Chapter 5

The Kernel Method

Before we can mine data, it is important to first find a suitable data representation that facilitates data analysis. For example, for complex data like text, sequences, images, and so on, we must typically extract or construct a set of attributes or features, so that we may 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 non-linear relationships among the attributes, then a non-linear mapping ϕ may be used, so that $\phi(\mathbf{x})$ represents a vector in the corresponding high-dimensional space comprising non-linear 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 $P(s) = \frac{n_s}{m}$ for $s \in \Sigma$ and n_s is the number of times symbol 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 $\mathbf{y} = AGCAAGCGAG$, 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, and 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,

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

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 (Non-linear Features): As an example of a non-linear 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 non-linear 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, since the features are non-linear combinations of the original attributes, this allows us to mine non-linear 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 (as we shall see below). Thus, transforming all the input points into feature space can be very expensive, or even impossible. Since the dimensionality is high, we also run into the curse of dimensionality highlighted 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 $n \times n$ pair-wise similarity values. The similarity function, called

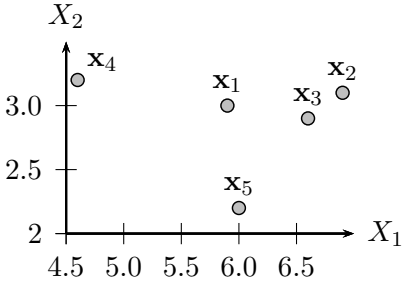
the *kernel*, is chosen so that it represents a dot product in some high-dimensional feature space, yet it can be computed without directly constructing $\phi(\mathbf{x})$. Let $\mathcal{I}$ denote the input space, which can comprise any arbitrary set of objects, and let $\mathbf{D} = \{\mathbf{x}_i\}_{i=1}^n \subset \mathcal{I}$ be a dataset comprising n objects in the input space. We can represent the pair-wise similarity values between point in $\mathbf{D}$ via the $n \times n$ *kernel matrix*, also called the *gram matrix*, defined as

$$\mathbf{K} = \begin{pmatrix} K(\mathbf{x}_1, \mathbf{x}_1) & K(\mathbf{x}_1, \mathbf{x}_2) & \cdots & K(\mathbf{x}_1, \mathbf{x}_n) \\ K(\mathbf{x}_2, \mathbf{x}_1) & K(\mathbf{x}_2, \mathbf{x}_2) & \cdots & K(\mathbf{x}_2, \mathbf{x}_n) \\ \vdots & \vdots & \ddots & \vdots \\ K(\mathbf{x}_n, \mathbf{x}_1) & K(\mathbf{x}_n, \mathbf{x}_2) & \cdots & K(\mathbf{x}_n, \mathbf{x}_n) \end{pmatrix} \tag{5.1}$$

where $K : \mathcal{I} \times \mathcal{I} \to \mathbb{R}$ 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 $\mathbf{x}_i, \mathbf{x}_j \in \mathcal{I}$, we have

$$K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \tag{5.2}$$

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 (5.2) remains valid, as discussed later.



(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

Example 5.3: Linear Kernel: 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 (5.2)

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

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,

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

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

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

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

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

We can rearrange the above to obtain the (homogeneous) quadratic kernel function as follows

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

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*, i.e., 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*, i.e., 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 $\mathbf{x}_i$, since 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, kernels methods allow much more flexibility, since we can just as easily perform non-linear analysis by employing non-linear 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, since 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

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

Since 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 (5.15)) 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 above. 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 the 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 $\mathbf{D} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\} \subset \mathcal{I}$ 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, i.e., $K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$, and let $\mathbf{K}$ be the $n \times n$ kernel matrix corresponding to the subset $\mathbf{D}$.

The function K is called a **positive semi-definite 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 semi-definite, that is,

$$\begin{aligned} & \mathbf{a}^T \mathbf{K} \mathbf{a} \geq 0, \text{ for all vectors } \mathbf{a} \in \mathbb{R}^n \\ \implies & \sum_{i=1}^{n} \sum_{j=1}^{n} a_i a_j K(\mathbf{x}_i, \mathbf{x}_j) \geq 0, \text{ for all } a_i \in \mathbb{R}, i \in [1, n] \end{aligned} \tag{5.3}$$

Other common names for a positive semi-definite kernel include *Mercer kernel* and *reproducing kernel*.

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 semi-definite 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 semi-definite since

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

Thus, K is a positive semi-definite kernel.

We now show that if we are given a positive semi-definite 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 any $\mathbf{x} \in \mathcal{I}$ in the input space, define the mapping $\phi(\mathbf{x})$ as follows

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

where the $\cdot$ stands for any argument in $\mathcal{I}$. What we have done is map each point $\mathbf{x}$ to a function $\phi_{\mathbf{x}} : \mathcal{I} \to \mathbb{R}$, so that for any particular argument $\mathbf{x}_i$, we have

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

In other words, if we evaluate the function $\phi_{\mathbf{x}}$ at the point $\mathbf{x}_i$, then the result is simply the kernel value between $\mathbf{x}$ and $\mathbf{x}_i$. Thus, each object in input space get mapped to a *feature point* which is in fact a function. Such a space is also called *functional space*, since it is a space of functions, but algebraically it is just a vector space where each point happens to be a function.

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

$$\mathcal{F} = \left\{ \mathbf{f} = f(\cdot) = \sum_{i=1}^{m} \alpha_i \, K(\mathbf{x}_i, \cdot) \;\middle|\; m \text{ is any natural number, and } \alpha_i \in \mathbb{R} \right\}$$

In other words, $\mathcal{F}$ includes all possible linear combinations (for any value of m) over feature points. We use the dual notation $\mathbf{f} = f(\cdot)$ to emphasize the fact that each point $\mathbf{f}$ in the feature space is in fact a function $f(\cdot)$. Note that the mapped point $\phi(\mathbf{x}) = \phi_{\mathbf{x}}(\cdot) = K(\mathbf{x}, \cdot)$ is itself a point in $\mathcal{F}$.

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

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

We can verify that the dot product is *bilinear*, i.e., linear in both arguments, since

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

The fact that K is positive semi-definite 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 normed inner product space, i.e., it is endowed with a symmetric bilinear dot product and a norm. In fact the functional space $\mathcal{F}$ is a *Hilbert space*, defined as a normed inner product space that is complete and separable.

The space $\mathcal{F}$ has the so called *reproducing property*, that is the dot product of any function $\mathbf{f} = f(\cdot)$ with any feature point $\phi(\mathbf{x}) = \phi_{\mathbf{x}}(\cdot)$ results in the same function $f(\cdot)$, since

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

Thus, the dot product of any function and any mapped point reproduces the function. 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, since for any two mapped points $\phi(\mathbf{x}_i)$ and $\phi(\mathbf{x}_j)$

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

their dot product in feature space is given as

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

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

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

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 $\mathbf{x}_i \in \mathbf{D}$. The dot product in the feature space is given 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 i-th row of $\mathbf{K}$, which is also the same as the i-th column of $\mathbf{K}$, since $\mathbf{K}$ 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 (5.4). One solution is to replace $\mathbf{K}_i^T\mathbf{K}_j$ in (5.4) with $\mathbf{K}_i^T\mathbf{A}\mathbf{K}_j$ for some positive semi-definite 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

$$\begin{aligned}\left\{\mathbf{K}_i^T\mathbf{A}\mathbf{K}_j\right\}_{i,j=1}^n &= \left\{K(\mathbf{x}_i, \mathbf{x}_j)\right\}_{i,j=1}^n \\ \implies \mathbf{K}\mathbf{A}\mathbf{K} &= \mathbf{K}\end{aligned}$$

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

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

so that the dot product yields

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

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.

