Week 7

Non-parametric methods

Emre Ugur, BM 33

emre.ugur@boun.edu.tr

http://www.cmpe.boun.edu.tr/~emre/courses/cmpe462

cmpe462@listeci.cmpe.boun.edu.tr

Acknowledgements

- Clustering:
  - adapted from Alexander Ihler's Machine Learning course material
- Nonparametric methods:
  - Textbook
  - http://www3.stat.sinica.edu.tw/stat2005w/download/NP-111805.pdf

Parametric vs nonparametric

- Parametric: Bernoulli, multinomial, normal, MLE,etc. Multivariate methods: parameter estimation, classification, regression .
- Semiparametric methods: mixture densities, clustering
- Non-parametric methods
  - Density estimation:
    - Histogram
    - Kernel estimator
    - K-nearest neighbour estimator
  - Classification and regression

Parametric vs nonparametric

- Parametric: data are drawn from a probability distribution of specific form up to unknown parameters.
- Semiparametric: in between, contains parametric and nonparametric components.
- Nonparametric: data are drawn from a certain unspecified probability distribution

Basic philosophy of nonparametric estimation

- The world is smooth and functions are changing slowly.
- Similar instances mean similar things.
- Unlike parametric methods, there is no single global model; local models are estimated as they are needed, affected only by closeby training data.
- Learn to know “similar patterns” from training set, and “interpolate” from them to find the right output (in prediction).
- Need a distance measure for similarity and interpolation.

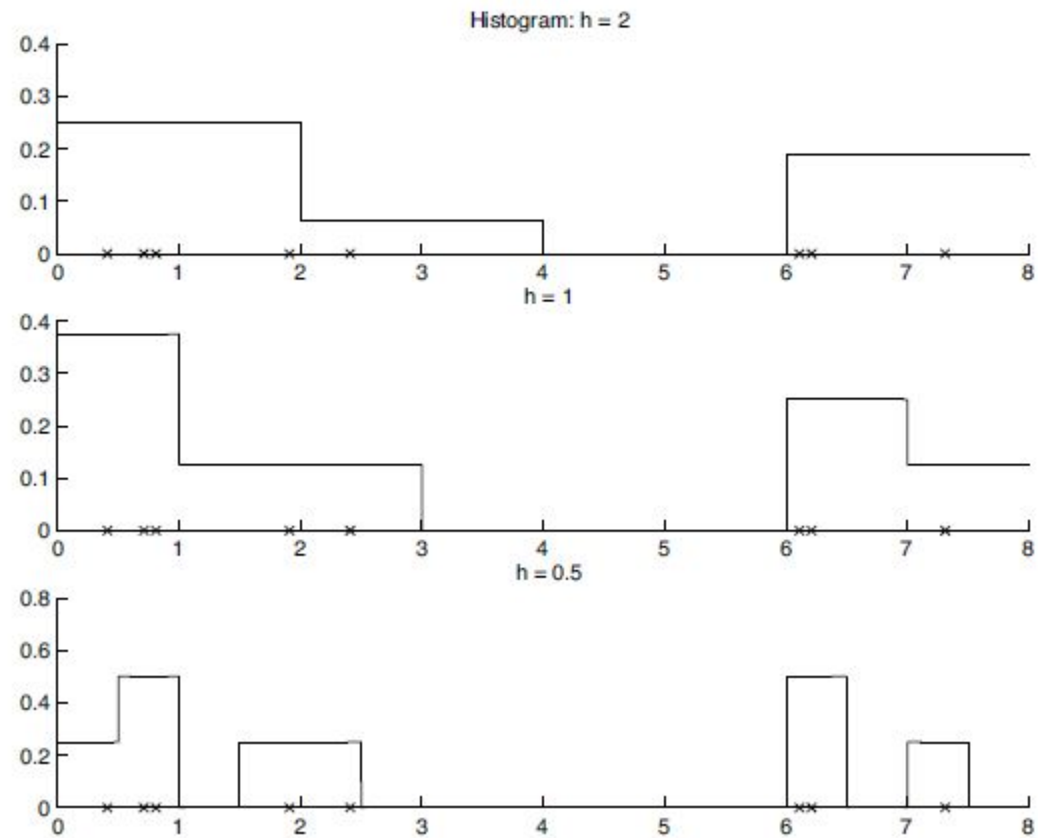
Heavier computation cost

- In machine learning literature, nonparametric methods are also call instance-based or memory-based learning algorithms.
- Store the training instances in a lookup table and interpolate from these for prediction
- Lazy learning algorithm, as opposed to the eager parametric methods, which have simple model and a small number of parameters

