

From Kernel Regression to Attention Mechanisms

A six decade journey (1960-2020)

Kernel Regression

(Statistics)

1960-90s

Data-Adaptive Filters

(Signal Processing)

1990-2010s

Attention

(Machine Learning)

2010-2020s

Kernel Regression/Smoothing (Nadaraya-Watson, '64)

Objective: fit a nonlinear relationship to paired data


$$y_i = y(x_i)$$

At any position x ,
one can estimate:

$$\hat{y}(x) = \frac{\sum_i K(x, x_i) y_i}{\sum_i K(x, x_i)}$$

$$K(x_i, x_j) = \exp \left\{ \frac{-\|x_i - x_j\|^2}{h_x^2} \right\}$$

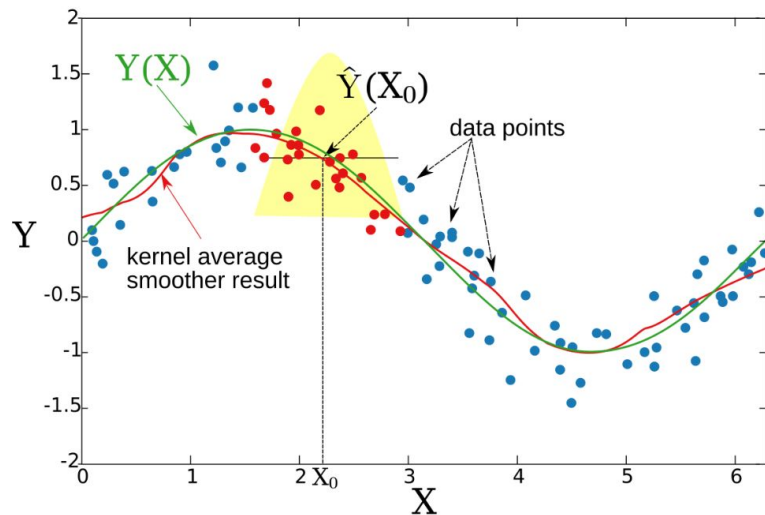
Positions



Kernel Regression/Smoothing (Nadaraya-Watson, '64)

$$K(x_i, x_j) = \exp \left\{ \frac{-\|x_i - x_j\|^2}{h_x^2} \right\}$$

Positions



Kernel Regression/Smoothing (Nadaraya-Watson, '64)

$$K(x_i, x_j) = \exp \left\{ \frac{-\|x_i - x_j\|^2}{h_x^2} \right\}$$

Positions

Bilateral Filter (Tomasi, Manduchi, '98)

$$K(x_i, x_j, y_i, y_j) = \exp \left\{ \frac{-\|x_i - x_j\|^2}{h_x^2} + \frac{-(y_i - y_j)^2}{h_y^2} \right\}$$

Positions

Pixels value

Bilateral Filter (Tomasi, Manduchi, '98)

$$K(x_i, x_j, y_i, y_j) = \exp \left\{ \frac{-\|x_i - x_j\|^2}{h_x^2} + \frac{-(y_i - y_j)^2}{h_y^2} \right\}$$

Diagram illustrating the Bilateral Filter kernel function. The equation shows the exponential of the sum of two squared differences, each divided by a spatial variance parameter (h_x^2 and h_y^2). Annotations indicate that x_i and x_j represent Positions, and y_i and y_j represent Pixels value.

Non-local Means (Buades, Coll, Morel, '05)

$$K(x_i, x_j, y_i, y_j) = \exp \left\{ \frac{-\|x_i - x_j\|^2}{h_x^2} + \frac{-\|\mathbf{y}_i - \mathbf{y}_j\|^2}{h_y^2} \right\}$$

Diagram illustrating the Non-local Means kernel function. The equation shows the exponential of the sum of two squared differences, each divided by a spatial variance parameter (h_x^2 and h_y^2). Annotations indicate that x_i and x_j represent Positions, and \mathbf{y}_i and \mathbf{y}_j represent Patches of Pixels. An infinity symbol (∞) is shown with an arrow pointing to the h_x^2 term, indicating a limit or constraint.

