Local Polynomial Regression

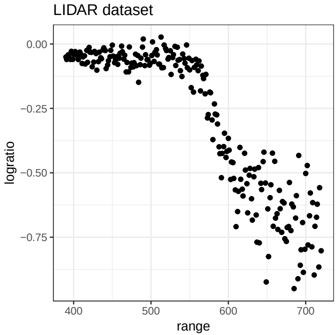
Statistical Machine Learning - individual project

Leonardo Stincone

Università degli Studi di Trieste

18th July 2019

Problem statement: LIDAR dataset



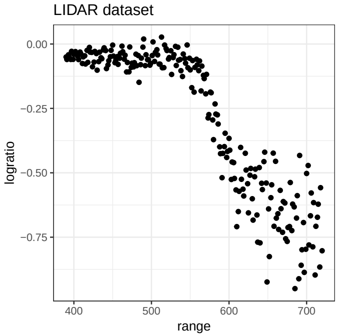
LIDAR = LIght Detection And Ranging

- it is a surveying method that measures distance to a target by illuminating the target with laser light and measuring the reflected light with a sensor
- x: distance travelled before the light is reflected back to its source
- y: logarithm of the ratio of received light from two laser sources

The objective is to estimate:

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

Problem statement: LIDAR dataset



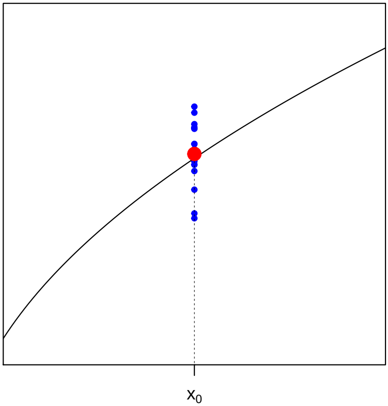
LIDAR = LIght Detection And Ranging

- it is a surveying method that measures distance to a target by illuminating the target with laser light and measuring the reflected light with a sensor
- x: distance travelled before the light is reflected back to its source
- y: logarithm of the ratio of received light from two laser sources

The objective is to estimate:

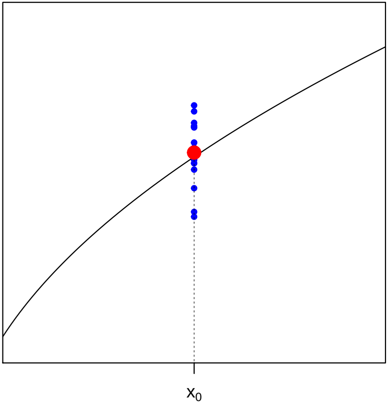
$$f(x) = E\left[Y \mid X = x\right]$$

If we had enough points with $x = x_0$

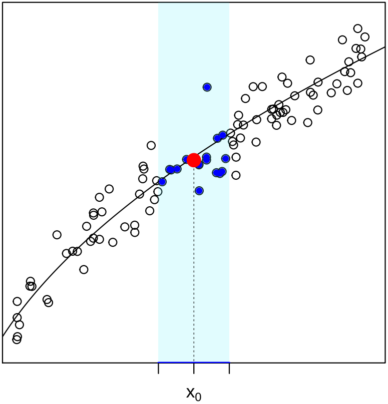


What does local mean?

