STAT 5630, Fall 2019

Kernel Methods

Xiwei Tang, Ph.D. <xt4yj@virginia.edu>

University of Virginia
November 12, 2019

Outline

- Kernel Methods
- Density Estimation

k-Nearest Neighbor Smoother

- k-Nearest Neighbor averaging

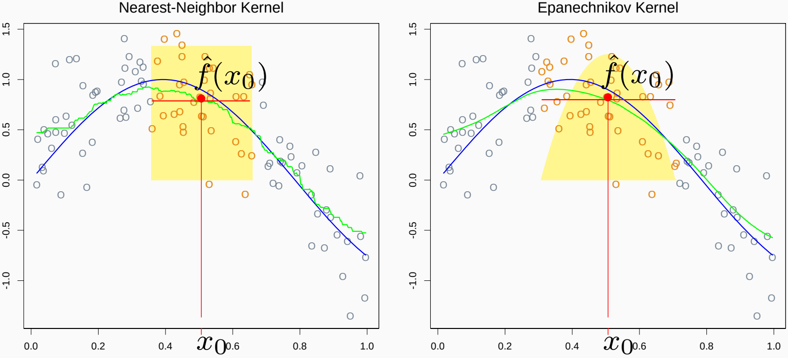
$$\widehat{f}(x) = \sum_{i=1}^{n} w(x, x_i) y_i$$

where

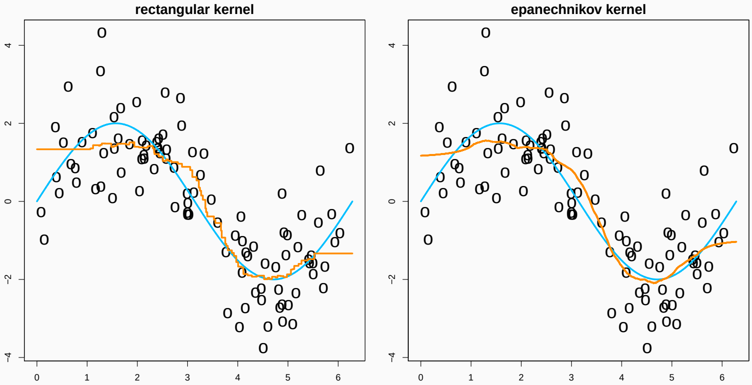
$$w(x, x_i) = \begin{cases} \frac{1}{k} & \text{if } x_i \in N_k(x) \\ 0 & \text{o.w.} \end{cases}$$

- The weights $w(x, x_i)$ drop off abruptly to zero outside the neighborhood of x.
- Rather than giving all the points in $N_k(x)$ equal weights, we can assign weights decaying smoothly according to the distance to x.

Epanechnikov vs. Rectangular Kernels



Epanechnikov vs. Rectangular Kernels



Kernel Smoother (Univariate)

- We can still use the local averaging idea: Fit a simple model locally at each point x using only those observations close to it.
- Localization via the weighting function $K(x, x_i)$, the weight of x_i is based on its distance from x
- For any point $x \in \mathcal{X}$,

$$\widehat{f}(x) = \frac{\sum_i K_\lambda(x, x_i) y_i}{\sum_i K_\lambda(x, x_i)}$$

where

$$K_\lambda(x, x_i) = K(|x - x_i|/\lambda)/\lambda$$

Kernel Smoother

- K is a kernel function (we have seen this in SVM)
- The kernel is indexed by λ, the "bandwidth", which defined the width of the neighborhood
- The estimator is called Nadaraya-Watson kernel estimator
- Requires little or no training time; all the work gets done at evaluation time (same as kNN).

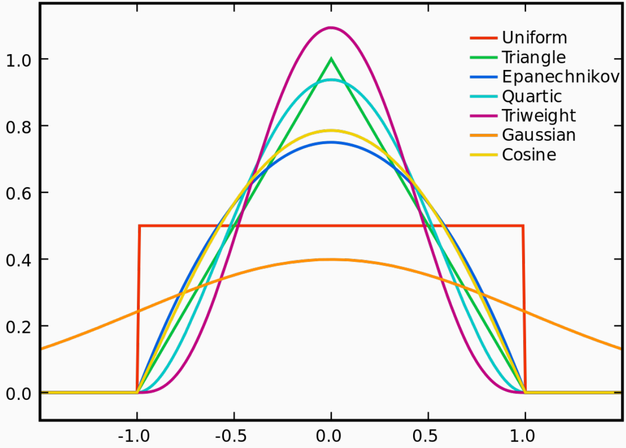
Popular Kernels

- $\int K(u)du = 1$; K is symmetric around 0; $\int u^2 K(u)du \leq \infty$
- Symmetric Beta family kernel