5.1.2 Mercer Kernel 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. Since $\mathbf{K}$ is a symmetric positive semi-definite matrix, it has real and non-negative eigenvalues, and it can be decomposed as follows

$$\mathbf{K} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T \tag{5.5}$$

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

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

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

$$\begin{aligned} \mathbf{K}(\mathbf{x}_i, \mathbf{x}_j) &= \lambda_1 \, u_{1i} \, u_{1j} + \lambda_2 \, u_{2i} \, u_{2j} \cdots + \lambda_n \, u_{ni} \, u_{nj} \\ &= \sum_{k=1} \lambda_k \, u_{ki} \, u_{kj} \end{aligned} \tag{5.6}$$

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

$$\phi(\mathbf{x}_i) = (\sqrt{\lambda_1} \, u_{i1}, \sqrt{\lambda_2} \, u_{2i}, \cdots, \sqrt{\lambda_n} \, u_{ni})^T \tag{5.7}$$

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

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

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

$$\phi(\mathbf{x}_i) = \sqrt{\mathbf{\Lambda}} U_i \tag{5.8}$$

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

$$\phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) = \left(\sqrt{\mathbf{\Lambda}} U_i \right)^T \left(\sqrt{\mathbf{\Lambda}} U_j \right) = U_i^T \mathbf{\Lambda} U_j \tag{5.9}$$

The Mercer map, defined equivalently in (5.7) and (5.8), is obviously restricted to the input data set $\mathbf{D}$, 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

$$\mathbf{U} = \begin{array}{c|rr} & \mathbf{u}_1 & \mathbf{u}_2 \\ \hline U_1 & -0.442 & 0.163 \\ U_2 & -0.505 & -0.134 \\ U_3 & -0.482 & -0.181 \\ U_4 & -0.369 & 0.813 \\ U_5 & -0.425 & -0.512 \end{array}$$

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

$$\phi(\mathbf{x}_1) = \sqrt{\mathbf{\Lambda}} U_1 = \begin{pmatrix} 14.965 & 0 \\ 0 & 1.135 \end{pmatrix} \begin{pmatrix} -0.442 \\ 0.163 \end{pmatrix} = \begin{pmatrix} -6.616 \\ 0.185 \end{pmatrix}$$

$$\phi(\mathbf{x}_2) = \sqrt{\mathbf{\Lambda}} U_2 = \begin{pmatrix} 14.965 & 0 \\ 0 & 1.135 \end{pmatrix} \begin{pmatrix} -0.505 \\ -0.134 \end{pmatrix} = \begin{pmatrix} -7.563 \\ -0.153 \end{pmatrix}$$

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 continuous spaces, analogous to the discrete case in (5.6), 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 $\{\lambda_1, \lambda_2, \cdots\}$ is the infinite set of eigenvalues, and $\{\mathbf{u}_1(\cdot), \mathbf{u}_2(\cdot), \cdots\}$ is the corresponding set of orthogonal and normalized *eigenfunctions*, i.e., 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 positive semi-definite kernel, i.e., for all functions $a(\cdot)$ with finite square integrals ($\int a(\mathbf{x})^2 \; d\mathbf{x} < 0$)

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

Contrast this with the discrete kernel in (5.3).

Analogously to the data-specific Mercer map (5.7), the general Mercer kernel map is given as

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

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

Kernels that map an (input) vector space into the 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 $\mathbf{D}$ consist of n input points $\mathbf{x}_i \in \mathbb{R}^d$, for $i = 1, 2, \ldots, n$. Some commonly used (non-linear) kernel functions over vector data include the polynomial and Gaussian kernels.

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

$$K_q(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x})^T \phi(\mathbf{y}) = (\mathbf{x}^T \mathbf{y})^q \tag{5.10}$$

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

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

$$K_q(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x})^T \phi(\mathbf{y}) = (c + \mathbf{x}^T \mathbf{y})^q \tag{5.11}$$

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, i.e.,

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

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 derive the 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 $|\mathbf{n}| = \sum_{i=0}^d n_i$. 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

$$\begin{aligned}
K_q(\mathbf{x}, \mathbf{y}) = (c + \mathbf{x}^T\mathbf{y})^q &= \left(c + \sum_{k=1}^d x_k y_k\right)^q = (c + x_1y_1 + \cdots + x_dy_d)^q \\
&= \sum_{|\mathbf{n}|=q} \binom{q}{\mathbf{n}} c^{n_0} \left(x_1y_1\right)^{n_1} \left(x_2y_2\right)^{n_2} \ldots \left(x_dy_d\right)^{n_d} \\
&= \sum_{|\mathbf{n}|=q} \binom{q}{\mathbf{n}} c^{n_0} \left(x_1^{n_1}x_2^{n_2}\ldots x_d^{n_d}\right)\left(y_1^{n_1}y_2^{n_2}\ldots y_d^{n_d}\right) \\
&= \sum_{|\mathbf{n}|=q} \left(\sqrt{a_{\mathbf{n}}} \prod_{k=1}^d x_k^{n_k}\right)\left(\sqrt{a_{\mathbf{n}}} \prod_{k=1}^d y_k^{n_k}\right) \\
&= \phi(\mathbf{x})^T\phi(\mathbf{y})
\end{aligned}$$

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 $\mathbf{x}, \mathbf{y} \in \mathbb{R}^2$. Let $c = 1$. The inhomogeneous quadratic polynomial kernel is given as

$$K(\mathbf{x}, \mathbf{y}) = (1 + \mathbf{x}^T\mathbf{y})^2 = (1 + x_1y_1 + x_2y_2)^2$$

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 $a_{\mathbf{n}} = \binom{q}{n_0, n_1, n_2} c^{n_0}$	variables $\mathbf{x}^{\mathbf{n}}\mathbf{y}^{\mathbf{n}} = \prod_{k=1}^{d} (x_i y_i)^{n_i}$
$(1, 1, 0)$	2	x_1y_1
$(1, 0, 1)$	2	x_2y_2
$(0, 1, 1)$	2	$x_1y_1x_2y_2$
$(2, 0, 0)$	1	1
$(0, 2, 0)$	1	$(x_1y_1)^2$
$(0, 0, 2)$	1	$(x_2y_2)^2$

Thus, the kernel can be written as

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

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

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

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

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

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

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

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

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

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 *Radial Basis Kernel*, is defined as

$$K(\mathbf{x}, \mathbf{y}) = \exp\left\{-\frac{\|\mathbf{x} - \mathbf{y}\|^2}{2\sigma^2}\right\} \tag{5.12}$$

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 proportional 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 the feature space for a Gaussian kernel has infinite dimensionality. To see this, note that the exponential function can be written as the infinite expansion

$$\exp\{a\} = \sum_{n=0}^{\infty} \frac{a^n}{n!} = 1 + a + \frac{1}{2!}a^2 + \frac{1}{3!}a^3 + \ldots$$

Further, using $\gamma = \frac{1}{2\sigma^2}$, and noting that $\|\mathbf{x} - \mathbf{y}\|^2 = \|\mathbf{x}\|^2 + \|\mathbf{y}\|^2 - 2\mathbf{x}^T\mathbf{y}$, we can rewrite the Gaussian kernel as follows

$$\begin{aligned} K(\mathbf{x}, \mathbf{y}) &= \exp\left\{-\gamma \|\mathbf{x} - \mathbf{y}\|^2\right\} \\ &= \exp\left\{-\gamma \|\mathbf{x}\|^2\right\} \cdot \exp\left\{-\gamma \|\mathbf{y}\|^2\right\} \cdot \exp\left\{2\gamma \mathbf{x}^T\mathbf{y}\right\} \end{aligned}$$

