Lec 8. Kernel Smoothing Methods and Other Kernel-based Methods

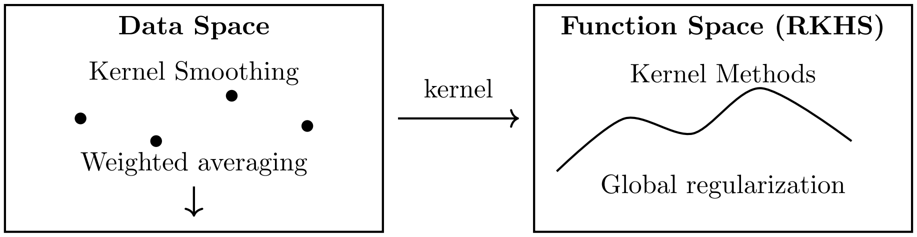
1 Kernel Functions

1.1 The Meaning of "Kernel" in Kernel Methods vs. Kernel Smoothing Methods

The term *kernel* appears in multiple areas of statistical learning, but its meaning depends on context. In *kernel methods*, the kernel defines a notion of similarity and induces a geometry on a function space. In *kernel smoothing*, the term kernel is used in a different but analogous way. Here, a kernel defines similarity through distance in the input space and is used to assign larger weights to nearby observations.

Key analogy:

- In kernel methods, similarity is defined via inner products.
- In kernel smoothing, similarity is defined via distance.



Kernel smoothing operates by assigning weights to data points in input space, while kernel methods operate by selecting a function in an RKHS through global regularization. Despite the difference, in both cases kernels encode how information is shared across observations.

1.2 Kernel in Kernel Methods

Recall that a kernel function can be written as

$$K(x, x') = \langle \phi(x), \phi(x') \rangle_{\mathcal{H}},$$

where $\phi(x)$ is a feature map into a (possibly infinite-dimensional) Hilbert space $\mathcal{H}$. Thus, $K(x, x')$ measures the inner product between the feature representations of x and x'.

Similarity in kernel methods. Two inputs x and x' are considered *similar* if $K(x, x')$ is large. This means that their feature representations $\phi(x)$ and $\phi(x')$ are well aligned in the Hilbert space. Importantly, this notion of similarity does not necessarily correspond to Euclidean distance in the input space. Different kernels encode different notions of similarity.

From similarity to geometry. Because the kernel defines an inner product in $\mathcal{H}$, it induces geometric concepts such as norms, angles, and distances in the associated function space. In particular, the RKHS norm

$$\|f\|_{\mathcal{H}_K} = \sqrt{\langle f, f \rangle_{\mathcal{H}_K}}$$

measures the complexity of a function relative to the kernel.

As a consequence, the kernel determines which functions are considered smooth or simple: functions that vary slowly across similar inputs (as defined by the kernel) have smaller RKHS norm and are therefore preferred by regularized kernel methods.

1.3 Kernel Functions in Kernel Smoothing

A *kernel* in kernel smoothing is a nonnegative function

$$K : \mathbb{R} \to \mathbb{R}_+$$

satisfying:

- $K(u) \geq 0$,
- $\int K(u)\, du = 1$,
- $K(u) = K(-u)$ (typically).

In practice, kernels are used with a *bandwidth* parameter $h > 0$:

$$K_h(x - x_0) = \frac{1}{h} K\left(\frac{x - x_0}{h}\right).$$

The bandwidth controls the size of the local neighborhood and plays a central role in the bias–variance tradeoff.