$$K(u,d) = \frac{(1-u^2)^d}{2^{2d+1}B(d+1,d+1)\mathbf{1}\{|u|<1\}}$$

  — Uniform kernel $d = 0$
  — Epanechnikov kernel $d = 1$
  — Bi/Tri weight $d = 2, 3$
- Tri-cube kernel: $K(u) = (1-u^3)^3\mathbf{1}\{|u|<1\}$
- Gaussian kernel: $K(u) = \phi(u) = 1/\sqrt{2\pi}\exp(-u^2/2)$

Kernels



Uniform
Triangle
Epanechnikov
Quartic
Triweight
Gaussian
Cosine
1.0
0.8
0.6
0.4
0.2
0.0
-1.0
-0.5
0.0
0.5
1.0

Choice of λ

- The bandwidth λ controls how “local” the estimator is

$$K_\lambda(u) = K(u/\lambda)/\lambda$$

- In many kernels (except Gaussian), only points within $[x-\lambda, x+\lambda]$ receive positive weights
- Small λ: rougher estimate, bias $\downarrow$, variance $\uparrow$
- Large λ: smoother estimate, bias $\uparrow$, variance $\downarrow$
- Besides choosing λ using CV, many works in the literature discuss the choice of λ theoretically, e.g., Fan and Gijbels (1992, 1995).

Local Modeling

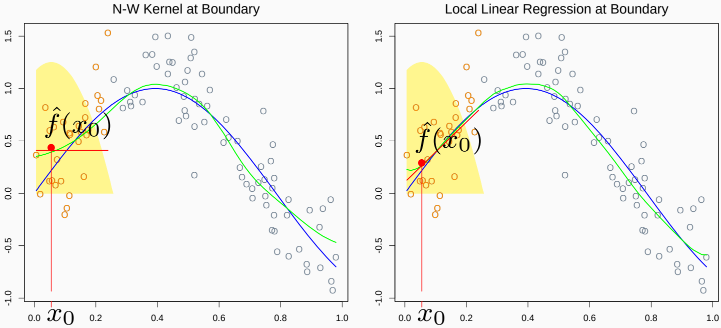
- Drawbacks: The kernel averaging formulation can be badly biased on the boundaries of the domain due to the asymmetry of the kernel in that region (we already seen this)

- Locally weighted linear regression can make a first order correction (straight lines vs. constants)
- Minimizing the objective function

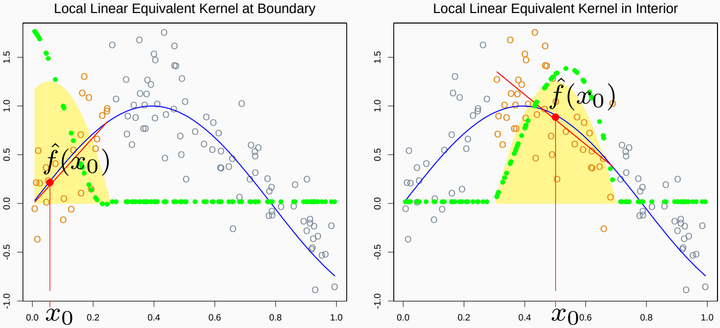
$$\underset{\beta_0(x),\beta_1(x)}{\text{minimize}} \sum_{i=1}^{n} K_\lambda(x, x_i)\left[y_i - \beta_0(x) - \beta_1(x)x_i\right]^2$$

- The estimation is extremely simple
- The solution $\widehat{f}(x_0) = \widehat{\beta}_0(x_0) + \widehat{\beta}_1(x_0)x_0$ is evaluated only at x_0
- Correct the boundary bias of the kernel estimator

