DSTA 2018-19, 2019-20:
Study the following parts:
---up to page 139
---(skip Sec. 5.1.1)
---Sec. 5.1.2 -(skip Mercer Kernel map for compact continuous spaces)
---Sec. 5.2 up to Example 5.7.

CHAPTER 5 Kernel Methods

Before we can mine data, it is important to first find a suitable data representation that facilitates data analysis. For example, for complex data such as text, sequences, images, and so on, we must typically extract or construct a set of attributes or features, so that we can represent the data instances as multivariate vectors. That is, given a data instance $\mathbf{x}$ (e.g., a sequence), we need to find a mapping ϕ, so that $\phi(\mathbf{x})$ is the vector representation of $\mathbf{x}$. Even when the input data is a numeric data matrix, if we wish to discover nonlinear relationships among the attributes, then a nonlinear mapping ϕ may be used, so that $\phi(\mathbf{x})$ represents a vector in the corresponding high-dimensional space comprising nonlinear attributes. We use the term *input space* to refer to the data space for the input data $\mathbf{x}$ and *feature space* to refer to the space of mapped vectors $\phi(\mathbf{x})$. Thus, given a set of data objects or instances $\mathbf{x}_i$, and given a mapping function ϕ, we can transform them into feature vectors $\phi(\mathbf{x}_i)$, which then allows us to analyze complex data instances via numeric analysis methods.

Example 5.1 (Sequence-based Features). Consider a dataset of DNA sequences over the alphabet $\Sigma = \{A, C, G, T\}$. One simple feature space is to represent each sequence in terms of the probability distribution over symbols in Σ. That is, given a sequence $\mathbf{x}$ with length $|\mathbf{x}| = m$, the mapping into feature space is given as

$$\phi(\mathbf{x}) = \{P(A), P(C), P(G), P(T)\}$$

where $P(s) = \frac{n_s}{m}$ is the probability of observing symbol $s \in \Sigma$, and n_s is the number of times s appears in sequence $\mathbf{x}$. Here the input space is the set of sequences Σ^*, and the feature space is $\mathbb{R}^4$. For example, if $\mathbf{x} = ACAGCAGTA$, with $m = |\mathbf{x}| = 9$, since A occurs four times, C and G occur twice, and T occurs once, we have

$$\phi(\mathbf{x}) = (4/9, 2/9, 2/9, 1/9) = (0.44, 0.22, 0.22, 0.11)$$

Likewise, for another sequence $\mathbf{y} = AGCAAGCGAG$, we have

$$\phi(\mathbf{y}) = (4/10, 2/10, 4/10, 0) = (0.4, 0.2, 0.4, 0)$$

The mapping ϕ now allows one to compute statistics over the data sample to make inferences about the population. For example, we may compute the mean symbol composition. We can also define the distance between any two sequences, for example,

$$\begin{aligned}\delta(\mathbf{x},\mathbf{y}) &= \left\|\phi(\mathbf{x})-\phi(\mathbf{y})\right\| \\ &= \sqrt{(0.44-0.4)^2+(0.22-0.2)^2+(0.22-0.4)^2+(0.11-0)^2} = 0.22\end{aligned}$$

We can compute larger feature spaces by considering, for example, the probability distribution over all substrings or words of size up to k over the alphabet Σ, and so on.

Example 5.2 (Nonlinear Features). As an example of a nonlinear mapping consider the mapping ϕ that takes as input a vector $\mathbf{x} = (x_1, x_2)^T \in \mathbb{R}^2$ and maps it to a "quadratic" feature space via the nonlinear mapping

$$\phi(\mathbf{x}) = (x_1^2, x_2^2, \sqrt{2}x_1x_2)^T \in \mathbb{R}^3$$

For example, the point $\mathbf{x} = (5.9, 3)^T$ is mapped to the vector

$$\phi(\mathbf{x}) = (5.9^2, 3^2, \sqrt{2}\cdot 5.9\cdot 3)^T = (34.81, 9, 25.03)^T$$

The main benefit of this transformation is that we may apply well-known linear analysis methods in the feature space. However, because the features are nonlinear combinations of the original attributes, this allows us to mine nonlinear patterns and relationships.

Whereas mapping into feature space allows one to analyze the data via algebraic and probabilistic modeling, the resulting feature space is usually very high-dimensional; it may even be infinite dimensional. Thus, transforming all the input points into feature space can be very expensive, or even impossible. Because the dimensionality is high, we also run into the curse of dimensionality highlighted later in Chapter 6.

Kernel methods avoid explicitly transforming each point $\mathbf{x}$ in the input space into the mapped point $\phi(\mathbf{x})$ in the feature space. Instead, the input objects are represented via their $n \times n$ pairwise similarity values. The similarity function, called a *kernel*, is chosen so that it represents a dot product in some high-dimensional feature space, yet it can be computed without directly constructing $\phi(\mathbf{x})$. Let $\mathcal{I}$ denote the input space, which can comprise any arbitrary set of objects, and let $\mathbf{D} = \{\mathbf{x}_i\}_{i=1}^n \subset \mathcal{I}$ be a dataset comprising n objects in the input space. We can represent the pairwise similarity values between points in $\mathbf{D}$ via the $n \times n$ *kernel matrix*, defined as

$$\mathbf{K} = \begin{pmatrix} K(\mathbf{x}_1,\mathbf{x}_1) & K(\mathbf{x}_1,\mathbf{x}_2) & \cdots & K(\mathbf{x}_1,\mathbf{x}_n) \\ K(\mathbf{x}_2,\mathbf{x}_1) & K(\mathbf{x}_2,\mathbf{x}_2) & \cdots & K(\mathbf{x}_2,\mathbf{x}_n) \\ \vdots & \vdots & \ddots & \vdots \\ K(\mathbf{x}_n,\mathbf{x}_1) & K(\mathbf{x}_n,\mathbf{x}_2) & \cdots & K(\mathbf{x}_n,\mathbf{x}_n) \end{pmatrix}$$

where $K\colon \mathcal{I}\times\mathcal{I}\to\mathbb{R}$ is a *kernel function* on any two points in input space. However, we require that K corresponds to a dot product in some feature space. That is, for any

$\mathbf{x}_i, \mathbf{x}_j \in \mathcal{I}$, the kernel function should satisfy the condition

$$K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \tag{5.1}$$

where $\phi: \mathcal{I} \to \mathcal{F}$ is a mapping from the input space $\mathcal{I}$ to the feature space $\mathcal{F}$. Intuitively, this means that we should be able to compute the value of the dot product using the original input representation $\mathbf{x}$, without having recourse to the mapping $\phi(\mathbf{x})$. Obviously, not just any arbitrary function can be used as a kernel; a valid kernel function must satisfy certain conditions so that Eq. (5.1) remains valid, as discussed in Section 5.1.

