Kernel Methods for Regression

University of California, San Diego
Instructor: Armin Schwartzman

Regression model

- Consider a regression model with additive noise

$$y = f(\mathbf{x}) + \varepsilon \ ,$$

  where $\mathbb{E}(\varepsilon|\mathbf{x}) = 0$.

- We have independent observations $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$ from that model.

Local average

- Note that

$$f(\mathbf{x}) = \mathbb{E}(y|\mathbf{x})$$

- A local average (aka moving average) attempts to approximate this conditional expectation directly.

  It takes the form:

$$\widehat{f}(\mathbf{x}) = \mathrm{Ave}\big(y_i|\mathbf{x}_i \in N(\mathbf{x})\big)$$

  where $N(\mathbf{x})$ is a neighborhood of $\mathbf{x}$.

- Note that there are two approximations here:

  1. The expectation is approximated by an average.
  2. The conditioning on an exact value for $\mathbf{x}$ is approximated by conditioning on a region for $\mathbf{x}$.

Choice of neighborhood type

The two main choices are:

- h-ball neighborhood where

$$N(\mathbf{x}) = N_h(\mathbf{x}) = \{\mathbf{x}' : \|\mathbf{x}' - \mathbf{x}\| \leq h\}$$

  This choice implies a constant window width, and this keeps the bias stable. Indeed, the bias comes from averaging over $N(\mathbf{x})$, a region around $\mathbf{x}$ instead of averaging responses precisely at $\mathbf{x}$.

- k-nearest neighbors where

$$N(\mathbf{x}) = N_k(\mathbf{x}) = \{k \text{ closest points } \mathbf{x}_i\text{'s to } \mathbf{x}\}$$

  This choice implies a constant variance, assuming the errors have the same variance independent of the predictors ($\mathrm{Var}(\varepsilon|\mathbf{x}) = \sigma^2$).

Kernel regression (aka weighted local average)

- Choose a kernel function, often of the form

$$K_h(\mathbf{x}, \mathbf{x}_0) = D(\|\mathbf{x} - \mathbf{x}_0\|/h)$$

  where $D : \mathbb{R}_+ \to \mathbb{R}$ is non-increasing.

- The Nadaraya-Watson estimator based on that kernel is:

$$\widehat{f}(\mathbf{x}) = \frac{\sum_i K_h(\mathbf{x}, \mathbf{x}_i) y_i}{\sum_i K_h(\mathbf{x}, \mathbf{x}_i)}$$

  The nearest neighbor version of this kernel estimator would be of the form:

$$\widehat{f}(\mathbf{x}) = \frac{\sum_i \mathbb{I}\{\mathbf{x}_i \in N_k(\mathbf{x})\} K_h(\mathbf{x}, \mathbf{x}_i) y_i}{\sum_i \mathbb{I}\{\mathbf{x}_i \in N_k(\mathbf{x})\} K_h(\mathbf{x}, \mathbf{x}_i)}$$

Examples of Kernels

Our most basic requirement of D is that it be non-increasing on $\mathbb{R}_+$.

- ☐ Uniform: $D(t) = \mathbb{I}\{t < 1\}$ [this leads to the local average]
- ☐ Triangle: $D(t) = (1-t)_+$
- ☐ Epanechnikov: $D(t) = (1-t^2)_+$
- ☐ Quartic: $D(t) = (1-t^2)_+^2$
- ☐ TriCube: $D(t) = (1-t^3)_+^3$ [used by the R function `loess`]
- ☐ Gaussian: $D(t) = e^{-t^2/2}$ [also called heat kernel]
- ☐ Cosine: $D(t) = \cos(\frac{\pi}{2}t)\mathbb{I}\{t < 1\}$

They are all supported on $[0, 1]$ except for the Gaussian kernel which is supported on the entire $\mathbb{R}^+$. However, the Gaussian kernel is fast-decaying.

Kernel methods are linear

- ☐ Let $\hat{y}_i = \hat{f}_h(\mathbf{x}_i)$ be the usual fitted value for observation i.

  We have

  $$\hat{y}_i = \frac{\sum_r K_h(\mathbf{x}_i, \mathbf{x}_r) y_r}{\sum_r K_h(\mathbf{x}_i, \mathbf{x}_r)} = \sum_r s_h(i, r) y_r$$

  where

  $$s_h(i, r) = \frac{K_h(\mathbf{x}_i, \mathbf{x}_r)}{\sum_t K_h(\mathbf{x}_i, \mathbf{x}_t)}$$

  Hence,

  $$\widehat{\mathbf{y}} = \mathbf{S}_h \mathbf{y}$$

  where $\mathbf{S}_h = (s_h(i, r) : i, r \in \{1, \ldots, n\})$ is the smoother matrix.

