightly more data to achieve the same $AMISE$

$$\frac{\sigma_K R(K)}{\sigma_{K_2^*} R(K_2^*)} = \frac{\sigma_K R(K)}{3/(5\sqrt{5})}$$

Table: Some Common and Some Unusual Kernels and Their Relative Efficiencies. All kernels are supported on $[-1, 1]$ unless noted otherwise.

Kernel	Equation	$\sigma_K R(K)$	Eff.
Uniform	$U(-1, 1)$	0.2887	1.0758
Triangle	$(1 - \lvert t\rvert)_+$	0.2722	1.0143
Epanechnikov	$\frac{3}{4}(1 - t^2)_+$	0.2683	1
Biweight	$\frac{15}{16}(1 - t^2)_+^2$	0.2700	1.0061
Triweight	$\frac{35}{32}(1 - t^2)_+^3$	0.2720	1.0135
Normal	$N(0, 1)$	0.2821	1.0513
Cosine arch	$\frac{\pi}{4}\cos\frac{\pi}{2}t$	0.2685	1.0005
Indifferent FP	See Problem	0.2750	1.0249
Dble. exp.	$\frac{1}{2}e^{-\lvert t\rvert}, \lvert t\rvert \leq \infty$	0.3536	1.3176
Skewed	$2860(t + \frac{2}{7})_+^3(\frac{5}{7} - t)_+^9$	0.2835	1.0567
Dble. Epan.	$3\lvert t\rvert(1 - \lvert t\rvert)_+$	0.3286	1.2247
Shifted exp.	$e^{-(t+1)}, t > -1$	0.5743	1.8634
FP	See Theorem	0.3405	1.2690

Equivalent Kernels

- It is easy enough to use the formulae for h^* so the kernel estimates are essentially the same for different kernels:

$$\frac{h_1^*}{h_2^*} = \left[\frac{R(K_1)/\sigma_{K_1}^4}{R(K_2)/\sigma_{K_2}^4}\right]^{1/5} = \frac{\sigma_{K_2}}{\sigma_{K_1}}\left[\frac{\sigma_{K_1}R(K_1)}{\sigma_{K_2}R(K_2)}\right]^{1/5}.$$

- Recall that the factors $\sigma_K R(K)$ are essentially the same for all kernels, we may use the approximation

Equivalent kernel rescaling: $\quad h_2^* \approx \frac{\sigma_{K_1}}{\sigma_{K_2}} h_1^*.$

Table: Factors for Equivalent Smoothing Among Popular Kernels[a]

From \ To	Normal	Uniform	Epan.	Triangle	Biwt.	Triwt.
Normal	1	1.740	2.214	2.432	2.623	2.978
Uniform	0.575	1	1.272	1.398	1.507	1.711
Epanech.	0.452	0.786	1	1.099	1.185	1.345
Triangle	0.411	0.715	0.910	1	1.078	1.225
Biwt.	0.381	0.663	0.844	0.927	1	1.136
Triwt.	0.336	0.584	0.743	0.817	0.881	1

[a]To go from h_1 to h_2, multiply h_1 by the factor in the table in the row labeled K_1 and in the column labeled K_2.

Boundary Kernels

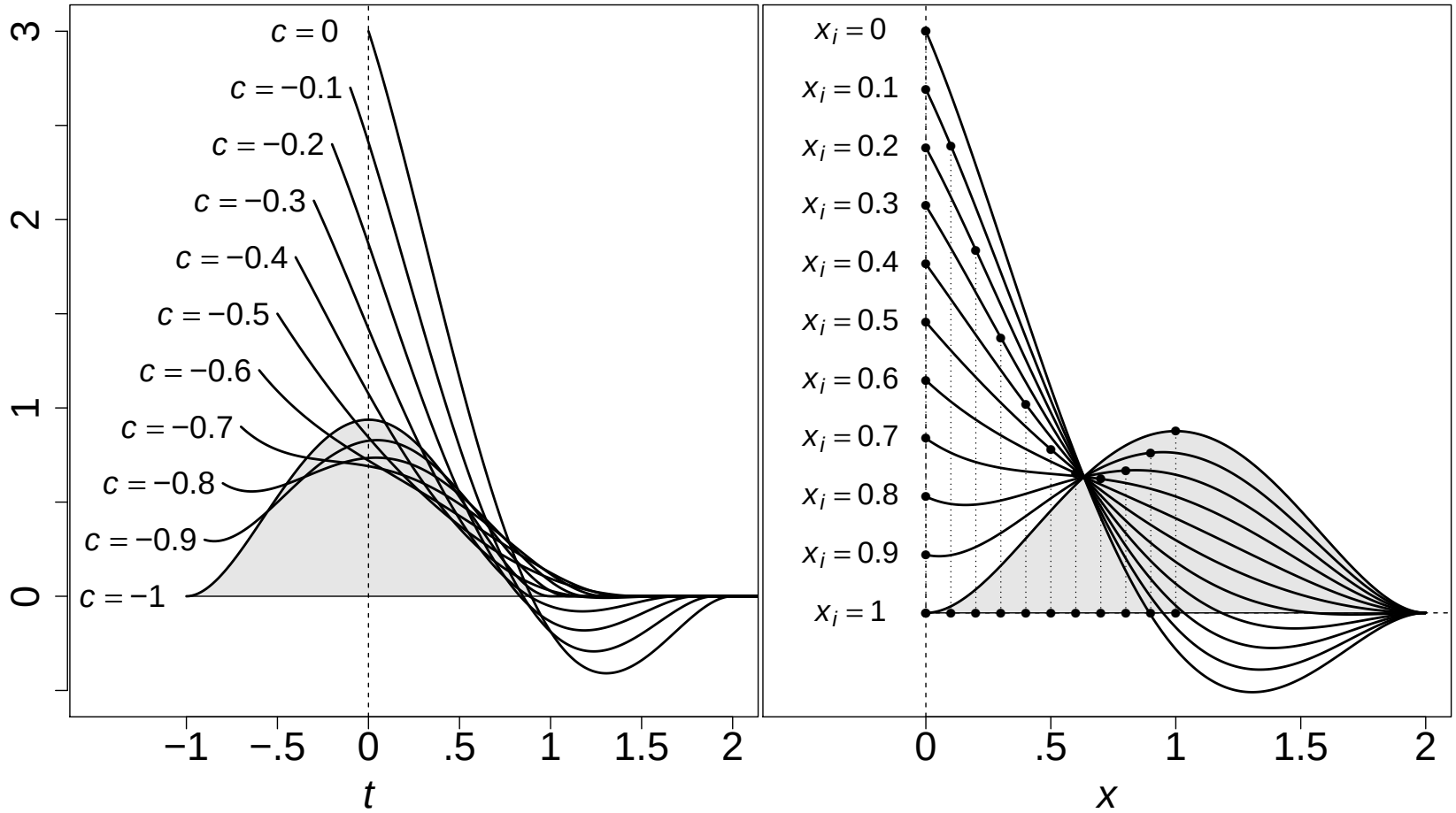

Figure: *(Left frame)* Examples of the "floating" boundary kernels $K_c(t)$, where $-1 < c < 0$. *(Right frame)* Assuming the boundary $x \geq 0$, each kernel $K_c(t)$ is drawn centered on the data point, $x_i = -c$, which is indicated by the dashed vertical line.