It is important to remark that the transpose operator for the dot product applies only when $\mathcal{F}$ is a vector space. When $\mathcal{F}$ is an abstract vector space with an inner product, the kernel is written as $K(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle$. However, for convenience we use the transpose operator throughout this chapter; when $\mathcal{F}$ is an inner product space it should be understood that

$$\phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \equiv \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle$$

Example 5.3 (Linear and Quadratic Kernels). Consider the identity mapping, $\phi(\mathbf{x}) \to \mathbf{x}$. This naturally leads to the *linear kernel*, which is simply the dot product between two input vectors, and thus satisfies Eq. (5.1):

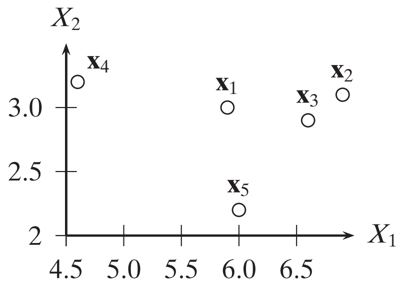
$$\phi(\mathbf{x})^T \phi(\mathbf{y}) = \mathbf{x}^T \mathbf{y} = K(\mathbf{x}, \mathbf{y})$$

For example, consider the first five points from the two-dimensional Iris dataset shown in Figure 5.1a:

$$\mathbf{x}_1 = \begin{pmatrix} 5.9 \\ 3 \end{pmatrix} \quad \mathbf{x}_2 = \begin{pmatrix} 6.9 \\ 3.1 \end{pmatrix} \quad \mathbf{x}_3 = \begin{pmatrix} 6.6 \\ 2.9 \end{pmatrix} \quad \mathbf{x}_4 = \begin{pmatrix} 4.6 \\ 3.2 \end{pmatrix} \quad \mathbf{x}_5 = \begin{pmatrix} 6 \\ 2.2 \end{pmatrix}$$

The kernel matrix for the linear kernel is shown in Figure 5.1b. For example,

$$K(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1^T \mathbf{x}_2 = 5.9 \times 6.9 + 3 \times 3.1 = 40.71 + 9.3 = 50.01$$



(a)

$\mathbf{K}$	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_3$	$\mathbf{x}_4$	$\mathbf{x}_5$
$\mathbf{x}_1$	43.81	50.01	47.64	36.74	42.00
$\mathbf{x}_2$	50.01	57.22	54.53	41.66	48.22
$\mathbf{x}_3$	47.64	54.53	51.97	39.64	45.98
$\mathbf{x}_4$	36.74	41.66	39.64	31.40	34.64
$\mathbf{x}_5$	42.00	48.22	45.98	34.64	40.84

(b)

Figure 5.1. (a) Example points. (b) Linear kernel matrix.

Consider the quadratic mapping $\phi : \mathbb{R}^2 \to \mathbb{R}^3$ from Example 5.2, that maps $\mathbf{x} = (x_1, x_2)^T$ as follows:

$$\phi(\mathbf{x}) = (x_1^2, x_2^2, \sqrt{2}x_1x_2)^T$$

The dot product between the mapping for two input points $\mathbf{x}, \mathbf{y} \in \mathbb{R}^2$ is given as

$$\phi(\mathbf{x})^T\phi(\mathbf{y}) = x_1^2y_1^2 + x_2^2y_2^2 + 2x_1y_1x_2y_2$$

We can rearrange the preceding to obtain the (homogeneous) *quadratic kernel* function as follows:

$$\begin{aligned}\phi(\mathbf{x})^T\phi(\mathbf{y}) &= x_1^2y_1^2 + x_2^2y_2^2 + 2x_1y_1x_2y_2 \\ &= (x_1y_1 + x_2y_2)^2 \\ &= (\mathbf{x}^T\mathbf{y})^2 \\ &= K(\mathbf{x}, \mathbf{y})\end{aligned}$$

We can thus see that the dot product in feature space can be computed by evaluating the kernel in input space, without explicitly mapping the points into feature space. For example, we have

$$\begin{aligned}\phi(\mathbf{x}_1) &= (5.9^2, 3^2, \sqrt{2}\cdot 5.9\cdot 3)^T = (34.81, 9, 25.03)^T \\ \phi(\mathbf{x}_2) &= (6.9^2, 3.1^2, \sqrt{2}\cdot 6.9\cdot 3.1)^T = (47.61, 9.61, 30.25)^T \\ \phi(\mathbf{x}_1)^T\phi(\mathbf{x}_2) &= 34.81\times 47.61 + 9\times 9.61 + 25.03\times 30.25 = 2501\end{aligned}$$

We can verify that the homogeneous quadratic kernel gives the same value

$$K(\mathbf{x}_1, \mathbf{x}_2) = (\mathbf{x}_1^T\mathbf{x}_2)^2 = (50.01)^2 = 2501$$

We shall see that many data mining methods can be *kernelized*, that is, instead of mapping the input points into feature space, the data can be represented via the $n \times n$ kernel matrix $\mathbf{K}$, and all relevant analysis can be performed over $\mathbf{K}$. This is usually done via the so-called *kernel trick*, that is, show that the analysis task requires only dot products $\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$ in feature space, which can be replaced by the corresponding kernel $K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$ that can be computed efficiently in input space. Once the kernel matrix has been computed, we no longer even need the input points $\mathbf{x}_i$, as all operations involving only dot products in the feature space can be performed over the $n \times n$ kernel matrix $\mathbf{K}$. An immediate consequence is that when the input data is the typical $n \times d$ numeric matrix $\mathbf{D}$ and we employ the linear kernel, the results obtained by analyzing $\mathbf{K}$ are equivalent to those obtained by analyzing $\mathbf{D}$ (as long as only dot products are involved in the analysis). Of course, kernel methods allow much more flexibility, as we can just as easily perform non-linear analysis by employing nonlinear kernels, or we may analyze (non-numeric) complex objects without explicitly constructing the mapping $\phi(\mathbf{x})$.

