**CS 378** **Introduction to Data Mining** **Spring 2009**

## Homework 4

Instructor: Inderjit Dhillon  Date Due: April 2, 2009

**Keywords:** *Linear Regression*

1. (5 points) Based on the lecture notes on linear regression (available on the course website), let us consider the linear regression problem for the case when dimensionality of the data $d = 1$. Suppose we are given a training data $\{(x_i, y_i), i = 1, \ldots, N\}$ with $N$ observations, $x_i \in \mathbb{R}$ is the data point and $y_i \in \mathbb{R}$ is the target value associated with $x_i$. We use a linear function of the form $f(x) = w_0 + w_1 x$ to fit the data. Show that

$$w_0 = \bar{y} - w_1 \bar{x} \quad \text{and} \quad w_1 = \frac{\sigma_{xy}}{\sigma_{xx}},$$

where $\bar{x} = \frac{1}{N}\sum_{i=1}^N x_i$, $\bar{y} = \frac{1}{N}\sum_{i=1}^N y_i$, $\sigma_{xy} = \frac{1}{N}\sum_{i}^N (x_i - \bar{x})(y_i - \bar{y})$, and $\sigma_{xx} = \frac{1}{N}\sum_{i=1}^N (x_i - \bar{x})^2$.

2. (4 points) In this problem, we are going to investigate the time complexity of solving a least squares problem by normal equation with different number of data points $N$ and dimensionality $d$. Suppose the data points are organized as a matrix $X \in \mathbb{R}^{N \times (d+1)}$, where each row of $X$ corresponds to a data point, and the associated target values are organized as column vector $\boldsymbol{y} \in \mathbb{R}^N$. The normal equation to solve for the $(d+1)$-dimensional parameter vector $\boldsymbol{w}$ can be written as $X^T X \boldsymbol{w} = X^T \boldsymbol{y}$, which can be solved in Matlab using "\" operator. ($A\boldsymbol{x} = \boldsymbol{b}$ can be solved in Matlab by $\boldsymbol{x} = A\backslash\boldsymbol{b}$.) To measure the elapsed time, you can use the Matlab commands "tic" and "toc".

   (a) (2 points) Use the provided Matlab code genRegression.m to generate synthetic data for linear regression, where the number of data points is varied from 1,000 to 10,000 with a step size of 100 and $d$ is fixed to 100. Solve normal equation (by using "\") for each generated data set and report a plot showing the elapsed time (in seconds) for solving the problem vs the number of data points. (Your results should be averaged over 10 different runs.) DO NOT COMPUTE AN INVERSE.

   (b) (2 points) Repeat the above procedure while the number of data points is fixed to $N = 5000$ and $d$ is varied from 100 to 1000 with a step size of 50.

3. (6 points) You are given the Iris dataset (`http://www.cs.utexas.edu/~wtang/cs378/iris_reg.tar.gz`) for the linear regression problem, where the dataset has been split into training and test set. Let's consider the problem of predicting the petal width using a linear combination of sepal length, sepal width, and petal length. In another word, we use the linear function $f(x_1, x_2, x_3) = \sum_{i=1}^3 w_i x_i$ to fit the training data, where $x_1, x_2, x_3$ represent sepal length, sepal width and petal length, respectively, and we assume $w_0 = 0$.

   (a) (2 points) Use normal equation to solve this least squares problem. What is the solution $\boldsymbol{w}$ you obtained? What is the RMSE on training set? What is the RMSE on test set?

   (b) (4 points) Now let us consider using gradient descent algorithm to solve this problem. The objective

$$J = \frac{1}{2}\|\boldsymbol{y} - X\boldsymbol{w}\|_2^2,$$

   where $X \in \mathbb{R}^{N\times 3}$, $\boldsymbol{y} \in \mathbb{R}^N$, $\boldsymbol{w} \in \mathbb{R}^3$, and $N$ is the number of data points in the training set.

The gradient of $J$ with respect to $\boldsymbol{w}$ is

$$\frac{\partial J}{\partial \boldsymbol{w}} = X^T(X\boldsymbol{w} - \boldsymbol{y}).$$

We can use the following updating rule to reach the optimal value of $\boldsymbol{w}$:

$$\boldsymbol{w}^{(t+1)} \leftarrow \boldsymbol{w}^{(t)} - \eta X^T(X\boldsymbol{w}^{(t)} - \boldsymbol{y}),$$

where $t$ is the current number of iterations and $\eta$ is the learning rate which is set heuristically based on the application. When using the gradient descent algorithm, we need an initial point $\boldsymbol{w}^{(0)}$ as well. For this problem, let $\boldsymbol{w}^{(0)} = [1,1,1]^T$ and $\eta = 10^{-4}$. Update $\boldsymbol{w}$ using the above updating rule until the criteria $\|\boldsymbol{w}^{(t+1)} - \boldsymbol{w}^{(t)}\|_2 / \|\boldsymbol{w}^{(t)}\|_2 \leq \epsilon$ is satisfied, where the tolerance $\epsilon$ is set to be $10^{-6}$ for this problem.

What is the solution $\boldsymbol{w}$ using the above gradient descent algorithm? Is it the same as in 3(a)?