Zero Constrained Boundary Kernels

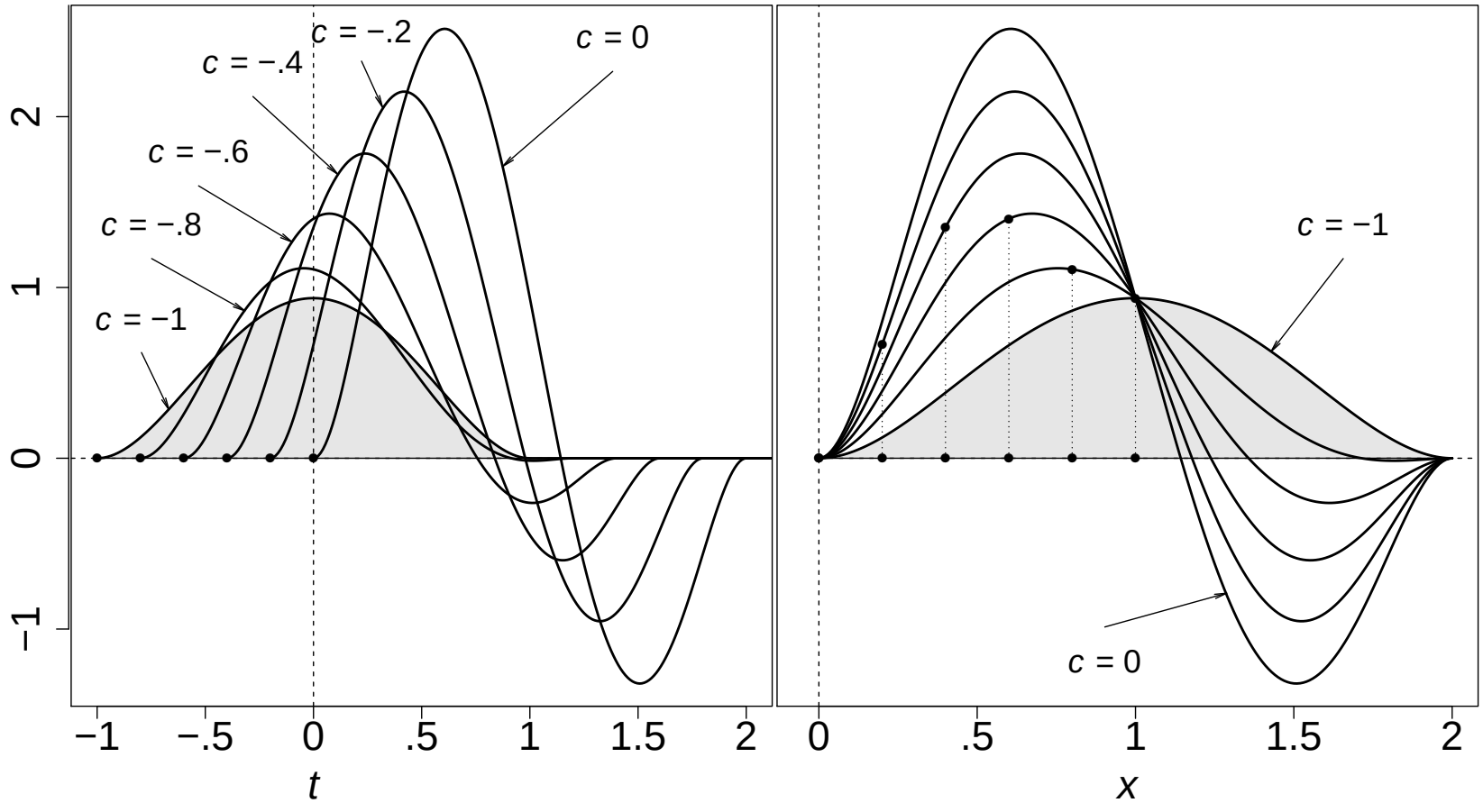

Figure: Examples of "zero" boundary kernels as in Figure **??**.

Boundary Kernels Applied

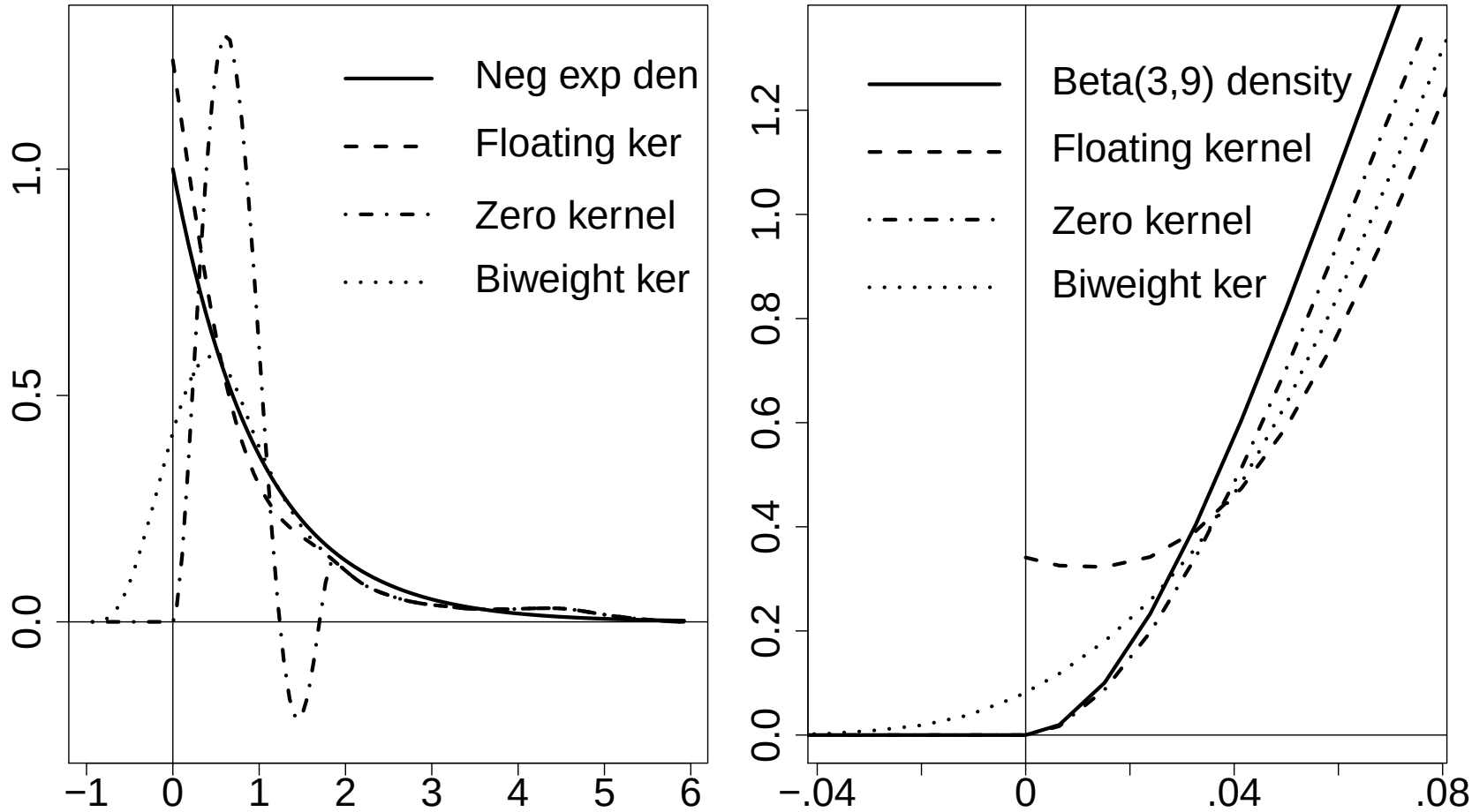

Figure: *(Left frame)* Example with negative exponential data—with and without boundary modification for $n = 100$ and $h = 0.93$. The "floating" and "zero" boundary kernels are defined in Equations (**??**) and (**??**), respectively. *(Right frame)* Example with Beta(3,9) density in a neighborhood of 0 for $n = 100$ and $h = 0.11$.