Non-local Means (Buades, Coll, Morel, '05)

$$K(x_i, x_j, y_i, y_j) = \exp \left\{ \frac{-\|x_i - x_j\|^2}{h_x^2} + \frac{-\|y_i - y_j\|^2}{h_y^2} \right\}$$

Diagram illustrating the Non-local Means kernel formula. The formula is shown with annotations:

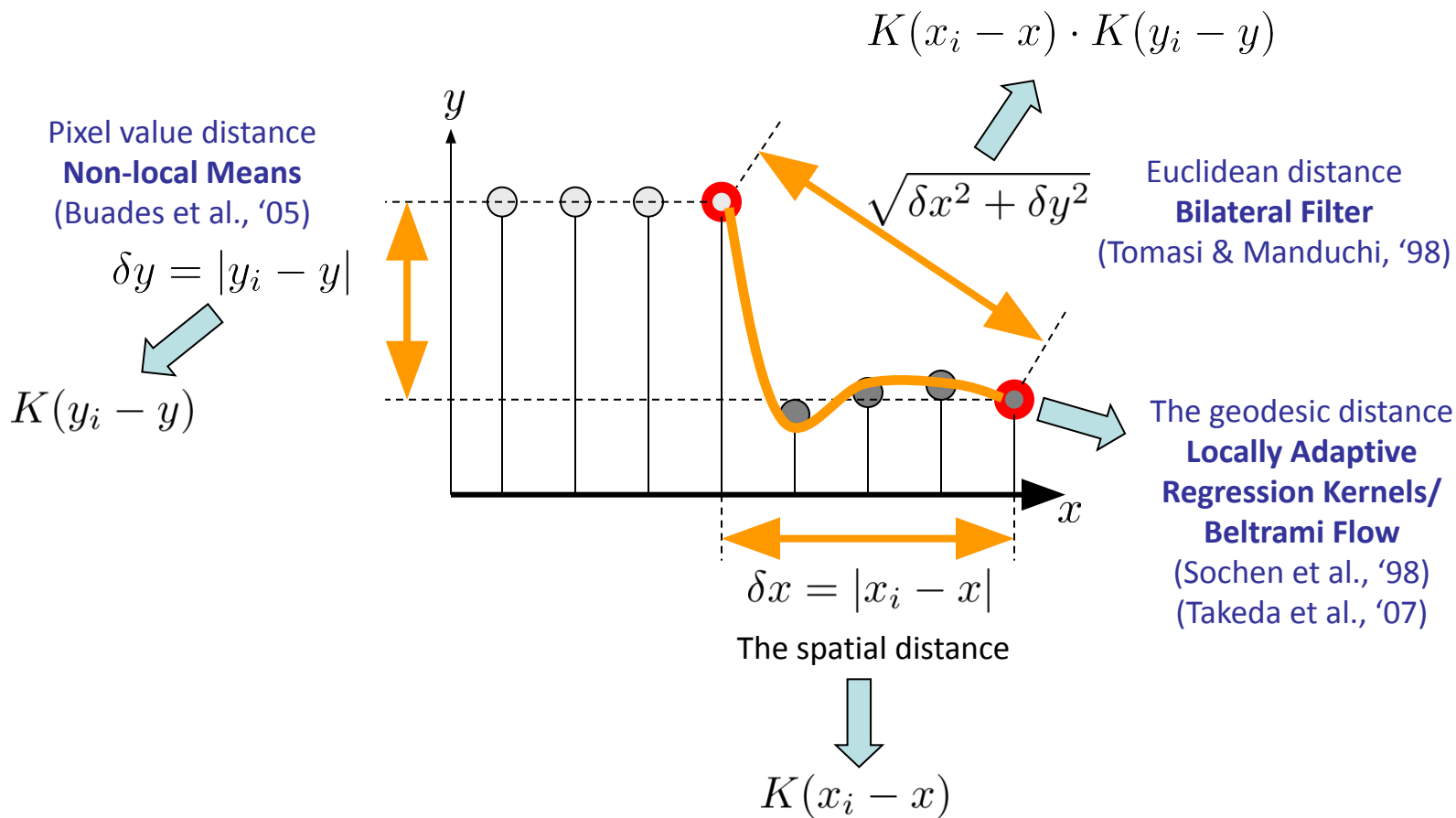
- Positions**: Points to the variables x_i and x_j in the first term of the exponent.
- Patches**: Points to the variables y_i and y_j in the second term of the exponent.
- ∞ : Points to the denominator h_x^2 , indicating a limit or scaling factor.