Example 5.4. Consider the five points from Example 5.3 along with the linear kernel matrix shown in Figure 5.1. The mean of the five points in feature space is simply the mean in input space, as ϕ is the identity function for the linear kernel:

$$\boldsymbol{\mu}_\phi = \sum_{i=1}^{5} \phi(\mathbf{x}_i) = \sum_{i=1}^{5} \mathbf{x}_i = (6.00, 2.88)^T$$

Now consider the squared magnitude of the mean in feature space:

$$\|\boldsymbol{\mu}_\phi\|^2 = \boldsymbol{\mu}_\phi^T \boldsymbol{\mu}_\phi = (6.0^2 + 2.88^2) = 44.29$$

Because this involves only a dot product in feature space, the squared magnitude can be computed directly from $\mathbf{K}$. As we shall see later [see Eq. (5.12)] the squared norm of the mean vector in feature space is equivalent to the average value of the kernel matrix $\mathbf{K}$. For the kernel matrix in Figure 5.1b we have

$$\frac{1}{5^2} \sum_{i=1}^{5} \sum_{j=1}^{5} K(\mathbf{x}_i, \mathbf{x}_j) = \frac{1107.36}{25} = 44.29$$

which matches the $\|\boldsymbol{\mu}_\phi\|^2$ value computed earlier. This example illustrates that operations involving dot products in feature space can be cast as operations over the kernel matrix $\mathbf{K}$.

Kernel methods offer a radically different view of the data. Instead of thinking of the data as vectors in input or feature space, we consider only the kernel values between pairs of points. The kernel matrix can also be considered as a weighted adjacency matrix for the complete graph over the n input points, and consequently there is a strong connection between kernels and graph analysis, in particular algebraic graph theory.

5.1 KERNEL MATRIX

Let $\mathcal{I}$ denote the input space, which can be any arbitrary set of data objects, and let $\mathbf{D} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\} \subset \mathcal{I}$ denote a subset of n objects in the input space. Let $\phi: \mathcal{I} \to \mathcal{F}$ be a mapping from the input space into the feature space $\mathcal{F}$, which is endowed with a dot product and norm. Let $K: \mathcal{I} \times \mathcal{I} \to \mathbb{R}$ be a function that maps pairs of input objects to their dot product value in feature space, that is, $K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$, and let $\mathbf{K}$ be the $n \times n$ kernel matrix corresponding to the subset $\mathbf{D}$.

The function K is called a **positive semidefinite kernel** if and only if it is symmetric:

$$K(\mathbf{x}_i, \mathbf{x}_j) = K(\mathbf{x}_j, \mathbf{x}_i)$$

and the corresponding kernel matrix $\mathbf{K}$ for any subset $\mathbf{D} \subset \mathcal{I}$ is positive semidefinite, that is,

$$\mathbf{a}^T \mathbf{K} \mathbf{a} \geq 0, \text{ for all vectors } \mathbf{a} \in \mathbb{R}^n$$

which implies that

$$\sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j K(\mathbf{x}_i, \mathbf{x}_j) \ge 0, \text{ for all } a_i \in \mathbb{R}, i \in [1, n] \tag{5.2}$$

We first verify that if $K(\mathbf{x}_i, \mathbf{x}_j)$ represents the dot product $\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$ in some feature space, then K is a positive semidefinite kernel. Consider any dataset $\mathbf{D}$, and let $\mathbf{K} = \{K(\mathbf{x}_i, \mathbf{x}_j)\}$ be the corresponding kernel matrix. First, K is symmetric since the dot product is symmetric, which also implies that $\mathbf{K}$ is symmetric. Second, $\mathbf{K}$ is positive semidefinite because

$$\begin{aligned}
\mathbf{a}^T\mathbf{K}\mathbf{a} &= \sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j K(\mathbf{x}_i, \mathbf{x}_j) \\
&= \sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j) \\
&= \left(\sum_{i=1}^{n} a_i \phi(\mathbf{x}_i)\right)^T \left(\sum_{j=1}^{n} a_j \phi(\mathbf{x}_j)\right) \\
&= \left\| \sum_{i=1}^{n} a_i \phi(\mathbf{x}_i) \right\|^2 \ge 0
\end{aligned}$$

Thus, K is a positive semidefinite kernel.

We now show that if we are given a positive semidefinite kernel $K : \mathcal{I} \times \mathcal{I} \to \mathbb{R}$, then it corresponds to a dot product in some feature space $\mathcal{F}$.

5.1.1 Reproducing Kernel Map

For the reproducing kernel map ϕ, we map each point $\mathbf{x} \in \mathcal{I}$ into a function in a *functional space* $\{f : \mathcal{I} \to \mathbb{R}\}$ comprising functions that map points in $\mathcal{I}$ into $\mathbb{R}$. Algebraically this space of functions is an abstract vector space where each point happens to be a function. In particular, any $\mathbf{x} \in \mathcal{I}$ in the input space is mapped to the following function:

$$\phi(\mathbf{x}) = K(\mathbf{x}, \cdot)$$

where the $\cdot$ stands for any argument in $\mathcal{I}$. That is, each object $\mathbf{x}$ in the input space gets mapped to a *feature point* $\phi(\mathbf{x})$, which is in fact a function $K(\mathbf{x}, \cdot)$ that represents its similarity to all other points in the input space $\mathcal{I}$.

