Advanced Statistical Methods.
P4-E1-K3.
Pointwise derivative estimation

Evgeny Lagutin

Skoltech
Data Science

March 20, 2022

github.com/sverdoot/pointwise-derivative-estimation

Data generation

We observe sample $S_n = \{(X_i, Y_i) : 1 \leq i \leq n\}$ where $X_i = \frac{i}{n},\ i \in \{1, \ldots, n\}$

$$Y_i = f^*(X_i) + \varepsilon_i,\ \varepsilon_i \sim \mathcal{N}(0,1), \quad 1 \leq i \leq n,$$

$$\psi_j(x) = \begin{cases} 1, & \text{if } j = 0 \\ \sin(\pi(j+1)x), & \text{if } j \text{ is odd} \\ \cos(\pi j x), & \text{if } j \text{ is even.} \end{cases}$$

The true function f^* is then equal to

$$f(x) = c_1\psi_1(x) + \cdots + c_n\psi_n(x)$$

where the coefficients $c_1, \ldots, c_n$ are chosen randomly: with γ_j i.i.d. standard normal,

$$c_j = \begin{cases} \gamma_j, & 1 \leq j \leq 10 \\ \frac{\gamma_j}{(j-10)^2}, & 11 \leq j \leq n \end{cases}$$

Data generation

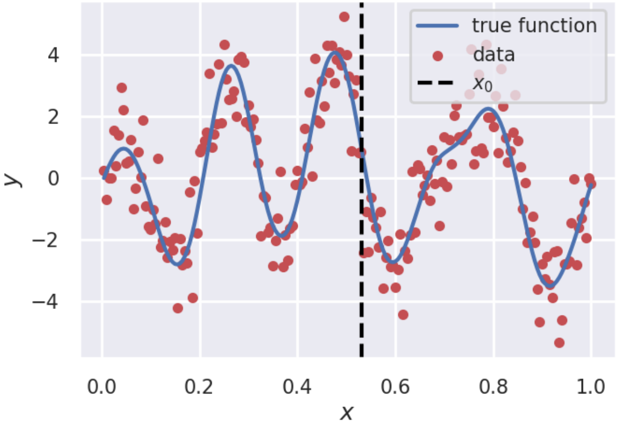


Figure 1: Function and sampled data

$$x_0 = 0.532$$

Local polynomial smoothing

Gaussian kernel: $K(x) = \exp(-x^2/2)$
Localizing weights: $w_i(x) = K\left(\frac{1}{h}(X_i - x)\right)$

$$\tilde{\theta}(x) = \mathcal{S}(x)\mathbf{Y}$$
$$S(x) = \{\Psi(x)W(x)\Psi(x)^\top\}^{-1}\Psi(x)W(x)$$
$$W(x) = \mathrm{diag}(\{w_i(x)\}_{i=1}^n)$$
$$\Psi(x) = \begin{pmatrix} 1 & \cdots & 1 \\ (X_1 - x) & \cdots & (X_n - x) \end{pmatrix}$$

Let $(\mathcal{S}(x))_j$ be a $\mathbb{R}^n$ vector constructed of j's row of $\mathcal{S}(x)$, then

$$\hat{f}'(x_0) = \tilde{\theta}_1(x_0) = (\mathcal{S}(x_0))_1^\top \mathbf{Y}$$

Unbiased risk estimation

$$\hat{\mathcal{R}}_m = \|\mathcal{K}_m \mathbf{Y} - \mathbf{Y}\|^2 + 2\sigma^2 \operatorname{tr}(\mathcal{K}_m)$$

$$\mathcal{K}_m = \begin{pmatrix} (\mathcal{S}(X_1))_0 \\ \cdots \\ (\mathcal{S}(X_n))_0 \end{pmatrix}$$

Unbiased risk estimation

Bandwidth	Function estimate	True value	Derivative estimate	True value	Unbiased risk
0.00028	0.196	0.898	-101	-86	225

