AIML - CS 337

Lecture 12: Kernel Methods

26/09/2022

Lecturer: Prof. Abir De *Scribe: Parshant, Shikhar Mundra, Sai Kiran, Meghana*

1 Introduction

For Support vector classification to work, the data needs to be linearly separable. When the data in the original space is not linearly separable, we transform the data into a higher dimensional space. The aim is that the classes become linearly separable in the higher dimensional space.

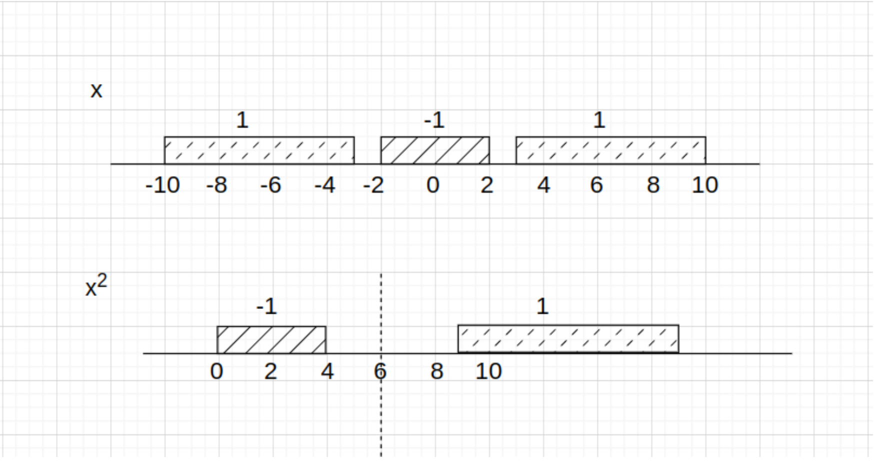
In kernel methods we represent data using pairwise similarity comparisons instead of applying transformation. Using this trick we can find the right decision boundary for the classes without having to calculate the actual transformation using only its dot product.

2 Transformation

The expressive power of linear separator is rather restricted – for example, the following training set is not separable by a linear function. Let the domain be the real line; consider the domain points $\{-10, -9, -8, . . . , 0, 1, ... , 9, 10\}$ where the labels are +1 for all x such that |x| > 2 and -1 otherwise. To classify this data we can first map the original instance space into another space (possibly of a higher dimension) and then learn a linear model (halfspace) in that space.

let us first define a mapping $\phi \ : \ R \hookrightarrow R^2$ as follows:

$$\phi(x) \ = \ (x, x^2)$$



Now this is linearaly separable with $h(x) = sgn(\boldsymbol{w}^T\phi(x) - b)$ with w = (0,1) and b = 6 (*Check*)

Note that dimension after transformation can be **infinity**, this is theoretically possible for non linear functions but it is practically hard to code, thus we try to calculate $w^T\phi(x)$ directly.
We can define a general loss function

$$L(w) \ = \ f(\{y_i\}, \{w^T\phi(x_i)\}_i) + \lambda R(w)$$

here R can be any monotonic norm

Theorem 2.1. *Representer Theorem: Assume that ϕ is a mapping on X, Then, there exists a vector $\alpha \in R^m$ such that $w = \sum_{i=1}^{m} \alpha_i \phi(x_i)$ is an optimal solution of L(w)*

Proof. Let w^* be an optimal solution of L(w). Because w^* is an element of a transformed(Hilbert) space, we can rewrite w^* as

$$w^* \ = \ \sum_i^m \alpha_i \phi(x_i) + u$$

where $< u, \phi(x_i) > = \ 0$ for all i. Set $w \ = w^* - u$ then $||w^*||^2 \ = \ ||w||^2 \ + \ ||u||^2$.
Since R is monotonic (increasing) $R(w) \leq R(w^*)$. Also observe that

$$< w, \phi(x_i) > = < w^* - u, \phi(x_i) > = < w^*, \phi(x_i) >$$

We have shown that the objective L at w cannot be larger than the objective at w^* and therefore w is also an optimal solution. Since

$$w = \sum_{i=1}^{m} \alpha_i \phi(x_i)$$

we conclude our proof.

□

2.1 Implications of this result

- This representation takes us beyond SVM
- w is linear combination of $\phi(x_i)$ thus $w^T\phi(x)$ can be easily calculated.

3 Kernels

Lets try to calculate what we started with, the predictor

$$w^{*T}\phi(x) \ = \ \textstyle\sum_{i=1}^{N} \alpha_i \phi(x_i)^T \phi(x)$$

What does this expression physically mean? We want to make a prediction at a new test point x, then $w^{*T}\phi(x)\ y(x)$ gives the **weighted mean** of labels $y(x_i)$ with weights as similarity

$$S(x, x_i) \ = \ \phi(x_i)^T\phi(x)$$

thus giving more weight to similar neighbours. This essentially represents the idea of K-Means clustering.