  In that sense, kernel regression is linear.

- ☐ The degrees of freedom are defined as

  $$\mathrm{df}(h) = \mathrm{trace}(\mathbf{S}_h)$$

  This is in analogy with least squares, where $\mathbf{S}$ is the *hat matrix*.

Local Linear Regression (LOESS)

- Assume the predictor x is one-dimensional.
- The local linear estimator is

$$\hat{f}_h(x) = \widehat{\beta}_{h,0}(x) + \widehat{\beta}_{h,1}(x)x$$

where

$$(\widehat{\beta}_{h,0}(x), \widehat{\beta}_{h,1}(x)) = \underset{\beta_0,\beta_1}{\arg\min} \sum_{i=1}^{n} K_h(x, x_i)\big[y_i - \beta_0 - \beta_1 x_i\big]^2$$

- Note that the estimate is linear in the response. Indeed,

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

The weights $\ell_{h,i}(x)$ are referred to as equivalent kernel. We have

$$\widehat{\mathbf{y}} = \mathbf{S}_h \mathbf{y}, \quad \mathbf{S}_h = (\ell_{h,r}(x_i) : i, r \in \{1, \ldots, n\})$$

- The degrees of freedom are defined as before.

Local Polynomial Regression

- The local degree p polynomial estimator is

$$\hat{f}_h(x) = \widehat{\beta}_{h,0}(x) + \widehat{\beta}_{h,1}(x)x + \cdots + \widehat{\beta}_{h,p}(x)x^p$$

where

$$(\widehat{\beta}_{h,0}(x), \ldots, \widehat{\beta}_{h,p}(x)) = \underset{\beta_0,\ldots,\beta_p}{\arg\min} \sum_{i=1}^{n} K_h(x, x_i)\big[y_i - \beta_0 - \beta_1 x_i - \cdots - \beta_p x_i^p\big]^2$$

- The resulting method is also linear in the response.
- This approach generalizes to the case where $\mathbf{x}$ is multi-dimensional.

Local Regression

- Suppose we assume a linear model in some basis $\{g_0, \ldots, g_p\}$:

$$f_\theta(\mathbf{x}) = \sum_{j=0}^{p} \theta_j g_j(\mathbf{x})$$

  (Now $\mathbf{x}$ can be multivariate.)

- The local linear estimator is $f_{\widehat{\theta}_h(\mathbf{x})}(\mathbf{x})$, where

$$\widehat{\theta}_h(\mathbf{x}) = \arg\min_\theta \sum_{i=1}^{n} K_h(\mathbf{x}, \mathbf{x}_i)\big[y_i - f_\theta(\mathbf{x})\big]^2$$

- The resulting method is still linear in the response.

Choosing of the tuning parameter

- Assuming a model (when there is one) has been chosen. Then the window width h (also called bandwidth) is the only tuning parameter.

  (This is replaced by the neighborhood size k in the k-NN variant.)

- This tuning parameter controls the degrees of freedom. The smaller h is, the larger the degrees of freedom. The range is from 1 ($h \to \infty$) to n ($h \to 0$).

- This parameter can be chosen to minimize an estimate of prediction error, for example, obtained by cross-validation. (Many other methods have been proposed.)

The curse of dimensionality

- Consider a regression setting as before, $y = f(\mathbf{x}) + \varepsilon$, where $\mathbf{x} \in [0,1]^p$. We have data $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$ from this model.

- In a typical nonparametric setting, where f is only assumed to have some degree of smoothness (e.g., f is C^1), we are bound to learn about f locally. In fact, it is known that kernel methods (with a proper choice of bandwidth) are optimal in some sense.

- The issue in high-dimensions is that a sample $\mathbf{x}_1, \ldots, \mathbf{x}_n$ is not dense in $[0,1]^p$ unless n is exponential in p. Indeed, to cover $[0,1]^p$ with precision δ requires on the order of $n \approx \delta^{-p}$ sample points.

  ($p \geq 3$ is already challenging and $p \geq 10$ is hopeless)

  This is a symptom of the so-called curse of dimensionality.