Density estimation – Histogram

- The input space is divided into equal-sized intervals named **bins**. Given an origin x_0 and a bin width h, density estimate:

$$\hat{p}(x) = \frac{\#\{x^t \text{ in the same bin as } x\}}{Nh}$$



Density estimation – Histogram

- Naive estimator frees us from setting an origin

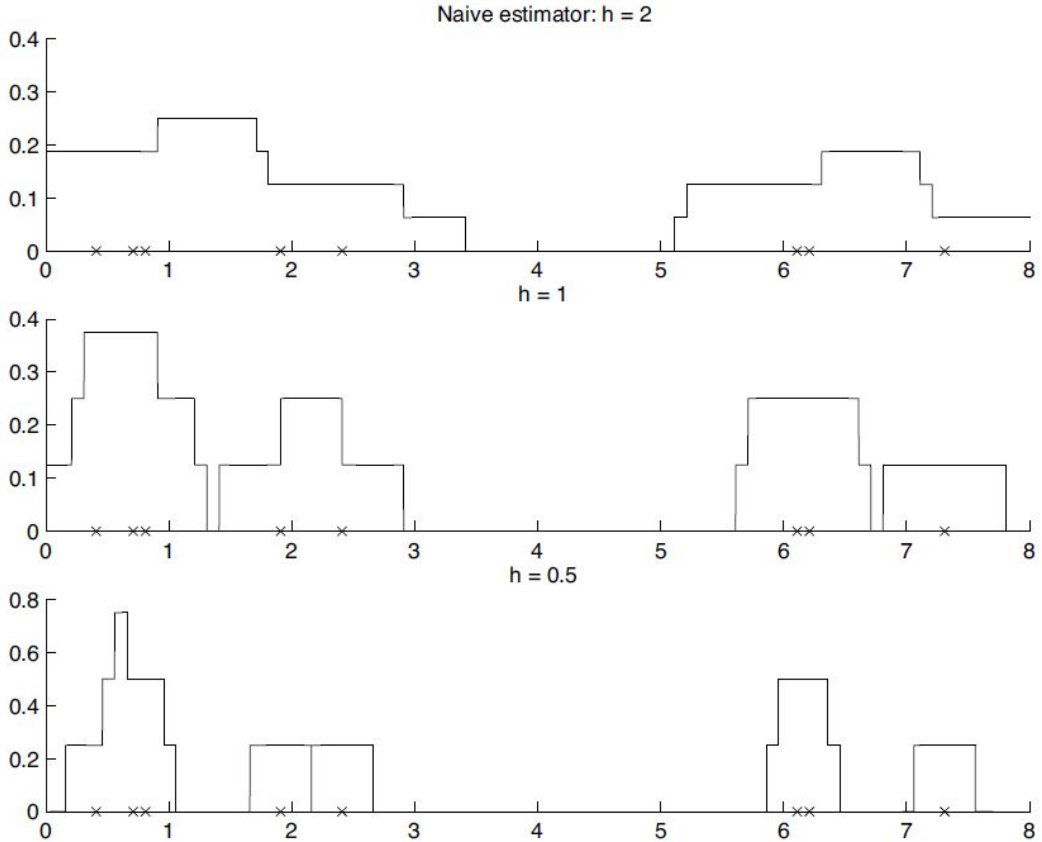
$$\hat{p}(x) = \frac{\#\{x - h/2 < x^t \le x + h/2\}}{Nh}$$

- As if each x^t has an influence of width h.

$$\hat{p}(x) = \frac{1}{Nh} \sum_{t=1}^{N} w\left(\frac{x - x^t}{h}\right)$$

with the *weight function* defined as

$$w(u) = \begin{cases} 1 & \text{if } |u| < 1/2 \\ 0 & \text{otherwise} \end{cases}$$



Density estimation – Kernel estimator

- To get a smooth estimate, we use a smooth weight function, called a kernel function kernel function.
- The most popular is the Gaussian kernel.

