SHIRAZ UNIVERSITY
Computer Science and Engineering Department
Machine Learning Lab

Introduction to Kernel Methods

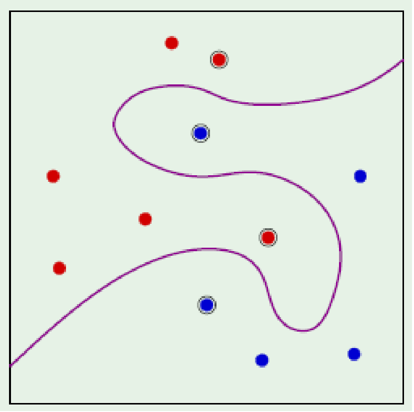
Dr Sattar Hashemi

Based on:

Christopher M. Bishop, ***Pattern recognition and machine learning*****.----- 6**

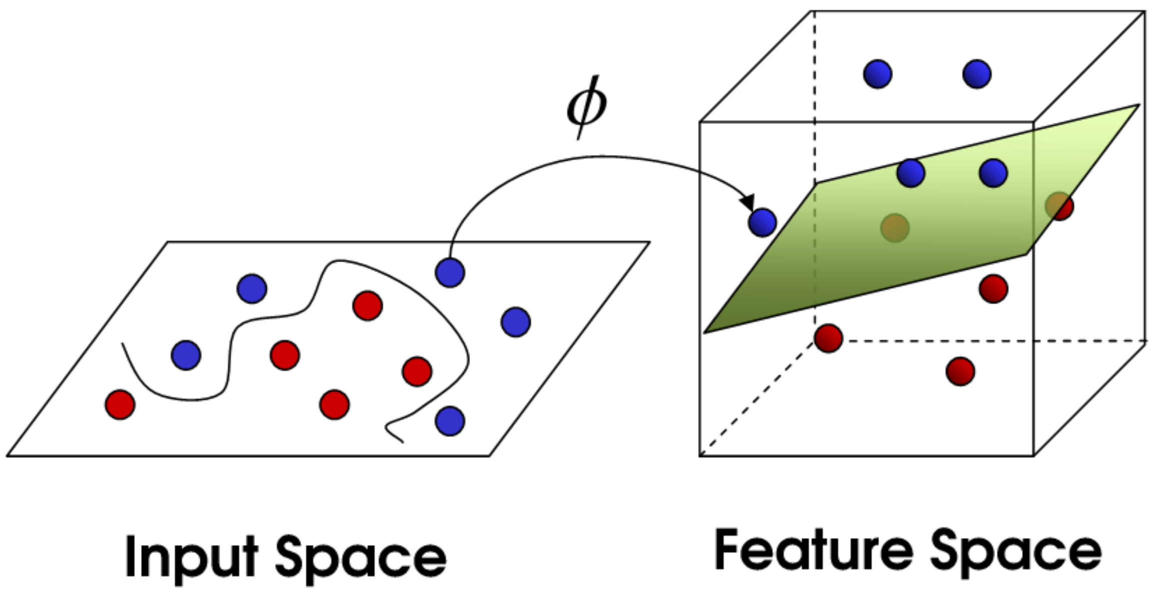
Motivation

- Given a set of vectors , there are many tools available to use to detect linear relations among the data.
  - Linear regression
  - Logistic regression
  - ...
- **But what if the relations are non-linear in the original space?**



Motivation

Solution*:* Map the data into a (possibly high dimensional) vector space where linear relations exist among the data, then apply a linear algorithm in this space



Problem

- Problem: Representing data in a high dimensional space is computationally difficult
- Alternative solution to the original problem:
  - Calculate a **similarity measure** in the feature space instead of the coordinates of the vectors there, then apply algorithms that only need the value of this measure
- Use dot product as **similarity measure**

Dot Product

- Algebraic View
- The dot product of two vector
  - $\boldsymbol{x} = [x_1, x_2, \dots, x_n]$
  - $\mathbf{z} = [z_1, z_2, \dots, z_n]$

Is:

$$\boldsymbol{x}^{\boldsymbol{T}}.\boldsymbol{z} = \sum_{i=1}^{n} x_i z_i$$
$$= x_1 z_1 + x_2 z_2 + \cdots + x_n z_n$$

