Examplar-based and Kernel Methods in Machine Learning

Sunita Sarawagi

September 13, 2023

Introduction to Examplar/Kernel Methods

- A nonparametric method where the prediction function does not have a fixed functional form but instead estimated directly from the data.
- Given training data $D = \{(\mathbf{x}^i, y^i) : i = 1 \dots N\}$ are observed N points of an unknown function $f(\mathbf{x})$, make prediction purely based on **similarity** of a text point $\mathbf{x}^*$ to training points. Need to keep around training data.
- Similarity measured as a kernel function $k(\mathbf{x}^*, \mathbf{x}^i)$ between any two points.
- Equivalently dissimilarity between two points specified in terms of a distance function $d(\mathbf{x}^*, \mathbf{x}^i)$
- No need to know coordinates of points in any fixed dimensional space.

Nearest neighbor classifier

Simplest kind of classifier. Given a R : # of nbrs :

Given training data: $D = \{(x^{i}, y^{i}) : i = 1 \text{ to } N\}$, a distance function $d(x, x') \to \mathbb{R}$.

x^{i} can be anything

No training:

During test-time:

Given a test example x^*

Find R nearest neighbors of x^* from D :

$Nbr(x^*) = \{(x^{i_1}, y^{i_1}) \ldots (x^{i_R}, y^{i_R})\}$

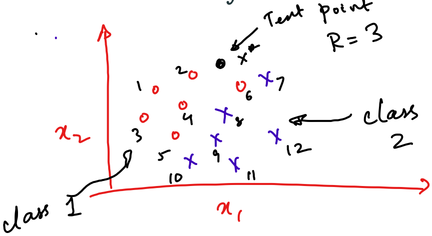
s.t $d(x^*, x^{i_j})$ is among the R shortest among all examples in D.

$$P(y = k \mid x^*) = \frac{1}{R} \sum_{j=1}^{R} \delta(y^{i_j} == k)$$

← fraction of examples in the nearest neighbors that are of class 'k'.

Example

https://drive.google.com/file/d/
1LihEFIe5fyFbWPS342PA9NZdy2Kxjkvy/view?usp=sharing



$$\text{R-Nbr}(\overset{x^*}{\bullet}) = \{(\overset{1}{2}, \overset{1}{6}, \overset{2}{7})\}$$

$$P(y=1 \mid x^*) = \frac{2}{3}$$

$$P(y=2 \mid x^*) = \frac{1}{3}$$

Limitations

1. KNN does not work well for high-dimensioanl data. As dimension increases, most points become equally far from each other.
2. Need to design a meaningful distance measure for KNN to be useful. Lots of work on metric learning.
3. Need to store the training data, and retrieve during deployment. Slow.

Application: Few-shot classification

Classification with dynamically changing class labels. Do not need to decide on the set of classes during training time.

Learning distance functions

Many methods have been proposed.

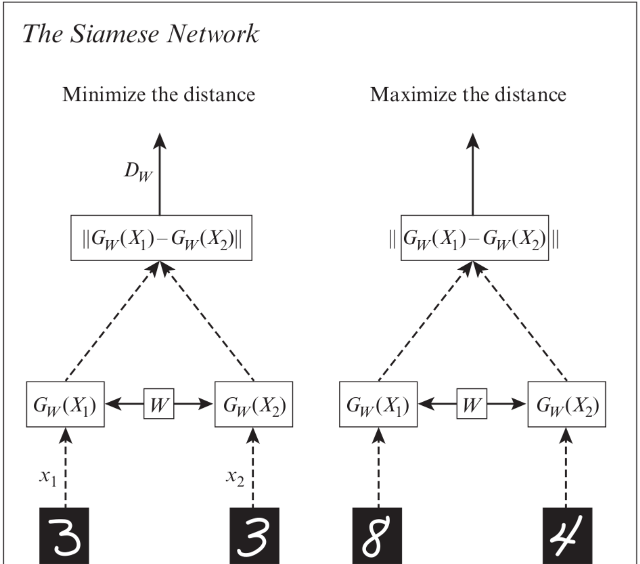
Assume we have supervision on pairs of objects that are similar and dissimilar.

