MCP: neuron

Perceptron in a binary classification problem:

Equivalent form

Redefine as

Then

This is actually the same as using a straight line (linear decision boundary) to separate data points.

Update rule: (why this form???)

Note that this is done for each example, one by one.

The perceptron is a binary classification model. To extend it to multi-classification, we relabel the data as ‘Class1’ and ‘Not Class1’ and use a perceptron. Then we repeat this for all other classes. Finally, we get 10 trained perceptrons. Given a new example, we collect the net inputs of all perceptrons. We assign the label with the highest net input to the example. The whole procedure is called OvA (one-versus-all) or OvR (one-versus-rest).

Adaline:

Adaline: adaptive linear neuron (why it is also called a single-layer neural network?)

A key difference between Adaline and perceptron:

--Adaline: it compares the output of the activation function with the true label, to update the parameters

--Perceptron: it compares the predicted label with the true label.

Loss function in Adaline:

is minimized by gradient descent.

Note that Adaline is merely linear regression used to solve a classification problem. We know that this is not a good choice. The logistic regression is much better.

Gradient of :

Feature scaling:

Stochastic Gradient Descent

The original gradient descent:

In stochastic gradient descent, we use one training example at a time:

We repeat this for all

We also use an adaptive learning rate:

We also shuffle the training set for every epoch.