Reproducing Kernel Hilbert Space

Zhe Li

November 12, 2014

1 Reproducing Kernel Hilbert Space

Firstly, we would like to use one concrete example to show the concept input space, feature space, feature map and kernel function. Given the data \( \{(x_i, y_i), i = 1, \ldots, n \} \), where \( x_i \in \mathcal{X} \subseteq \mathbb{R}^2 \) is the representation of feature and \( y_i \) is the target value. Here, \( x_i \) is two dimension. Often it is very hard to find a linear classifier in this two dimensional space to classifier \( x \) properly, therefore one attempt to map the original input space \( \mathcal{X} \) to a high dimension space \( \mathcal{H} \), corresponding every \( x \in \mathcal{X} \) mapping to \( \Phi(x) \in \mathcal{H} \). For specifical \( x_1 \in \mathcal{X} \), we show the feature map from \( x_1 \) to \( \Phi(x_1) \):

\[
x_1 = \begin{bmatrix} x_{11} \\ x_{12} \end{bmatrix} \xrightarrow{\Phi(\cdot)} \begin{bmatrix} x_{11}^2 \\ x_{12}^2 \\ \sqrt{2}x_{11}x_{12} \\ \sqrt{2}x_{11} \\ \sqrt{2}x_{12} \\ 1 \end{bmatrix}
\]

In this example, two dimensional space \( \mathcal{X} \) is the input space; six dimensional space \( \mathcal{H} \) is the feature space (very often \( X \) is also called feature space in many book, in this case, we can consider the feature map is linear map), the mapping \( \Phi(\cdot) \) from this two dimensional input space \( \mathcal{X} \) to six dimensional feature space \( \mathcal{H} \) is called feature map. In most linear classifier or regression model, we need to deal with the term \( x_i^T x_j \), which can be seen as the similarity between \( x_i \) and \( x_j \) in the input space. Similar, for the feature space \( \mathcal{H} \), one needs to compute \( \Phi(x_i)^T \Phi(x_j) \), which is the similarity between \( x_i \) and \( x_j \) in the feature space. For the above example, Let’s compute \( \Phi(x_i)^T \Phi(x_j) \),

\[
\Phi(x_i)^T \Phi(x_j) = x_{i1}^2x_{j1} + x_{i2}^2x_{j2} + \sqrt{2}x_{i1}x_{i2}\sqrt{2}x_{j1}x_{j2} + \sqrt{2}x_{i1}\sqrt{2}x_{j1} + \sqrt{2}x_{i2}\sqrt{2}x_{j2} + 1
\]

\[
= (1 + x_{i1}x_{j1} + x_{i2}x_{j2})^2
\]

\[
= (1 + x_i^T x_j)^2
\]
It turns out that $\Phi(x_i)^T \Phi(x_j)$ can be computed via the quadratic function in the input space with less computational cost compared with computing it in the feature space. Thus, we define this quadratic function

$$k(x, x') = (1 + x^T x')^2$$

as kernel function. There are two benefits following this, one is less computation cost to compute $\Phi(x_i)^T \Phi(x_j)$ for high dimensional feature space and another is that one does not necessary explicitly construct feature space $\mathcal{H}$, just use the similarity between two objects $x_i$ and $x_j$ as $k(x_i, x_j)$. Here, we explain some notations, used later on.

$$k(x_i, \cdot) = (x_{i1}^2, x_{i2}^2, \sqrt{2}x_{i1}x_{i2}, \sqrt{2}x_{i1}, \sqrt{2}x_{i2}, 1)^T$$

Since $k(\cdot, \cdot)$ is symmetric, we have $k(x_i, \cdot) = k(\cdot, x_i) = \Phi(x_i)$. Note that they are vectors in this six dimention feature space. There is slight difference between $k(x_i, \cdot)$ and $\Phi(x_i)$, we will explain that later.

In this high dimension feature space, we define a linear function (Here, linear is in terms of this high dimensional feature space)

$$f(x) = \langle f(\cdot), k(x, \cdot) \rangle$$

The notation $f(\cdot), f$ are same and $k_x(\cdot), k(x, \cdot), k(\cdot, x)$ are same. $f(\cdot)$ is a vector $(f_1 \ f_2 \ \cdots \ f_6)^T$ representing linear function,

$$f_1 \ast \text{dim}_1 + f_2 \ast \text{dim}_2 + \cdots + f_6 \ast \text{dim}_6$$

So the $f(x)$ is

$$f(x_i) = f_1 x_{i1}^2 + f_2 x_{i2}^2 + f_3\sqrt{2}x_{i1}x_{i2} + f_4\sqrt{2}x_{i1} + f_5\sqrt{2}x_{i2} + f_6$$

The vector $f$ is kind of similar to $w$ in linear case. $f$ is vector, but representing a linear function, so is $k(x, \cdot)$. Previously we say there is slight difference between $k(x, \cdot)$ and $\Phi(x)$, even though both of them are same vector, but the vector $k(x, \cdot)$ represents a linear function in that six dimension feature space. How can we find vector $f$ or function $f(x)$ so that

$$\min_{f \in \mathcal{F}} \sum_{i=1}^{n} l(y_i, f^T \Phi_i(x_i)) + \frac{\lambda}{2} \|f\|^2$$