- Geometric View
  - In Euclidean space, a Euclidean vector is a geometrical object that possesses both a magnitude and a direction.
  - $\boldsymbol{x}^{\boldsymbol{T}}.\boldsymbol{z} = \|\boldsymbol{x}\|.\|\boldsymbol{z}\|.\cos\theta$
  - Where θ is angle between $\boldsymbol{x}$, $\boldsymbol{z}$

Kernel Function

- A function that takes as its inputs vectors in the original space and returns the dot product of the vectors in the feature space is called a kernel function .
- More formally, if we have data $\boldsymbol{x}, \boldsymbol{z} \in \mathbb{X}$ and a map $\varphi: \boldsymbol{x} \rightarrow \varphi(\boldsymbol{x})$ then

$$k(\boldsymbol{x}, \boldsymbol{z}) = \varphi(\boldsymbol{x})^T . \varphi(\boldsymbol{z})$$

  Is a kernel function .
- Using kernels, we do not need to embed the data into the Feature space explicitly, because a number of algorithms only require the inner products between the mapped vectors!

Kernel Function

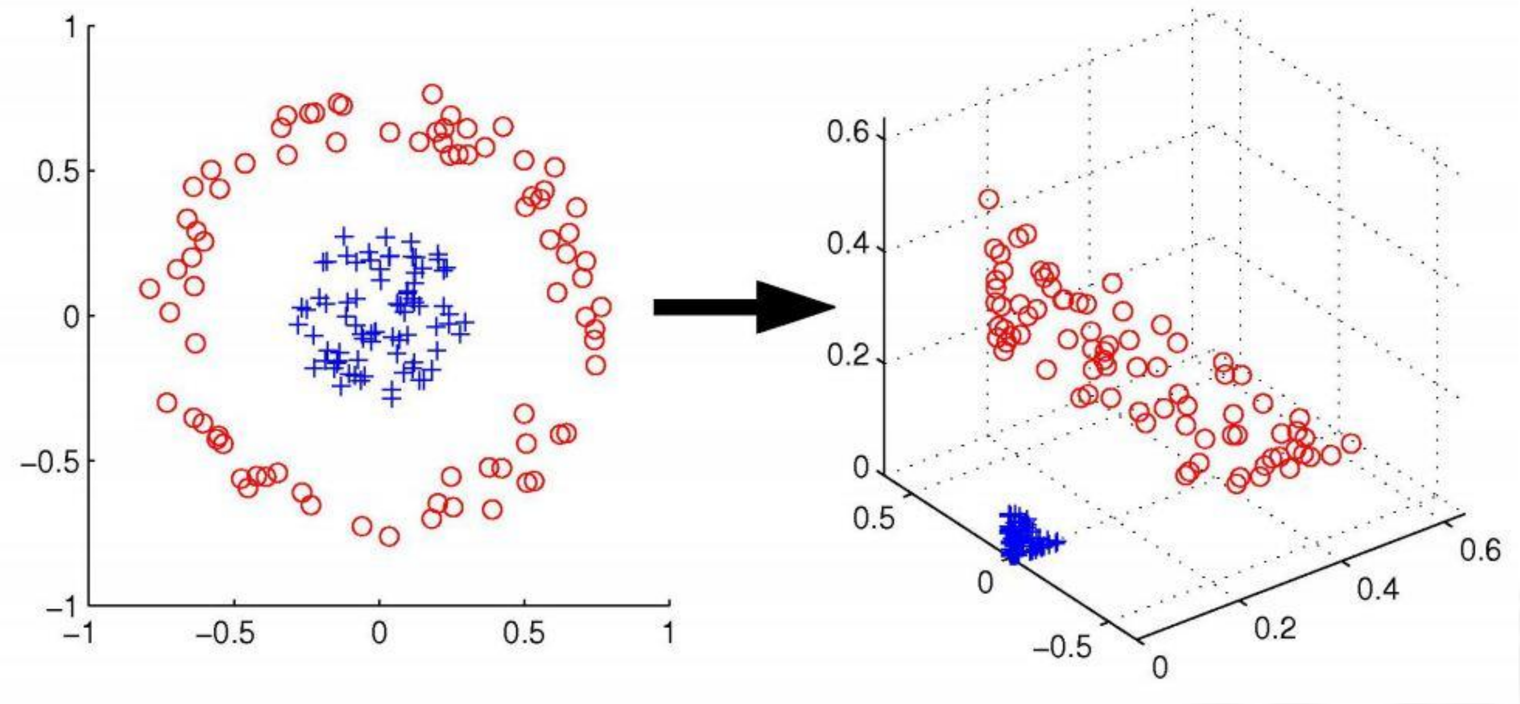
- Consider the two dimensional space $\mathbb{X}$ with the feature map :