If we had enough points with $x = x_0$

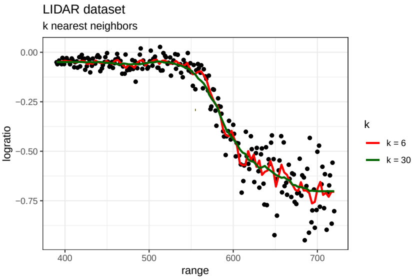


We can consider points "close" to x_0



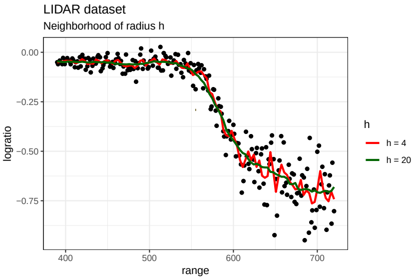
k nearest neighbors

$$\hat{f}(x) = \frac{1}{k}\sum_{i=1}^{n} y_i I_{N_k(x)}(x_i)$$



Neighborhood of radius h

$$\hat{f}(x) = \frac{\sum_{i=1}^{n} y_i I_{[0,h]}(|x - x_i|)}{\sum_{i=1}^{n} I_{[0,h]}(|x - x_i|)}$$



$$\hat{f}(x) = \sum_{i=1}^{n} \ell_i(x) y_i$$

with:

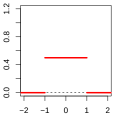
$$\ell_i(x) = \frac{K\left(\frac{x - x_i}{h}\right)}{\sum_{j=1}^{n} K\left(\frac{x - x_j}{h}\right)}$$

where $K(\cdot)$ is a kernel function that satisfies:

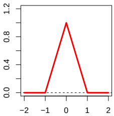
- $K(x) \geq 0$
- $\int K(x)dx = 1$
- $\int xK(x)dx = 0$
- $\int x^2 K(x)dx > 0$

Some proposed kernels

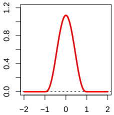
Uniform
$\frac{1}{2}I_{[-1,1]}(u)$



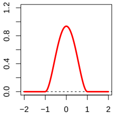
Triangle
$(1-|u|)I_{[-1,1]}(u)$



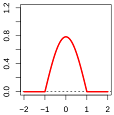
Triweight
$\frac{35}{32}(1-u^2)^3 I_{[-1,1]}(u)$



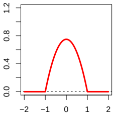
Quartic
$\frac{15}{16}(1-u^2)^2 I_{[-1,1]}(u)$



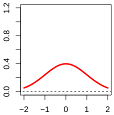
Cosine
$\frac{\pi}{4}\cos\left(\frac{\pi}{2}u\right) I_{[-1,1]}(u)$



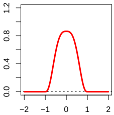
Epanechnikov
$\frac{3}{4}(1-u^2) I_{[-1,1]}(u)$



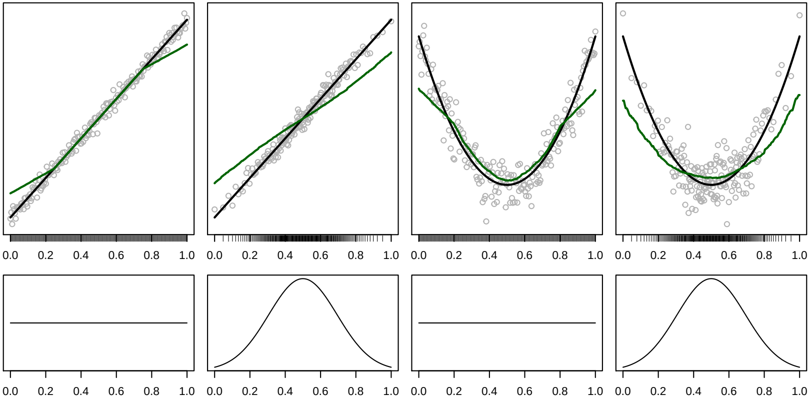
Gaussian
$\frac{1}{\sqrt{2\pi}}e^{-u^2/2}$