Kernel Boundary Bias



Kernel Boundary Bias



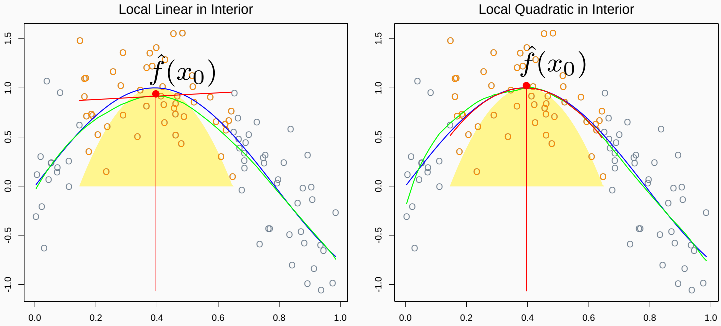
Local Polynomial Regression

- Locally weighted d polynomial regression
- Minimizing the objective function

$$\underset{\beta_0(x),\beta_r(x)}{\text{minimize}} \sum_{i=1}^{n} K_\lambda(x, x_i)\Big[y_i - \beta_0(x) - \sum_{r=1}^{d} \beta_j(x) x_i^r\Big]^2$$

- Still a weighed linear regression problem at each x
- $\widehat{f}(x_0) = \widehat{\beta}_0(x_0) + \sum_{r=1}^{d} \widehat{\beta}_j(x_0) x_0^r$
- Correct the boundary bias of the kernel estimator
- Reduce bias in regions of curvature, however, at a price of higher variance

Kernel Boundary Bias



R implementation

- R function loess provides fitting of the local polynomial regressions
- The most important parameter span = α controls the degree of smoothing: only αn number of closest points are used based on the distance $|x - x_i|$, forming the neighborhood "$N(x)$"
- A weighted least-square linear regression is fit within the neighborhood
- The weights uses tri-cube kernel: $w_{x,i} = (1 - u^3)^3$ with

$$u_i = \frac{|x_i - x|}{\max_{N(x)} |x_j - x|}$$

- degree specifies the degree of the polynomial
- Other implementations such as locfit and locpoly (use Gaussian kernel)

Kernel Density Estimation

Kernel Density Estimation

- Another area where we often use the kernel methods is for estimating the density
- Given some observations from an unknown distribution, we want to estimate the pdf of that distribution (unsupervised)

$$X_1, \ldots, X_n \quad \overset{\text{i.i.d}}{\sim} \quad f(\cdot)$$

- Some density estimation methods
  - Histograms
  - Assume a family of distributions and estimate parameters
  - Kernel density estimator

Histogram Estimator of Density Functions

- If $X \sim f(\cdot)$ the following are some facts:
  - $f(u) > 0$ and $\int f(u)du = 1$
  - $\mathsf{P}(x - \lambda/2 \leq X \leq x + \lambda/2) = \int_{x-\lambda/2}^{x+\lambda/2} f(u)du$
  - $f(x) = \lim_{\lambda \to 0} \frac{1}{\lambda}\mathsf{P}(x - \lambda/2 \leq X \leq x + \lambda/2)$
- A natural estimator is

$$\widehat{f}(x) = \frac{1}{hn}\#\{x : x \in [x - h/2, x + h/2]\}$$

- However, this estimation is bumpy and non-smooth

Kernel Density Estimation

- Parzen estimate

$$\widehat{f}(x) = \frac{1}{n} \sum_{i=1}^{n} K_\lambda(x, x_i)$$

- Usually $K_\lambda(x, x_i) = K(\frac{|x-x_i|}{\lambda})/\lambda$, and $K(u) \geq 0$.
- $K(u) = K(-u)$, and $\int K(u)du = 1$
- Popular choice: Gaussian kernel $K_\lambda(x - x_i) = \phi(|x - x_i|/\lambda)/\lambda$

$$\widehat{f}(x) = \frac{1}{n} \sum_{i=1}^{n} \phi_\lambda(|x - x_i|)$$

where ϕ_λ is the Gaussian density with mean zero and standard deviation λ

Popular Kernels

- Performance of a kernel is measured by MISE (mean integrated squared error) or AMISE (asymptotic MISE).
- Epanechnikov kernel minimizes AMISE and is therefore optimal.
- Kernel efficiency is measured in comparison to Epanechnikov kernel:
  - Biweight 0.994; Triangular 0.986; Normal 0.951; Uniform 0.930
- However, choosing kernel is not as important as choosing the bandwidth!
- The rule of thumb (Silverman 1986) for the bandwidth λ in univariate case is

$$\widehat{\lambda} = 1.06\widehat{\sigma}n^{-1/5}$$

R implementations

- hist makes histograms
- density for kernel density estimator
- bw.nrd and a set of related functions for bandwidth selection
- Library locfit: function locfit can perform both local polynomial regressions and density estimation