Soft Computing:ISCO630E

Report

Assignment 2.1

Anant Chaturvedi(IIT2016506)

Question Description

1. Use housing price dataset. Using various parameters in the datasets we need to predict the housing price.

   1. Predict housing price using normal equations with regularizer.

   2. Predict housing price using gradient descent with regulariser.

   3. Compare the results for both approaches

   Design alpha and lambda parameters so that hypothesis will perform better.

   Case 1 : Use all dataset for training.

   Case 2 : 70% to design hypothesis and 30% to test hypothesis

   Find the differences and conclude analysis

Introduction

We are given a dataset comprising of housing prices with various features(continuous as well as categorical), for which we need to find the parameters that will help us find an appropriate hypothesis for the model.We have to apply regularization in contrast to the previous assignment.

Concepts Used

Hypothesis Function

$$h(x) = \sum_{i=0}^{n} \theta_i x_i = \theta^T x, \qquad X = \begin{bmatrix} — (x^{(1)})^T — \\ — (x^{(2)})^T — \\ \vdots \\ — (x^{(m)})^T — \end{bmatrix}. \qquad \vec{y} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{bmatrix}.$$

The above formula approximates y as a linear function of x, called hypothesis. The hypothesis considers both Θ and x as (n + 1) size vectors.

(1) X = m X (n + 1) matrix, containing the training samples features in the rows, where each $x^{(i)}$ is a (n + 1) X 1 matrix.
(2) m = number of training samples
(3) Θ_i = the parameters
(4) x_0 = 1 for every sample.
(5) n = features count, not including x_0 for every training sample
(6) Y contains all the m target values from the training samples, so it's m X 1 matrix

Cost Function(Regularized)

$$J(\theta) = \frac{1}{2m}\left[\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 + \lambda \sum_{j=1}^{n} \theta_j^2\right]$$

$$\text{where } \lambda \text{ is the regularization parameter}$$

The regularization parameter λ is a control on your fitting parameters. As the magnitudes of the fitting parameters increase, there will be an increasing penalty on the cost function. This penalty is dependent on the squares of the parameters as well as the magnitude of λ

Normal Equation(Regularized)

$$\theta = \left(X^T X + \lambda \begin{bmatrix} 0 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix}\right)^{-1} X^T \vec{y}$$

Gradient Descent(Regularized)

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta).$$

$$\frac{\partial J(\theta)}{\partial \theta_0} = \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)} \qquad \text{for } j = 0$$

$$\frac{\partial J(\theta)}{\partial \theta_j} = \left(\frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)} \right) + \frac{\lambda}{m} \theta_j \quad \text{for } j \geq 1$$

α = Learning rate

The convergence arises when $J_t(\Theta) - J_{t+1}(\Theta) <=$ ERROR_THRESHOLD

Since the cost function is convex in nature, the local minima obtained for the cost function would be the global minima.

RESULTS

The convergence appeared to be faster for

α =0.001

Error_Threshold=0.001

λ =0.00025

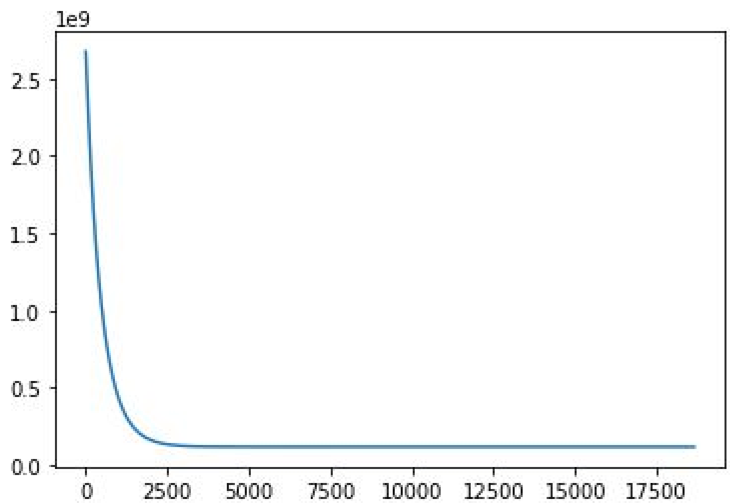
For the approach of Gradient Descent it took 18642 iterations to arrive at a final cost of 116323395.6

The final theta values using Gradient Descent obtained were

Final theta values : [68121.59652816 7678.75690636 1352.26974984 7189.94739908

5684.45181292 2328.40900684 1725.21246225 2598.33007529

2680.76454173 5876.27380775 3652.58217771 3968.60446529]



Using Normal Equation the Final Cost came out to be 116323378.54

And The Final theta values obtained were

Final theta values : [68121.5970696 7681.90052904 1349.66168159 7192.11581214

5687.53881382 2327.66854324 1724.31539331 2600.30648789

2682.07092653 5877.42976373 3652.74931055 3969.33193691]

It can be seen that the Final cost and the theta values using both gradient descent and normal equation are almost similar.