In particular, the last term is given as the infinite expansion

$$\exp\left\{2\gamma \mathbf{x}^T\mathbf{y}\right\} = \sum_{q=0}^{\infty} \frac{(2\gamma)^q}{q!} \left(\mathbf{x}^T\mathbf{y}\right)^q = 1 + (2\gamma)\mathbf{x}^T\mathbf{y} + \frac{(2\gamma)^2}{2!} \left(\mathbf{x}^T\mathbf{y}\right)^2 + \ldots$$

Using the multinomial expansion of $(\mathbf{x}^T\mathbf{y})^q$, we can write the Gaussian kernel as

$$\begin{aligned}
&K(\mathbf{x},\mathbf{y}) \\
&\quad = \exp\left\{-\gamma\left\|\mathbf{x}\right\|^2\right\}\exp\left\{-\gamma\left\|\mathbf{y}\right\|^2\right\}\sum_{q=0}^{\infty}\frac{(2\gamma)^q}{q!}\left(\sum_{|\mathbf{n}|=q}\binom{q}{\mathbf{n}}\prod_{k=1}^{d}(x_k y_k)^{n_k}\right) \\
&\quad = \sum_{q=0}^{\infty}\sum_{|\mathbf{n}|=q}\left(\sqrt{a_{q,\mathbf{n}}}\exp\left\{-\gamma\left\|\mathbf{x}\right\|^2\right\}\prod_{k=1}^{d}x_k^{n_k}\right)\left(\sqrt{a_{q,\mathbf{n}}}\exp\left\{-\gamma\left\|\mathbf{y}\right\|^2\right\}\prod_{k=1}^{d}y_k^{n_k}\right) \\
&\quad = \phi(\mathbf{x})^T\phi(\mathbf{y})
\end{aligned}$$

where $a_{q,\mathbf{n}} = \frac{(2\gamma)^q}{q!}\binom{q}{\mathbf{n}}$, and $\mathbf{n} = (n_1, n_2, \ldots, n_d)$, with $|\mathbf{n}| = n_1 + n_2 + \cdots + n_d = q$. The mapping into feature space corresponds to the function $\phi : \mathbb{R}^d \to \mathbb{R}^\infty$

$$\phi(\mathbf{x}) = \left(\ldots,\ \sqrt{\frac{(2\gamma)^q}{q!}\binom{q}{\mathbf{n}}}\exp\left\{-\gamma\left\|\mathbf{x}\right\|^2\right\}\prod_{k=1}^{d}x_k^{n_k},\ \ldots\right)^T$$

with the dimensions ranging over all degrees $q = 0, \ldots, \infty$, and with the variable $\mathbf{n} = (n_1, \ldots, n_d)$ ranging over all possible assignments such that $|\mathbf{n}| = q$ for each value of q. Since ϕ maps the input space into an infinite dimensional feature space, we obviously cannot compute $\phi(\mathbf{x})$, yet computing the Gaussian kernel $K(\mathbf{x},\mathbf{y})$ is straightforward.

5.3 Basic Kernel Operations in Feature Space

Let us look at some of the basic data analysis tasks that can be performed solely via kernels, without instantiating $\phi(\mathbf{x})$.

Norm of a Point: We can compute the norm of a point $\phi(\mathbf{x})$ in feature space as follows

$$\left\|\phi(\mathbf{x})\right\|^2 = \phi(\mathbf{x})^T\phi(\mathbf{x}) = K(\mathbf{x},\mathbf{x})$$

which implies that $\left\|\phi(\mathbf{x})\right\| = \sqrt{K(\mathbf{x},\mathbf{x})}$.

Distance between Points: The distance between two points $\phi(\mathbf{x}_i)$ and $\phi(\mathbf{x}_j)$ can be computed as follows

$$\begin{aligned}
\left\|\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)\right\|^2 &= \left\|\phi(\mathbf{x}_i)\right\|^2 + \left\|\phi(\mathbf{x}_j)\right\|^2 - 2\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j) \\
&= K(\mathbf{x}_i,\mathbf{x}_i) + K(\mathbf{x}_j,\mathbf{x}_j) - 2K(\mathbf{x}_i,\mathbf{x}_j)
\end{aligned} \tag{5.13}$$

which implies that

$$\delta\big(\phi(\mathbf{x}_i), \phi(\mathbf{x}_j)\big) = \|\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)\| = \sqrt{K(\mathbf{x}_i, \mathbf{x}_i) + K(\mathbf{x}_j, \mathbf{x}_j) - 2K(\mathbf{x}_i, \mathbf{x}_j)}$$

Rearranging (5.13), we can see that the kernel value can be considered as a measure of the similarity between two points, since

$$K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) = \frac{1}{2}\left(\|\phi(\mathbf{x}_i)\|^2 + \|\phi(\mathbf{x}_j)\|^2 - \|\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)\|^2\right) \quad (5.14)$$

Thus, the more the distance $\|\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)\|$ between the two points in feature space, the less the kernel value, i.e., the less the similarity.

Example 5.9: Consider the two points $\mathbf{x}_1$ and $\mathbf{x}_2$

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

Assuming the homogeneous quadratic kernel, the norm of $\phi(\mathbf{x}_1)$ can be computed as

$$\|\phi(\mathbf{x}_1)\|^2 = K(\mathbf{x}_1, \mathbf{x}_1) = (\mathbf{x}_1^T \mathbf{x}_1)^2 = 43.81^2 = 1919.32$$

which implies $\|\phi(\mathbf{x}_1)\| = \sqrt{43.81^2} = 43.81$.

The distance between $\phi(\mathbf{x}_1)$ and $\phi(\mathbf{x}_2)$ is

$$\begin{aligned} \delta\big(\phi(\mathbf{x}_1), \phi(\mathbf{x}_2)\big) &= \sqrt{K(\mathbf{x}_1, \mathbf{x}_1) + K(\mathbf{x}_2, \mathbf{x}_2) - 2K(\mathbf{x}_1, \mathbf{x}_2)} \\ &= \sqrt{1919.32 + 3274.13 - 2 \cdot 2501} = \sqrt{191.45} = 13.84 \end{aligned}$$

Mean in Feature Space: The mean of the points in feature space is given as

$$\boldsymbol{\mu}_\phi = \frac{1}{n}\sum_{i=1}^{n} \phi(\mathbf{x}_i)$$

Since we do not, in general, have access to $\phi(\mathbf{x})$, we cannot explicitly compute the mean point in feature space.

Nevertheless we can compute the norm of the mean as follows

$$\begin{aligned}\|\boldsymbol{\mu}_\phi\|^2 &= (\boldsymbol{\mu}_\phi)^T \boldsymbol{\mu}_\phi \\ &= \left(\frac{1}{n}\sum_{i=1}^{n}\phi(\mathbf{x}_i)\right)^T \left(\frac{1}{n}\sum_{j=1}^{n}\phi(\mathbf{x}_j)\right) \\ &= \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j) \\ &= \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}K(\mathbf{x}_i,\mathbf{x}_j) \end{aligned} \tag{5.15}$$

The above derivation implies that the squared norm of the mean in feature space is simply the average of the values in the kernel matrix $\mathbf{K}$.

