Lecture 18: 04 April, 2022

Madhavan Mukund

https://www.cmi.ac.in/~madhavan

Data Mining and Machine Learning
January–May 2022

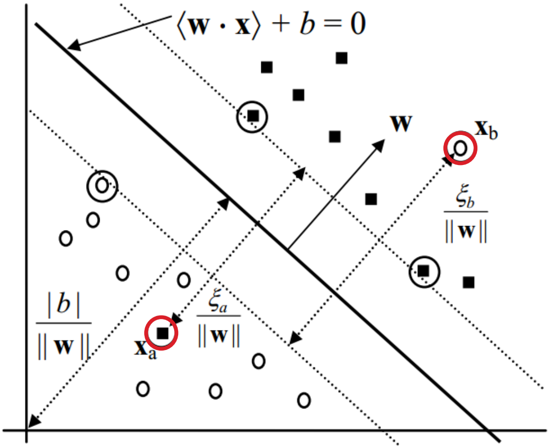
Soft margin optimization

Minimize $\frac{||w||}{2} + \sum_{i=1}^{N} \xi_i^2$

Subject to

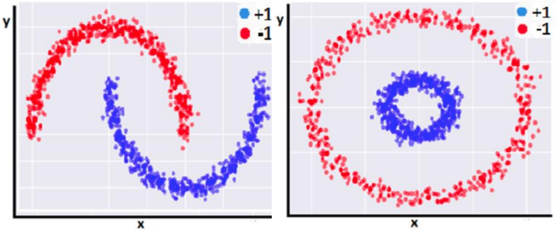
$$\begin{array}{ll} \xi_i \geq 0 & \\ \langle w \cdot x \rangle + b > 1 - \xi_i, & \text{if } y_i = 1 \\ \langle w \cdot x \rangle + b < -1 + \xi_i, & \text{if } y_i = -1 \end{array}$$

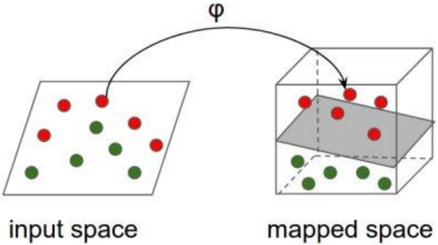
- Constraints include requirement that error terms are non-negative
- Again the objective function is quadratic



The non-linear case

- How do we deal with datasets where the separator is a complex shape?
- Geometrically transform the data
  - Typically, add dimensions
- For instance, if we can "lift" one class, we can find a planar separator between levels



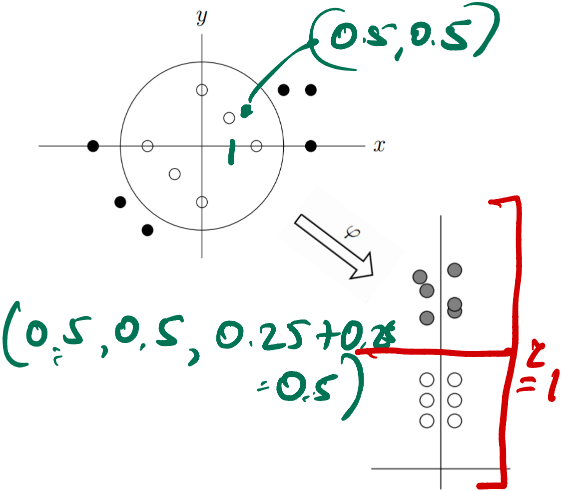


Geometric tranformation

- Consider two sets of points separated by a circle of radius 1
- Equation of circle is $x^2 + y^2 = 1$
- Points inside the circle $x^2 + y^2 < 1$
- Points outside circle $x^2 + y^2 > 1$
- Transformation

$$\varphi : (x, y) \mapsto (x, y, \underline{x^2 + y^2})$$

- Points inside circle lie below z = 1
- Point outside circle lifted above z = 1



SVM after transformation

- SVM in original space

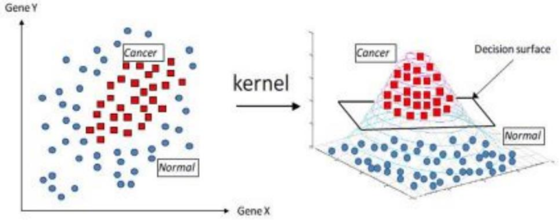
unknown input to classify

$$\operatorname{sign}\left[\sum_{i\in sv} y_i\alpha_i\langle x_i\cdot z\rangle + b\right]$$

soln

- After transformation

$$\operatorname{sign}\left[\sum_{i\in sv'} y_i\alpha_i\langle \varphi(x_i)\cdot \varphi(z)\rangle + b\right]$$