Application of Both Boundary Kernels

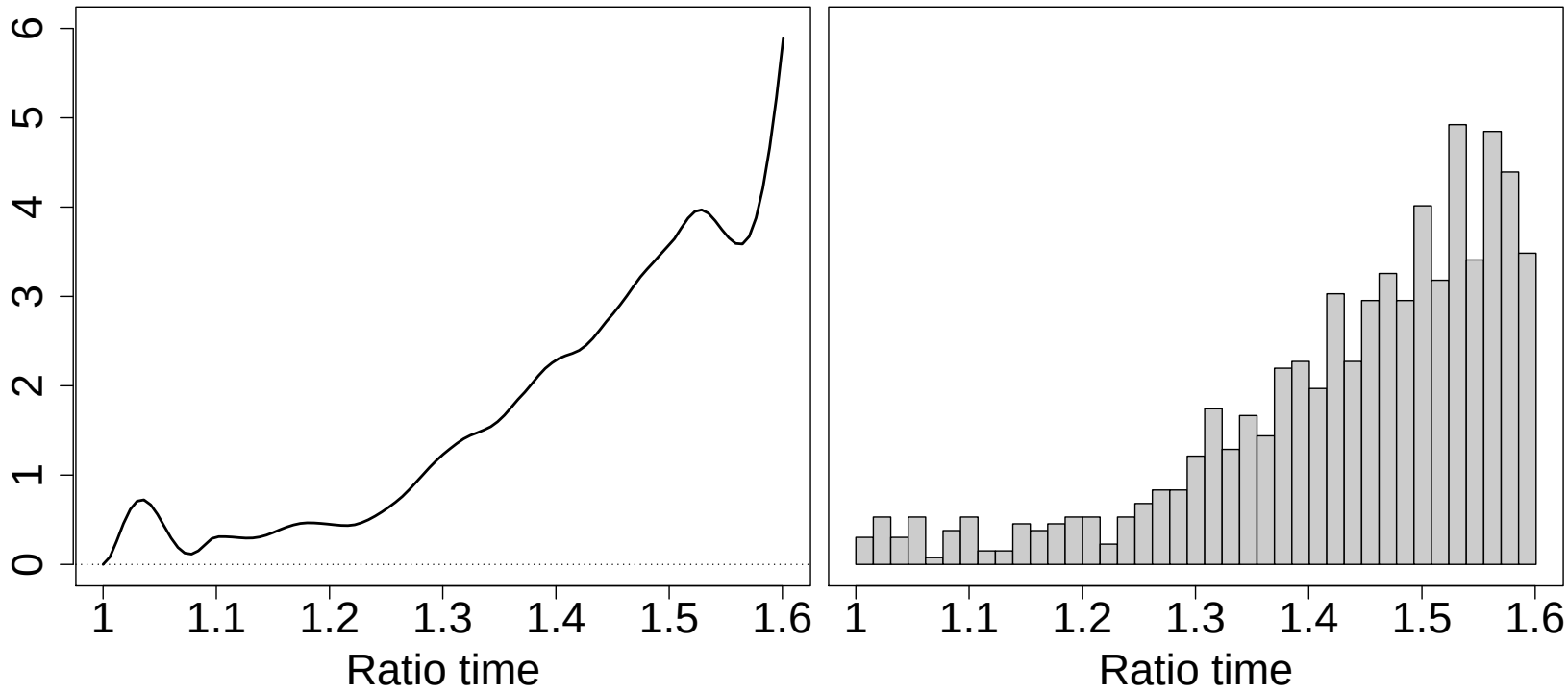

Figure: Density estimate of 857 fastest times in the 1991 Houston Tenneco Marathon. The data are the ratio to the leader's time for the race. Different boundary kernels were used on each extreme. A histogram is shown for comparison.

Product Kernels

- The general form of a product kernel estimator is given by

$$\hat{f}(\mathbf{x}) = \frac{1}{nh_1 \cdots h_d} \sum_{i=1}^{n} \left\{ \prod_{j=1}^{d} K\left(\frac{x_j - x_{ij}}{h_j} \right) \right\}.$$

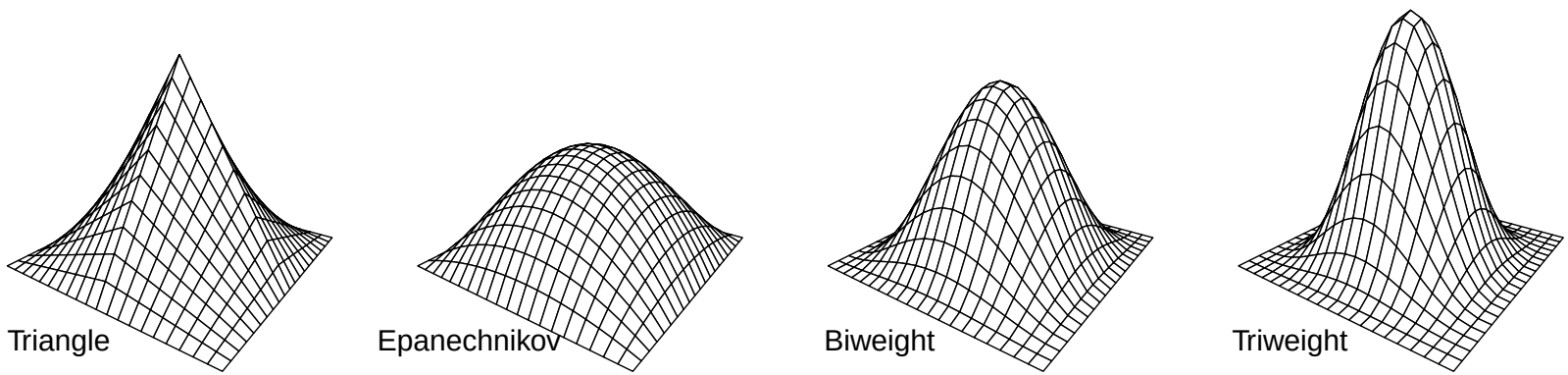

Figure: Product kernel examples for 4 kernels.

a

Multivariate Rules of Thumb

- We won't go into the *AMISE* details, but those results are useful

$$\text{normal reference rule :} \qquad h_i^* = \left(\frac{4}{d+2}\right)^{1/(d+4)} \sigma_i \; n^{-1/(d+4)}$$

- The coefficient ranges over the interval (0.924, 1.059), with a limit equal to 1 as $d \to \infty$; hence, an easy-to-remember formula is

$$\text{Scott's rule in } \Re^d : \qquad \hat{h}_i = \hat{\sigma}_i \, n^{-1/(d+4)}$$

More General Kernels

- Let H be a $d \times d$ nonsingular matrix and $K : R^d \to R^1$ be a kernel satisfying conditions given below.
- Then the general multivariate kernel estimator is

$$\hat{f}(\mathbf{x}) = \frac{1}{n|H|} \sum_{i=1}^{n} K(H^{-1}(\mathbf{x} - \mathbf{x}_i)) .$$

- Kernel conditions

$$\int_{\Re^d} K(\mathbf{w})\, d\mathbf{w} = 1$$

$$\int_{\Re^d} \mathbf{w} K(\mathbf{w})\, d\mathbf{w} = \mathbf{0}_d$$

$$\int_{\Re^d} \mathbf{w}\mathbf{w}^T K(\mathbf{w})\, d\mathbf{w} = I_d$$

Easy Out

- Consider the kernel $N(\mathbf{0}, \mathbf{\Sigma})$ $\quad K(\mathbf{t}) \propto \exp\left(-\frac{1}{2}\mathbf{t}^t\mathbf{\Sigma}^{-1}\mathbf{t}\right)$
- Recall the eigen (spectral) representation of $\mathbf{\Sigma}$, which gives us several useful formulae

$$\begin{aligned}\mathbf{\Sigma} &= \mathbf{\Gamma}\mathbf{\Lambda}\mathbf{\Gamma}^t \\ \mathbf{\Sigma}^{-1} &= \mathbf{\Gamma}\mathbf{\Lambda}^{-1}\mathbf{\Gamma}^t \\ \mathbf{\Sigma}^{-1/2} &= \mathbf{\Gamma}\mathbf{\Lambda}^{-1/2}\mathbf{\Gamma}^t\end{aligned}$$

  where $\mathbf{\Lambda}$ contains eigenvalues and $\mathbf{\Gamma}$ contains eigenvectors
- Then a typical term in the KDE becomes

$$\exp\left(-\frac{1}{2}(\mathbf{x}-\mathbf{x}_i)^t\mathbf{\Sigma}^{-1}(\mathbf{x}-\mathbf{x}_i)\right) = \exp\left(-\frac{1}{2}\mathbf{y}^t\mathbf{y}\right)$$

  letting $\mathbf{y} = \mathbf{\Sigma}^{-1/2}(\mathbf{x}-\mathbf{x}_i)$
- So equivalent to use a product kernel on the transformed data!