Let $\mathcal{F}$ be the set of all functions or points that can be obtained as a linear combination of any subset of feature points, defined as

$$\begin{aligned}
\mathcal{F} &= span\big\{K(\mathbf{x}, \cdot) \mid \mathbf{x} \in \mathcal{I}\big\} \\
&= \left\{ \mathbf{f} = f(\cdot) = \sum_{i=1}^{m} \alpha_i\, K(\mathbf{x}_i, \cdot) \;\middle|\; m \in \mathbb{N}, \alpha_i \in \mathbb{R}, \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subseteq \mathcal{I} \right\}
\end{aligned}$$

We use the dual notation $\mathbf{f}$ and $f(\cdot)$ interchangeably to emphasize the fact that each point $\mathbf{f}$ in the feature space is in fact a function $f(\cdot)$. Note that by definition the feature point $\phi(\mathbf{x}) = K(\mathbf{x}, \cdot)$ belongs to $\mathcal{F}$.

Let $\mathbf{f}, \mathbf{g} \in \mathcal{F}$ be any two points in feature space:

$$\mathbf{f} = f(\cdot) = \sum_{i=1}^{m_a} \alpha_i \, K(\mathbf{x}_i, \cdot) \qquad\qquad \mathbf{g} = g(\cdot) = \sum_{j=1}^{m_b} \beta_j \, K(\mathbf{x}_j, \cdot)$$

Define the dot product between two points as

$$\mathbf{f}^T \mathbf{g} = f(\cdot)^T g(\cdot) = \sum_{i=1}^{m_a} \sum_{j=1}^{m_b} \alpha_i \beta_j K(\mathbf{x}_i, \mathbf{x}_j) \tag{5.3}$$

We emphasize that the notation $\mathbf{f}^T \mathbf{g}$ is only a convenience; it denotes the inner product $\langle \mathbf{f}, \mathbf{g} \rangle$ because $\mathcal{F}$ is an abstract vector space, with an inner product as defined above.

We can verify that the dot product is *bilinear*, that is, linear in both arguments, because

$$\mathbf{f}^T \mathbf{g} = \sum_{i=1}^{m_a} \sum_{j=1}^{m_b} \alpha_i \, \beta_j \, K(\mathbf{x}_i, \mathbf{x}_j) = \sum_{i=1}^{m_a} \alpha_i \, g(\mathbf{x}_i) = \sum_{j=1}^{m_b} \beta_j \, f(\mathbf{x}_j)$$

The fact that K is positive semidefinite implies that

$$\|\mathbf{f}\|^2 = \mathbf{f}^T \mathbf{f} = \sum_{i=1}^{m_a} \sum_{j=1}^{m_a} \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}) \geq 0$$

Thus, the space $\mathcal{F}$ is a *pre-Hilbert space*, defined as a normed inner product space, because it is endowed with a symmetric bilinear dot product and a norm. By adding the limit points of all Cauchy sequences that are convergent, $\mathcal{F}$ can be turned into a *Hilbert space*, defined as a normed inner product space that is complete. However, showing this is beyond the scope of this chapter.

The space $\mathcal{F}$ has the so-called *reproducing property*, that is, we can evaluate a function $f(\cdot) = \mathbf{f}$ at a point $\mathbf{x} \in \mathcal{I}$ by taking the dot product of $\mathbf{f}$ with $\phi(\mathbf{x})$, that is,

$$\mathbf{f}^T \phi(\mathbf{x}) = f(\cdot)^T K(\mathbf{x}, \cdot) = \sum_{i=1}^{m_a} \alpha_i \, K(\mathbf{x}_i, \mathbf{x}) = f(\mathbf{x})$$

For this reason, the space $\mathcal{F}$ is also called a *reproducing kernel Hilbert space*.

All we have to do now is to show that $K(\mathbf{x}_i, \mathbf{x}_j)$ corresponds to a dot product in the feature space $\mathcal{F}$. This is indeed the case, because using Eq. (5.3) for any two feature points $\phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \in \mathcal{F}$ their dot product is given as

$$\phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) = K(\mathbf{x}_i, \cdot)^T K(\mathbf{x}_j, \cdot) = K(\mathbf{x}_i, \mathbf{x}_j)$$

The reproducing kernel map shows that any positive semidefinite kernel corresponds to a dot product in some feature space. This means we can apply well known algebraic and geometric methods to understand and analyze the data in these spaces.

Empirical Kernel Map

The reproducing kernel map ϕ maps the input space into a potentially infinite dimensional feature space. However, given a dataset $\mathbf{D} = \{\mathbf{x}_i\}_{i=1}^n$, we can obtain a finite dimensional mapping by evaluating the kernel only on points in $\mathbf{D}$. That is, define the map ϕ as follows:

$$\phi(\mathbf{x}) = \Big((K(\mathbf{x}_1, \mathbf{x}), K(\mathbf{x}_2, \mathbf{x}), \ldots, K(\mathbf{x}_n, \mathbf{x})\Big)^T \in \mathbb{R}^n$$

which maps each point $\mathbf{x} \in \mathcal{I}$ to the n-dimensional vector comprising the kernel values of $\mathbf{x}$ with each of the objects $\mathbf{x}_i \in \mathbf{D}$. We can define the dot product in feature space as

$$\phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) = \sum_{k=1}^{n} K(\mathbf{x}_k, \mathbf{x}_i) K(\mathbf{x}_k, \mathbf{x}_j) = \mathbf{K}_i^T \mathbf{K}_j \tag{5.4}$$

where $\mathbf{K}_i$ denotes the ith column of $\mathbf{K}$, which is also the same as the ith row of $\mathbf{K}$ (considered as a column vector), as $\mathbf{K}$ is symmetric. However, for ϕ to be a valid map, we require that $\phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) = K(\mathbf{x}_i, \mathbf{x}_j)$, which is clearly not satisfied by Eq. (5.4). One solution is to replace $\mathbf{K}_i^T \mathbf{K}_j$ in Eq. (5.4) with $\mathbf{K}_i^T \mathbf{A} \mathbf{K}_j$ for some positive semidefinite matrix $\mathbf{A}$ such that

$$\mathbf{K}_i^T \mathbf{A} \mathbf{K}_j = \mathbf{K}(\mathbf{x}_i, \mathbf{x}_j)$$

