Statistical Pattern Recognition

Introduction to Kernel Methods

Hamid R. Rabiee
Mohammad H. Rohban

Spring 2013
http://ce.sharif.edu/courses/91-92/2/ce725-1/
Agenda

✧ Motivations
✧ Kernel Definition
✧ Mercer’s Theorem
✧ Kernel Matrix
✧ Kernel Construction
Motivations

✧ Learning linear classifiers can be done effectively (SVM, Perceptron, …).
  ✧ How to generalize existing efficient linear classifiers to non-linear ones.
✧ It may be hard to classify data points in the original feature space.
  ✧ Use an appropriate high dimensional non-linear map to change the feature space.
Kernel Definition

✧ Consider data \( x \) lying in \( \mathbb{R}^n \).

✧ Use a high dimensional mapping \( \Phi: \mathbb{R}^n \rightarrow \mathbb{R}^N \), with \( N > n \).

✧ Define the kernel function \( K(x,x') = \Phi(x)^T \Phi(x') \).

✧ That is the kernel function is the dot product in the new feature space.

✧ Dot product measures the similarity of two data points.

✧ \( K(x,x') \) shows the similarity of \( x \) and \( x' \).

✧ It is efficient to use \( K \) instead of \( \Phi \) if the dimensionality of \( \Phi \) is high (Why?).
Kernel Definition

✧ A simple example:

✧ Consider \( x = (x_1, x_2) \) lies in 2 dimensional plane and \( \Phi : \mathbb{R}^2 \rightarrow \mathbb{R}^3 \) with the following definition

\[
\Phi(x_1, x_2) = (z_1, z_2, z_3) = (x_1^2, \sqrt{2}x_1x_2, x_2^2)
\]

✧ A linear classifier in new space will become (\( w' \) is a vector in new space):

\[
g(x) = w'^T x' + w'_0 = w'^T \Phi(x) + w'_0 = w'_1 x_1^2 + \sqrt{2}w'_2 x_1x_2 + w'_3 x_2^2 + w'_0
\]

✧ What will be the shape of separating curve in the original space?

\[
w'_1 x_1^2 + \sqrt{2}w'_2 x_1x_2 + w'_3 x_2^2 + w'_0 = 0
\]
Kernel Definition

✧ What will be the kernel function in the previous example?

\[
K(u, v) = \Phi(u)^T \Phi(v) = \begin{pmatrix} u_1^2 & \sqrt{2}u_1u_2 \\ u_2^2 & \sqrt{2}v_1v_2 \end{pmatrix} \begin{pmatrix} v_1^2 \\ v_2^2 \end{pmatrix}
\]

\[
= (u_1v_1)^2 + 2(u_1v_1)(u_2v_2) + (u_2v_2)^2
\]

\[
= (u_1v_1 + u_2v_2)^2 = (u^Tv)^2
\]

The dot product in the new space is squared of the dot product in the original space.

✧ Can we construct an arbitrary conic section in original feature space? Why?

We instead use \((u^Tv + 1)^2\)
Kernel Definition

✧ Some typical kernels include:
  ✧ Polynomial: \( K(u,v) = (u^T v + c)^d \)
  ✧ Sigmoid: \( K(u,v) = \tanh(\kappa u^T v + \theta) \)
  ✧ Gaussian RBF: \( K(u,v) = \exp\left\{ -\|u-v\|^2 / 2\sigma^2 \right\} \)

✧ Can any function \( K(u,v) \) be a valid kernel function?
  ✧ That is, does there exist a function \( \Phi \) with \( K(u,v) = \Phi(u)^T \Phi(v) \)?
  ✧ In the case of Mercer’s condition, it is a valid kernel function.
Mercer’s Theorem

✧ If for any squared integrable function \( f(.) \), we have
\[
\int_{\mathbb{R}^{2n}} K(x, x') f(x) f(x') dx dx' \geq 0
\]
then the function \( K(x, x') \) is a valid kernel function.

✧ In this case the components of the corresponding function \( \Phi \) are proportional to the eigenfunctions of \( K(x, x') \), that is
\[
\Phi(x) = \left( \sqrt{\lambda_1} \varphi_1(x), \sqrt{\lambda_2} \varphi_2(x), \ldots \right)
\]
\[
\int_{\mathbb{R}^n} K(u, v) \varphi_i(v) dv = \lambda_i \varphi_i(u)
\]

In fact Mercer’s theorem checks that if \( K(x, y) \) is positive semi-definite and hence all \( \lambda_i \geq 0 \).
Kernel Matrix

✧ Restricting the kernel function to a set of points \( \{x_1, \ldots, x_k\} \), the kernel function can be represented with a matrix:

\[
K = \begin{bmatrix}
K(x_1,x_1) & K(x_1,x_2) & \cdots & K(x_1,x_k) \\
K(x_2,x_1) & K(x_2,x_2) & \cdots & \\
\vdots & \vdots & \ddots & \\
K(x_k,x_1) & K(x_k,x_2) & \cdots & K(x_k,x_k)
\end{bmatrix}
\]

✧ A matrix \( K \) is a valid kernel matrix if it is a positive semi-definite matrix,

✧ That is, all its eigenvalues are greater or equal to zero.