When is an Estimator Nonparametric?

- Terrell showed that **all** density estimators, even parametric estimators may be expressed as a **kernel estimator**
- Without going into details, the kernel estimator form of the MLE $N(\bar{x}, 1)$ is

$$K(x, y, F_n) = \frac{1 + (y - \bar{x})(x - \bar{x})}{\sqrt{2\pi}} e^{-\frac{1}{2}(x-\bar{x})^2}$$

so that the density estimate may be written as

$$\hat{f}(x) = \frac{1}{n} \sum_{i=1}^{n} \frac{1 + (x_i - \bar{x})(x - \bar{x})}{\sqrt{2\pi}} e^{-(x-\bar{x})^2/2} = \frac{1}{\sqrt{2\pi}} e^{-(x-\bar{x})^2/2}$$

- Terrell showed that a **parametric kernel** was not local; that is, the influence of the point x on y could be large, even when they were far apart
- A **nonparametric kernel** is **local**; for example, what happens in one histogram bin has almost no impact anywhere else.

Brief Overview of Cross-Validation (BCV and UCV)

- The same ideas that worked for the histogram work for kernel estimators
- For example, a plug-in estimator of $R(f'')$ is

$$R(\hat{f}_h'') = \frac{3}{8\sqrt{\pi}n^2h^5}\sum_{i=1}^{n}\sum_{j=1}^{n}\left(1 - \Delta_{ij}^2 + \frac{1}{12}\Delta_{ij}^4\right)e^{-\frac{1}{4}\Delta_{ij}^2}$$

where

$$\Delta_{ij} = \frac{x_i - x_j}{h}$$

- As with the histogram, this is a little too big

$$ER(\hat{f}_h'') = R(f'') + \frac{R(K'')}{nh^5} + O(h^2)$$

so we subtract off that constant term

UCV and *BCV* Formulae

- For a $N(0,1)$ kernel, the magic formulae are

$$UCV(h) = \frac{1}{2nh\sqrt{\pi}} + \frac{1}{n^2 h\sqrt{\pi}} \sum_{i<j} \left(e^{-\Delta_{ij}^2/4} - \sqrt{8}\, e^{-\Delta_{ij}^2/2} \right)$$

$$BCV(h) = \frac{1}{2nh\sqrt{\pi}} + \frac{1}{64n^2 h\sqrt{\pi}} \sum_{i<j} \left(\Delta_{ij}^4 - 12\Delta_{ij}^2 + 12 \right) e^{-\Delta_{ij}^2/4}$$

- Given their quite different motivations, it is remarkable how similar these are?
- Charles Taylor also proposed a bootstrap risk estimator:

$$\text{BMISE}_*(h) = \frac{1 + \frac{\sqrt{2}}{n} \sum_{i<j} \left[\sqrt{2}\, e^{-\frac{\Delta_{ij}^2}{4}} - \frac{4}{\sqrt{3}} e^{-\frac{\Delta_{ij}^2}{6}} + e^{-\frac{\Delta_{ij}^2}{8}} \right]}{2nh\sqrt{\pi}}$$

Examples

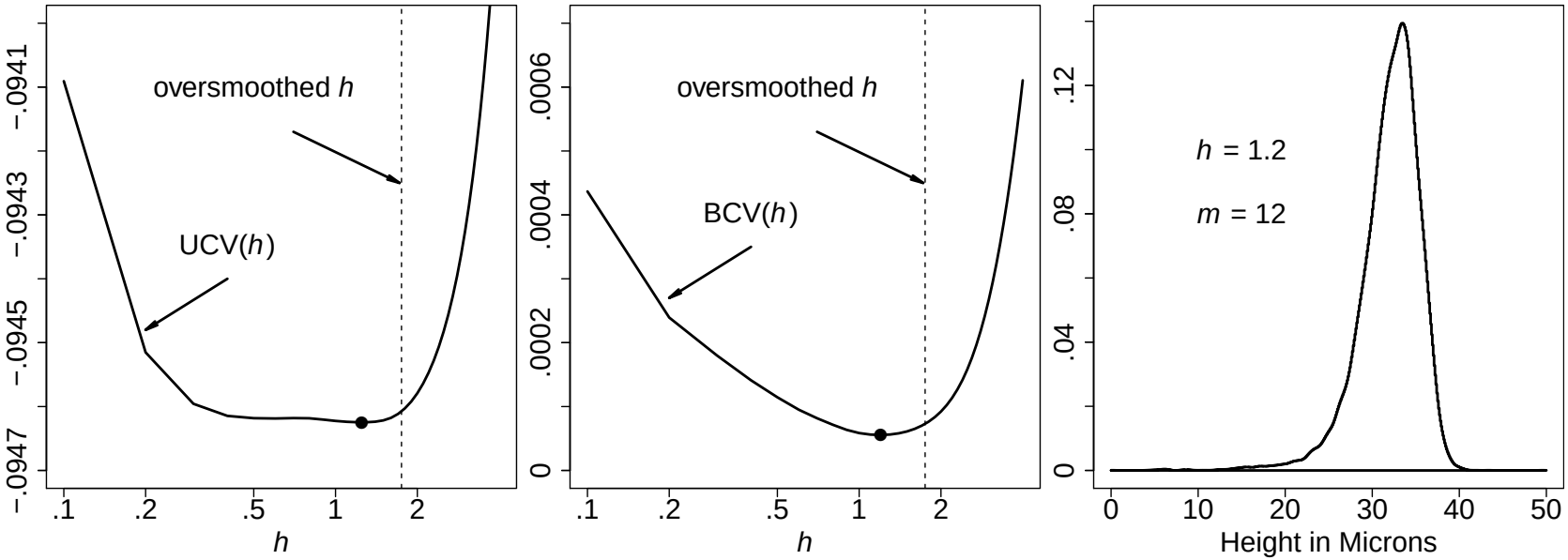

Figure: UCV and BCV estimates of the steel surface data ($n = 15,000$) using the triweight ASH. The UCV bandwidth was tied at 1.2 and 1.3, while the BCV bandwidth was 1.2 (shown): both estimates were virtually identical.