If we can find such an $\mathbf{A}$, it would imply that over all pairs of mapped points we have

$$\Big\{\mathbf{K}_i^T \mathbf{A} \mathbf{K}_j\Big\}_{i,j=1}^{n} = \Big\{K(\mathbf{x}_i, \mathbf{x}_j)\Big\}_{i,j=1}^{n}$$

which can be written compactly as

$$\mathbf{KAK} = \mathbf{K}$$

This immediately suggests that we take $\mathbf{A} = \mathbf{K}^{-1}$, the (pseudo) inverse of the kernel matrix $\mathbf{K}$. The modified map ϕ, called the *empirical kernel map*, is then defined as

$$\phi(\mathbf{x}) = \mathbf{K}^{-1/2} \cdot \Big((K(\mathbf{x}_1, \mathbf{x}), K(\mathbf{x}_2, \mathbf{x}), \ldots, K(\mathbf{x}_n, \mathbf{x})\Big)^T \in \mathbb{R}^n$$

so that the dot product yields

$$\begin{aligned} \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) &= \Big(\mathbf{K}^{-1/2}\, \mathbf{K}_i\Big)^T \Big(\mathbf{K}^{-1/2}\, \mathbf{K}_j\Big) \\ &= \mathbf{K}_i^T \left(\mathbf{K}^{-1/2} \mathbf{K}^{-1/2}\right) \mathbf{K}_j \\ &= \mathbf{K}_i^T\, \mathbf{K}^{-1}\, \mathbf{K}_j \end{aligned}$$

Over all pairs of mapped points, we have

$$\left\{\mathbf{K}_i^T \mathbf{K}^{-1}\, \mathbf{K}_j\right\}_{i,j=1}^{n} = \mathbf{K}\, \mathbf{K}^{-1}\, \mathbf{K} = \mathbf{K}$$

as desired. However, it is important to note that this empirical feature representation is valid only for the n points in $\mathbf{D}$. If points are added to or removed from $\mathbf{D}$, the kernel map will have to be updated for all points.

5.1.2 Mercer Kernel Map

In general different feature spaces can be constructed for the same kernel K. We now describe how to construct the Mercer map.

Data-specific Kernel Map

The Mercer kernel map is best understood starting from the kernel matrix for the dataset $\mathbf{D}$ in input space. Because $\mathbf{K}$ is a symmetric positive semidefinite matrix, it has real and non-negative eigenvalues, and it can be decomposed as follows:

$$\mathbf{K} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$$

where $\mathbf{U}$ is the orthonormal matrix of eigenvectors $\mathbf{u}_i = (u_{i1}, u_{i2}, \ldots, u_{in})^T \in \mathbb{R}^n$ (for $i = 1, \ldots, n$), and $\mathbf{\Lambda}$ is the diagonal matrix of eigenvalues, with both arranged in non-increasing order of the eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \geq 0$:

$$\mathbf{U} = \begin{pmatrix} | & | & & | \\ \mathbf{u}_1 & \mathbf{u}_2 & \cdots & \mathbf{u}_n \\ | & | & & | \end{pmatrix} \qquad \mathbf{\Lambda} = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{pmatrix}$$

The kernel matrix $\mathbf{K}$ can therefore be rewritten as the spectral sum

$$\mathbf{K} = \lambda_1 \mathbf{u}_1 \mathbf{u}_1^T + \lambda_2 \mathbf{u}_2 \mathbf{u}_2^T + \cdots + \lambda_n \mathbf{u}_n \mathbf{u}_n^T$$

In particular the kernel function between $\mathbf{x}_i$ and $\mathbf{x}_j$ is given as

$$\begin{aligned} \mathbf{K}(\mathbf{x}_i, \mathbf{x}_j) &= \lambda_1\, u_{1i}\, u_{1j} + \lambda_2\, u_{2i}\, u_{2j} \cdots + \lambda_n\, u_{ni}\, u_{nj} \\ &= \sum_{k=1}^{n} \lambda_k\, u_{ki}\, u_{kj} \end{aligned} \tag{5.5}$$

where u_{ki} denotes the ith component of eigenvector $\mathbf{u}_k$. It follows that if we define the Mercer map ϕ as follows:

$$\phi(\mathbf{x}_i) = \left(\sqrt{\lambda_1}\, u_{1i}, \sqrt{\lambda_2}\, u_{2i}, \ldots, \sqrt{\lambda_n}\, u_{ni}\right)^T \tag{5.6}$$

then $\mathbf{K}(\mathbf{x}_i, \mathbf{x}_j)$ is a dot product in feature space between the mapped points $\phi(\mathbf{x}_i)$ and $\phi(\mathbf{x}_j)$ because

$$\begin{aligned} \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) &= \left(\sqrt{\lambda_1}\, u_{1i}, \ldots, \sqrt{\lambda_n}\, u_{ni}\right)\left(\sqrt{\lambda_1}\, u_{1j}, \ldots, \sqrt{\lambda_n}\, u_{nj}\right)^T \\ &= \lambda_1\, u_{1i}\, u_{1j} + \cdots + \lambda_n\, u_{ni}\, u_{nj} = K(\mathbf{x}_i, \mathbf{x}_j) \end{aligned}$$

Noting that $\mathbf{U}_i = (u_{1i}, u_{2i}, \ldots, u_{ni})^T$ is the ith row of $\mathbf{U}$, we can rewrite the Mercer map ϕ as

$$\phi(\mathbf{x}_i) = \sqrt{\mathbf{\Lambda}}\, \mathbf{U}_i \tag{5.7}$$

Thus, the kernel value is simply the dot product between scaled rows of $\mathbf{U}$:

$$\phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) = \left(\sqrt{\mathbf{\Lambda}}\, \mathbf{U}_i\right)^T \left(\sqrt{\mathbf{\Lambda}}\, \mathbf{U}_j\right) = \mathbf{U}_i^T \mathbf{\Lambda} \mathbf{U}_j$$

The Mercer map, defined equivalently in Eqs. (5.6) and (5.7), is obviously restricted to the input dataset $\mathbf{D}$, just like the empirical kernel map, and is therefore called the *data-specific Mercer kernel map*. It defines a data-specific feature space of dimensionality at most n, comprising the eigenvectors of $\mathbf{K}$.

