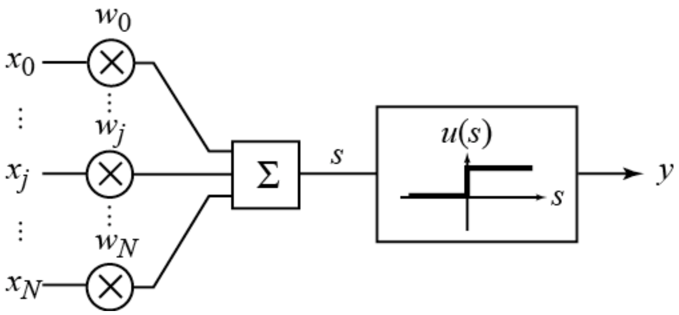
TOOL: Proportional Increment Training (PIT) is a learning algorithm for updating synaptic weight w_j in a Perceptron, (shown below).



The (PIT) equation for weight updates is a modified form of gradient descent that treats the step-function squashing function as the identity function $y = s$ when calculating a derivative:

$$\Delta w_j = \eta(y_d - y)x_j$$

DERIVE: The derivation proceeds initially as in modified gradient descent learning.

$$\Delta w_j = -\eta \frac{\partial E}{\partial w_j}$$

where $E \equiv \frac{1}{2}(y_d - y)^2$

Part way through the derivation, however, we replace the derivative of step function, $u(s)$, (which would be zero almost everywhere), with the derivative of an identity squashing function, $y = s$, which is unity.

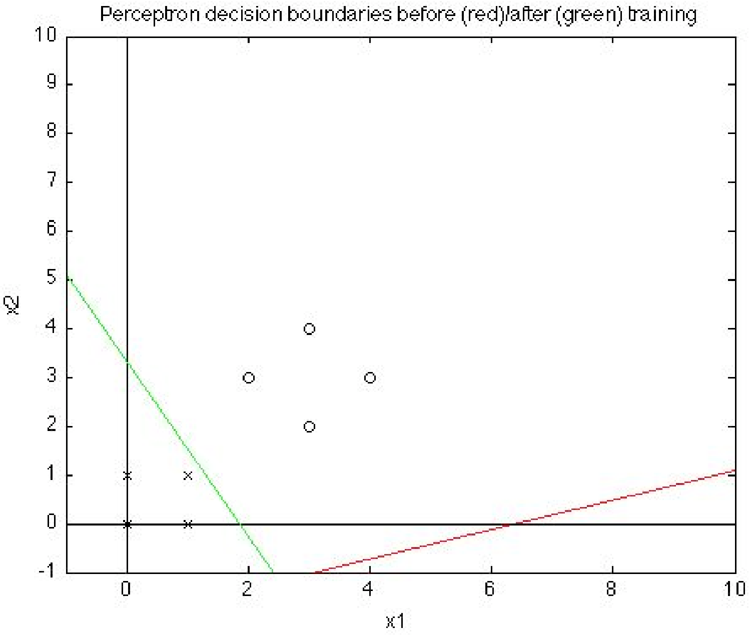
$$-\eta \frac{\partial E}{\partial w_j} = -\eta \frac{\partial E}{\partial err} \frac{\partial err}{\partial y} \frac{\partial s}{\partial s} \frac{\partial s}{\partial w_j} = -\eta(err)(-1)(1)(x_j) = \eta(y_d - y)x_j$$

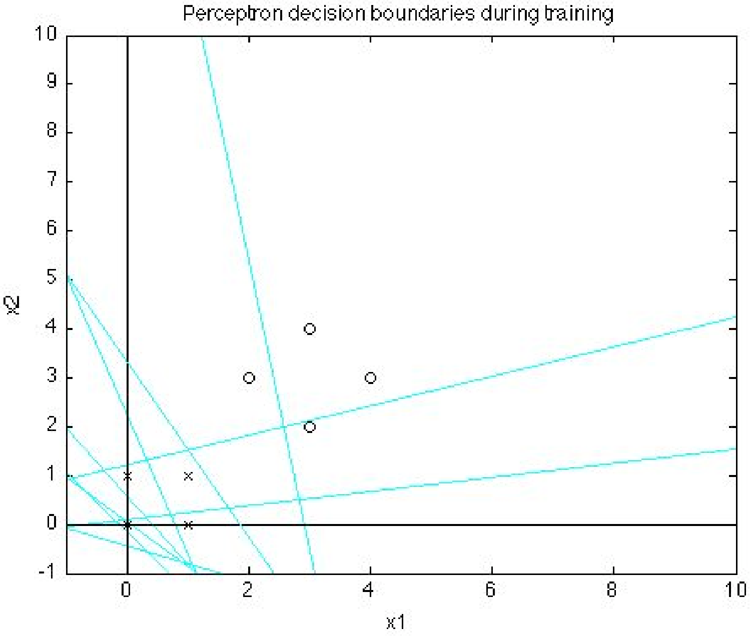
where $err \equiv y_d - y$

NOTE: Typically, $\eta = 1$ is used.

Initial weight values are chosen at random in the PIT algorithm, and if the training patterns are linearly separable, the PIT algorithm will converge after a finite number of weight updates.

The first figure below shows the decision boundary (where the perceptron output changes from 0 to 1) before and after training to distinguish zeros (o's) and ones (x's). The second figure shows decision boundaries during training. Note that the final decision boundary appears in the second figure, and once the training pattern is in correct position, weight updates cease because errors are zero.


Perceptron decision boundaries before (red)/after (green) training
x2
x1


Perceptron decision boundaries during training
x2
x1