$$K(u) = \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{u^2}{2}\right]$$

- The *kernel estimator* (*Parzen windows*):

$$\hat{p}(x) = \frac{1}{Nh} \sum_{t=1}^{N} K\left(\frac{x - x^t}{h}\right)$$

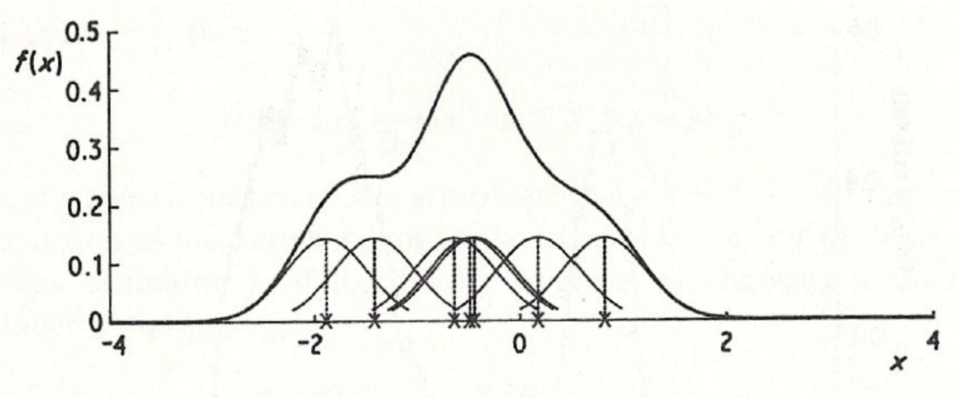


Fig. 2.4 *Kernel estimate showing individual kernels. Window width 0.4.*

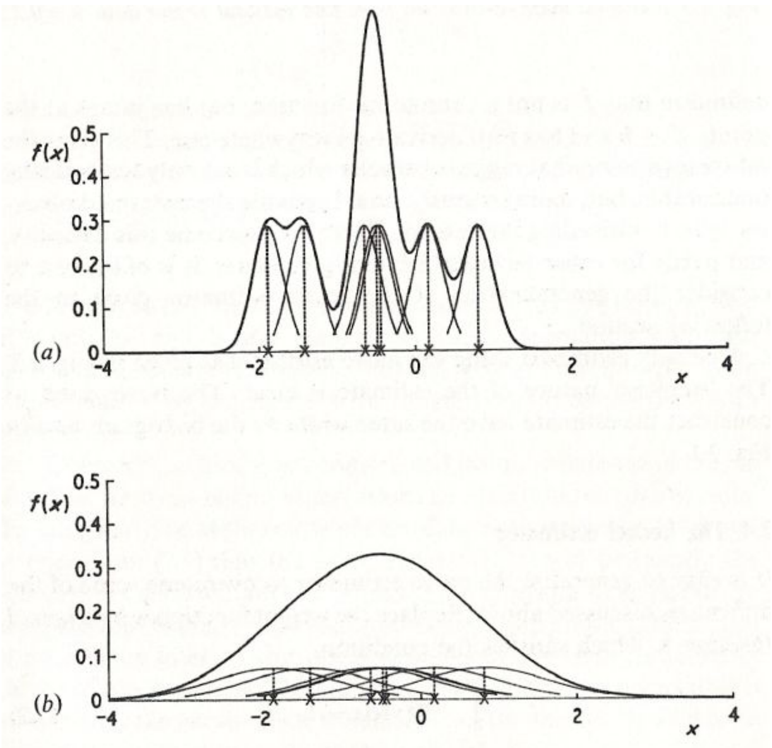
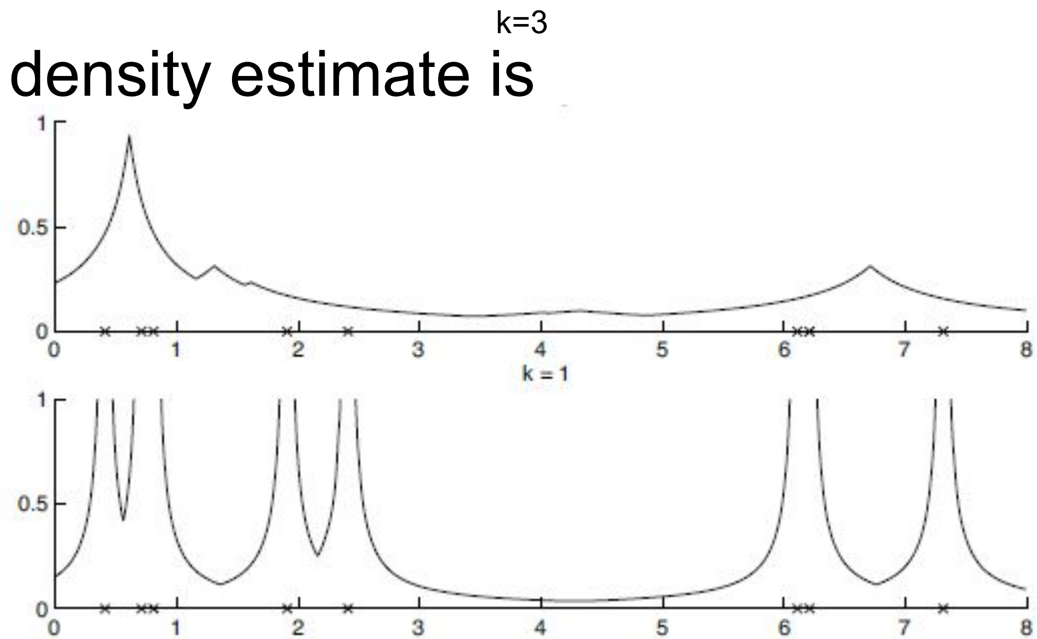