Buffalo Snowfall Data Example

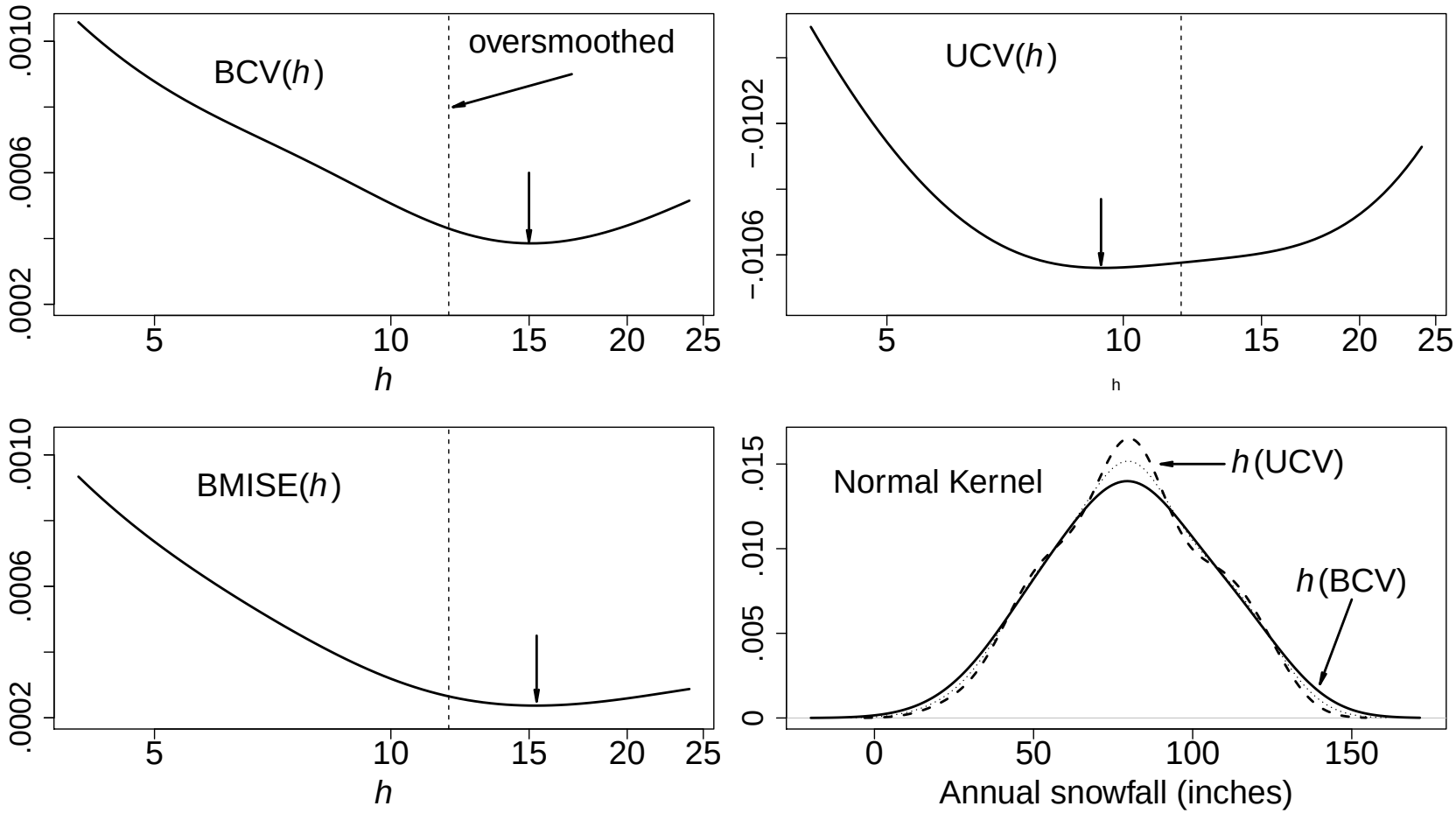

Figure: Normal kernel cross-validation algorithms and density estimates for the snowfall data ($n = 63$). The CV bandwidths are indicated by arrows and the oversmoothed bandwidth by the dashed line. The UCV, BCV, and oversmoothed density estimates are represented by the dashed, solid, and dotted lines, respectively.

Old Faithful Geyser Data Example

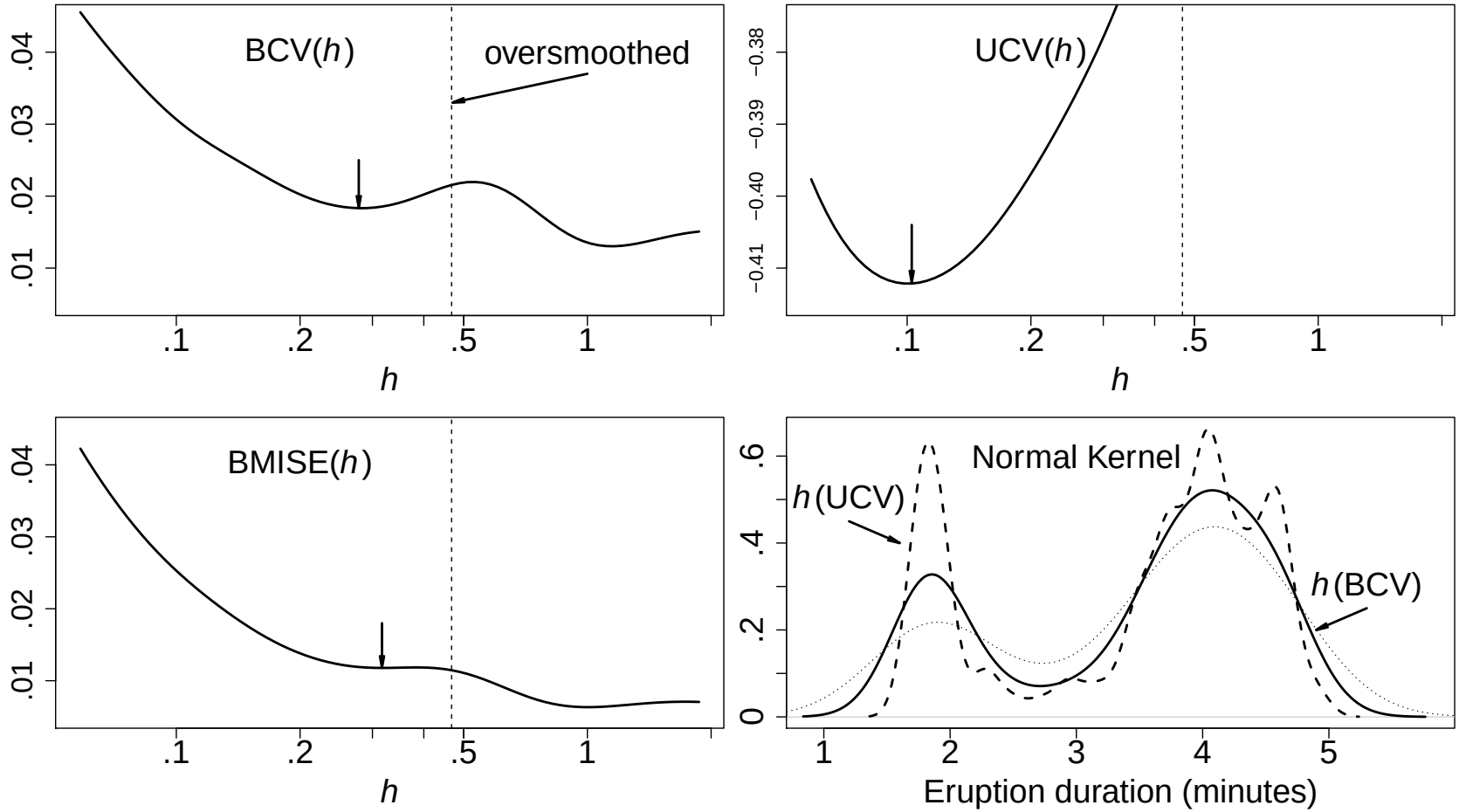

Figure: Normal kernel cross-validation algorithms and density estimates for the geyser dataset ($n = 107$). The CV bandwidths are indicated by arrows and the oversmoothed bandwidth by a solid line. The UCV, BCV, and oversmoothed density estimates are represented by the dashed, solid, and dotted lines, respectively.

