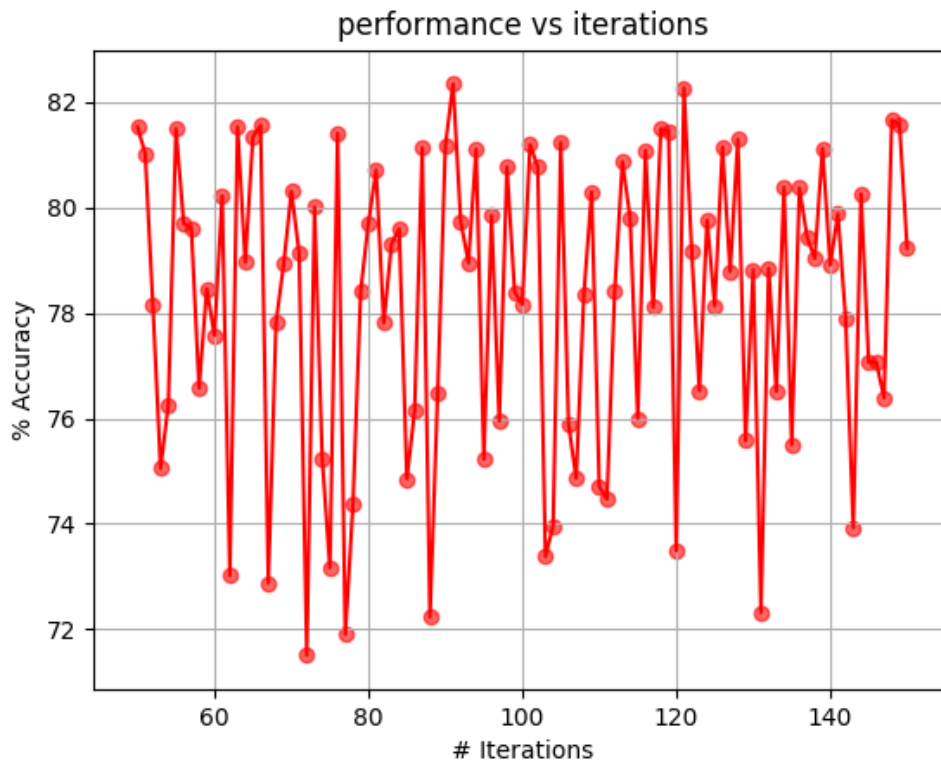


ReadMe

Perceptron algorithm is a linear classifier which makes progress by using stochastic gradient descent. Based on the change of the number of iterations, the performance (accuracy) of the perceptron algorithm on development set goes as follows:



It is obvious that the accuracy is oscillated (with high variance) in vertical direction even when the number of iterations is pretty big. Here is my interpretation of this result: we apply stochastic gradient descent to calculate the derivative of weights and make progress. The advantage of SGD is that it can start making progress right away, and continues to make progress with each example it looks at, whereas batch gradient descent has to scan through the entire training set before making a single step – a costly operation if the size of training set is large. Besides, SGD often gets close the minimum much faster than batch gradient descent. However, SGD may never converge to the minimum because it makes every step just according to the current single example and this example may contain some noise which makes SGD often ‘overshooting’, that’s why the plot oscillates vertically.

There are some approaches to improve SGD and makes it even faster. For instance, we can adopt SGD with momentum to dampen oscillations. Besides, RMSprop is also a good improvement of SGD which divides the learning rate by an exponentially decaying average of squared gradient.