- All we need to know is how to compute dot products in transformed space

Transform: $x \longmapsto \varphi(x)$

Dot products

$\varphi(z) = (1, \sqrt{2}z_1, \sqrt{2}z_2, z_1^2, \sqrt{2}z_1z_2, z_2^2)$

- Consider the transformation

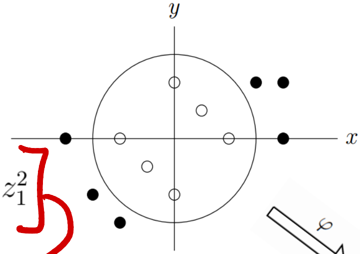
$$\varphi : \underset{2}{(x_1, x_2)} \mapsto (1, \sqrt{2}x_1, \underset{6}{\sqrt{2}x_2}, x_1^2, \sqrt{2}x_1x_2, x_2^2)$$

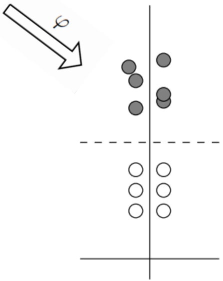
- Dot product in transformed space

$$\begin{aligned} \langle \varphi(x) \cdot \varphi(z) \rangle &= 1 + 2x_1z_1 + 2x_2z_2 + x_1^2z_1^2 \\ &\quad + 2x_1x_2z_1z_2 + x_2^2z_2^2 \\ &= (1 + x_1z_1 + x_2z_2)^2 \end{aligned}$$

$x = \langle x_1, x_2 \rangle$

$z = \langle z_1, z_2 \rangle$





$\langle \varphi(x) \cdot \varphi(z) \rangle$

in terms of x, z

- Transformed dot product can be expressed in terms of original inputs

$$\langle \varphi(x) \cdot \varphi(z) \rangle = K(x, z) = (1 + x_1z_1 + x_2z_2)^2$$

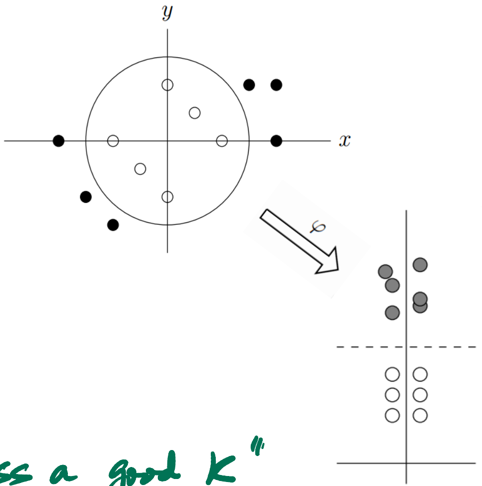
Kernels

- K is a *kernel* for transformation φ if

$$K(x,z) = \langle \varphi(x) \cdot \varphi(z) \rangle$$

- If we have a kernel, we don't need to explicitly compute transformed points
- All dot products can be computed implicitly using the kernel on original data points

$$\mathrm{sign}\left[\sum_{i \in sv'} y_i \alpha_i \underbrace{\langle \varphi(x_i) \cdot \varphi(z) \rangle}_{K(x,z)} + b \right]$$



"Guess a good K"