✧ The eigenvectors multiplied by squared roots of eigenvalues will be the restrictions of \( \Phi_i \) to the set \( \{x_1, \ldots, x_k\} \).
Polynomial Kernel

✧ 2nd degree polynomial:

\[ K(u,v) = (u^T v)^2 = (u_1 v_1 + u_2 v_2)^2 \]

\[ = \begin{pmatrix} u_1^2 \\ \sqrt{2}u_1 u_2 \\ u_2^2 \end{pmatrix}^T \begin{pmatrix} v_1^2 \\ \sqrt{2}v_1 v_2 \\ v_2^2 \end{pmatrix} \]

✧ Up to 2nd degree polynomial:

✧ Can construct any 2nd order function in original feature space

\[ K(u,v) = (u^T v + 1)^2 = (u_1 v_1 + u_2 v_2 + 1)^2 \]

\[ = \begin{pmatrix} u_1^2 \\ \sqrt{2}u_1 u_2 \\ u_2^2 \end{pmatrix}^T \begin{pmatrix} v_1^2 \\ \sqrt{2}v_1 v_2 \\ v_2^2 \end{pmatrix} \]
RBF Kernel

✧ An example

✧ That is the input space \(-5 < u < 5\) will be mapped to a curve using only 2 dimensions of \(\phi\).
**RBF Kernel**

✧ An example (cont.)

✧ **Consider the Gaussian kernel**: 

\[ K(u,v) = \exp \left\{-\frac{\|u-v\|^2}{2\sigma^2}\right\} \]

✧ Where \( u \) lies in a subset of \( \mathbb{R} \), \(-5<u<5\).

✧ The eigenfunctions of \( K \) are illustrated. \( \Phi = (\varphi_1, ..., \varphi_{10}, ...) \).
RBF Kernel

✧ An example (cont.)

✧ Consider a linear classifier in the new space.

✧ The corresponding classifier in the u space is clearly non-linear in the original space.
RBF Kernel

- RBF kernel considers a Gaussian around each data point.
- Linear discriminant function cuts through the surface in embedding function.
- Therefore any arbitrary set of points can be classified by RBF kernels.
- Training error is zero when $\sigma \to 0$. 

Template designed by Jafar Malekmohammadi
Kernel Construction

✦ How to build valid kernels from existing kernels?
✦ According to Mercer’s theorem if $c > 0$ and $k_1$, $k_2$ are valid kernels, and $\psi$ is an arbitrary function, then following functions will also be valid kernels:
  ✦ $K(u,v) = ck_1(u,v)$
  ✦ $K(u,v) = k_1(u,v) + k_2(u,v)$
  ✦ $K(u,v) = k_1(u,v) k_2(u,v)$
  ✦ $K(u,v) = k_1(\psi(u), \psi(v))$ (here $\psi(x)$ is a function from $x \rightarrow \mathbb{R}^M$ and $k_1$ is a valid kernel in $\mathbb{R}^M$)
Kernel Construction

✧ Construct kernels from probabilistic generative models (class conditional probabilities, HMM, ...) and then use the kernel in a discriminative model (such as SVM or linear discriminant functions, ...).

✧ $K(x,x') = p(x)p(x')$ is clearly a valid kernel, which states that $x$ and $x'$ are similar if they both have high probability (Why it is valid?).

✧ A better kernel can be constructed in the same way:

$$K(u,v) = \sum_{i=1}^{n} p(u|c_i)p(v|c_i)p(c_i)$$

✧ That is $u$ and $v$ are similar if they have high probabilities under same classes.
Kernel Construction

✧ State of the arts methods tries to learn the kernel from (probably many) training points.

✧ The simplest one is the multiple kernel learning.

✧ Consider \( \{k_1, \ldots, k_n\} \) as \( n \) valid kernels.

✧ Find an appropriate kernel, \( k(u,v) \), from the training data

\[
K(u,v) = \sum_{i=1}^{n} c_i k_i(u,v), \quad c_i \geq 0
\]

✧ Minimize training loss (MSE) by changing \( c_i \) and simultaneously minimize trace of the kernel matrix on training data to avoid overfitting.

✧ Many variations of the algorithm are developed.
Any Question?

End of Lecture 11

Thank you!

Spring 2013

http://ce.sharif.edu/courses/91-92/2/ce725-1/