Example 5.10: Consider the five points from Example 5.3, also shown in Figure 5.1. Example 5.4 showed the norm of mean for the linear kernel. Let us consider the Guassian kernel with $\sigma = 1$. The Gaussian kernel matrix is given as

$$\mathbf{K} = \begin{pmatrix} 1.00 & 0.60 & 0.78 & 0.42 & 0.72 \\ 0.60 & 1.00 & 0.94 & 0.07 & 0.44 \\ 0.78 & 0.94 & 1.00 & 0.13 & 0.65 \\ 0.42 & 0.07 & 0.13 & 1.00 & 0.23 \\ 0.72 & 0.44 & 0.65 & 0.23 & 1.00 \end{pmatrix}$$

The squared norm of the mean in feature space is therefore

$$\|\boldsymbol{\mu}_\phi\|^2 = \frac{1}{25}\sum_{i=1}^{5}\sum_{j=1}^{5}K(\mathbf{x}_i,\mathbf{x}_j) = \frac{14.98}{25} = 0.599$$

which implies that $\|\boldsymbol{\mu}_\phi\| = \sqrt{0.599} = 0.774$.

Total Variance in Feature Space: Let us first derive the formula for the distance of a point to the mean in feature space

$$\begin{aligned}\|\phi(\mathbf{x}_i) - \boldsymbol{\mu}_\phi\|^2 &= \|\phi(\mathbf{x}_i)\|^2 - 2\phi(\mathbf{x}_i)^T\boldsymbol{\mu}_\phi + \|\boldsymbol{\mu}_\phi\|^2 \\ &= K(\mathbf{x}_i,\mathbf{x}_i) - \frac{2}{n}\sum_{j=1}^{n}K(\mathbf{x}_i,\mathbf{x}_j) + \frac{1}{n^2}\sum_{a=1}^{n}\sum_{b=1}^{n}K(\mathbf{x}_a,\mathbf{x}_b)\end{aligned}$$

The total variance (1.9) in feature space is obtained by taking the average deviation of points from the mean in feature space

$$\begin{aligned}
\sigma_\phi^2 &= \frac{1}{n}\sum_{i=1}^{n}\|\phi(\mathbf{x}_i) - \boldsymbol{\mu}_\phi\|^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}\left(K(\mathbf{x}_i,\mathbf{x}_i) - \frac{2}{n}\sum_{j=1}^{n}K(\mathbf{x}_i,\mathbf{x}_j) + \frac{1}{n^2}\sum_{a=1}^{n}\sum_{b=1}^{n}K(\mathbf{x}_a,\mathbf{x}_b)\right) \\
&= \frac{1}{n}\sum_{i=1}^{n}K(\mathbf{x}_i,\mathbf{x}_i) - \frac{2}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}K(\mathbf{x}_i,\mathbf{x}_j) + \frac{n}{n^3}\sum_{a=1}^{n}\sum_{b=1}^{n}K(\mathbf{x}_a,\mathbf{x}_b) \\
&= \frac{1}{n}\sum_{i=1}^{n}K(\mathbf{x}_i,\mathbf{x}_i) - \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}K(\mathbf{x}_i,\mathbf{x}_j) \qquad (5.16)
\end{aligned}$$

In other words, the total variance in feature space is given as the difference between the average of the diagonal entries and the average of the entire kernel matrix $\mathbf{K}$.

Example 5.11: Continuing Example 5.10, the total variance in feature space for the five points for the Gaussian kernel is given as

$$\sigma_\phi^2 = \left(\frac{1}{n}\sum_{i=1}^{n}K(\mathbf{x}_i,\mathbf{x}_i)\right) - \|\boldsymbol{\mu}_\phi\|^2 = \frac{1}{5}\times 5 - 0.599 = 0.401$$

The distance between $\phi(\mathbf{x}_1)$ from the mean $\boldsymbol{\mu}_\phi$ in feature space is given as

$$\begin{aligned}
\|\phi(\mathbf{x}_1) - \boldsymbol{\mu}_\phi\|^2 &= K(\mathbf{x}_1,\mathbf{x}_1) - \frac{2}{5}\sum_{j=1}^{5}K(\mathbf{x}_1,\mathbf{x}_j) + \|\boldsymbol{\mu}_\phi\|^2 \\
&= 1 - \frac{2}{5}(1 + 0.6 + 0.78 + 0.42 + 0.72) + 0.599 \\
&= 1 - 1.410 + 0.599 = 0.189
\end{aligned}$$

Centering in Feature Space: We can center each point in feature space by subtracting the mean from it, as follows

$$\hat{\phi}(\mathbf{x}_i) = \phi(\mathbf{x}_i) - \boldsymbol{\mu}_\phi$$

Since we do not have explicit representation of $\phi(\mathbf{x}_i)$ or $\boldsymbol{\mu}_\phi$, we cannot explicitly center the points. However, we can still compute the *centered kernel matrix*, i.e., the kernel matrix over centered points.

The centered kernel matrix is then given as

$$\hat{\mathbf{K}} = \left\{\hat{K}(\mathbf{x}_i, \mathbf{x}_j)\right\}_{i,j=1}^{n}$$

where each cell corresponds to the kernel between centered points, given as

$$\begin{aligned}
\hat{K}(\mathbf{x}_i, \mathbf{x}_j) &= \hat{\phi}(\mathbf{x}_i)^T \hat{\phi}(\mathbf{x}_j) \\
&= (\phi(\mathbf{x}_i) - \boldsymbol{\mu}_\phi)^T (\phi(\mathbf{x}_j) - \boldsymbol{\mu}_\phi) \\
&= \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) - \phi(\mathbf{x}_i)^T \boldsymbol{\mu}_\phi - \phi(\mathbf{x}_j)^T \boldsymbol{\mu}_\phi + (\boldsymbol{\mu}_\phi)^T \boldsymbol{\mu}_\phi \\
&= K(\mathbf{x}_i, \mathbf{x}_j) - \frac{1}{n}\sum_{k=1}^{n} \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_k) - \frac{1}{n}\sum_{k=1}^{n} \phi(\mathbf{x}_j)^T \phi(\mathbf{x}_k) + \|\boldsymbol{\mu}_\phi\|^2 \\
&= K(\mathbf{x}_i, \mathbf{x}_j) - \frac{1}{n}\sum_{k=1}^{n} K(\mathbf{x}_i, \mathbf{x}_k) - \frac{1}{n}\sum_{k=1}^{n} K(\mathbf{x}_j, \mathbf{x}_k) + \frac{1}{n^2}\sum_{a=1}^{n}\sum_{b=1}^{n} K(\mathbf{x}_a, \mathbf{x}_b)
\end{aligned}$$

In other words, we can compute the centered kernel matrix using only the kernel function. Over all the pairs of points, we can write all n^2 entries in compact matrix notation as follows

$$\begin{aligned}
\hat{\mathbf{K}} &= \mathbf{K} - \frac{1}{n}\mathbf{1}_{n\times n}\mathbf{K} - \frac{1}{n}\mathbf{K}\mathbf{1}_{n\times n} + \frac{1}{n^2}\mathbf{1}_{n\times n}\mathbf{K}\mathbf{1}_{n\times n} \\
&= \left(\mathbf{I} - \frac{1}{n}\mathbf{1}_{n\times n}\right)\mathbf{K}\left(\mathbf{I} - \frac{1}{n}\mathbf{1}_{n\times n}\right) \qquad (5.17)
\end{aligned}$$

where $\mathbf{1}_{n\times n}$ is the $n \times n$ singular matrix, all of whose entries equal one.