Example 5.5. Let the input dataset comprise the five points shown in Figure 5.1a, and let the corresponding kernel matrix be as shown in Figure 5.1b. Computing the eigen-decomposition of $\mathbf{K}$, we obtain $\lambda_1 = 223.95$, $\lambda_2 = 1.29$, and $\lambda_3 = \lambda_4 = \lambda_5 = 0$. The effective dimensionality of the feature space is 2, comprising the eigenvectors $\mathbf{u}_1$ and $\mathbf{u}_2$. Thus, the matrix $\mathbf{U}$ is given as follows:

$$\mathbf{U} = \begin{pmatrix} & \mathbf{u}_1 & \mathbf{u}_2 \\ \hline \mathbf{U}_1 & -0.442 & 0.163 \\ \mathbf{U}_2 & -0.505 & -0.134 \\ \mathbf{U}_3 & -0.482 & -0.181 \\ \mathbf{U}_4 & -0.369 & 0.813 \\ \mathbf{U}_5 & -0.425 & -0.512 \end{pmatrix}$$

and we have

$$\mathbf{\Lambda} = \begin{pmatrix} 223.95 & 0 \\ 0 & 1.29 \end{pmatrix} \qquad \sqrt{\mathbf{\Lambda}} = \begin{pmatrix} \sqrt{223.95} & 0 \\ 0 & \sqrt{1.29} \end{pmatrix} = \begin{pmatrix} 14.965 & 0 \\ 0 & 1.135 \end{pmatrix}$$

The kernel map is specified via Eq. (5.7). For example, for $\mathbf{x}_1 = (5.9, 3)^T$ and $\mathbf{x}_2 = (6.9, 3.1)^T$ we have

$$\phi(\mathbf{x}_1) = \sqrt{\mathbf{\Lambda}}\mathbf{U}_1 = \begin{pmatrix} 14.965 & 0 \\ 0 & 1.135 \end{pmatrix} \begin{pmatrix} -0.442 \\ 0.163 \end{pmatrix} = \begin{pmatrix} -6.616 \\ 0.185 \end{pmatrix}$$

$$\phi(\mathbf{x}_2) = \sqrt{\mathbf{\Lambda}}\mathbf{U}_2 = \begin{pmatrix} 14.965 & 0 \\ 0 & 1.135 \end{pmatrix} \begin{pmatrix} -0.505 \\ -0.134 \end{pmatrix} = \begin{pmatrix} -7.563 \\ -0.153 \end{pmatrix}$$

Their dot product is given as

$$\begin{aligned} \phi(\mathbf{x}_1)^T\phi(\mathbf{x}_2) &= 6.616 \times 7.563 - 0.185 \times 0.153 \\ &= 50.038 - 0.028 = 50.01 \end{aligned}$$

which matches the kernel value $K(\mathbf{x}_1, \mathbf{x}_2)$ in Figure 5.1b.

Mercer Kernel Map

For compact continuous spaces, analogous to the discrete case in Eq. (5.5), the kernel value between any two points can be written as the infinite spectral decomposition

$$K(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{\infty} \lambda_k \, \mathbf{u}_k(\mathbf{x}_i) \, \mathbf{u}_k(\mathbf{x}_j)$$

where $\{\lambda_1, \lambda_2, \ldots\}$ is the infinite set of eigenvalues, and $\{\mathbf{u}_1(\cdot), \mathbf{u}_2(\cdot), \ldots\}$ is the corresponding set of orthogonal and normalized *eigenfunctions*, that is, each function $\mathbf{u}_i(\cdot)$ is a solution to the integral equation

$$\int K(\mathbf{x}, \mathbf{y}) \, \mathbf{u}_i(\mathbf{y}) \, d\mathbf{y} = \lambda_i \mathbf{u}_i(\mathbf{x})$$

and K is a continuous positive semidefinite kernel, that is, for all functions $a(\cdot)$ with a finite square integral (i.e., $\int a(\mathbf{x})^2\, d\mathbf{x} < 0$) K satisfies the condition

$$\int\int K(\mathbf{x}_1,\mathbf{x}_2)\, a(\mathbf{x}_1)\, a(\mathbf{x}_2)\, d\mathbf{x}_1\, d\mathbf{x}_2 \geq 0$$

We can see that this positive semidefinite kernel for compact continuous spaces is analogous to the the discrete kernel in Eq. (5.2). Further, similarly to the data-specific Mercer map [Eq. (5.6)], the general Mercer kernel map is given as

$$\phi(\mathbf{x}_i) = \left(\sqrt{\lambda_1}\, \mathbf{u}_1(\mathbf{x}_i), \sqrt{\lambda_2}\, \mathbf{u}_2(\mathbf{x}_i), \ldots\right)^T$$

with the kernel value being equivalent to the dot product between two mapped points:

$$K(\mathbf{x}_i,\mathbf{x}_j) = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$$

5.2 VECTOR KERNELS

We now consider two of the most commonly used vector kernels in practice. Kernels that map an (input) vector space into another (feature) vector space are called *vector kernels*. For multivariate input data, the input vector space will be the d-dimensional real space $\mathbb{R}^d$. Let $\mathbf{D}$ comprise n input points $\mathbf{x}_i \in \mathbb{R}^d$, for $i = 1, 2, \ldots, n$. Commonly used (nonlinear) kernel functions over vector data include the polynomial and Gaussian kernels, as described next.