We can note that this similarity function can be any form, not necessarily dot product, The term "kernels" is used in this context to describe inner products in the feature space

$$K(\boldsymbol{x}, \boldsymbol{x}^{'}) \ = < \phi(\boldsymbol{x}), \phi(\boldsymbol{x}^{'}) >$$

$$w^{*T}\phi(\boldsymbol{x}) \ = \ \textstyle\sum_{i=1}^{N} \alpha_i K(\boldsymbol{x_i}, \boldsymbol{x})$$

Here we have finite (N^2) terms thus we have dealt with the problem of infinite dimensions.

It turns out that many learning algorithms can be carried out just on the basis of the values of the kernel function over pairs of domain points. Such algorithms implement linear separators in high dimensional feature spaces without having to specify points in that space or expressing the embedding ϕ explicitly.

There are many different choice of kernels, another possibility is **RBF Kernel** also known as Gaussian kernel.

$$K(\boldsymbol{x}, \boldsymbol{x}^{'}) \ = \ e^{\frac{-||\boldsymbol{x}-\boldsymbol{x}^{'}||_2^2}{2\sigma^2}}$$

Or polynomial kernels of degree k

$$K(\boldsymbol{x}, \boldsymbol{x}^{'}) \ = \ (1 + \ < \boldsymbol{x}, \boldsymbol{x}^{'} >)^k$$

3.1 SVM rewritten in terms of kernels

$$L(\boldsymbol{w}) = \lambda||\boldsymbol{w}||^2 + \sum_i (1 - y_i \boldsymbol{w}^T \phi(\boldsymbol{x_i}))_+$$

$$L(\boldsymbol{w}) = \lambda < \boldsymbol{w}, \boldsymbol{w} > + \sum_i (1 - y_i < \boldsymbol{w}, \phi(\boldsymbol{x_i}) >)_+$$

$$L(\boldsymbol{w}) = \lambda < \sum_j \alpha_j \phi(\boldsymbol{x_j}), \sum_i \alpha_i \phi(\boldsymbol{x_i}) > + \sum_i (1 - y_i < \sum_j \alpha_j \phi(\boldsymbol{x_j}), \phi(\boldsymbol{x_i}) >)_+$$

$$L(\boldsymbol{w}) = \lambda \sum_j \sum_i \alpha_j \alpha_i < \phi(\boldsymbol{x_j}), \phi(\boldsymbol{x_i}) > + \sum_i (1 - y_i \sum_j \alpha_j < \phi(\boldsymbol{x_j}), \phi(\boldsymbol{x_i}) >)_+$$

$$L(\boldsymbol{w}) = \lambda \sum_{i,j} \alpha_i \alpha_j K(\boldsymbol{x_j}, \boldsymbol{x_i}) + \sum_i (1 - y_i \sum_j \alpha_j K(\boldsymbol{x_j}, \boldsymbol{x_i}))_+$$

$$L(\boldsymbol{w}) = \lambda \alpha^T G \alpha + \sum_i (1 - y_i (G\alpha)_i)_+$$

Where G is Gram Matrix, $G_{ij} = K(x_i, x_j)$

Theorem 3.1. *A symmetric kernel K : X × X ⟶ R implements an inner product in some Hilbert space if and only if it is positive semidefinite; namely, for all $x_1, \ . \ . \ ., \ x_m$ the Gram matrix, $G_{i,j} = K(x_i, x_j)$, is a positive semidefinite matrix.*

K is said to be non-negative definite (or positive semidefinite) if and only if

$$\sum_i \sum_j c_i c_j K(x_i, x_j) \geq 0$$

for all finite sequences of points $x_1, ..., x_n$ all choices of real numbers $c_1, ..., c_n$

3.2 Excercise

3.2.1 Show that RBF can be written as inner product

It is easy to see that RBF(or Gaussian) Kernel satisfies the positive semidefinite condition, we can infact state and prove a possible inner product representation

Consider

$$\phi(x) \ = \ \sum_{n=0}^{\infty} \frac{1}{\sqrt{n!}} e^{-\frac{x^2}{2}} x^n$$

Observe that

$$< \phi(x), \phi(x^{'}) > = \ \sum_{n=0}^{\infty} \left(\frac{1}{\sqrt{n!}} e^{-\frac{x^2}{2}} x^n \right) \left(\frac{1}{\sqrt{n!}} e^{-\frac{x^{'2}}{2}} x^{'n} \right)$$

$$< \phi(x), \phi(x^{'}) > = \; e^{-\frac{x^2+x'^2}{2}} \sum_{n=0}^{\infty} \left(\frac{(xx^{'})^n}{n!} x^n \right)$$

$$< \phi(x), \phi(x^{'}) > = \; e^{-\frac{x^2+x'^2}{2} + xx^{'}}$$

$$< \phi(x), \phi(x^{'}) > = \; e^{-\frac{(x-x')^2}{2}}$$

3.2.2 Do the same for polynomial kernel

Left for reader
Solution can be checked in book Understanding machine learning chapter Kernel Methods