Better Plug-In Cross-Validation

- Start with a slightly longer version of $AMISE(h)$:

$$AMISE(h) = \frac{R(K)}{nh} + \frac{1}{4}h^4\mu_2^2 R(f'') - \frac{1}{24}h^6\mu_2\mu_4 R(f''')$$

- Need to find estimates of $R(f''')$ as well as $R(f'')$
- Hall, Marron, Sheather, Jones introduce two auxiliary smoothing parameters, λ_1 and λ_2 and separate kernel estimates just for those functionals, resulting in the risk estimator

$$\boxed{\widehat{AMISE}(h) = \frac{R(K)}{nh} + \frac{1}{4}h^4\mu_2^2 \hat{R}_{\lambda_1}(f'') - \frac{1}{24}h^6\mu_2\mu_4 \hat{R}_{\lambda_2}(f''')}$$

- Rather than plotting, have the approximation

$$\boxed{\hat{h}_{\mathrm{PI}} = \left[\frac{\hat{J}_1}{n}\right]^{\frac{1}{5}} + \left[\frac{\hat{J}_1}{n}\right]^{\frac{3}{5}} \cdot \hat{J}_2\,; \quad \hat{J}_1 = \frac{R(K)}{\mu_2^2 \hat{R}_{\lambda_1}(f'')},\ \hat{J}_2 = \frac{\mu_4 \hat{R}_{\lambda_2}(f''')}{\mu_2 \hat{R}_{\lambda_1}(f'')}}$$

How Well Does This Work In Practice? Simulations

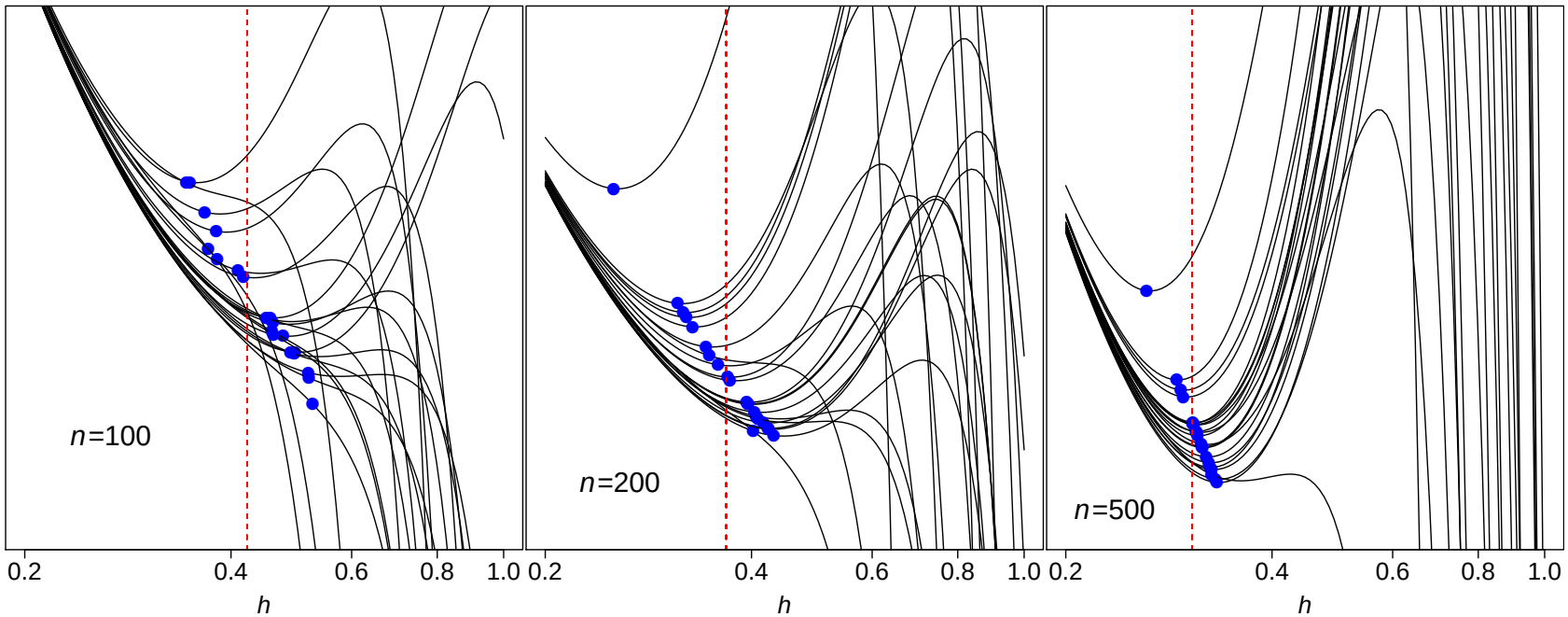

Figure: Twenty-one examples of the AMISE approximation of the plug-in rule with $N(0,1)$ data and a normal kernel. The plug-in bandwidth for each simulation is shown by the blue dot on the risk curve. The vertical dotted line indicates the normal reference rule (with $\sigma = 1$). Note that the horizontal axis is the same for each sample size, but the vertical scale (not labeled) zooms in on the relevant area.

How Well Does This Work In Practice? Examples

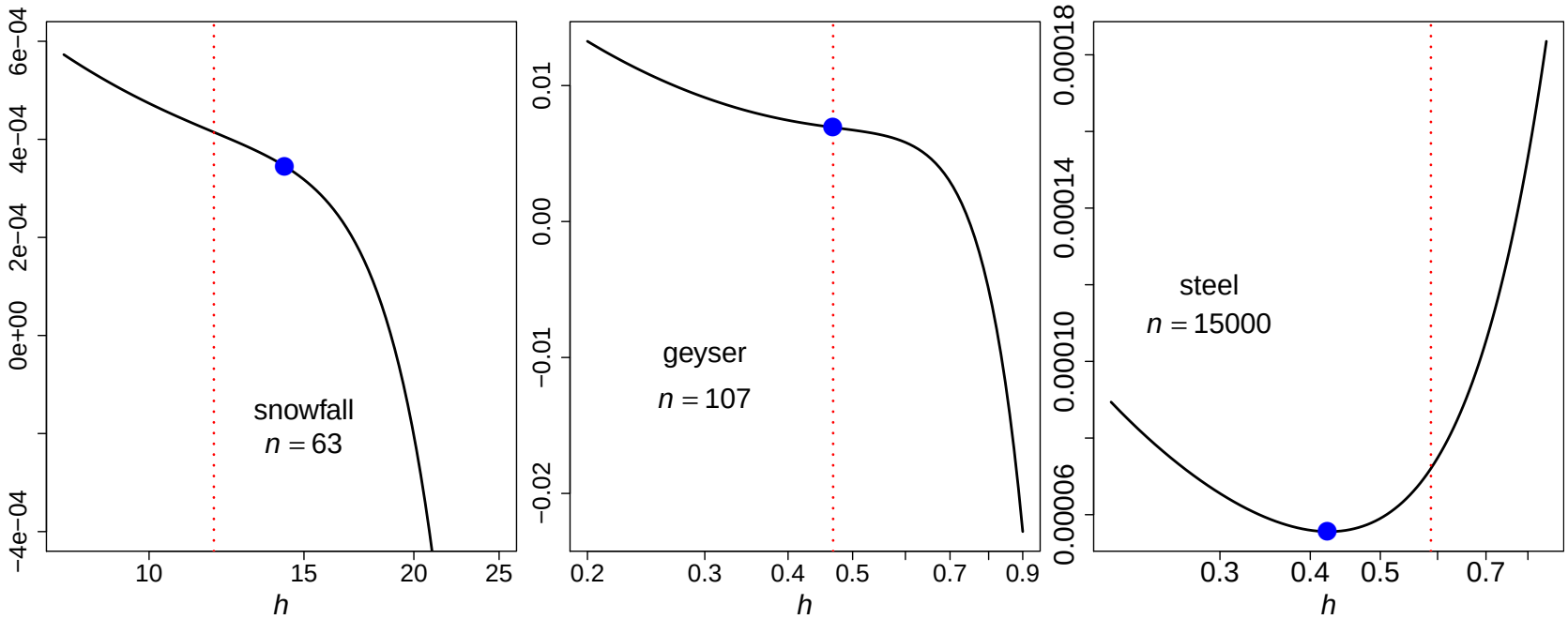

Figure: Plug-in cross-validation curves for the snowfall data ($n = 63$), the geyser dataset ($n = 107$) and the steel surface data ($n = 15{,}000$) for the normal kernel. The plug-in bandwidth obtained by the formula is indicated by the blue dot, and the oversmoothed bandwidth by the dashed red line.