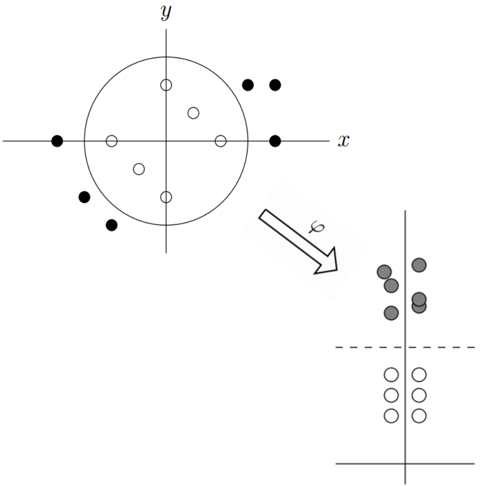
Kernels

- K is a *kernel* for transformation φ if

$$K(x,z) = \langle \varphi(x) \cdot \varphi(z) \rangle$$

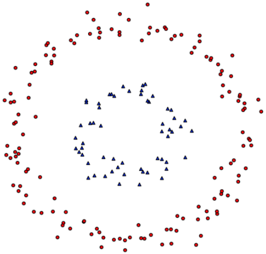
- If we have a kernel, we don't need to explicitly compute transformed points
- All dot products can be computed implicitly using the kernel on original data points

$$\text{sign}\left[\sum_{i \in sv'} y_i \alpha_i K(x_i, z) + b\right]$$



Kernels

- If we know K is a kernel for some transformation φ, we can blindly use K without even knowing what φ looks like!
- When is a function a valid kernel?
- Has been studied in mathematics – Mercer's Theorem
    - Criteria are non-constructive
- Can define sufficient conditions from linear algebra

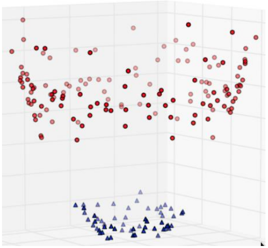


$x \mapsto \varphi(x)$

$z \mapsto \varphi(z)$

$\varphi(x) \cdot \varphi(z)$

some expn in x & z



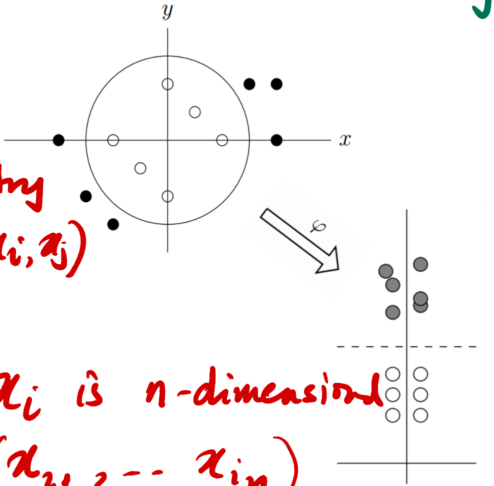
Kernels

If K is some kind of ~~distance~~ function
similarity

- Kernel over training data $x_1, x_2, \ldots, x_N$ can be represented as a *gram matrix*

$$K = \begin{array}{c c} & \begin{array}{cccc} x_1 & x_2 & \cdots & x_N \end{array} \\ \begin{array}{c} x_1 \\ x_2 \\ \vdots \\ x_n \end{array} & \left[\begin{array}{cccc} & & & \\ & & & \\ & & & \\ & & & \end{array} \right] \end{array}$$

(i,j) entry is $K(x_i, x_j)$



- Entries are values $K(x_i, x_j)$
- Gram matrix should be *positive semi-definite* for all $x_1, x_2, \ldots, x_N$

x_i is n-dimensional
$(x_{i1}, \ldots, x_{in})$

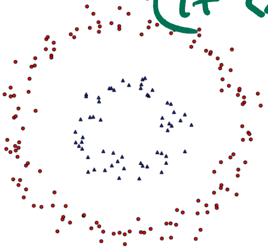
Known kernels

$$(1 + \underline{x_1 z_1 + x_2 z_2})^2$$

$$(1 + \langle x \cdot z \rangle)^2$$

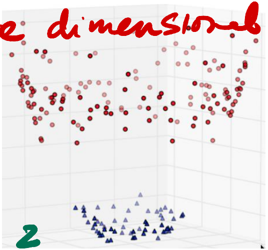
- Fortunately, there are many known kernels
- Polynomial kernels

$$K(x,z) = (1 + \langle x \cdot z \rangle)^k$$



- Any K(x,z) representing a similarity measure
- Gaussian radial basis function – similarity based on inverse exponential distance

— Theoretically, infinite dimensional



$$K(x,z) = \boxed{e^{-c|x-z|^2}}$$

$|x - z|$

distance from x to z

SVM + Kernels

Disadvantage: Manually discover good kernels

Advantage: Good kernels work "very well"

Best known models till $\approx$ 2010

Modern times

Neural networks have taken over

"Finding a kernel" is done automatically