$$\varphi: \boldsymbol{x} = (x_1, x_2) \rightarrow \varphi(\boldsymbol{x}) = \left(x_1^2, \sqrt{2}x_1 x_2, x_2^2\right) \in R^3$$

- Now consider the inner product in feature space :

$$\begin{aligned} &\varphi(\boldsymbol{x})^T.\,\varphi(\boldsymbol{y}) \\ &= \left(x_1^2, \sqrt{2}x_1 x_2, x_2^2\right).\,\left(z_1^2, \sqrt{2}z_1 z_2, z_2^2\right) \\ &= x_1^2 z_1^2 + 2x_1 x_2 z_1 z_2 + x_2^2 z_2^2 \\ &= (x_1 z_1 + x_2 z_2)^2 \\ &= \left(\boldsymbol{x}^{\boldsymbol{T}}.\,\boldsymbol{z}\right)^2 \end{aligned}$$

Kernel Example (cont'd)



Effect of the map $\varphi(\boldsymbol{x}) = \left(x_1^2, \sqrt{2}x_1x_2, x_2^2\right)$

Kind of Kernels

Linear	$k(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{x}^{\mathrm{T}}\boldsymbol{x}' + \mathrm{c}$
polynomial	$k(\boldsymbol{x}, \boldsymbol{x}') = (\alpha\boldsymbol{x}^{T}x' + \mathrm{c})^{\mathrm{d}}$
Exponential	$k(\boldsymbol{x}, \boldsymbol{x}') = \exp(\dfrac{\lVert\boldsymbol{x} - \boldsymbol{x}'\rVert}{2\sigma^2})$
Gaussian	$k(\boldsymbol{x}, \boldsymbol{x}') = \exp(\dfrac{\lVert\boldsymbol{x} - \boldsymbol{x}'\rVert^2}{2\sigma^2})$
power	$k(\boldsymbol{x}, \boldsymbol{x}') = -\lVert\boldsymbol{x} - \boldsymbol{x}'\rVert^{d}$

Linear Regression: Primal Form

Learn $\hat{f}(\boldsymbol{x}) = \sum_{i=1}^{N} x_i w_i =< \boldsymbol{x}, \boldsymbol{w} > = \boldsymbol{x}^T \boldsymbol{w}$

Where $\boldsymbol{w} = \arg\min_{w} \| \boldsymbol{y} - X\boldsymbol{w} \|^2 + \lambda \| \boldsymbol{w} \|^2$

solve by taking derivative wrt **w**, setting to zero…

$$w = (X^T X + \lambda I)^{-1} X^T \boldsymbol{y}$$

So: $\hat{f}(\boldsymbol{x}_{new}) = \boldsymbol{x}_{new}^T \boldsymbol{w} = \boldsymbol{x}_{new}^T (X^T X + \lambda I)^{-1} X^T y$

Aha!

Learn $\hat{f}(X) = \sum_{i=1}^{N} x_i w_i =< \boldsymbol{x}, \boldsymbol{w} > = \boldsymbol{x}^T \boldsymbol{w}$

Where $\boldsymbol{w} = \arg\min_{w} \| \boldsymbol{y} - X\boldsymbol{w} \|^2 + \lambda \| \boldsymbol{w} \|^2$

Solution: $\boldsymbol{w} = (X^T X + \lambda I)^{-1} X^T \boldsymbol{y}$

But notice $\boldsymbol{w}$ lies in the space spanned by training examples (why?)

Aha!