Example 5.12: Consider the first five points from the two-dimensional Iris dataset shown in Figure 5.1a

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

Consider the linear kernel matrix shown in Figure 5.1b. We can center it by first computing

$$\mathbf{I} - \frac{1}{5}\mathbf{1}_{5\times 5} = \begin{pmatrix} 0.8 & -0.2 & -0.2 & -0.2 & -0.2 \\ -0.2 & 0.8 & -0.2 & -0.2 & -0.2 \\ -0.2 & -0.2 & 0.8 & -0.2 & -0.2 \\ -0.2 & -0.2 & -0.2 & 0.8 & -0.2 \\ -0.2 & -0.2 & -0.2 & -0.2 & 0.8 \end{pmatrix}$$

The centered kernel matrix (5.17) is given as

$$\hat{\mathbf{K}} = \left(\mathbf{I} - \frac{1}{5}\mathbf{1}_{5\times 5}\right) \cdot \begin{pmatrix} 43.81 & 50.01 & 47.64 & 36.74 & 42.00 \\ 50.01 & 57.22 & 54.53 & 41.66 & 48.22 \\ 47.64 & 54.53 & 51.97 & 39.64 & 45.98 \\ 36.74 & 41.66 & 39.64 & 31.40 & 34.64 \\ 42.00 & 48.22 & 45.98 & 34.64 & 40.84 \end{pmatrix} \cdot \left(\mathbf{I} - \frac{1}{5}\mathbf{1}_{5\times 5}\right)$$

$$= \begin{pmatrix} 0.02 & -0.06 & -0.06 & 0.18 & -0.08 \\ -0.06 & 0.86 & 0.54 & -1.19 & -0.15 \\ -0.06 & 0.54 & 0.36 & -0.83 & -0.01 \\ 0.18 & -1.19 & -0.83 & 2.06 & -0.22 \\ -0.08 & -0.15 & -0.01 & -0.22 & 0.46 \end{pmatrix}$$

To verify that $\hat{\mathbf{K}}$ is the same as the kernel matrix for the centered points, let us first center the points by subtracting the mean $\boldsymbol{\mu} = (6.0, 2.88)^T$. The centered points in feature space are given as

$$\mathbf{z}_1 = \begin{pmatrix} -0.1 \\ 0.12 \end{pmatrix} \quad \mathbf{z}_2 = \begin{pmatrix} 0.9 \\ 0.22 \end{pmatrix} \quad \mathbf{z}_3 = \begin{pmatrix} 0.6 \\ 0.02 \end{pmatrix} \quad \mathbf{z}_4 = \begin{pmatrix} -1.4 \\ 0.32 \end{pmatrix} \quad \mathbf{z}_5 = \begin{pmatrix} 0.0 \\ -0.68 \end{pmatrix}$$

For example, the kernel between $\phi(\mathbf{z}_1)$ and $\phi(\mathbf{z}_2)$ is

$$\phi(\mathbf{z}_1)^T\phi(\mathbf{z}_2) = \mathbf{z}_1^T\mathbf{z}_2 = -0.09 + 0.03 = -0.06$$

which matches $\hat{\mathbf{K}}(\mathbf{x}_1, \mathbf{x}_2)$ as expected. The other entries can be verified in a similar manner. Thus, the kernel matrix obtained by centering the data and then computing the kernel is the same as that obtained via (5.17).

Normalizing in Feature Space: A common form of normalization is to ensure that points in feature space have unit length by replacing $\phi(\mathbf{x}_i)$ with the corresponding unit vector $\phi_n(\mathbf{x}_i) = \frac{\phi(\mathbf{x}_i)}{\|\phi(\mathbf{x}_i)\|}$. The dot product in feature space then corresponds to the cosine of the angle between the two mapped points, since

$$\phi_n(\mathbf{x}_i)^T\phi_n(\mathbf{x}_j) = \frac{\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)}{\|\phi(\mathbf{x}_i)\| \cdot \|\phi(\mathbf{x}_j)\|} = \cos_\phi(\theta)$$

If the mapped points are both centered and normalized, then a dot product corresponds to the correlation between the two points in feature space.

The normalized kernel matrix, $\mathbf{K}_n$, can be computed using only the kernel func-

tion K, since

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

$\mathbf{K}_n$ has all diagonal elements as one.

Let $\mathbf{W}$ denote the diagonal matrix comprising the diagonal elements of $\mathbf{K}$

$$\mathbf{W} = \text{diag}(\mathbf{K}) = \begin{pmatrix} K(\mathbf{x}_1, \mathbf{x}_1) & 0 & \cdots & 0 \\ 0 & K(\mathbf{x}_2, \mathbf{x}_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & K(\mathbf{x}_n, \mathbf{x}_n) \end{pmatrix}$$

The normalized kernel matrix can then be expressed compactly as

$$\mathbf{K}_n = \mathbf{W}^{-1/2} \cdot \mathbf{K} \cdot \mathbf{W}^{-1/2} \tag{5.18}$$

where $\mathbf{W}^{-1/2}$ is the diagonal matrix, defined as $\mathbf{W}^{-1/2}(\mathbf{x}_i, \mathbf{x}_i) = \frac{1}{\sqrt{K(\mathbf{x}_i, \mathbf{x}_i)}}$, with all other elements being zero.

Example 5.13: Consider the five points and the linear kernel matrix shown in Figure 5.1. We have

$$\mathbf{W} = \begin{pmatrix} 43.81 & 0 & 0 & 0 & 0 \\ 0 & 57.22 & 0 & 0 & 0 \\ 0 & 0 & 51.97 & 0 & 0 \\ 0 & 0 & 0 & 31.40 & 0 \\ 0 & 0 & 0 & 0 & 40.84 \end{pmatrix}$$

And the normalized kernel is given as

$$\mathbf{K}_n = \mathbf{W}^{-1/2} \cdot \mathbf{K} \cdot \mathbf{W}^{-1/2} = \begin{pmatrix} 1.0000 & 0.9988 & 0.9984 & 0.9906 & 0.9929 \\ 0.9988 & 1.0000 & 0.9999 & 0.9828 & 0.9975 \\ 0.9984 & 0.9999 & 1.0000 & 0.9812 & 0.9980 \\ 0.9906 & 0.9828 & 0.9812 & 1.0000 & 0.9673 \\ 0.9929 & 0.9975 & 0.9980 & 0.9673 & 1.0000 \end{pmatrix}$$

The same kernel would have been obtained, if we had first normalized feature vectors to have unit length, and then taken the dot products. For example, with the linear kernel, the normalized point $\phi_n(\mathbf{x}_1)$ is given as

$$\phi_n(\mathbf{x}_1) = \frac{\phi(\mathbf{x}_1)}{\|\phi(\mathbf{x}_1)\|} = \frac{\mathbf{x}_1}{\|\mathbf{x}_1\|} = \frac{1}{\sqrt{43.81}} \begin{pmatrix} 5.9 \\ 3 \end{pmatrix} = \begin{pmatrix} 0.8914 \\ 0.4532 \end{pmatrix}$$

Likewise, we have $\phi_n(\mathbf{x}_2) = \frac{1}{\sqrt{57.22}}\begin{pmatrix}6.9\\3.1\end{pmatrix} = \begin{pmatrix}0.9122\\0.4098\end{pmatrix}$. Their dot product is

$$\phi_n(\mathbf{x}_1)^T\phi_n(\mathbf{x}_2) = 0.8914 \cdot 0.9122 + 0.4532 \cdot 0.4098 = 0.9988$$

which matches $\mathbf{K}_n(\mathbf{x}_1, \mathbf{x}_2)$.