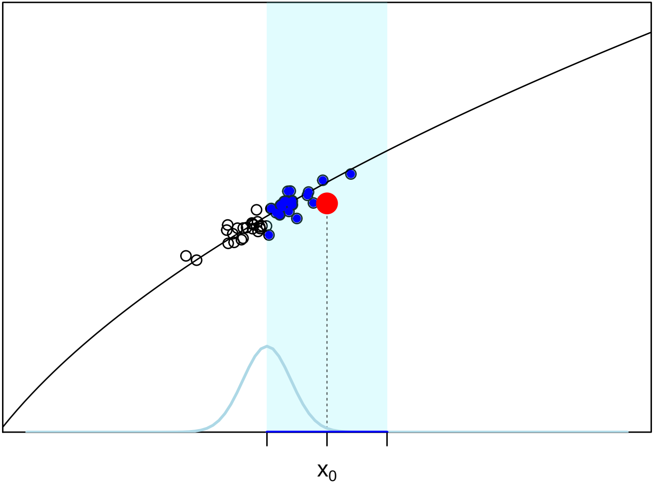
Tricube
$\frac{70}{81}(1-|u|^3)^3 I_{[-1,1]}(u)$



Design bias, boundary bias and concavity bias



Design bias: what's happening?



Local polynomial regression: idea

Locally the Nadaraya-Watson estimator is a Weighted Least Square Estimator:

$$\hat{f}_{NW}(x_0) = \underset{a}{\text{argmin}} \sum_{i=1}^{n} K\left(\frac{x_i - x_0}{h}\right)(y_i - a)^2$$

Idea: instead of approximating $f(x_0)$ with a constant value a, we could approximate it with a polynomial $p_{x_0}(u, \boldsymbol{a})$.

Taylor polynomial approximation:

$$p_{x_0}(u; \boldsymbol{a}) = a_0 + a_1(u - x_0) + \frac{a_2}{2!}(u - x_0)^2 + \ldots + \frac{a_d}{d!}(u - x_0)^d$$

Local polynomial regression: idea

Locally the Nadaraya-Watson estimator is a Weighted Least Square Estimator:

$$\hat{f}_{NW}(x_0) = \underset{a}{\text{argmin}} \sum_{i=1}^{n} K\left(\frac{x_i - x_0}{h}\right)(y_i - a)^2$$

Idea: instead of approximating $f(x_0)$ with a constant value a, we could approximate it with a polynomial $p_{x_0}(u, \boldsymbol{a})$.

Taylor polynomial approximation:

$$p_{x_0}(u; \boldsymbol{a}) = a_0 + a_1(u - x_0) + \frac{a_2}{2!}(u - x_0)^2 + \ldots + \frac{a_d}{d!}(u - x_0)^d$$

We can estimate the coefficients of $p_{x_0}(u; \boldsymbol{a})$ as:

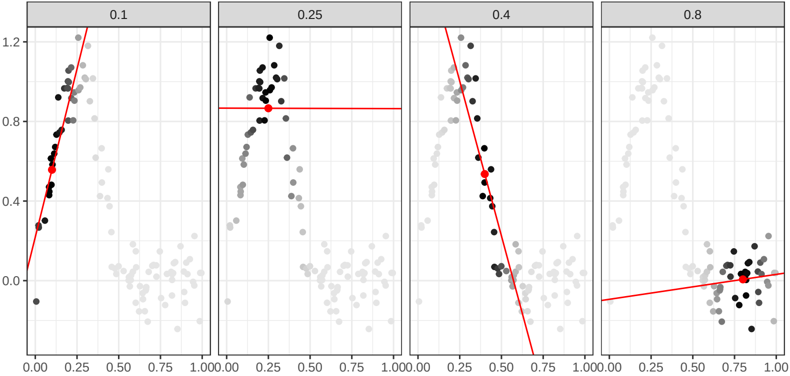
$$\hat{\boldsymbol{a}}(x_0) = \underset{\boldsymbol{a}}{\text{argmin}} \sum_{i=1}^{n} K\left(\frac{x_i - x_0}{h}\right)(y_i - p_{x_0}(x_i; \boldsymbol{a}))^2$$

Thus, we can define the estimator for $f(x)$ in x_0 just computing $p_{x_0}(u; \hat{\boldsymbol{a}})$ in x_0:

