CHENNAI MATHEMATICAL INSTITUTE

Machine Learning

Aug 30, 2015.

(1) The MATLAB function *rand*() which samples a random variable uniformly from $(0, 1)$. There is a corresponding function in numpy, random.uniform. Show how this can be used to get samples from a Bernoulli distribution with parameter p. Write a program which does this.

Use this to sample from a binomial distribution with parameters n, p. Get N samples from the binomial and compare on the same figure the histogram of values obtained and the binomial distribution itself.

(2) Consider the equation of a hyperplane given by $y = c_0 + c_1 * x_1 + \ldots + c_4 * x_4$. Let ϵ be a random variable that is normally distributed with mean 0 and standard deviation σ. Generate points x^i and for each of them let ϵ^i be a sample point of the normal distribution. Compute $y^i = c^T * x^i + \epsilon_i$ and use this as the value of the observed for the given data point. Use linear regression to estimate the values of $c_0, \ldots, c_4$. Do this for 5 different data sets each containing 102 data points. Also output the mean of 5 values of c_0, c_1, c_2, c_3, c_4 from the five data sets. Use stochastic gradient or batch gradient to compute these values.

Compare this with the closed form solution computed by Python.

(3) Consider the data set in ex3.dat. Now the feature space is two dimensional. Estimate the parameters for linear regression using batch gradient descent. Does the algorithm converge?

Normalize the data set so that the values in the first column are adjusted so that the mean is zero. Calculate μ the mean of the values and the standard deviation σ. Replace each value x^i by $(x^i - \mu)/\sigma$. Now run batch gradient for various values of α and plot the number of iterations it takes to converge as a function of α.

Use also stochastic gradient by taking a random permutation and see how quickly we converge to the values you get from above. What does the closed form solution give?