If we start with the centered kernel matrix $\hat{\mathbf{K}}$ from Example 5.12, and then normalize it, we obtain the normalized and centered kernel matrix $\hat{\mathbf{K}}_n$

$$\hat{\mathbf{K}}_n = \begin{pmatrix} 1.00 & -0.44 & -0.61 & 0.80 & -0.77 \\ -0.44 & 1.00 & 0.98 & -0.89 & -0.24 \\ -0.61 & 0.98 & 1.00 & -0.97 & -0.03 \\ 0.80 & -0.89 & -0.97 & 1.00 & -0.22 \\ -0.77 & -0.24 & -0.03 & -0.22 & 1.00 \end{pmatrix}$$

As noted earlier, the kernel value $\hat{\mathbf{K}}_n(\mathbf{x}_i, \mathbf{x}_j)$ denotes the correlation between $\mathbf{x}_i$ and $\mathbf{x}_j$ in feature space, i.e., it is cosine of the angle between the centered points $\phi(\mathbf{x}_i)$ and $\phi(\mathbf{x}_j)$.

5.4 Kernels for Complex Objects

We conclude this chapter with some examples of kernels defined for complex data like strings and graphs. The use of kernels for dimensionality reduction, classification, and clustering will be considered in later chapters.

5.4.1 Spectrum Kernel for Strings

Consider text or sequence data defined over an alphabet Σ. The l-spectrum feature map is the mapping $\phi : \Sigma^* \to \mathbb{R}^{|\Sigma|^l}$ from the set of substrings over Σ to the $|\Sigma|^l$-dimensional space representing the number of occurrences of all possible substrings of length l, defined as

$$\phi(\mathbf{x}) = \Big(\cdots, \#(\alpha), \cdots\Big)^T_{\alpha \in \Sigma^l}$$

where $\#(\alpha)$ is the number of occurrences of the l-length string α in $\mathbf{x}$.

The (full) spectrum map is an extension of the l-spectrum map, obtained by considering all lengths from $l = 0$ to $l = \infty$, leading to an infinite dimensional feature map $\phi : \Sigma^* \to \mathbb{R}^\infty$

$$\phi(\mathbf{x}) = \Big(\cdots, \#(\alpha), \cdots\Big)^T_{\alpha \in \Sigma^*}$$

where $\#(\alpha)$ is the number of occurrences of the string α in $\mathbf{x}$.

The (l-)spectrum kernel between two strings $\mathbf{x}_i, \mathbf{x}_j \in \Sigma^*$ is simply the dot product between their (l-)spectrum maps

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

A naive computation of the l-spectrum kernel takes $O(|\Sigma|^l)$ time. However, for a given string $\mathbf{x}$ of length n, the vast majority of the l-length strings have an occurrence count of zero, which can be ignored. The l-spectrum map can be effectively computed in $O(n)$ time for a string of length n (assuming $n \gg l$), since there can be at most $n - l + 1$ substrings of length l, and the l-spectrum kernel can thus be computed in $O(n + m)$ time for any two strings of length n and m, respectively.

The feature map for the (full) spectrum kernel is infinite dimensional, but once again, for a given string $\mathbf{x}$ of length n, the vast majority of the strings will have an occurrence count of zero. A straightforward implementation of the spectrum map for a string $\mathbf{x}$ of length n can be computed in $O(n^2)$ time, since $\mathbf{x}$ can have at most $\sum_{l=1}^{n} n - l + 1 = n(n+1)/2$ distinct non-empty substrings. The spectrum kernel can then be computed in $O(n^2 + m^2)$ time for any two strings of length n and m, respectively. However, a much more efficient computation is enabled via suffix trees (see Chapter **??**), with a total time of $O(n + m)$.

Example 5.14: Consider sequences over the DNA alphabet $\Sigma = \{A, C, G, T\}$. Let $\mathbf{x}_1 = ACAGCAGTA$, and let $\mathbf{x}_2 = AGCAAGCGAG$. For $l = 3$, the feature space has dimensionality $|\Sigma|^l = 4^3 = 64$. Nevertheless, we do not have to map the input points into the full feature space; we can compute the reduced 3-spectrum mapping by counting the number of occurrences for only the length 3 substrings that occur in each input sequence, as follows

$$\phi(\mathbf{x}_1) = (ACA : 1, AGC : 1, AGT : 1, CAG : 2, GCA : 1, GTA : 1)$$
$$\phi(\mathbf{x}_2) = (AAG : 1, AGC : 2, CAA : 1, CGA : 1, GAG : 1, GCA : 1, GCG : 1)$$

where $\alpha : \#(\alpha)$ denotes that substring α has $\#(\alpha)$ occurrences in $\mathbf{x}_i$. We can then compute the dot product by considering only the common substrings, as follows

$$K(\mathbf{x}_1, \mathbf{x}_2) = 1 \times 2 + 1 \times 1 = 2 + 1 = 3$$

The first term in the dot product is due to the substring AGC, and the second due to GCA, which are the only common length 3 substrings between $\mathbf{x}_1$ and $\mathbf{x}_2$.

The full spectrum can be computed by considering the occurrences of all common substrings. For $\mathbf{x}_1$ and $\mathbf{x}_2$, the common substrings and their occurrence counts are given as

α	A	C	G	AG	CA	AGC	GCA	$AGCA$
$\#(\alpha)$ in $\mathbf{x}_1$	4	2	2	2	2	1	1	1
$\#(\alpha)$ in $\mathbf{x}_2$	4	2	4	3	1	2	1	1

Thus, the full spectrum kernel value is given as

$$K(\mathbf{x}_1, \mathbf{x}_2) = 16 + 4 + 8 + 6 + 2 + 2 + 1 + 1 = 40$$

5.4.2 Diffusion Kernels on Graph Nodes

Let $\mathbf{S}$ be some symmetric similarity matrix between nodes of a graph $G = (V, E)$. For instance $\mathbf{S}$ can be the (weighted) adjacency matrix $\mathbf{A}$ (4.2) or the negated Laplacian matrix $-\mathbf{L} = \mathbf{A} - \mathbf{\Delta}$ (17.7), where $\mathbf{\Delta}$ is the degree matrix for an undirected graph G.