Fig. 2.5 *Kernel estimates showing individual kernels. Window widths: (a) 0.2; (b) 0.8.*

K-Nearest Neigbour Estimator

- It adapts the amount of smoothing to the local density of data.
- The degree of smoothing is controlled by k, the number of neighbours taken into account
- The k-nearest neighbour (k-nn) density estimate is



$$\hat{p}(x) = \frac{k}{2N d_k(x)}$$

Distance of the k[th] closest instance

- Equivalent to $h=2d_k(x)$
  - Histogram: Fixing h, compute how many instances fall in the bin
  - k-nn: Fix k, number of closest instances fall in the bin, compute the bin size.
    - Density high: bins are small.

Multivariate data

- Given sample: $\mathcal{X} = \{\mathbf{x}^t\}_{t=1}^N,$
- Kernel density estimator: $\hat{p}(\mathbf{x}) = \frac{1}{Nh^d} \sum_{t=1}^{N} K\left(\frac{\mathbf{x} - \mathbf{x}^t}{h}\right)$
- Gaussian kernel: $K(\mathbf{u}) = \left(\frac{1}{\sqrt{2\pi}}\right)^d \exp\left[-\frac{\|\mathbf{u}\|^2}{2}\right]$

- Care with high-dimensional data and histograms!
- Inputs are discrete, might use *Hamming distance:*

$$HD(\mathbf{x}, \mathbf{x}^t) = \sum_{j=1}^{d} 1(x_j \neq x_j^t)$$

where

$$1(x_j \neq x_j^t) = \begin{cases} 1 & \text{if } x_j \neq x_j^t \\ 0 & \text{otherwise} \end{cases}$$

Nonparametric classification

- The kernel estimator of the class-conditional density

$$\hat{p}(\mathbf{x}|C_i) = \frac{1}{N_i h^d} \sum_{t=1}^{N} K\left(\frac{\mathbf{x} - \mathbf{x}^t}{h}\right) r_i^t$$

- Discriminant function

$$\begin{aligned} g_i(\mathbf{x}) &= \hat{p}(\mathbf{x}|C_i)\hat{P}(C_i) \\ &= \frac{1}{N h^d} \sum_{t=1}^{N} K\left(\frac{\mathbf{x} - \mathbf{x}^t}{h}\right) r_i^t \end{aligned}$$

- k-nn classifier (exercise)