Table 1: Result for chosen bandwidth

True risk I

$$\mathcal{R}(\hat{f}) = \mathbb{E}((f^*)'(x_0) - \hat{f}'(x_0))^2 = \mathbb{E}((f^*)'(x_0) - S\left(\mathcal{S}(x_0)\right)_1^\top \mathbf{Y})^2 =$$
$$\mathbb{E}\left((f^*)'(x_0) - \left(\mathcal{S}(x_0)\right)_1^\top (\mathbf{f} + \boldsymbol{\varepsilon})\right)^2 =$$
$$\left((f^*)'(x_0) - \left(\mathcal{S}(x_0)\right)_1^\top \mathbf{f}\right)^2 + \left(\mathcal{S}(x_0)\right)_1^\top \mathbb{E}\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^\top \left(\mathcal{S}(x_0)\right)_1 -$$
$$2(f^*)'(x_0)\left(\mathcal{S}(x_0)\right)_1^\top (x_0)\mathbb{E}\boldsymbol{\varepsilon} =$$
$$\left((f^*)'(x_0) - \left(\mathcal{S}(x_0)\right)_1^\top \mathbf{f}\right)^2 + \sigma^2 \| \left(\mathcal{S}(x_0)\right)_1 \|^2$$

True risk II

$$
(\mathcal{S}(x_0))_1^\top = \frac{1}{D}\left[\sum_{i=1}^n w_i(x_0)(X_i - x_0)\sum_{j=1}^n Y_j w_j(x_0) + \right.
$$

$$
\left.\sum_{j=1}^n Y_j w_j(x_0)(X_j - x_0)\sum_{i=1}^n w_i(x_0)\right]
$$

$$
D = \left[\sum_{i=1}^n w_i(x_0)\right]\left[\sum_{i=1}^n w_i(x_0)(X_i - x_0)^2\right] - \left[\sum_{i=1}^n w_i(x_0)(X_i - x_0)\right]^2
$$

True risk

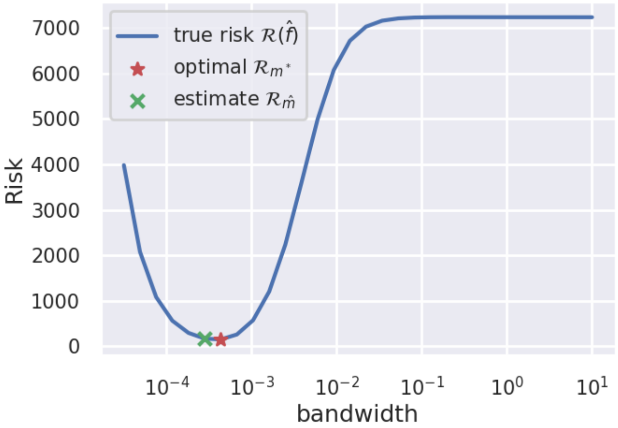


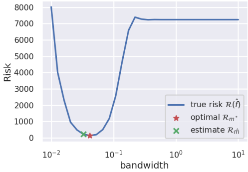
Figure 2: Dependence of true risk on bandwidth.

Comparison

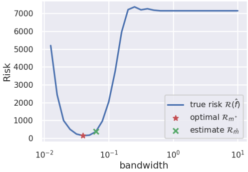
Oracle bandwidth	Oracle risk	Chosen bandwidth	Estimate risk
0.00043	144	0.00028	163

Table 2: Comparison with oracle

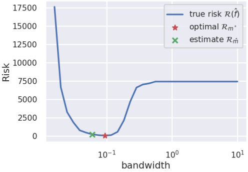
Additional results



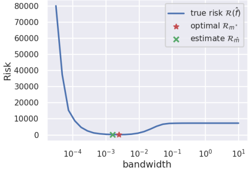
(a) Rectangular kernel, locally linear est.



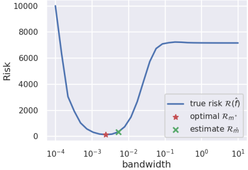
(b) Rectangular kernel, locally quadratic est.



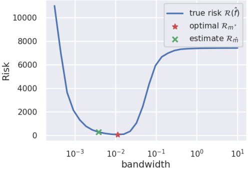
(c) Rectangular kernel, locally cubic est.



(d) Epanechnikov kernel, locally linear est.



(e) Epanechnikov kernel, locally quadratic est.



(f) Epanechnikov kernel, locally cubic est.