Fall 2016 CPS 571/STA 561: Homework 1

Duke University

August 31, 2016

1 Perceptron Algorithm

Consider applying the perceptron algorithm (through the origin) based on a small training set containing three points: $\mathbf{x}_1 = [-1, 1]$, $\mathbf{x}_2 = [0, -1]$, and $\mathbf{x}_3 = [1.5, 1]$ with labels $y_1 = 1$, $y_2 = -1$, and $y_3 = 1$. Given that the algorithm starts with $\mathbf{w} \equiv \mathbf{0}$, the first point that the algorithm sees is always considered a mistake. The algorithm starts with one data point and then cycles through the data until it makes no further mistakes.

(a) How many mistakes does the algorithm make until convergence if the algorithm starts with data point $\mathbf{x}_1$? How many mistakes does the algorithm make if it starts with data point $\mathbf{x}_2$? Draw a diagram showing the progression of the plane as the algorithm cycles.

(b) Now assume that $\mathbf{x}_3 = [10, 1]$. How many mistakes does the algorithm make until convergence if cycling starts with data point $\mathbf{x}_1$? How many if it starts with data point $\mathbf{x}_2$? Draw a diagram showing the progression of the plane as the algorithm cycles. You can draw it on a piece of paper and take a picture of the picture with your phone or you could do it in powerpoint or another drawing software.

(c) Explain why there is a difference between the number of mistakes made by the algorithm in part (a) and part (b). In particular, explain why there is a difference between the number of mistakes when the algorithm starts with $\mathbf{x}_1$ and when it starts with $\mathbf{x}_2$, and explain why there is a difference in the number of mistakes when $\mathbf{x}_3$ is moved from $[1.5, 1]$ to $[10, 1]$.

(d) Briefly describe an adversarial procedure for selecting the order of labeled data points so as to maximize the number of mistakes the perceptron algorithm makes before converging. Assume that the data is indeed linearly separable, by a hyperplane that you (but not the algorithm) know.

2 Handwritten Digits Classification

Machine vision is a key problem in machine learning. When you send a check into an ATM machine or a letter through the post office, a machine learning algorithm reads your handwriting. Let's use the popular MNIST database to tell the difference between 0's and 1's, using the perceptron algorithm.

Download the MNIST dataset[1]. Each observation is an image. Filter out all observations that are not labeled '0' or '1'. Program your own version of the perceptron algorithm to create a classifier for 0's and 1's. Report the accuracy of the classifier.

Come up with your own feature reduction techniques - how many features can you reduce while maintaining similar accuracy? Describe your new features. Why might these features be the only ones you need to keep?

[1]Matlab users can refer `mnist_all.mat`, others can use the `mnist.zip` in `.txt` format or download the raw dataset from `http://yann.lecun.com/exdb/mnist/`.