Polynomial Kernel

Polynomial kernels are of two types: homogeneous or inhomogeneous. Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$. The *homogeneous polynomial kernel* is defined as

$$K_q(\mathbf{x},\mathbf{y}) = \phi(\mathbf{x})^T \phi(\mathbf{y}) = (\mathbf{x}^T \mathbf{y})^q \tag{5.8}$$

where q is the degree of the polynomial. This kernel corresponds to a feature space spanned by all products of exactly q attributes.

The most typical cases are the *linear* (with $q = 1$) and *quadratic* (with $q = 2$) kernels, given as

$$K_1(\mathbf{x},\mathbf{y}) = \mathbf{x}^T \mathbf{y}$$
$$K_2(\mathbf{x},\mathbf{y}) = (\mathbf{x}^T \mathbf{y})^2$$

The *inhomogeneous polynomial kernel* is defined as

$$K_q(\mathbf{x},\mathbf{y}) = \phi(\mathbf{x})^T \phi(\mathbf{y}) = (c + \mathbf{x}^T \mathbf{y})^q \tag{5.9}$$

where q is the degree of the polynomial, and $c \geq 0$ is some constant. When $c = 0$ we obtain the homogeneous kernel. When $c > 0$, this kernel corresponds to the feature space spanned by all products of at most q attributes. This can be seen from the binomial expansion

$$K_q(\mathbf{x},\mathbf{y}) = (c + \mathbf{x}^T \mathbf{y})^q = \sum_{k=1}^{q} \binom{q}{k} c^{q-k} \left(\mathbf{x}^T \mathbf{y}\right)^k$$

For example, for the typical value of $c = 1$, the inhomogeneous kernel is a weighted sum of the homogeneous polynomial kernels for all powers up to q, that is,

$$(1+\mathbf{x}^T\mathbf{y})^q = 1 + q\mathbf{x}^T\mathbf{y} + \binom{q}{2}\left(\mathbf{x}^T\mathbf{y}\right)^2 + \cdots + q\left(\mathbf{x}^T\mathbf{y}\right)^{q-1} + \left(\mathbf{x}^T\mathbf{y}\right)^q$$

Example 5.6. Consider the points $\mathbf{x}_1$ and $\mathbf{x}_2$ in Figure 5.1.

$$\mathbf{x}_1 = \begin{pmatrix} 5.9 \\ 3 \end{pmatrix} \qquad \mathbf{x}_2 = \begin{pmatrix} 6.9 \\ 3.1 \end{pmatrix}$$

The homogeneous quadratic kernel is given as

$$K(\mathbf{x}_1, \mathbf{x}_2) = (\mathbf{x}_1^T\mathbf{x}_2)^2 = 50.01^2 = 2501$$

The inhomogeneous quadratic kernel is given as

$$K(\mathbf{x}_1, \mathbf{x}_2) = (1 + \mathbf{x}_1^T\mathbf{x}_2)^2 = (1 + 50.01)^2 = 51.01^2 = 2602.02$$

For the polynomial kernel it is possible to construct a mapping ϕ from the input to the feature space. Let $n_0, n_1, \ldots, n_d$ denote non-negative integers, such that $\sum_{i=0}^d n_i = q$. Further, let $\mathbf{n} = (n_0, n_1, \ldots, n_d)$, and let $|\mathbf{n}| = \sum_{i=0}^d n_i = q$. Also, let $\binom{q}{\mathbf{n}}$ denote the multinomial coefficient

$$\binom{q}{\mathbf{n}} = \binom{q}{n_0, n_1, \ldots, n_d} = \frac{q!}{n_0! n_1! \ldots n_d!}$$

The multinomial expansion of the inhomogeneous kernel is then given as

$$\begin{aligned}
K_q(\mathbf{x}, \mathbf{y}) = (c + \mathbf{x}^T\mathbf{y})^q &= \left(c + \sum_{k=1}^d x_k y_k\right)^q = (c + x_1 y_1 + \cdots + x_d y_d)^q \\
&= \sum_{|\mathbf{n}|=q} \binom{q}{\mathbf{n}} c^{n_0} (x_1 y_1)^{n_1} (x_2 y_2)^{n_2} \ldots (x_d y_d)^{n_d} \\
&= \sum_{|\mathbf{n}|=q} \binom{q}{\mathbf{n}} c^{n_0} \left(x_1^{n_1} x_2^{n_2} \ldots x_d^{n_d}\right)\left(y_1^{n_1} y_2^{n_2} \ldots y_d^{n_d}\right) \\
&= \sum_{|\mathbf{n}|=q} \left(\sqrt{a_\mathbf{n}} \prod_{k=1}^d x_k^{n_k}\right)\left(\sqrt{a_\mathbf{n}} \prod_{k=1}^d y_k^{n_k}\right) \\
&= \phi(\mathbf{x})^T \phi(\mathbf{y})
\end{aligned}$$

where $a_\mathbf{n} = \binom{q}{\mathbf{n}} c^{n_0}$, and the summation is over all $\mathbf{n} = (n_0, n_1, \ldots, n_d)$ such that $|\mathbf{n}| = n_0 + n_1 + \cdots + n_d = q$. Using the notation $\mathbf{x}^\mathbf{n} = \prod_{k=1}^d x_k^{n_k}$, the mapping $\phi : \mathbb{R}^d \to \mathbb{R}^m$ is given as the vector

$$\phi(\mathbf{x}) = (\ldots, a_\mathbf{n}\mathbf{x}^\mathbf{n}, \ldots)^T = \left(\ldots, \sqrt{\binom{q}{\mathbf{n}} c^{n_0}} \prod_{k=1}^d x_k^{n_k}, \ldots\right)^T$$

where the variable $\mathbf{n} = (n_0, \ldots, n_d)$ ranges over all the possible assignments, such that $|\mathbf{n}| = q$. It can be shown that the dimensionality of the feature space is given as

$$m = \binom{d+q}{q}$$

Example 5.7 (Quadratic Polynomial Kernel). Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^2$ and let $c = 1$. The inhomogeneous quadratic polynomial kernel is given as

$$K(\mathbf{x}, \mathbf{y}) = (1 + \mathbf{x}^T\mathbf{y})^2 = (1 + x_1 y_1 + x_2 y_2)^2$$

The set of all assignments $\mathbf{n} = (n_0, n_1, n_2)$, such that $|\mathbf{n}| = q = 2$, and the corresponding terms in the multinomial expansion are shown below.