1.3.1 Common Kernels

Uniform (Rectangular) Kernel This kernel assigns equal weight to all observations within a fixed window and corresponds to simple local averaging.

$$K(u) = \begin{cases} \frac{1}{2}, & |u| \leq 1, \\ 0, & \text{otherwise.} \end{cases}$$

Triangular Kernel The triangular kernel gives more weight to points closer to the target location and less weight to points near the boundary of the window.

$$K(u) = \begin{cases} 1 - |u|, & |u| \leq 1, \\ 0, & \text{otherwise.} \end{cases}$$

Gaussian Kernel The Gaussian kernel has infinite support and assigns positive weight to all observations, with weights decaying exponentially with distance. While not compactly supported, it is smooth and widely used in practice.

$$K(u) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{u^2}{2}\right).$$

Remark. In practice, the choice of kernel function typically has much less impact on the estimator than the choice of bandwidth h. Most kernel functions yield similar results when used with an appropriately selected bandwidth.

2 One-Dimensional Kernel Smoother

We observe data

$$(x_1, y_1), \ldots, (x_n, y_n), \qquad y_i = f(x_i) + \varepsilon_i,$$

where $\mathbb{E}[\varepsilon_i] = 0$ and $\text{Var}(\varepsilon_i) = \sigma^2$.

Our goal is to estimate

$$f(x_0) = \mathbb{E}[Y \mid X = x_0]$$

without specifying a parametric form for f. The fundamental assumption behind kernel smoothing is that f is locally smooth so observations near x_0 are more informative than distant ones.

2.1 k-Nearest Neighbor Regression as a Kernel Smoother

Let $\mathcal{N}_k(x_0)$ denote the set of k nearest neighbors of x_0. The kNN regression estimator is

$$\hat{f}_{\text{kNN}}(x_0) = \frac{1}{k} \sum_{i \in \mathcal{N}_k(x_0)} y_i.$$

This estimator can be written as a weighted average:

$$\hat{f}_{\text{kNN}}(x_0) = \sum_{i=1}^{n} w_i(x_0) y_i, \quad w_i(x_0) = \frac{1}{k} \mathbf{1}\{x_i \in \mathcal{N}_k(x_0)\}.$$

Thus, kNN corresponds to a kernel smoother with a *uniform kernel* and an *adaptive bandwidth.*

2.2 The Nadaraya–Watson Kernel Estimator

A more general kernel smoother is the *Nadaraya–Watson estimator*

$$\hat{f}_{\mathrm{NW}}(x_0) = \frac{\sum_{i=1}^n K_h(x_i - x_0)\, y_i}{\sum_{i=1}^n K_h(x_i - x_0)}.$$

This estimator computes a locally weighted average of the responses, where weights decay smoothly with distance from x_0.

2.3 Local Linear Regression

The Nadaraya–Watson estimator can exhibit substantial bias near boundaries and when the regression function has nonzero slope. By fitting straight lines rather than constants locally, we can remove the bias exactly to first order. We fit a local linear model around x_0:

$$(\hat{\beta}_0, \hat{\beta}_1) = \arg\min_{\beta_0, \beta_1} \sum_{i=1}^n K_h(x_i - x_0)\big(y_i - \beta_0 - \beta_1(x_i - x_0)\big)^2.$$

The estimator of $f(x_0)$ is

$$\hat{f}(x_0) = \hat{\beta}_0.$$

Local linear regression automatically reduces boundary bias and generally outperforms local constant smoothing.

2.4 Local Polynomial Regression

Local polynomial regression extends the local linear idea by fitting a polynomial of degree p locally:

$$\min_{\beta_0, \ldots, \beta_p} \sum_{i=1}^n K_h(x_i - x_0) \left[y_i - \sum_{j=0}^p \beta_j (x_i - x_0)^j \right]^2.$$

Special cases include:

- $p = 0$: Nadaraya–Watson estimator,
- $p = 1$: Local linear regression.

Higher-order polynomials reduce bias but increase variance. In practice, $p = 1$ or $p = 2$ is usually sufficient.

3 Bandwidth Selection

The bandwidth h controls the bias–variance tradeoff and is the most important tuning parameter in kernel smoothing.