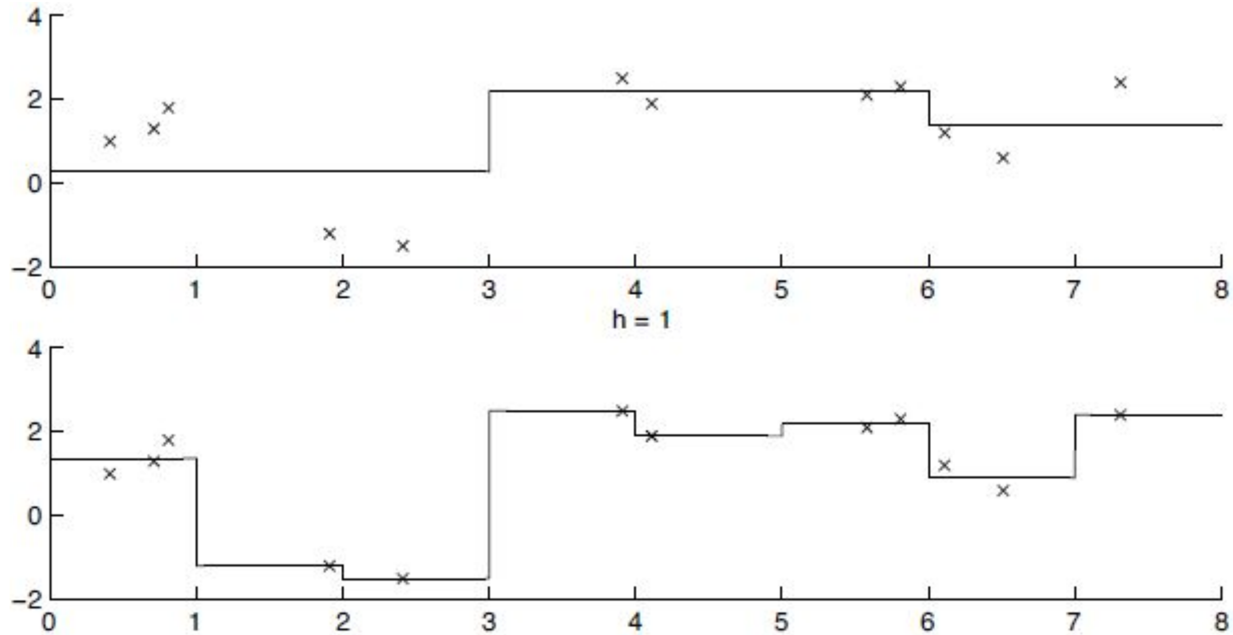
$$\hat{P}(C_i|\mathbf{x}) = \frac{\hat{p}(\mathbf{x}|C_i)\hat{P}(C_i)}{\hat{p}(\mathbf{x})} = \frac{k_i}{k}$$

Nonparametric regression: Smoother

- Assume that close x have close g(x) values.
- Find the neighborhood of x and average the r values in the neighborhood to calculate ˆg(x).
- Regressogram

$$b(x, x^t) = \begin{cases} 1 & \text{if } x^t \text{ is the same bin with } x \\ 0 & \text{otherwise} \end{cases}$$