Learn a distance function that brings together similar objects and keeps apart dissimilar objects.

$$L(\mathbf{x}^i, \mathbf{x}^j) = \delta(y_i = y_j) d(\mathbf{x}^i, \mathbf{x}^j) + \delta(y_i \neq y_j) \max(0, m - d(\mathbf{x}^i, \mathbf{x}^j))$$

Many different models proposed in traditional and deep learning methods for representing distances.

Siamese networks for learning distance functions

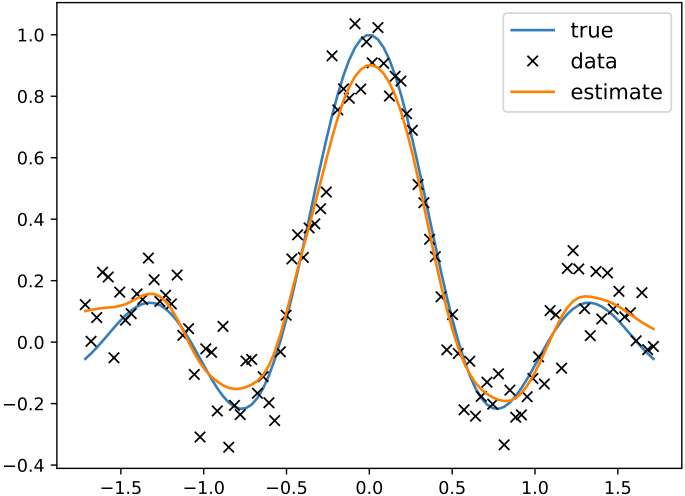


Kernel Regression

Predict real-values based on similarity weighted values of training examples.

$f(\mathbf{x}|D) = \sum_{i=1}^{N} \frac{K(\mathbf{x},\mathbf{x}^i)}{\sum_{j=1}^{N} K(\mathbf{x},\mathbf{x}_j)} y_i$

Kernel regression example



Mercer's Kernel function

The (Mercer's) kernel function $K : \mathcal{X} \times \mathcal{X} \mapsto R$ is a symmetric function such that for any set of N points and any choice of numbers $c_i \in R$

$$\sum_{i=1}^{N} \sum_{j=1}^{N} K(\mathbf{x}^i, \mathbf{x}^j) c_i c_j \geq 0$$

Equivalent to matrix of kernel values (Gram) matrix being positive semi-definite.

Commonly Used Kernels

- Linear Kernel: $K(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T\mathbf{x}'$
- Polynomial Kernel: $K(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T\mathbf{x}' + 1)^d$
- Radial Basis Function (RBF) or Gaussian Kernel:
  $K(\mathbf{x}, \mathbf{x}') = \exp(-\gamma||\mathbf{x} - \mathbf{x}'||^2)$
- Sigmoid Kernel: $K(x, y) = \tanh(\alpha x^T y + c)$

Kernel embedding

For every valid Kernel, there exists an embedding function $\phi(\mathbf{x})$ such that $K(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x}).\phi(\mathbf{x}')$ (Mercer's theorem)

(1cm) Example: Quadratic Kernel: $K(\mathbf{x}, \mathbf{x}') = (\mathbf{x}.\mathbf{x}')^2$.

If original data is 2-d, then $K(\mathbf{x}, \mathbf{x}') = (x_1 x_1' + x_2 x_2')^2 =$

$\phi(\mathbf{x}) = [x_1^2, \sqrt{2} x_1 x_2, x_2^2]$

Applications in Machine Learning

- Support Vector Machines (SVM)
- Kernel Principal Component Analysis (PCA)
- Gaussian Processes
- Radial Basis Function Networks (RBFN)

Support Vector Machines

Non-probabilistic models for classification and regression of the form:

$$f(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i K(\mathbf{x}, \mathbf{x}^i)$$

Learning fixes values of α_i. These are chosen in such a way that only a few *i*s have non-zero α_i. These are called support vectors.

Kernel Trick

The kernel trick allows for operations in the high-dimensional space without explicitly computing the coordinates in that space.

Given a kernel K, the trick involves replacing the dot product with the kernel function.

Summary

- Kernel methods transform data to higher dimensions for linear separability.
- Various kernels are available, each with its own characteristics.
- Kernel methods find applications in several ML algorithms.