

HW03: Perceptron

Hand in via moodle at: <https://moodle.umass.edu/course/view.php?id=20836>. Remember that only PDF submissions are accepted. We encourage using L^AT_EX to produce your writeups. See `hw00.tex` for an example of how to do so. You can make a `.pdf` out of the `.tex` by running “`pdflatex hw00.tex`”. You’ll need `mydefs.sty` and `notes.sty` which can be downloaded from the course page.

1. A common way to get rid of having to deal with the threshold separately on a perceptron is to add a new feature. This feature always has value one, and you learn a weight for it. Thus, if you have a 100 dimensional problem with a threshold, we solve it as a 101 dimensional problem with threshold at zero. Draw a picture for one dimensional data and a linear separator with a (non-zero) threshold and draw the corresponding picture for the same data, “lifted” into two dimensions, with the corresponding linear separator with threshold zero. (Please make sure that the two separators are actually equivalent!)
2. The perceptron will only converge if the data is linearly separable. For linearly separable data with margin δ , if $\|\mathbf{x}\| \leq R$ for all data points, then it will converge in at most $\frac{R^2}{\delta^2}$ iterations (In class we assumed that $\|\mathbf{x}\| \leq 1$). It is possible to force your data to be linearly separable as follows. If you have N data points in D dimensions, map data point \mathbf{x}_n to the $(D + N)$ -dimensional point $(\mathbf{x}_n, \mathbf{e}_n)$, where \mathbf{e}_n is a N -dimensional vector of zeros, except for the n^{th} position, which is 1. (Eg., $\mathbf{e}_4 = (0, 0, 0, 1, 0, \dots)$.)
 - (a) Show that if you apply this mapping the data becomes linearly separable (you may wish to do so by providing a weight vector \mathbf{w} in $(D + N)$ -dimensional space that successfully separates the data).
 - (b) How long will it take the perceptron algorithm to converge on this augmented data?
 - (c) How does this mapping affect generalization (i.e., test performance)?
3. Why is averaging preferred over voting perceptron?