Consider the similarity between any two nodes obtained by summing up the product of the similarities over paths of length 2

$$S^{(2)}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{a=1}^{n} S(\mathbf{x}_i, \mathbf{x}_a) S(\mathbf{x}_a, \mathbf{x}_j) = \mathbf{S}_i^T \mathbf{S}_j$$

where

$$\mathbf{S}_i = \Big(S(\mathbf{x}_i, \mathbf{x}_1), S(\mathbf{x}_i, \mathbf{x}_2), \cdots, S(\mathbf{x}_i, \mathbf{x}_n) \Big)^T$$

denotes the vector representing the i-th row of $\mathbf{S}$ (and since $\mathbf{S}$ is symmetric, it also denotes the i-th column of $\mathbf{S}$). Over all pairs of nodes the 2 length path similarity matrix $\mathbf{S}^{(2)}$ is thus given as the square of the base similarity matrix $\mathbf{S}$

$$\mathbf{S}^{(2)} = \mathbf{S} \times \mathbf{S} = \mathbf{S}^2$$

In general, if we sum up the product of the base similarities over all l-length paths between two nodes we obtain the l-length similarity matrix $\mathbf{S}^{(l)}$, which can be shown to be simply the l-th power of $\mathbf{S}$

$$\mathbf{S}^{(l)} = \mathbf{S}^l$$

Power Kernels: Even path lengths lead to positive semi-definite kernels, but odd path lengths are not guaranteed to do so, unless the base matrix $\mathbf{S}$ is itself a positive semi-definite matrix. In particular, $\mathbf{K} = \mathbf{S}^2$ is a valid kernel. To see this, assume that the i-th row of $\mathbf{S}$ denotes the feature map for $\mathbf{x}_i$, i.e., $\phi(\mathbf{x}_i) = \mathbf{S}_i$. The kernel value between any two points is then a dot product in feature space

$$K(\mathbf{x}_i, \mathbf{x}_j) = S^{(2)}(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{S}_i^T \mathbf{S}_j = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$$

For a general path length l, let $\mathbf{K} = \mathbf{S}^l$. Consider the eigen-decomposition of $\mathbf{S}$

$$\mathbf{S} = \mathbf{U} \mathbf{\Lambda} \mathbf{U}^T = \sum_{i=1}^{n} \mathbf{u}_i \lambda_i \mathbf{u}_i^T$$

where $\mathbf{U}$ is the orthogonal matrix of eigenvectors and $\mathbf{\Lambda}$ is the diagonal matrix of eigenvalues of $\mathbf{S}$

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

The eigen-decomposition of $\mathbf{K}$ can be obtained as follows

$$\mathbf{K} = \mathbf{S}^l = \big(\mathbf{U}\mathbf{\Lambda}\mathbf{U}^T\big)^l = \mathbf{U}\big(\mathbf{\Lambda}^l\big)\mathbf{U}^T$$

where we used the fact that eigenvectors of $\mathbf{S}$ and $\mathbf{S}^l$ are identical, and further that eigenvalues of $\mathbf{S}^l$ are given as $(\lambda_i)^l$ (for all $i = 1, \cdots, n$), where λ_i is an eigenvalue of $\mathbf{S}$. For $\mathbf{K} = \mathbf{S}^l$ to be a positive semi-definite matrix, all its eigenvalues must be non-negative, which is guaranteed for all even path lengths. Since $(\lambda_i)^l$ will be negative if l is odd and λ_i is negative, odd path lengths lead to a positive semi-definite kernel only if $\mathbf{S}$ is positive semi-definite.

Exponential Diffusion Kernel: Instead of fixing the path length *a priori*, we can obtain a new kernel between nodes of a graph by considering paths of all possible lengths, but by damping the contribution of longer paths, which leads to the *exponential diffusion kernel*, defined as

$$\begin{aligned} \mathbf{K} &= \sum_{l=0}^{\infty} \frac{1}{l!}\beta^l \mathbf{S}^l \\ &= \mathbf{I} + \beta\mathbf{S} + \frac{1}{2!}\beta^2\mathbf{S}^2 + \frac{1}{3!}\beta^3\mathbf{S}^3 + \cdots \\ &= \exp\big\{\beta\mathbf{S}\big\} \end{aligned} \tag{5.19}$$

where $\beta \in \mathbb{R}$ is a damping factor, and $\exp\{\beta\mathbf{S}\}$ is the matrix exponential.

Substituting $\mathbf{S} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T = \sum_{i=1}^n \lambda_i \mathbf{u}_i \mathbf{u}_i^T$ in (5.19), and utilizing the fact that $\mathbf{U}\mathbf{U}^T = \sum_{i=1}^n \mathbf{u}_i\mathbf{u}_i^T = \mathbf{I}$, we have

$$\begin{aligned} \mathbf{K} &= \mathbf{I} + \beta\mathbf{S} + \frac{1}{2!}\beta^2\mathbf{S}^2 + \cdots \\ &= \left(\sum_{i=1}^n \mathbf{u}_i\mathbf{u}_i^T\right) + \left(\sum_{i=1}^n \mathbf{u}_i\beta\lambda_i\mathbf{u}_i^T\right) + \left(\sum_{i=1}^n \mathbf{u}_i\frac{1}{2!}\beta^2\lambda_i^2\mathbf{u}_i^T\right) + \cdots \\ &= \sum_{i=1}^n \mathbf{u}_i\big(1 + \beta\lambda_i + \frac{1}{2!}\beta^2\lambda_i^2 + \cdots\big)\mathbf{u}_i^T \\ &= \sum_{i=1}^n \mathbf{u}_i e^{\beta\lambda_i}\ \mathbf{u}_i^T \end{aligned}$$

$$= \mathbf{U} \begin{pmatrix} e^{\beta\lambda_1} & 0 & \cdots & 0 \\ 0 & e^{\beta\lambda_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & e^{\beta\lambda_n} \end{pmatrix} \mathbf{U}^T \tag{5.20}$$

Thus, the eigenvectors of $\mathbf{K}$ are the same as those for $\mathbf{S}$, whereas its eigenvalues are given as $e^{\beta\lambda_i}$, where λ_i is an eigenvalue of $\mathbf{S}$. $\mathbf{K}$ is symmetric, since $\mathbf{S}$ is symmetric, and its eigenvalues are real and non-negative, since the exponential of a real number is non-negative. $\mathbf{K}$ is thus a positive semi-definite kernel matrix. The complexity of computing the diffusion kernel is $O(n^3)$ corresponding to the complexity of computing the eigen-decomposition.

Von Neumann Diffusion Kernel: A related kernel based on powers of $\mathbf{S}$ is the *von Neumann diffusion kernel*, defined as

$$\mathbf{K} = \sum_{l=0}^{\infty} \beta^l \mathbf{S}^l \tag{5.21}$$

Expanding the above, we have

$$\begin{aligned} \mathbf{K} &= \mathbf{I} + \beta\mathbf{S} + \beta^2\mathbf{S}^2 + \beta^3\mathbf{S}^3 + \cdots \\ &= \mathbf{I} + \beta\mathbf{S}(\mathbf{I} + \beta\mathbf{S} + \beta^2\mathbf{S}^2 + \cdots) \\ &= \mathbf{I} + \beta\mathbf{S}\mathbf{K} \end{aligned}$$

Rearranging the terms above we obtain a closed form expression for the von Neumann kernel

$$\begin{aligned} \mathbf{K} - \beta\mathbf{S}\mathbf{K} &= \mathbf{I} \\ \implies (\mathbf{I} - \beta\mathbf{S})\mathbf{K} &= \mathbf{I} \\ \implies \mathbf{K} &= (\mathbf{I} - \beta\mathbf{S})^{-1} \end{aligned} \tag{5.22}$$

Plugging in the eigen-decomposition $\mathbf{S} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$, and rewriting $\mathbf{I} = \mathbf{U}\mathbf{U}^T$, we have

$$\begin{aligned} \mathbf{K} &= \left(\mathbf{U}\mathbf{U}^T - \mathbf{U}(\beta\mathbf{\Lambda})\mathbf{U}^T\right)^{-1} \\ &= \left(\mathbf{U}\left(\mathbf{I} - \beta\mathbf{\Lambda}\right)\mathbf{U}^T\right)^{-1} \\ &= \mathbf{U}\left(\mathbf{I} - \beta\mathbf{\Lambda}\right)^{-1}\mathbf{U}^T \end{aligned}$$

where $(\mathbf{I} - \beta\mathbf{\Lambda})^{-1}$ is the diagonal matrix whose i-th diagonal entry is $(1 - \beta\lambda_i)^{-1}$. The eigenvectors of $\mathbf{K}$ and $\mathbf{S}$ are identical, but the eigenvalues of $\mathbf{K}$ are given as

