Extension to several dimension (of the non-parametric techniques seen on Monday)

$x \in \mathbb{R} \longrightarrow x \in \mathbb{R}^p$

$$f(x) = \underbrace{f(x_0)}_{\beta_0} + \underbrace{f'(x_0)}_{\beta_1}(x - x_0) + \dots$$

E.g., when $p=2$

$$y = f(x_1, x_2) + \varepsilon$$

where $f: \mathbb{R}^2 \to \mathbb{R}$

Let $y_i \in \mathbb{R}$, $x_i = (x_{i1}, x_{i2}) \in \mathbb{R}^2$, $i = 1, \dots, n$

The problem is to find the parameters β which minimize

$$\min_{\beta_0, \beta_1} \sum_{i=1}^{n} \left[y_i - \beta_0 - \beta_1 (x_{i1} - x_{01}) - \beta_2 (x_{i2} - x_{02}) \right]^2 w_i$$

where w_i has form

$$w_i = \frac{1}{h_1 h_2} K\left(\frac{x_{i1} - x_{01}}{h_1}\right) K\left(\frac{x_{i2} - x_{02}}{h_2}\right)$$

Note that there are two smoothing tuning parameters, h_1, h_2, to take into account the different variability in the two dimensions

Again, we obtain β through the weighted least squares

$$\hat{\beta} = (X^T W X)^{-1} X^T W y$$

where $y = (y_1 \dots y_n)^T$

$W = \operatorname{diag}(w_1, \dots, w_n)$

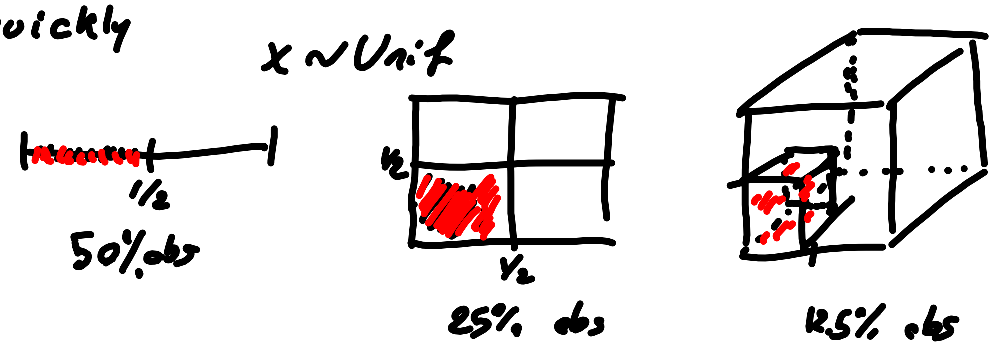
$X = (1, x_{i1} - x_{01}, x_{i2} - x_{02})$

While there is no conceptual difference from $p=2$ to a general $p=p$, in practice $p>2$ is almost never used

- difficulties in plotting/visualizing mentally
- hard to interpret the results
- <u>curse of dimensionality</u>

  when the number of dimensions increase, the number of observations close to the point of interest is decreases very quickly

E.g. $x \sim Unif$



In order to compensate for the increase dimensionality, we need a number of observation that increases of order n^p

(e.g., if we want to use 100 observations in 1 dimension, we need $100^5 =$ 10 000 000 000 (ten billions) observations with 5 variables, with 10 variables, we would need 100^{10})

The problem holds for all non-parametric techniques

- number of observations
- computational cost

Alternative: use principal components
(information concentrated on a few dimensions)

Splines : piecewise polynomial functions

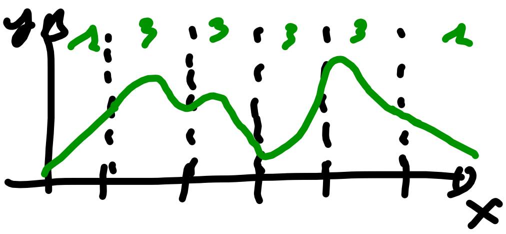
- split the support of x into several regions
  ($\text{fix } K \text{ points}, \ \xi_1 < \xi_2 < \dots < \xi_K$) ← ξ_j are called knots

- fit a polynomial inside each interval
  ↳ of the preferred degree d (it is almost always $d=3$)
  ↳ cubic splines
  - we force the polynomials to have the same value at the knots
    $$f(\xi_i^-) = f(\xi_i^+)$$
  - same with the first derivative
    $$f'(\xi_i^-) = f(\xi_i^+)$$
  - and with the second derivative
    $$f''(\xi_i^-) = f''(\xi_i^+)$$

Due to issues related to the variability, in the first and in the last regions ($x < \xi_1$ and $x > \xi_K$, respectively) we force

$$f''(x) = 0$$ (straight line)

