
Homework 2

Kevin Lei

Department of Computer Science and Engineering
Texas A&M University
College Station, TX 77843
kevinlei@tamu.edu

1 Tweaking Epoch Count and Learning Rate

The following are the decision boundaries plotted with the training data for variations on epoch and learning rate.

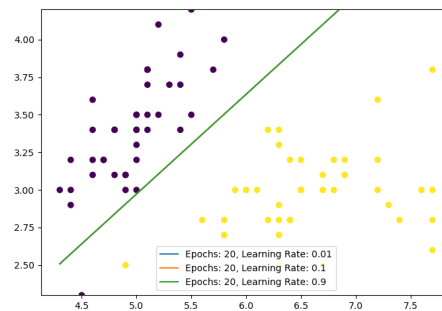
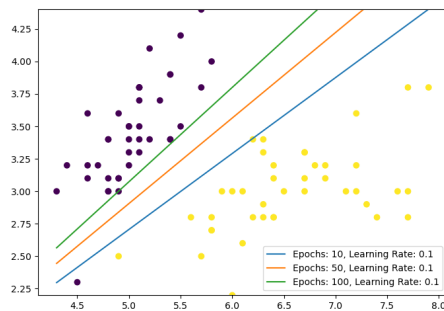


Figure 1: Perceptron with variable epochs. Figure 2: Perceptron with variable learning rate.

As we can see from Figure 1, having a suboptimal number of epochs can lead to a decision boundary that does not accurately separate the two classes. Compared to the initial settings (1000 epochs and 0.01 learning rate), the decision boundaries produced by 10, 50, and 100 epochs are not as accurate. Additionally, notice that the decision boundary produced by 50 epochs seems to have the best margin of separation between the two classes, compared to the other epoch values. This suggests that for a fixed learning rate, having too many or too few epochs can lead to a suboptimal decision boundary. Theoretically, less epochs would lead to a decision boundary that is not as accurate, while more epochs would lead to a decision boundary that is overfit to the training data.

Speaking in terms of learning rate, Figure 2 shows that changing the learning rate between 0.01, 0.1, and 0.9 does not produce significant differences in the decision boundary. However, that is not to say that learning rate is generally not an important hyperparameter. For more complex datasets, we may see a more significant impact of learning rate on the decision boundary.

2 Error Accumulation

The following is a graph of the accumulated number of misclassifications over all the epochs for the perceptron algorithm.

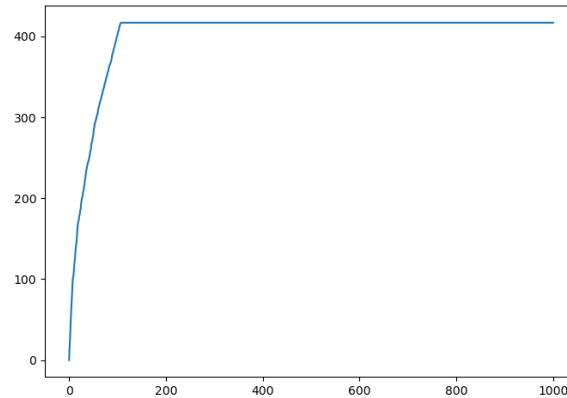


Figure 3: Accumulated number of misclassifications over all epochs.

As we can see from the graph, the number of misclassifications converges to a value slightly over 400 after about 100 epochs with a learning rate of 0.01. Also, the rate of change in the number of misclassifications decreases as the number of epochs increases. This suggests that the perceptron algorithm is converging to a solution that is not perfect, but is close to the optimal solution.

3 Linearly Inseparable Data

Using Versicolor instead of Setosa, the following is the decision boundary plotted with the training data.

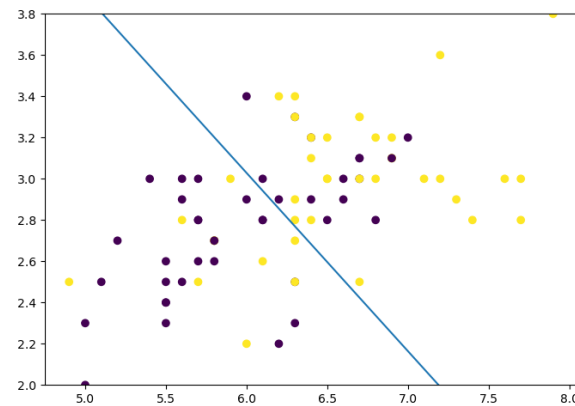


Figure 4: Perceptron with Versicolor instead of Setosa.

As we can see from the graph, the perceptron algorithm is unable to converge to a solution that separates the two classes due to the fact that there is no linear decision boundary that can separate the two classes. The points seem to be scattered within each other, which makes it impossible for a single perceptron to separate the two classes.