Multivariate Cross-Validation (Simulation)

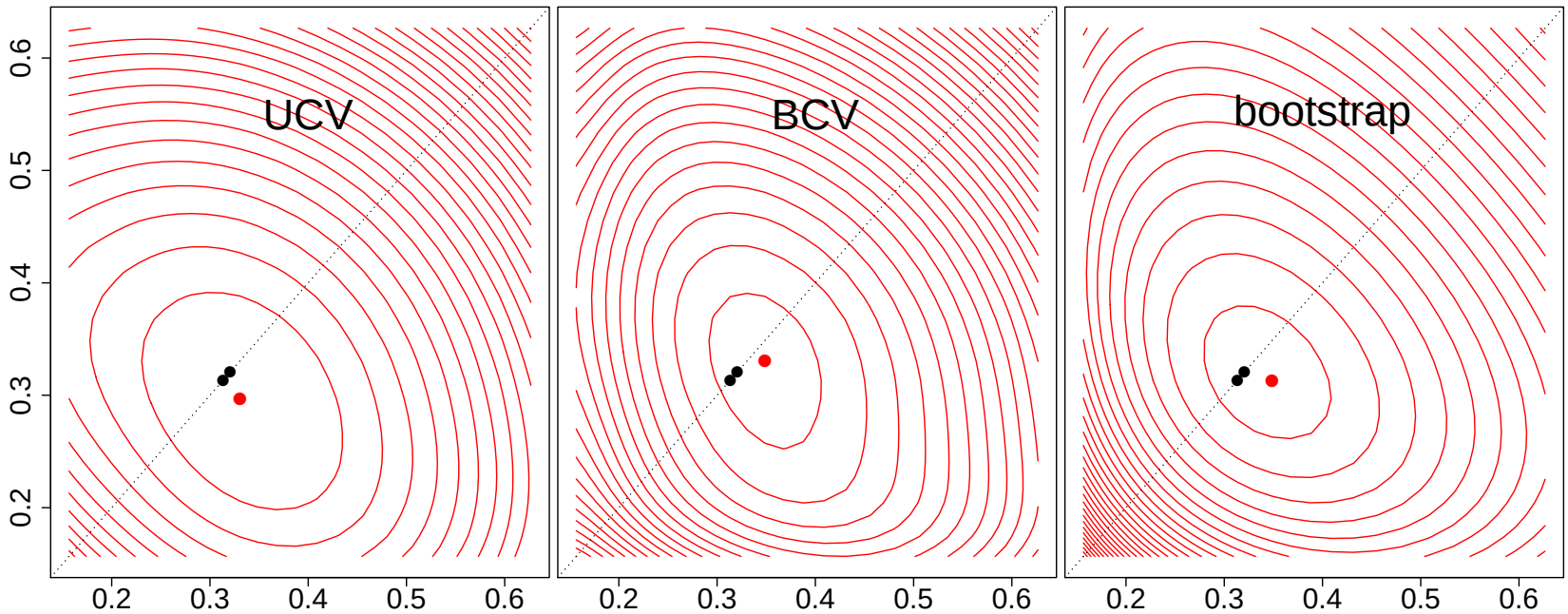

Figure: Estimated MISE(h_x, h_y) using UCV, BCV, and the bootstrap algorithms on 1,500 $N(\mathbf{0}_2, I_2)$ points. The two dots on each diagonal are h^* and the oversmoothed bandwidths. The dot locating the minimizer of each criterion is below the diagonal.

Multivariate Cross-Validation (Blood Fat Data)

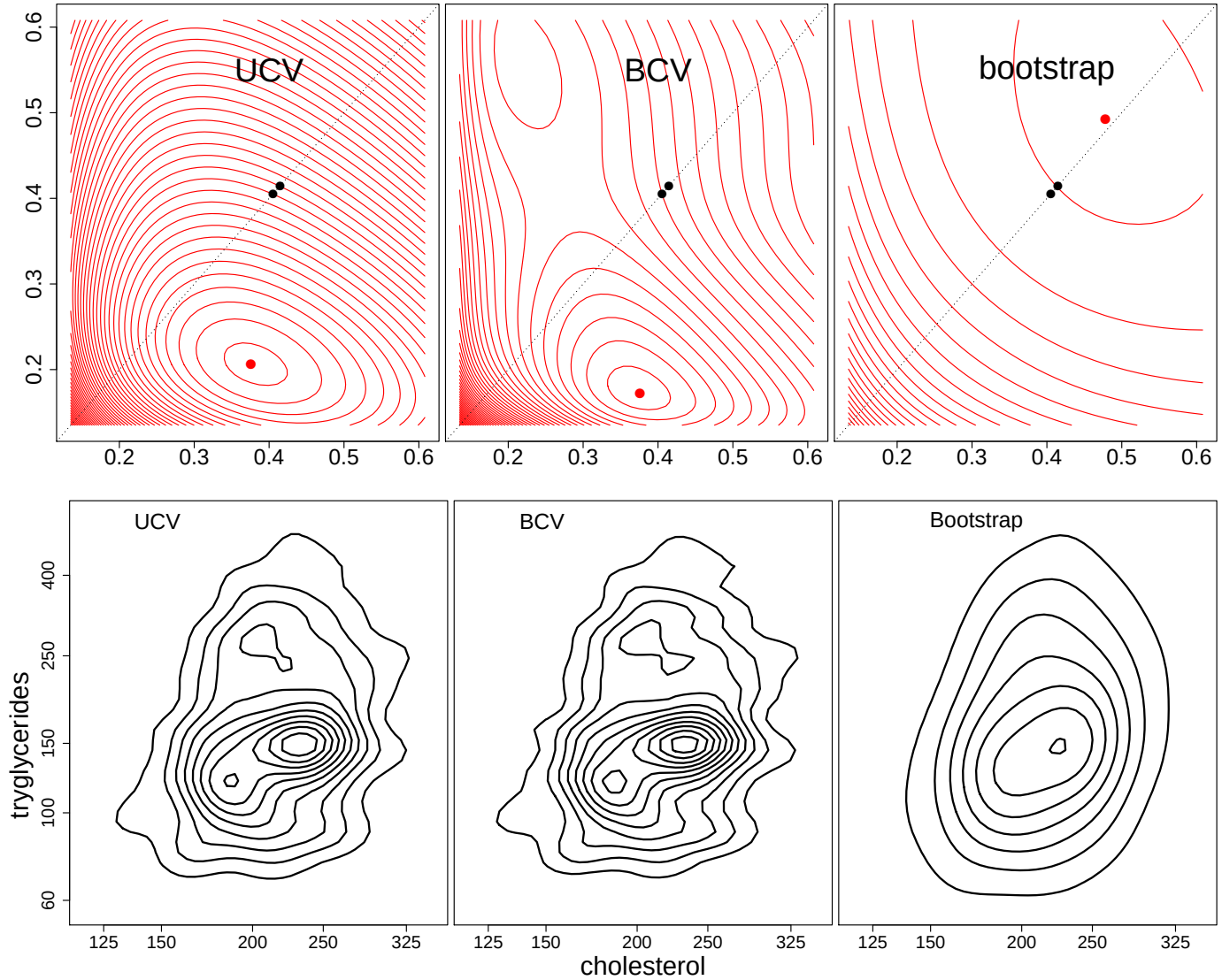

Figure: Same criterion for the standardized log lipid dataset ($n = 320$), together with the corresponding kernel estimates.