For the Gaussian kernel, the bandwidth h is the standard deviation of the kernel and determines how quickly the weights decay with distance. In k-nearest neighbor smoothing, the number of neighbors k serves as the bandwidth by controlling how many observations are included in the local average. For compactly supported kernels such as the uniform or tri-cube kernel, the bandwidth h determines the radius of support, beyond which observations receive zero weight.

3.1 Bias–Variance Intuition

- Large h: high bias, low variance.
- Small h: low bias, high variance.

The choice of bandwidth h typically has a much greater impact than the choice of kernel shape. Cross-validation is the most widely used in practice.

4 From One Dimension to $\mathbb{R}^p$

So far, we have focused on kernel methods in the one-dimensional setting, where inputs take the form $x \in \mathbb{R}$. Many of the key ideas—locality, smoothness, and similarity—extend naturally to the multivariate case. However, moving from 1D to $\mathbb{R}^p$ introduces both conceptual and statistical challenges.

4.1 Multivariate Kernel Smoothing

Let $x \in \mathbb{R}^p$. A multivariate kernel smoother can be written as

$$\hat{f}(x_0) = \frac{\sum_{i=1}^n K_h(x_i - x_0)\, y_i}{\sum_{i=1}^n K_h(x_i - x_0)},$$

where $K_h(\cdot)$ is a kernel defined on $\mathbb{R}^p$.

A common construction is the *product kernel*

$$K_h(x) = \prod_{j=1}^p \frac{1}{h_j} K\left(\frac{x_j}{h_j}\right),$$

where $h = (h_1, \ldots, h_p)$ are dimension-specific bandwidths.

Another widely used choice is a *radial kernel*

$$K_h(x) = \frac{1}{h^p} K\left(\frac{\|x\|}{h}\right),$$

which depends only on the Euclidean distance $\|x\|$.

4.2 Bandwidth as a Geometry in $\mathbb{R}^p$

In one dimension, the bandwidth h controls the width of a neighborhood around a target point x_0. In $\mathbb{R}^p$, the bandwidth determines the *shape and scale* of the local neighborhood, such as hyper-rectangles, hyperspheres, or more general ellipsoids.

A general formulation uses a positive-definite bandwidth matrix $H \in \mathbb{R}^{p \times p}$:

$$K_H(x) = |H|^{-1/2} K\Big(H^{-1/2} x\Big),$$

which allows for anisotropic smoothing and correlations across dimensions.

4.3 Curse of Dimensionality

A fundamental distinction between 1D and $\mathbb{R}^p$ is the *curse of dimensionality.* As the dimension p increases:

- local neighborhoods become sparse,
- distances between points become less informative,
- substantially more data are required to achieve the same level of accuracy.

The effective number of observations in a neighborhood of radius h scales as

$$nh^p,$$

which decreases rapidly with p unless the sample size n grows exponentially.

4.4 Relation to Kernel Methods in Function Space

The extension to $\mathbb{R}^p$ also highlights the conceptual difference between:

- *kernel smoothing*, which performs local averaging in data space, and
- *kernel methods* (e.g., kernel ridge regression), which operate in a function space induced by the kernel.

In high-dimensional settings, kernel methods often remain effective because they rely on global regularization in a reproducing kernel Hilbert space (RKHS), rather than explicit local neighborhoods.

5 Positioning Kernel Smoothing Among Kernel-Based Methods

Kernel smoothing methods are:

- local,
- nonparametric,
- based on explicit weighting in input space,
- not formulated as global optimization problems.

6 How Kernels Are Used in Modern Methods: Technical Connections

Although kernel-based methods appear in many modern areas, the role played by the kernel differs across settings. In all cases, the kernel serves as a mechanism for transferring local similarity into global structure, but the object on which the kernel acts and the mathematical operation it induces vary.