Learn $\hat{f}(X) = \sum_{i=1}^{N} x_i w_i =< \boldsymbol{x}, \boldsymbol{w} > = \ \boldsymbol{x}^T \boldsymbol{w}$

Where $\boldsymbol{w} = \arg\min_{w} \| \boldsymbol{y} - X\boldsymbol{w} \|^2 + \lambda \| \boldsymbol{w} \|^2$

Solution: $\boldsymbol{w} = (X^T X + \lambda I)^{-1} X^T \boldsymbol{y}$

But notice $\boldsymbol{w}$ lies in the space spanned by training examples (why?)

$X^T X\boldsymbol{w} + \lambda \boldsymbol{w} = X^T y$ implies

$$\boldsymbol{w} = \frac{1}{\lambda}(X^T y - X^T X\boldsymbol{w}) = X^T \frac{1}{\lambda}(\boldsymbol{y} - X\boldsymbol{w}) = X^T \boldsymbol{\alpha},$$

Where

$$\boldsymbol{\alpha} = \frac{1}{\lambda}(\boldsymbol{y} - X\boldsymbol{w})$$

Linear Regression: Dual Form

Primal form:

Learn $\hat{f}(X) = \sum_{i=1}^{n} x_i w_i =< \boldsymbol{x}, \boldsymbol{w} > = \ \boldsymbol{x}^T \boldsymbol{w}$

$$\boldsymbol{w} = \arg\min_{w} \parallel \boldsymbol{y} - X\boldsymbol{w} \parallel^2 + \lambda \parallel \boldsymbol{w} \parallel^2$$

Solution: $\boldsymbol{w} = (X^T X + \lambda I)^{-1} X^T \boldsymbol{y}$

Dual form: use the fact that $w = \sum_{i=1}^{m} \alpha_i \boldsymbol{x}^i$

Learn $\hat{f}(X) = \sum_{i=1}^{m} \alpha_m < \boldsymbol{x}, \boldsymbol{x}^i >$

$$\boldsymbol{\alpha} = \arg\min_{w} \parallel \boldsymbol{y} - XX^T \boldsymbol{\alpha} \parallel^2 + \lambda \parallel X^T \boldsymbol{\alpha} \parallel^2$$

Solution: $\boldsymbol{\alpha} = (XX^T + \lambda I)^{-1} \boldsymbol{y}$

A dual solution expresses the weight vector w as a linear combination of the training examples:

$$X^T X\boldsymbol{w} + \lambda \boldsymbol{w} = X^T y \quad \text{implies}$$

$$\boldsymbol{w} = \frac{1}{\lambda}(X^T y - X^T X\boldsymbol{w}) = X^T \frac{1}{\lambda}(\boldsymbol{y} - X\boldsymbol{w}) = X^T \boldsymbol{\alpha},$$

Where

$$\boldsymbol{\alpha} = \frac{1}{\lambda}(\boldsymbol{y} - X\boldsymbol{w}) \qquad (1)$$

Or equivalently

$$\boldsymbol{w} = \sum_{i=1}^{m} \alpha_i \, \boldsymbol{x}^i$$

The vector $\boldsymbol{\alpha}$ is the dual solution

Substituting $\boldsymbol{w} = X^T\boldsymbol{\alpha}$ into equation (1) we obtain:

$$\lambda\boldsymbol{\alpha} = \boldsymbol{y} - XX^T\boldsymbol{\alpha}$$

Implying

$$(XX^T + \lambda I)\boldsymbol{\alpha} = \boldsymbol{y}$$

This means the dual solution can be computed as:

$$\boldsymbol{\alpha} = (XX^T + \lambda I)^{-1}\boldsymbol{y}$$

With the regression function

$$g(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{w} = \boldsymbol{x}^T X^T\boldsymbol{\alpha} =< \boldsymbol{x}, \sum_{i=1}^{m} \alpha_i \boldsymbol{x}^i > = \sum_{i=1}^{m} \alpha_i < \boldsymbol{x}, \boldsymbol{x}^i >$$

Using Kernel

Step 1:Compute

$$\boldsymbol{\alpha} = (K + \lambda I)^{-1}\boldsymbol{y}$$

Where $K = XX^T$ that is $K_{ij} =< \boldsymbol{x}^i, \boldsymbol{x}^j >$

Step 2:Evaluate on new point x by

$$g(\boldsymbol{x}) = \sum_{i=1}^{m} \alpha_i < \boldsymbol{x}, \boldsymbol{x}^i >$$

Important observation: Both steps only involve inner products between input data points