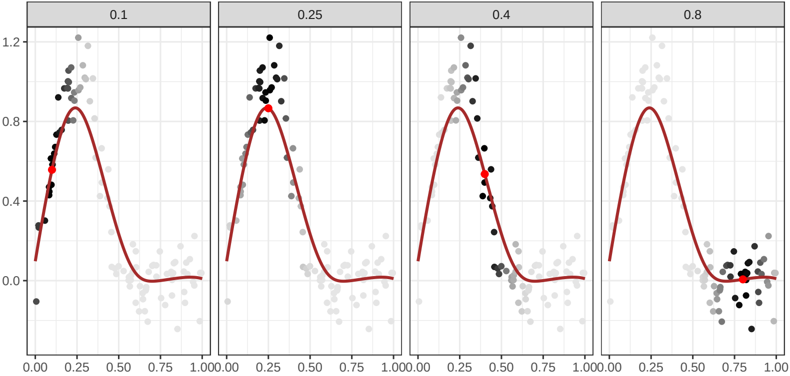
$$\hat{f}(x_0) = p_{x_0}(x_0; \hat{\boldsymbol{a}})$$

Then we can repeat the process for each value of x in a grid and obtain $\hat{f}(x)$.

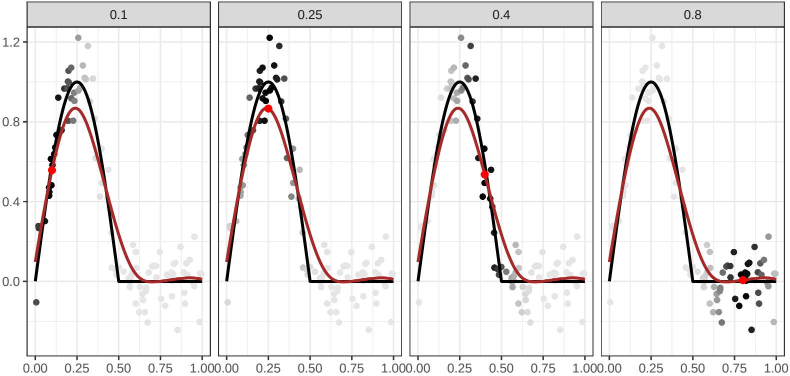
Local polynomial regression: example



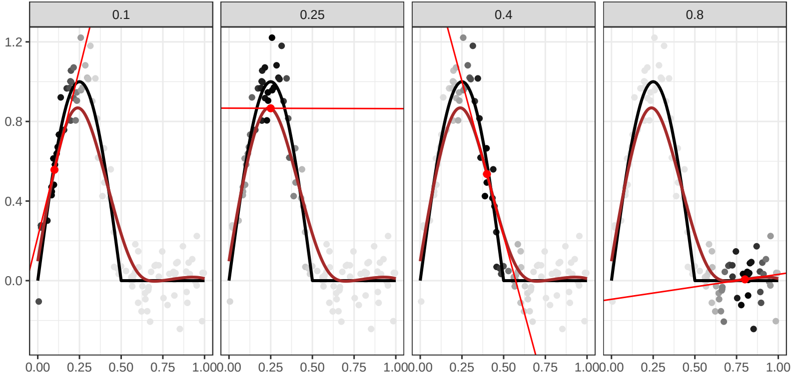
Local polynomial regression: example



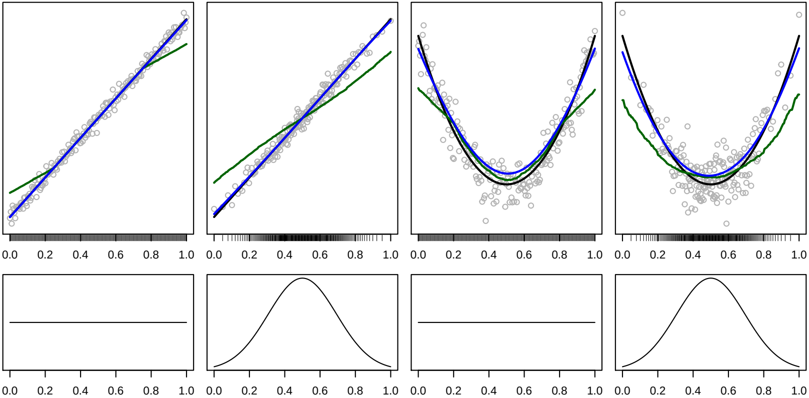
Local polynomial regression: example