$1/(1-\beta\lambda_i)$. For $\mathbf{K}$ to be a positive semi-definite kernel, all its eigenvalues should be non-negative, which in turn implies that

$$\begin{aligned} & (1-\beta\lambda_i)^{-1} \geq 0 \\ \implies & 1-\beta\lambda_i \leq 0 \\ \implies & \beta \leq 1/\lambda_i \end{aligned}$$

Furthermore, the inverse matrix $(\mathbf{I}-\beta\mathbf{\Lambda})^{-1}$ exists only if $det(\mathbf{I}-\beta\mathbf{\Lambda}) = \prod_{i=1}^{n}(1-\beta\lambda_i) \neq 0$, which implies that $\beta \neq 1/\lambda_i$ for all i. Thus, for $\mathbf{K}$ to be a valid kernel, we require that $\beta < 1/\lambda_i$ for all $i = 1, \cdots, n$. The von Neumann kernel is thus guaranteed to be positive semi-definite if $|\beta| < 1/\rho(\mathbf{S})$, where $\rho(\mathbf{S}) = \max_i\{|\lambda_i|\}$, the largest eigenvalue in absolute value, is called the *spectral radius* of $\mathbf{S}$.

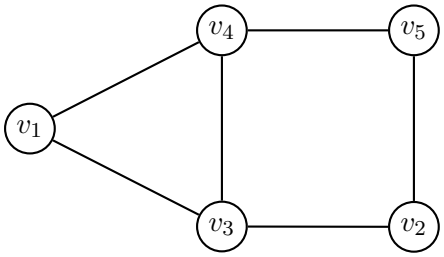


Figure 5.2: Graph Diffusion Kernel

Example 5.15: Consider the graph in Figure 5.2. It's adjacency matrix and degree matrix is given as

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{pmatrix} \qquad \mathbf{D} = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{pmatrix}$$

The negated Laplacian matrix for the graph is therefore

$$\mathbf{S} = -\mathbf{L} = \mathbf{A} - \mathbf{D} = \begin{pmatrix} -2 & 0 & 1 & 1 & 0 \\ 0 & -2 & 1 & 0 & 1 \\ 1 & 1 & -3 & 1 & 0 \\ 1 & 0 & 1 & -3 & 1 \\ 0 & 1 & 0 & 1 & -2 \end{pmatrix}$$

The eigenvalues of $\mathbf{S}$ are as follows

$$\lambda_1 = 0 \qquad \lambda_2 = -1.38 \qquad \lambda_3 = -2.38 \qquad \lambda_4 = -3.62 \qquad \lambda_5 = -4.62$$

and the eigenvectors of $\mathbf{S}$ are

$$\mathbf{U} = \begin{pmatrix} \mathbf{u}_1 & \mathbf{u}_2 & \mathbf{u}_3 & \mathbf{u}_4 & \mathbf{u}_5 \\ \hline 0.45 & -0.63 & 0.00 & 0.63 & 0.00 \\ 0.45 & 0.51 & -0.60 & 0.20 & -0.37 \\ 0.45 & -0.20 & -0.37 & -0.51 & 0.60 \\ 0.45 & -0.20 & 0.37 & -0.51 & -0.60 \\ 0.45 & 0.51 & 0.60 & 0.20 & 0.37 \end{pmatrix}$$

Assuming $\beta = 0.2$, the exponential diffusion kernel matrix is given as

$$\mathbf{K} = \exp\{0.2\mathbf{S}\} = \mathbf{U} \begin{pmatrix} e^{0.2\lambda_1} & 0 & \cdots & 0 \\ 0 & e^{0.2\lambda_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & e^{0.2\lambda_n} \end{pmatrix} \mathbf{U}^T$$

$$= \begin{pmatrix} 0.70 & 0.01 & 0.14 & 0.14 & 0.01 \\ 0.01 & 0.70 & 0.13 & 0.03 & 0.14 \\ 0.14 & 0.13 & 0.59 & 0.13 & 0.03 \\ 0.14 & 0.03 & 0.13 & 0.59 & 0.13 \\ 0.01 & 0.14 & 0.03 & 0.13 & 0.70 \end{pmatrix}$$

For the von Neumann diffusion kernel, we have

$$(\mathbf{I} - 0.2\mathbf{\Lambda})^{-1} = \begin{pmatrix} 1 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0 & 0.78 & 0.00 & 0.00 & 0.00 \\ 0 & 0.00 & 0.68 & 0.00 & 0.00 \\ 0 & 0.00 & 0.00 & 0.58 & 0.00 \\ 0 & 0.00 & 0.00 & 0.00 & 0.52 \end{pmatrix}$$

For instance, since $\lambda_2 = -1.38$, we have $1 - \beta\lambda_2 = 1 + 0.2 \times 1.38 = 1.28$, and therefore the second diagonal entry is $(1 - \beta\lambda_2)^{-1} = 1/1.28 = 0.78$. The von Neumann kernel is given as

$$\mathbf{K} = \mathbf{U}(\mathbf{I} - 0.2\mathbf{\Lambda})^{-1}\mathbf{U}^T = \begin{pmatrix} 0.75 & 0.02 & 0.11 & 0.11 & 0.02 \\ 0.02 & 0.74 & 0.10 & 0.03 & 0.11 \\ 0.11 & 0.10 & 0.66 & 0.10 & 0.03 \\ 0.11 & 0.03 & 0.10 & 0.66 & 0.10 \\ 0.02 & 0.11 & 0.03 & 0.10 & 0.74 \end{pmatrix}$$

5.5 Annotated References

5.6 Exercises

1. Prove that the dimensionality of the feature space for the inhomogeneous polynomial kernel of degree q is

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

i	$\mathbf{x}_i$		i	$\mathbf{x}_i$
$\mathbf{x}_1$	(4,2.9)		$\mathbf{x}_6$	(1.9,1.9)
$\mathbf{x}_2$	(4,4)		$\mathbf{x}_7$	(3.5,4)
$\mathbf{x}_3$	(1,2.5)		$\mathbf{x}_8$	(0.5,1.5)
$\mathbf{x}_4$	(2.5,1)		$\mathbf{x}_9$	(2,2.1)
$\mathbf{x}_5$	(4.9,4.5)		$\mathbf{x}_{10}$	(4.5,2.5)

Table 5.1: Dataset for exercise 2.

2. Based only on four points $\mathbf{x}_1, \mathbf{x}_4, \mathbf{x}_7, \mathbf{x}_9$ in Table 5.1, and using the following kernel function: $K(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{x}_i - \mathbf{x}_j\|^2$. Compute the kernel matrix $\mathbf{K}$.

3. Show that eigenvectors of $\mathbf{S}$ and $\mathbf{S}^l$ are identical, and further that eigenvalues of $\mathbf{S}^l$ are given as $(\lambda_i)^l$ (for all $i = 1, \cdots, n$), where λ_i is an eigenvalue of $\mathbf{S}$.

4. show that the exponential diffusion kernel is given as

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

5. Show that a solution to the von Neumann diffusion kernel is a valid positive semi-definite kernel if $|\beta| < \frac{1}{\rho(\mathbf{S})}$, where $\rho(\mathbf{S})$ is the spectral radius of $\mathbf{S}$. Can you derive better bounds for cases when $\beta > 0$ and when $\beta < 0$.