Locally Adaptive Regression Kernel (Takeda et al. '07)(Sochen et al., '98)

$$K(x_i, x_j, y_i, y_j) = \exp \left\{ - (x_i - x_j)^T \hat{\mathbf{C}}_{ij}(y) (x_i - x_j) \right\}$$

Diagram illustrating the Locally Adaptive Regression Kernel formula. The formula is shown with an annotation:

- Learned Metric**: Points to the matrix $\hat{\mathbf{C}}_{ij}(y)$ in the exponent, which is highlighted by a red box.



Consider Kernel with Augmented Variable:

$$\mathbf{t} = \begin{bmatrix} x \\ \mathbf{y} \end{bmatrix}$$

Positional
"encoding"

Pixel Values

$$K(\mathbf{t}_i, \mathbf{t}_j) = \exp \left\{ -(\mathbf{t}_i - \mathbf{t}_j)^T \mathbf{Q}_{i,j} (\mathbf{t}_i - \mathbf{t}_j) \right\}$$

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_x & 0 \\ 0 & \mathbf{Q}_y \end{bmatrix}$$

Symmetric, positive-definite

- Classical: $\mathbf{Q}_x = \frac{1}{h_x^2} \mathbf{I}$ and $\mathbf{Q}_y = 0$
- Bilateral: $\mathbf{Q}_x = \frac{1}{h_x^2} \mathbf{I}$ and $\mathbf{Q}_y = \frac{1}{h_y^2} \text{diag}[0, 0, \dots, 1, \dots, 0, 0]$
- Non-local Means: $\mathbf{Q}_x = 0$ and $\mathbf{Q}_y = \frac{1}{h_y^2} \mathbf{G}$
- LARK: $\mathbf{Q}_x = \mathbf{C}_{ij}$ and $\mathbf{Q}_y = 0$.

1. Choose any kernel from a
Reprod Kernel Hilbert Space

$$K(\mathbf{t}_i, \mathbf{t}_j) = \exp \left\{ -(\mathbf{t}_i - \mathbf{t}_j)^T \mathbf{Q}_{i,j} (\mathbf{t}_i - \mathbf{t}_j) \right\}$$

2. Define \mathbf{t} as a feature
vector (latent space)

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_x & 0 \\ 0 & \mathbf{Q}_y \end{bmatrix}$$

3. Allow off-diagonal
blocks and a more
general factored form:

$$\mathbf{Q} = \mathbf{Q}^T \mathbf{K}$$

$$\tilde{\mathbf{y}} = \mathbf{V} \mathbf{t}$$

4. Allow output to have
different dimensions
than the input or the
features

Attention

The diagram illustrates the Attention mechanism formula. It features three labels in blue text with arrows pointing to specific parts of the equation:
 - **QUERY** points to the \mathbb{Q} matrix in the exponent.
 - **KEY** points to the \mathbb{K} matrix in the exponent.
 - **VALUE** points to the \mathbb{V} matrix in the numerator.
 The formula is:
$$\sum_i \frac{e^{\langle \mathbb{Q}\mathbf{t}_i, \mathbb{K}\mathbf{t}_j \rangle}}{\sum_i e^{\langle \mathbb{Q}\mathbf{t}_i, \mathbb{K}\mathbf{t}_j \rangle}} \mathbb{V}\mathbf{t}_i$$