$$\hat{g}(x) = \frac{\sum_{t=1}^{N} b(x, x^t) r^t}{\sum_{t=1}^{N} b(x, x^t)}$$

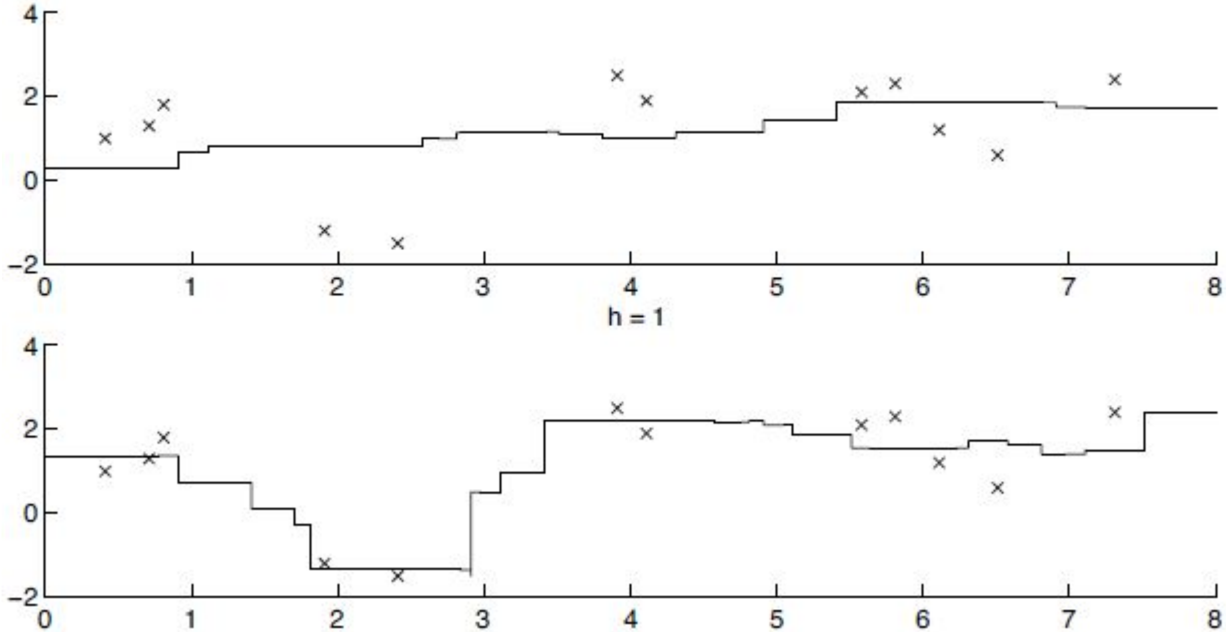


Nonparametric regression: Smoother

- Assume that close x have close g(x) values.
- Find the neighborhood of x and average the r values in the neighborhood to calculate ˆg(x).
- Mean smoother

$$\hat{g}(x) = \frac{\sum_{t=1}^{N} w\left(\frac{x-x^t}{h}\right) r^t}{\sum_{t=1}^{N} w\left(\frac{x-x^t}{h}\right)}$$

$$w(u) = \begin{cases} 1 & \text{if } |u| < 1 \\ 0 & \text{otherwise} \end{cases}$$

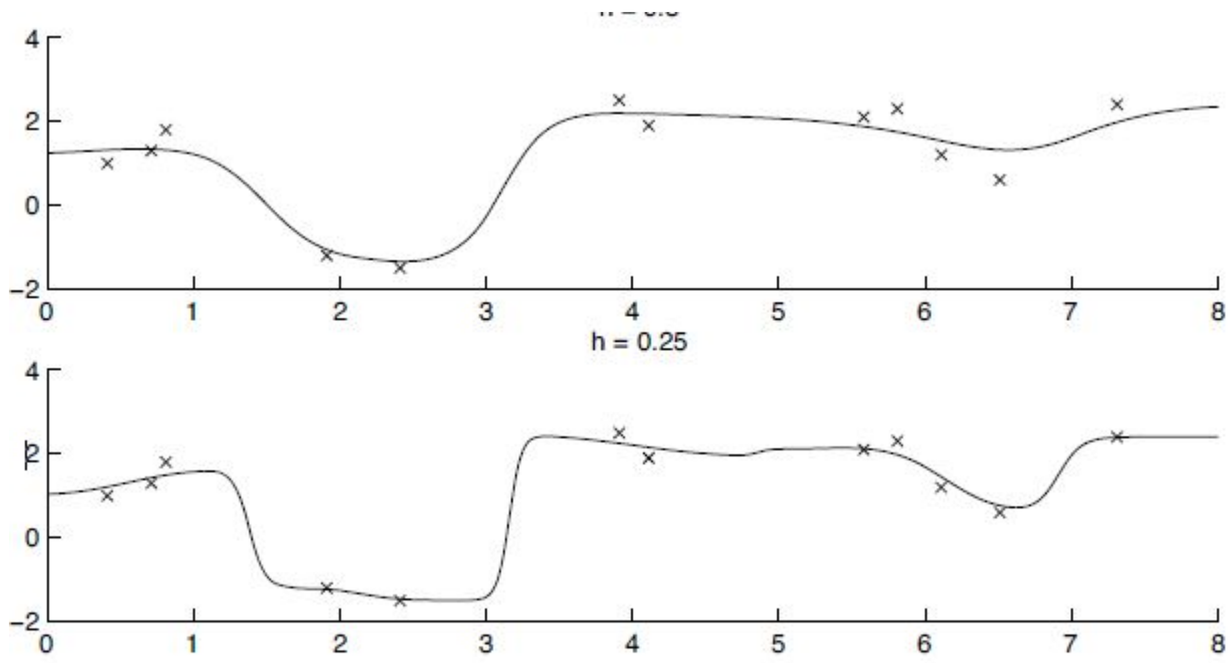


Nonparametric regression: Smoother

- Assume that close x have close g(x) values.
- Find the neighborhood of x and average the r values in the neighborhood to calculate ˆg(x).
- Kernel smoother

$$\hat{g}(x) = \frac{\sum_t K\left(\frac{x-x^t}{h}\right) r^t}{\sum_t K\left(\frac{x-x^t}{h}\right)}$$

$$K(u) = \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{u^2}{2}\right]$$



How to choose h or k?

- When k or h is small:
    - Single instances matter; bias is small, variance is large
    - Undersmoothing: High complexity
- As k or h increases,
    - we average over more instances and variance decreases but bias increases
    - Oversmoothing: Low complexity
- Use cross-validation