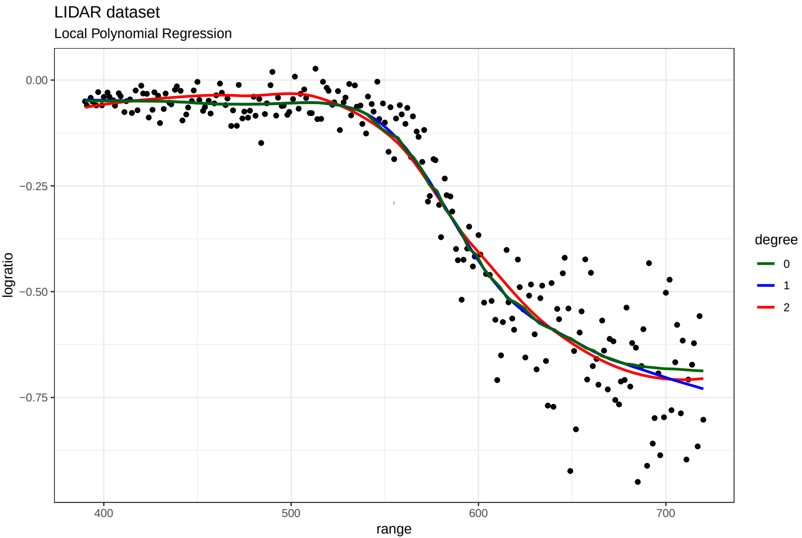
Local polynomial regression: example



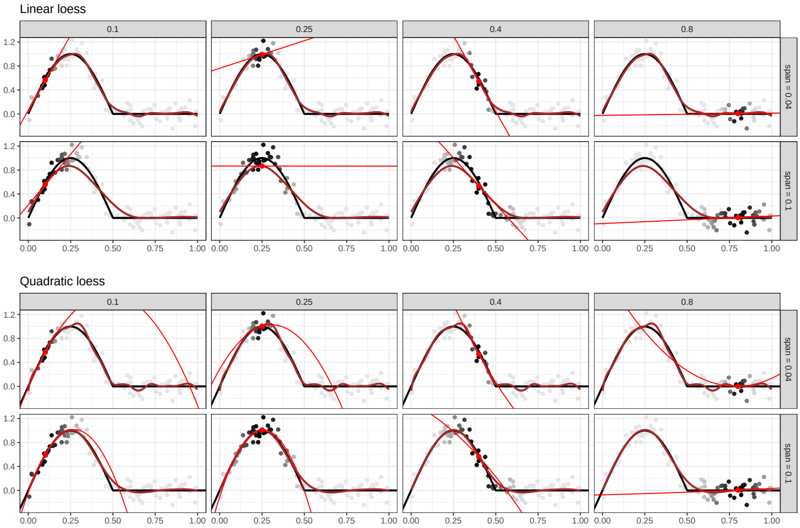
Design bias, boundary bias and concavity bias



Local Polynomial Regression on LIDAR dataset



Local polynomial regression: linear vs quadratic



Bibliography

Fan, Gijbels
Local Polynomial Modelling and its Applications
Springer (1996)

Ruppert, Wand, Carroll
Semiparametric Regression
Cambridge University Press (2003)