# Nonparametric Regression

• Section 5.6 gives a rank-based procedure for estimating a regression function when the function is <u>unknown</u> and <u>nonlinear</u> BUT known to be <u>monotonic</u>.

• Here we will examine a distribution-free method of estimating a very general type of regression function.

• In nonparametric regression, we assume very little about the functional form of the regression function.

• We assume the model:

$$E(Y \mid X = x) = f(x)$$

where $f(\cdot)$ is unknown but is typically assumed to be a smooth and continuous function.

• We also assume independence for the residuals

$$Y_i - f(X_i), \quad i = 1, \ldots, n$$

<u>Goal</u>: Estimate the mean response function $f(\cdot)$.

## Advantages of Nonparametric Regression

• Useful when we cannot know the relationship between $Y$ and $X$

• More flexible type of regression model

• Can account for unusual behavior in the data

• Less likely to have bias resulting from wrong model being chosen

## Disadvantages of Nonparametric Regression

- **Not as easy to interpret**
- **No easy way to describe relationship between *Y* and *X* with a formula (must be done with a graph)**
- **Inference is not as straightforward**

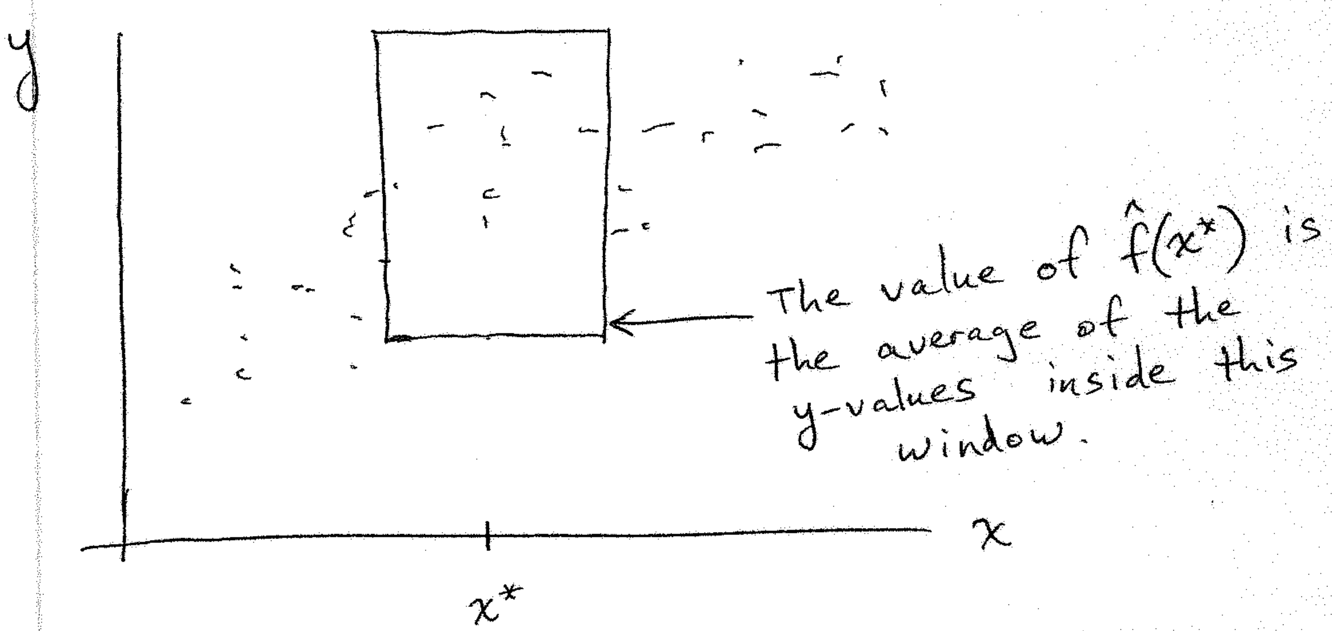
**<u>Note</u>: Nonparametric regression is sometimes called** scatterplot smoothing.

## Kernel Regression

- **The idea behind kernel regression is to estimate $f(x)$ <u>at</u> <u>each</u> value $x^*$ along the horizontal axis.**

- **At each value $x^*$, the estimate** $\hat{f}(x^*)$ **is simply an** average of the Y-values of the observations <u>near</u> $x^*$.

- **Consider a "window' of points centered at $x^*$:**



- **The width of this window is called the** <u>bandwidth</u>**.**

- **At each different $x^*$, the window of points** <u>moves</u> **to the left or right** (moving average)

- **Better idea: Use** weighted average of Y-values, with more weight on points <u>near</u> $x^*$.