The representer theorem tells us that the vector $f$ is the combination of the vectors $k(\cdot, x_1), k(\cdot, x_2), \cdots, k(\cdot, x_n)$. Using the linear algebra language, the six dimensional vector
f lies in the space \( \mathcal{H}_s = \text{span}\{k(\cdot, x_1), \cdots, k(\cdot, x_i), \cdots, k(\cdot, x_n)\} \). The entire six dimensional space \( \mathcal{H} \) can be orthogonally decomposed into \( \mathcal{H}_s \) and \( \mathcal{H}_\perp \), we can write \( f \) as

\[
f = \alpha_1 k(\cdot, x_1) + \alpha_2 k(\cdot, x_2) + \cdots + \alpha_n k(\cdot, x_n) = \sum_{i=1}^{n} k(\cdot, x_i)\alpha_i
\]

So we can write

\[
f(x_j) = f^T\Phi(x_j) = \alpha_1 k(\cdot, x_1)^T k(\cdot, x_j) + \alpha_2 k(\cdot, x_2)^T k(\cdot, x_j) + \cdots + \alpha_n k(\cdot, x_n)^T k(\cdot, x_j)
\]

Via representer theorem, the Eq. (6) can be rewritten as

\[
\min_{\alpha \in \mathbb{R}^n} \sum_{i=1}^{n} l((K\alpha)_i) + \frac{\lambda}{2} \alpha^T K\alpha
\]  

(7)

Combined the dual form of Eq. (6), we summarize the different form for machine learning problem

<table>
<thead>
<tr>
<th>Form</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Primal form</strong></td>
<td>( \min_{f \in \mathcal{F}} \sum_{i=1}^{n} l(y_i, f^T\Phi_i(x_i)) + \frac{\lambda}{2}</td>
</tr>
<tr>
<td><strong>Primal form + Representer Theorem</strong></td>
<td>( \min_{\alpha \in \mathbb{R}^n} \sum_{i=1}^{n} l((K\alpha)_i) + \frac{\lambda}{2} \alpha^T K\alpha )</td>
</tr>
<tr>
<td><strong>Dual form</strong></td>
<td>( \max_{\alpha \in \mathbb{R}^n} - \sum_{i=1}^{n} l^*(\lambda \alpha_i) - \frac{\lambda}{2} \lambda \alpha^T K\alpha )</td>
</tr>
</tbody>
</table>

Since the concepts in Reproducing Kernel Hilbert Space is very hard to understand, we use one concrete example to illustrate the important concepts for that.

## 2 The Proof of Representer Theorem

From linear algebra, the space \( \mathcal{H} \) can be divided into two orthogonal space \( \mathcal{H}_s \) and \( \mathcal{H}_\perp \), assume \( \mathcal{H}_s = \text{span}\{k(\cdot, x_1), \cdots, k(\cdot, x_i), \cdots, k(\cdot, x_n)\} \). For any \( f \) in the space \( \mathcal{H} \), \( f \) can be considered being composing from two part \( f_s \) and \( f_\perp \), put that formally, \( f = f_s + f_\perp \)
• 1). $\mathcal{H}_s = \text{span}\{k(\cdot, x_1), \ldots, k(\cdot, x_i), \ldots, k(\cdot, x_n)\}$

• 2). orthogonal decomposition: $\mathcal{H} = \mathcal{H}_s \oplus \mathcal{H}_\perp, \forall f \in \mathcal{H}, f = f_s + f_\perp$

• 3). pointwise evaluation decomposition

\[
f(x_i) = f_s(x_i) + f_\perp(x_i) = \langle f_s, k(\cdot, x_i) \rangle + \langle f_\perp(\cdot), k(\cdot, x_i) \rangle = \langle f_s, k(\cdot, x_i) \rangle \rangle = f_s(x_i)
\]

Since $f_\perp$ and $k(\cdot, x_i)$ are orthogonal.

• 4). norm decomposition $||f||^2 = ||f_s||^2 + ||f_\perp||^2 \geq ||f_s||^2$

• 5). decompose the global cost

\[
\sum_{i=1}^{n} l(y_i, f(x_i)) + ||f||^2 = \sum_{i=1}^{n} l(y_i, f_s(x_i)) + ||f_s||^2 + ||f_\perp||^2 = \sum_{i=1}^{n} l(y_i, f_s(x_i)) + ||f_s||^2
\]

So, we have $\argmin_{f \in \mathcal{H}} \text{Obj func} = \argmin_{f \in \mathcal{H}_s} \text{Obj func}$. In a word, if in small space, we can find a function satisfying the requirement and if we go beyond that small space, definitely we will increase the cost. If you can finish your job in a small space, don’t go to the bigger space, which makes worse.

3 Reference

http://www.gatsby.ucl.ac.uk/~gretton/coursefiles/Slides4A.pdf
http://www.di.ens.fr/~fbach/eccv08_fbach.pdf
http://www.cs.berkeley.edu/~bartlett/courses/281b-sp08/7.pdf