Homework 3

May 3, 2017

Problem 1. In this problem we will study the difficulty of back-propagation in training deep neural networks. For simplicity, we consider the simplest deep neural network: one with just a single neuron in each layer, where the output of the neuron in the jth layer is $z_j = \sigma(a_j) = \sigma(w_j z_{j-1} + b_j)$. Here σ is some activation function whose derivative on x is $\sigma'(x)$. Let m be the number of layers in the neural network, L the training loss.

1. Derive the derivative of L w.r.t. b_1 (the bias of the neuron in the first layer).
2. Assume the activation function is the usual sigmoid function $\sigma(x) = 1/(1+\exp\{-x\})$. The weights $\boldsymbol{w}$ are initialized to be $|w_j| < 1$ ($j = 1, \ldots, m$).
   (a) Explain why the above gradient ($\partial L/\partial b_1$) tends to vanish ($\to 0$) when m is large.
   (b) Even if $|w|$ is large, the above gradient would also tend to vanish, rather than explode ($\to \infty$). Explain why. (A rigorous proof is not required.)
3. One of the approaches to (partially) address the gradient vanishing/explosion problem is to use the rectified linear (ReL) activation function instead of the sigmoid. The ReL activation function is $\sigma(x) = \max\{0, x\}$. Explain why ReL can alleviate the gradient vanishing problem as faced by sigmoid.

Problem 2. To show a concept class H has VC dimension d, we need to prove both the lower bound $\mathrm{VCdim}(H) \geq d$ and the upper bound $\mathrm{VCdim}(H) \leq d$.
Show that linear classifiers $h(x) = \mathbf{1}_{\{\mathbf{a^T x + b \geq 0}\}}$ in $\mathbb{R}^n$ has VC dimension $n+1$.

Hint: the following theorem might be useful in proving the upper bound. A set of $n+2$ points in $\mathbb{R}^n$ can be partitioned into two disjoint subsets S_1 and S_2 such that their convex hulls intersect. The convex hull $\mathbf{conv(C)}$ of a set C is defined as the set of all convex combinations of points in C:

$$\mathbf{conv}(\mathbf{C}) = \{\sum_{\mathbf{i=1}}^{\mathbf{k}} \alpha_{\mathbf{i}} \mathbf{x_i} : \mathbf{x_i} \in \mathbf{C}, \alpha_{\mathbf{i}} \geq \mathbf{0}, \sum_{\mathbf{i=1}}^{\mathbf{k}} \alpha_{\mathbf{i}} = \mathbf{1}\}. \tag{1}$$

You do not need to know anything about convexity beyond this hint to solve this problem.

Problem 3. *Coding assignment.* You are required to build a typical MLP with 1 hidden layer in this task. The number of nodes in the hidden layer is your choice. Please use the data provided to build two different classifiers, one for distinguishing between **O** and **X**, the other for distinguishing between **O** and **D**. You are encourage to do feature selection instead of using all attributes provided. Please use the first 70% data as training set and set aside the last 30% as testing set. The details of the implementation and the classification accuracies (train and test) should be included in your report.