Convergence Rates (Relative)

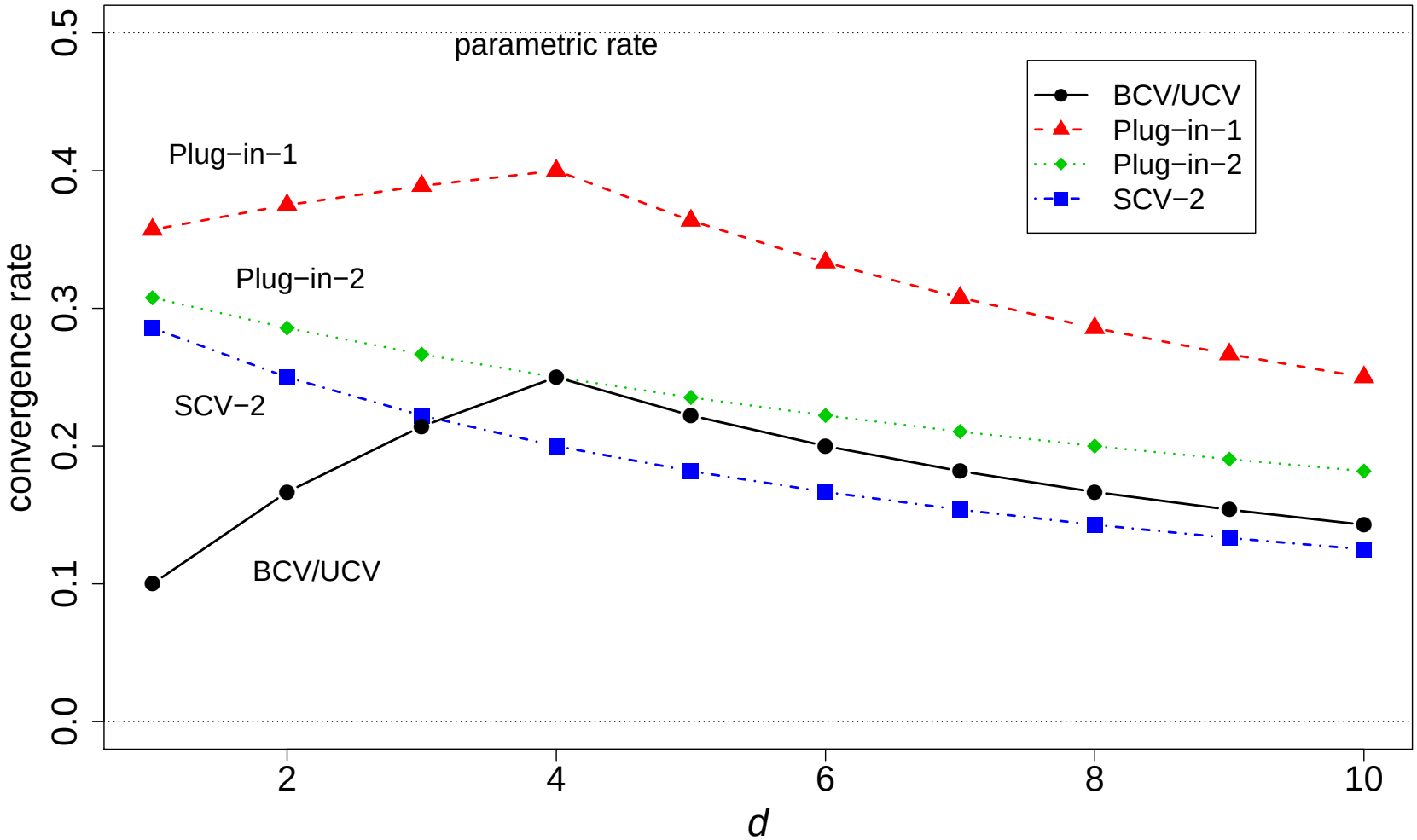

Figure: Magnitude of convergence rate exponents of several cross-validation algorithms. The best rate of $O(n^{-1/2})$ for parametric models would appear as $1/2$ on this graph.

Adaptive Kernel Smoothing

- There are two different (and intuitive) ways of defining an adaptive kernel estimator compared to the fixed bandwidth

$$\hat{f}(\mathbf{x}) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{\mathbf{x} - \mathbf{x}_i}{h}\right) = \frac{1}{n} \sum_{i=1}^{n} K_h(\mathbf{x} - \mathbf{x}_i)$$

- (1) A **different (but fixed)** bandwidth at each $\mathbf{x}$:

$$\boxed{\hat{f}_1(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} K_{h_{\mathbf{x}}}(\mathbf{x} - \mathbf{x}_i) \quad \text{where} \quad h_{\mathbf{x}} \equiv h(\mathbf{x}, \mathbf{x}, f)}$$

- (2) A **different bandwidth** at each data point $\mathbf{x}_i$:

$$\boxed{\hat{f}_2(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} K_{h_i}(\mathbf{x} - \mathbf{x}_i) \quad \text{where} \quad h_i \equiv h(\mathbf{x}_i, \mathbf{x}_i, f)}$$

Sain and Scott Example of $\hat{f}_2(\mathbf{x})$ (lipid data)

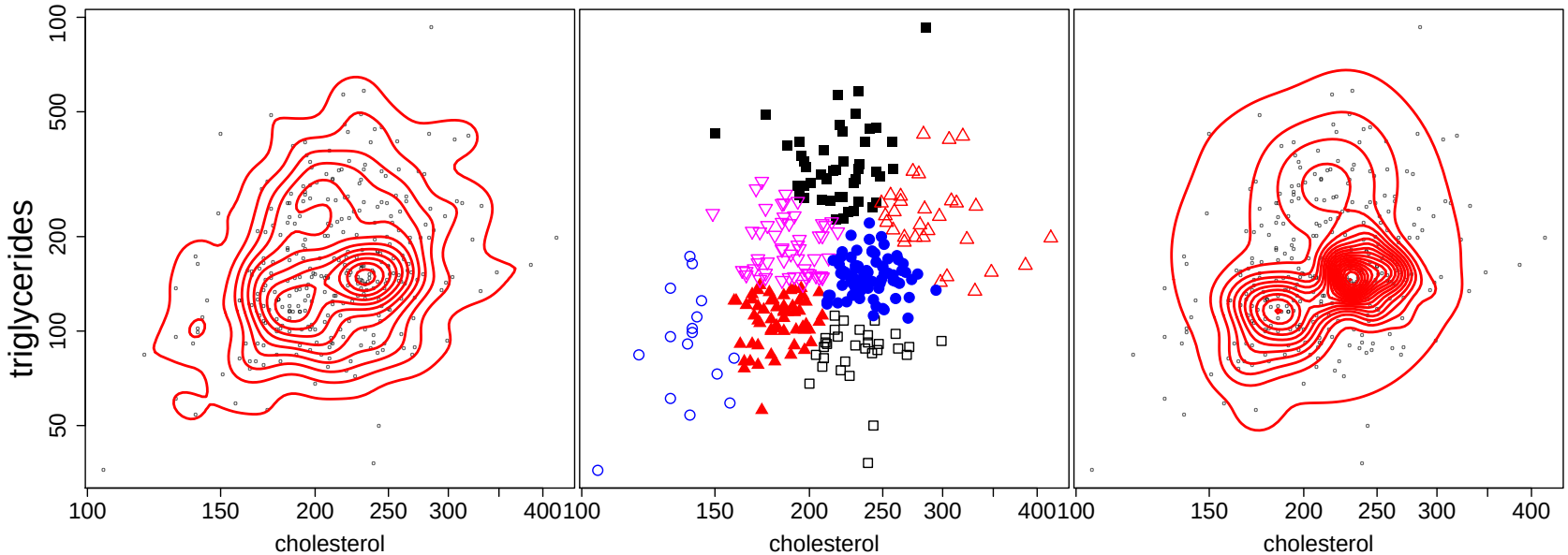

Figure: (Left) Twelve contours of the UCV-calibrated ($\hat{h} = 0.276$) bivariate Gaussian fixed-kernel estimate of the standardized log cholesterol and triglyceride data. (Middle) Seven clusters from k-means. (Right) The adaptive kernel estimator. The 7 bandwidths range from 0.174 to 2.36. The mode is 54% greater than in the left frame. The 19 contour levels are the same as in the left frame plus 7 more at higher levels.

Wrapping Up

- Kernel estimates are very general and very effective in dimensions 1–5.
- The ASH is a useful way to actually compute because of the pre-binning
- There are some FFT tricks, but these require lots of padding with zero's and full estimation (whereas, the ASH can estimates slices without going to the full dimension)
- In the interest of time, we skipped over interesting topics
  - oversmoothing
  - zero-bias estimation
  - nearest-neighbor density estimation (another adaptive procedure)