Distribution Learning and Two-Sample Testing

In distribution learning, kernels are used to embed probability distributions into a reproducing kernel Hilbert space (RKHS). Given a kernel K, the mean embedding of a distribution P is defined as

$$\mu_P = \mathbb{E}_{X \sim P}[K(X, \cdot)] \in \mathcal{H}_K.$$

Distances between distributions are then measured by the RKHS norm,

$$\mathrm{MMD}^2(P, Q) = \|\mu_P - \mu_Q\|^2_{\mathcal{H}_K}.$$

Here, the kernel determines which discrepancies between distributions are detectable, without requiring explicit density estimation. The comparison is purely geometric in the induced function space.

Causal Discovery and Dependence Testing

Kernel-based dependence measures, such as the Hilbert–Schmidt Independence Criterion (HSIC), quantify statistical dependence through covariance operators in RKHSs. Given kernels K_X and K_Y on X and Y, HSIC measures the squared norm of the cross-covariance operator,

$$\|\mathcal{C}_{XY}\|^2_{\mathrm{HS}} = \mathbb{E}[K_X(X, X')K_Y(Y, Y')] - \mathbb{E}[K_X(X, X')]\mathbb{E}[K_Y(Y, Y')].$$

The kernel enables detection of nonlinear dependence without specifying a parametric model, and RKHS norms provide a natural measure of dependence strength.

Manifold Learning and Geometric Data Analysis

In manifold learning, kernels are used to construct similarity graphs,

$$W_{ij} = K(x_i, x_j),$$

from which graph Laplacians or diffusion operators are derived. Eigenvalue problems are then solved to recover low-dimensional representations of the data.

In this setting, kernels encode local neighborhood structure and approximate geometric operators on an underlying manifold, rather than defining an RKHS for function regularization or prediction.

Bayesian Optimization and Scientific Machine Learning

In Gaussian process models, a kernel defines the covariance operator of a random function,

$$f \sim \mathcal{GP}(0, K).$$

Posterior inference corresponds to solving a regularized optimization problem in the associated RKHS,

$$\min_{f \in \mathcal{H}_K} \sum_i (y_i - f(x_i))^2 + \sigma^2 \|f\|_{\mathcal{H}_K}^2.$$

Here, the kernel encodes prior assumptions about smoothness and correlation, linking kernel methods with Bayesian uncertainty quantification.

Functional Data Analysis

Functional data analysis (FDA) provides a setting in which multiple roles of kernels appear simultaneously. When observations are functions $X_i(t)$, kernels are used for smoothing, regularization, and similarity measurement in function spaces.

Kernel smoothing is widely used to estimate mean and covariance functions from noisy or irregularly observed curves. For example, the mean function may be estimated as

$$\hat{\mu}(t) = \frac{\sum_{i,j} K_h(t_{ij} - t)\, Y_{ij}}{\sum_{i,j} K_h(t_{ij} - t)},$$

where the kernel controls local averaging over the domain.

In functional regression, regression functions are often assumed to lie in an RKHS, leading to estimators of the form

$$\min_{\beta \in \mathcal{H}_K} \sum_i \left(Y_i - \int X_i(t) \beta(t)\, dt \right)^2 + \lambda \|\beta\|_{\mathcal{H}_K}^2,$$

where the RKHS norm enforces smoothness of the regression function. Kernels may also be defined directly between functional observations, enabling kernel-based classification, clustering, and dimension reduction for functional data.

Scalable and Hybrid Kernel Methods

Modern kernel methods often rely on approximations such as random Fourier features or Nyström methods, which approximate the kernel by an explicit finite-dimensional feature map,

$$K(x, x') \approx \phi(x)^\top \phi(x').$$

Hybrid approaches, such as deep kernel learning and neural tangent kernels, use kernels to analyze or regularize function spaces induced by deep models while retaining kernel-based geometric structure.

Summary. Across modern applications, kernels encode similarity and smoothness, but their technical role varies: kernels may define local weights, RKHS geometries, covariance operators, graph structures, or feature approximations. Functional data analysis highlights how these roles can coexist within a single domain.