- **This can be done using a** <u>weighting</u> **function known as a <u>kernel</u>.**

- **Then, for any $x^*$,**

$$\hat{f}_\lambda(x^*) = \frac{1}{n}\sum_{i=1}^{n} w_i Y_i$$

**where the weights** $w_i = \frac{1}{\lambda} K\left(\frac{x^* - x_i}{\lambda}\right)$

***K*(·) is a kernel function, which typically is a <u>density</u> function symmetric about 0.**

**$\lambda$ = bandwidth, which controls the <u>smoothness</u> of the estimate of $f(x)$.**

**Possible choices of kernel:**

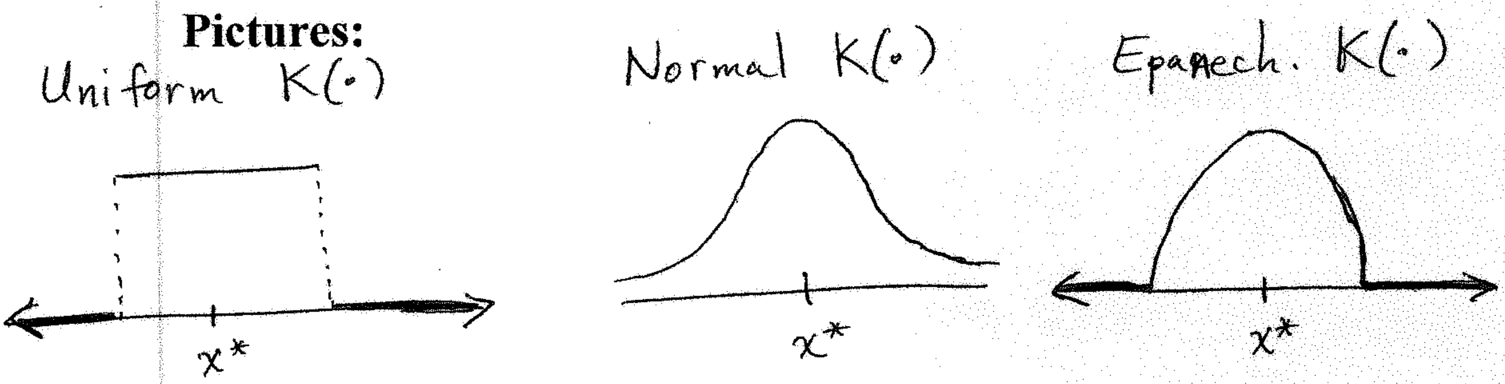
Uniform (box) kernel: Gives all points in window <u>equal</u> weight; Gives all points outside window <u>no</u> weight.

Normal kernel: Gives points <u>near</u> $x^*$ more weight; Gives points far from $x^*$ <u>less</u> weight.

Epanechnikov kernel: $K(x) = \begin{cases} 0.75(1-x^2) & \text{for } |x| < 1 \\ 0 & \text{otherwise} \end{cases}$

<u>Compromise</u>: Gives points closer to $x^*$ <u>more</u> weight, but gives points <u>very far</u> from $x^*$ <u>no</u> weight.

**Pictures:**




<u>**Note**</u>**: The Nadaraya-Watson estimator**

$$\hat{f}_\lambda(x) = \sum_{i=1}^{n} \frac{w_i}{\sum_{j=1}^{n} w_j} Y_i$$

**is a modification that assures that the weights for the $Y_i$'s will sum to one.**

- **The choice of <u>bandwidth</u> $\lambda$ is of more practical importance than the choice of kernel.**

- **The bandwidth controls how many data values are used to compute $f(x^*)$ at each $x^*$.**

**Large $\lambda$ →** many data values used at each estimation

⇒ low variability of estimate, smoother-looking curve

**Small $\lambda$ →** fewer data values used at each estimation

⇒ high variability of estimate, wiggly-looking curve

• **Choosing λ too large results in an estimate that** oversmooths **the true nature of the relationship between *Y* and *X*.**

• **Choosing λ too small results in an estimate that follows the "noise" in the data too closely.**

• **Often the best choice of λ is made through visual inspection (pick the roughest estimate that does not fluctuate implausibly?).**

• **Automatic bandwidth selection methods such as <u>cross-validation</u> are also available – this chooses the λ that minimizes a mean squared prediction error.**

**Example: We have data on the horsepower (*X*) and gas mileage (*Y*, in miles per gallon) of 82 cars, from Heavenrich et al. (1991).**

• **On computer: The R function `ksmooth` performs kernel regression (see web page for examples with various kernel functions and bandwidths).**

- Best choice appears to be the normal kernel with bw = 60.

- Other smoothing functions may produce better results.