Assignments $\mathbf{n} = (n_0, n_1, n_2)$	Coefficient $a_\mathbf{n} = \binom{q}{\mathbf{n}} c^{n_0}$	Variables $\mathbf{x}^\mathbf{n}\mathbf{y}^\mathbf{n} = \prod_{k=1}^d (x_i y_i)^{n_i}$
$(1, 1, 0)$	2	$x_1 y_1$
$(1, 0, 1)$	2	$x_2 y_2$
$(0, 1, 1)$	2	$x_1 y_1 x_2 y_2$
$(2, 0, 0)$	1	1
$(0, 2, 0)$	1	$(x_1 y_1)^2$
$(0, 0, 2)$	1	$(x_2 y_2)^2$

Thus, the kernel can be written as

$$\begin{aligned}
K(\mathbf{x}, \mathbf{y}) &= 1 + 2x_1y_1 + 2x_2y_2 + 2x_1y_1x_2y_2 + x_1^2y_1^2 + x_2^2y_2^2 \\
&= \left(1, \sqrt{2}x_1, \sqrt{2}x_2, \sqrt{2}x_1x_2, x_1^2, x_2^2\right)\left(1, \sqrt{2}y_1, \sqrt{2}y_2, \sqrt{2}y_1y_2, y_1^2, y_2^2\right)^T \\
&= \phi(\mathbf{x})^T\phi(\mathbf{y})
\end{aligned}$$

When the input space is $\mathbb{R}^2$, the dimensionality of the feature space is given as

$$m = \binom{d+q}{q} = \binom{2+2}{2} = \binom{4}{2} = 6$$

In this case the inhomogeneous quadratic kernel with $c = 1$ corresponds to the mapping $\phi : \mathbb{R}^2 \to \mathbb{R}^6$, given as

$$\phi(\mathbf{x}) = \left(1, \sqrt{2}x_1, \sqrt{2}x_2, \sqrt{2}x_1x_2,\ x_1^2,\ x_2^2\right)^T$$

For example, for $\mathbf{x}_1 = (5.9, 3)^T$ and $\mathbf{x}_2 = (6.9, 3.1)^T$, we have

$$\begin{aligned}
\phi(\mathbf{x}_1) &= \left(1, \sqrt{2}\cdot 5.9, \sqrt{2}\cdot 3, \sqrt{2}\cdot 5.9\cdot 3,\ 5.9^2,\ 3^2\right)^T \\
&= \left(1, 8.34, 4.24, 25.03, 34.81, 9\right)^T \\
\phi(\mathbf{x}_2) &= \left(1, \sqrt{2}\cdot 6.9, \sqrt{2}\cdot 3.1, \sqrt{2}\cdot 6.9\cdot 3.1,\ 6.9^2,\ 3.1^2\right)^T \\
&= \left(1, 9.76, 4.38, 30.25, 47.61, 9.61\right)^T
\end{aligned}$$

Thus, the inhomogeneous kernel value is

$$\phi(\mathbf{x}_1)^T\phi(\mathbf{x}_2) = 1 + 81.40 + 18.57 + 757.16 + 1657.30 + 86.49 = 2601.92$$

On the other hand, when the input space is $\mathbb{R}^2$, the homogeneous quadratic kernel corresponds to the mapping $\phi : \mathbb{R}^2 \to \mathbb{R}^3$, defined as

$$\phi(\mathbf{x}) = \left(\sqrt{2}x_1x_2,\ x_1^2,\ x_2^2\right)^T$$

because only the degree 2 terms are considered. For example, for $\mathbf{x}_1$ and $\mathbf{x}_2$, we have

$$\phi(\mathbf{x}_1) = \left(\sqrt{2}\cdot 5.9\cdot 3,\ 5.9^2,\ 3^2\right)^T = (25.03, 34.81, 9)^T$$

$$\phi(\mathbf{x}_2) = \left(\sqrt{2}\cdot 6.9\cdot 3.1,\ 6.9^2,\ 3.1^2\right)^T = (30.25, 47.61, 9.61)^T$$

and thus

$$K(\mathbf{x}_1, \mathbf{x}_2) = \phi(\mathbf{x}_1)^T\phi(\mathbf{x}_2) = 757.16 + 1657.3 + 86.49 = 2500.95$$

These values essentially match those shown in Example 5.6 up to four significant digits.

Gaussian Kernel

The Gaussian kernel, also called the Gaussian radial basis function (RBF) kernel, is defined as

$$K(\mathbf{x}, \mathbf{y}) = \exp\left\{-\frac{\|\mathbf{x} - \mathbf{y}\|^2}{2\sigma^2}\right\} \tag{5.10}$$

where $\sigma > 0$ is the spread parameter that plays the same role as the standard deviation in a normal density function. Note that $K(\mathbf{x}, \mathbf{x}) = 1$, and further that the kernel value is inversely related to the distance between the two points $\mathbf{x}$ and $\mathbf{y}$.

Example 5.8. Consider again the points $\mathbf{x}_1$ and $\mathbf{x}_2$ in Figure 5.1:

$$\mathbf{x}_1 = \begin{pmatrix} 5.9 \\ 3 \end{pmatrix} \qquad \mathbf{x}_2 = \begin{pmatrix} 6.9 \\ 3.1 \end{pmatrix}$$

The squared distance between them is given as

$$\|\mathbf{x}_1 - \mathbf{x}_2\|^2 = \left\|(-1, -0.1)^T\right\|^2 = 1^2 + 0.1^2 = 1.01$$

With $\sigma = 1$, the Gaussian kernel is

$$K(\mathbf{x}_1, \mathbf{x}_2) = \exp\left\{-\frac{1.01^2}{2}\right\} = \exp\{-0.51\} = 0.6$$

It is interesting to note that a feature space for the Gaussian kernel has infinite dimensionality. To see this, note that the exponential function can be written as the