Machine Learning
September 21, 2016

News of the day: solar industry uses machine learning to figure out where to install solar panels

Soft margins- not as strict with keeping as far apart as possible. Epsilon for each point…optimization problem becomes still minimizing w (normal to plane). Parameter C controls how soft do you want margins to be. If 0, want it to be very hard, and the larger it is the softer it is. Kernel defines the plane.

Most common kernels: linear, polynomial, RBF (Gaussian)

"Kernel trick": avoids the explicit mapping needed to get linear learning algorithms (find a line) to learn a nonlinear function or decision boundary. The word kernel is used in mathematics to denote a weighting function for a weighted sum or integral (distance). For all x and x' in input space X, certain functions k(x,x') can be expressed as an inner product in another space V. Function k does X x X -> R (real numbers).

Most popular kernel is default is the RBF (radial basis function), also called Gaussian. $K(x,x') = \exp(-||x-x'||^2 / 2\sigma^2)$ people like calling the sigma as gamma and then do a variable change: $K(x,x') = \exp(-\gamma || x- x'||^2)$ – so what is the variable gamma for a specific dataset? Variable C tells us how soft margin is…now also has variable gamma for RBF…have only 2 variables to have to play around with. Large gamma (small sigma) amplifies fact they are close… close by points very close and are significant to the algorithm… so sort of like a clustering

Don't let pandas become a crutch for loading in data… load_digits() python dictionary.

In remote sensing, OCR for example if you have 8x 8 you concatenate each row of pixels from top to bottom to get a 1x64 row…if do exactly the same manipulation of each of these 1x8 rows (column switching variance), support vector machine works. No sense of order, just shift weights. convolution neural networks are built ad hoc and are different but everything else is invariant for these manipulations.

SVC support vector classifier, SVR: support vector regression (basically same concept but find margin that passes through all the points)

If you have cancer patients and 2/100 have a certain type and asked to build classifier for detecting that type… if only get paid to detect if accuracy is >90%, then could just say no cancer for all classifier because would be right 98% time…which would be bad (unbalanced dataset- have few examples of one class)! So that's why accuracy is not necessary best metric. It also depends on amount of classifiers (if 50% accuracy and only 2 classes, youre just guessing…but if 3 classifiers, 50% accuracy is better)…have to explore your data to know if its unbalanced. Need to know your data.

Get very different accuracy values for different kernels, different gamma values, etc. don't want to check by hand… so do a grid search (an optimization problem) of values to test for different kernels and then tells you which kernel and gamma gives best accuracy. Grid search is brute force so computationally exhaustive.