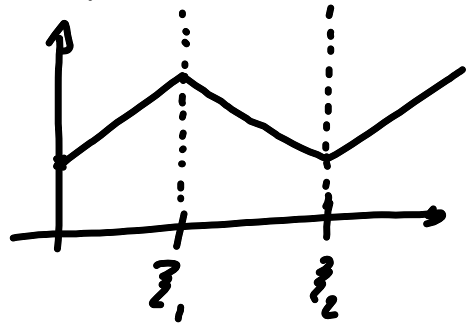
natural cubic splines



Use of splines to evaluate the relationship between x and y

$$y = f(x;\beta) + \varepsilon$$

Simplest case: $K=2$, $d=1$

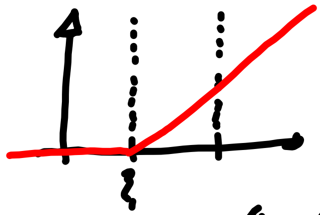


$$f(x;\beta) = \beta_0 + \beta_1 x + \beta_2 (x-\xi_1)_+ + \beta_3 (x-\xi_2)_+$$

$h_1(x) = 1$ $\qquad$ $h_3(x) = (x-\xi_1)_+$

$h_2(x) = x$ $\qquad$ $h_4(x) = (x-\xi_2)_+$

where $(x-\xi_i)_+ = \max(0, x-\xi_i)$



$$f(x;\beta) = \beta_0 h_1(x) + \beta_1 h_2(x) + \beta_2 h_3(x) + \beta_3 h_4(x) = \sum_{j=1}^{4} \hat{\beta}_{j-1} \underbrace{h_j(x)}_{\text{basis}}$$

In the case of cubic splines, with a general number of knots K,

$$f(x) = \sum_{j=1}^{K+4} \hat{\beta}_j h_j(x)$$

where $h_j(x) = x^{j-1}$ for $j = 1, \dots, 4$

$h_1(x) = 1$
$h_2(x) = x$
$h_3(x) = x^2$
$h_4(x) = x^3$

$h_{j+4}(x) = (x-\xi_j)_+^3$ for $j = 1, \dots, K$

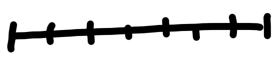
We need to decide K, the number of knots, and where to place them.

K is our complexity parameter: higher values, more complex function

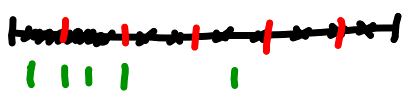
↳ find by cross-validation

Once we selected K, we can position the knots

- uniformly along the x: range



- using the quantiles of the empirical distribution of x



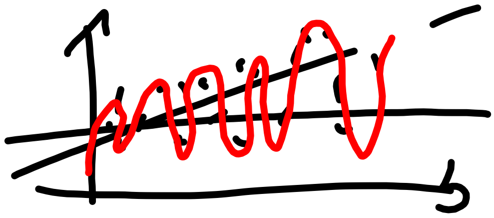
Smoothing splines

Consider the penalized least squares criterion

$$D(f,\lambda) = \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \int_{-\infty}^{+\infty} (f''(t))^2 dt \quad , \lambda > 0$$

λ is the smoothing parameter

The penalty penalizes the "bumpyness" of the curve



$\lambda \to 0$, more and more curvature is allowed, until for $\lambda = 0 \to$ interpolation

$\lambda \to \infty$, curvature is penalized more and more, so for a sufficiently large $\lambda \to$ straight line

The important result (Green & Silverman, 1994) is that the minimizer of $D(f,\lambda)$ is a natural cubic spline, which can be rewritten as

$$\hat{f}(x) = \sum_{j=1}^{n_0} \hat{\theta}_j N_j(x)$$

n_0 = # of unique points x_i

$N_j(x)$ are the basis functions of a natural cubic splines,

$$N_1(x) = 1, \quad N_2(x) = x, \quad N_{k+2}(x) = d_k(x) - d_{K-1}(x)$$

with

$$d_k(x) = \frac{(x-\xi_k)^3_+ - (x-\xi_K)^3_+}{\xi_K - \xi_k}$$

we will derive them in STK-IN4300

The nice part is that we can rewrite $D(f,\lambda)$ as

$$D(f,\lambda) = (y - N\theta)^T (y - N\theta) + \lambda \theta^T \Omega \theta$$

where $\{N\}_{ij} = N_j(x_i)$ and $\{\Omega\}_{jk} = \int N_j''(t) N_k''(t) dt$

This formula remind us that of the ridge regression, so

$$\hat{\theta} = (N^T N + \lambda \Omega)^{-1} N y$$

← generalized ridge estimator

$I \to \Omega$

the solution depends on λ ← find by cross-validation