Machine Learning and Artificial Intelligence - COS 402

Written Homework Assignment 3

Due Date: one week from announcement in class, due in class

(1) Consulting other students from this course is allowed. In this case - clearly state whom you consulted with for each problem separately.

(2) Searching the internet or literature for solutions is NOT allowed.

(3) Submit your homework in separate pages for the different questions, each including your name and email address (this is to help the graders). Typing solutions up is strongly advised.

I (20 points) In this exercise we'll see how to train a simple 1-layer neural net with with a single sigmoid gate based on the cross-entropy loss function. Such a classifier (with parameter w) gets as input a feature vector x, and outputs $h_w(x) = \sigma(w^T x) = \frac{1}{1+e^{-w^T x}}$. The loss is measured according to the logistic loss which we now define.

Let $x \in \mathbb{R}^d, y \in \mathbb{R}$ be a feature vector and label. Consider the cross-entropy loss $\ell : \mathbb{R}^d \mapsto \mathbb{R}$ defined as:

$$\ell_{x,y}(w) = -y \log(h_w(x)) - (1 - y) \log(1 - h_w(x))$$

(a) (10 points) Prove that the cross-entropy loss function is convex (as a function of w). Note that $\sigma(x)$ is not a convex function.

(b) (5 points) Consider a given dataset of feature vectors and labels $\{(x_i, y_i)\}$ such that the norm of all feature vectors is bounded by one, i.e. $\|x_i\| \le 1$, and the labels are in $y_i \in \{0, 1\}$. Consider the optimization problem of finding the optimal 1-layer neural net classifier, with norm at most one, with respect to the cross-entropy, i.e.

$$\min_{\|w\| \le 1} \frac{1}{m} \sum_{i=1}^{m} \ell_{x_i,y_i}(w)$$

Note that the function above is convex since the sum of convex functions is convex. (Can you see why? There is no need to state a proof for this.) In part A, we saw that $l_{x,y}(w)$ is convex.

Compute the gradient of the objective function, and spell out the gradient descent algorithm for solving this optimization problem. Given an upper bound on the number of iterations to attain ε-precision in the solution. (Hint: You might want to invoke the theorem on lecture 5, slide 24.)

(c) (5 points) Spell out the stochastic gradient descent algorithm for this setting. Give an upper bound on the number of iterations to attain ε-precision in the solution. (Hint: You might want to invoke the theorem on lecture 6, slide 18.)

II (10 points) In this exercise, we'll see that a neural net with hidden layers can represent non-convex functions. Consider a neural net $h(x)$ with a single 2-node hidden layer and the threshold gate. The threshold gate (with parameter a) outputs $T_a(s) = 1$ if $s \geq a$ and $T_a(s) = 0$ otherwise. Identify a setting of weights and thresholds for this neural network so that $h(x)$ is non-convex.

III (10 points) Blabber is a popular Martian language consisting of 3 words – "Blah, Bah, Meh". Consider the following sentences from a text discovered recently by a Mars rover:

(a) *Blah Blah Bah Meh Blah Bah Blah Bah Meh.*

(b) *Meh Bah Bah Blah Meh Bah Bah.*

(c) *Bah Meh Blah Bah Meh Bah.*

We wish to answer the following questions:

(a) (3 points) Compute the unigram and bigram counts for this corpus. (There is no need to pad with START, STOP symbols.)

(b) (5 points) Compute the 3×3 table of conditional probability of a word occurring given the previous word – $P(w_i|w_{i-1})$– with Add-1 Laplace smoothing.

(c) (2 points) Using the conditional probabilities in the second part, compute the probability of the sentence "Blah Blah Blah Meh". Please express this answer on a log scale.

IV (10 points) (Optional) Redo Q2 with sigmoid output gates.