ASSIGNMENT 1: LOGISTIC REGRESSION

ABHISHEK BOSE

Instructor: Marco Montes De Oca

June 06, 2019

Abstract

The aim of this assignment is to illustrate the concept of Logistic Regression.We have been given a set of image to perform the classification using logistic Regression.

Contents

Introduction

Logistic Regression Logistic Regression is used to solve the problem of classification. The main purpose behind logistic Regression is it gives the probability of y being belong to a particular class. Which is diddicult to acheive using linear regression since we would like to obatain the probability if the data belongs to a class or not. The **sigmoid function** is used in order to have a probabilistic value between 0 to 1 rather than keeping a value between negetive infinity to positive infinity and is used to represent the odds, the logistic function is represent as:

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T.X}}$$

Assuming that if we have m training examples we can write the likelihood of the parameters as:

$$L(\theta) = \prod_{i=1}^{m} (h_\theta(x^{(i)}))^{y^{(i)}} (1 - h_\theta(x^{(i)}))^{1-y^{(i)}}$$

which is the costfunction of logistic regression but unlike linear regression we would like to maximize after we take take log on both sides:

$$l(\theta) = \sum_{i=1}^{m} y^{(i)} log(h_\theta(x^{(i)})) + 1 - y^{(i)} log(1 - h_\theta(x^{(i)}))$$

The above function is our link function.

Our objective is to maximize the above function by applying gardient descent to obtain the parameter as discussed in last assignment.

Analysis:

- We are a set a of images in form of gray scale images for which we need to implement a logistic regression.
- We have implemented the code in R in order to generate an confusion matrix.

Accuracy Obtained: 73.75%

Observation:

- Initially we have created a sample taking all the positive data for a given label and diviving the remaining sample among the other label for which we get an accuracy of 68%. the reason being of high biased data and a a result it misclassfied between 0,7 and 9. The reson is we dont have enough information about 7 and 9 while we run the traing model for label 1 classification.
- We try to improve our accuracy taking random sample from the data set here we have alomost equal number of records for each label. So the model has equal weightage for classifing positive and negetive sample. Problem happens if we give particular weightage for a single label it will train itself perfectly for that label but for the remaining label it will be inefficiently train now if a certain number for example 7 comes up for classification model 1 it might have maximum characterstics which is similar to 1 and our algorithm is taking highest probability among the 10 iteration hence it will predict 7 as 1 since it don't have enough information for 7 in training model 1 to classifiy it as o.

Predicted Labels Actual Labels	0	1	2	3	4	5	6	7	8	9
0	380	23	9	132	9	13	420	2	10	2
1	1	966	2	16	3	1	9	0	2	0
2	2	11	556	26	180	10	202	3	10	0
3	22	57	11	858	8	6	34	1	3	0
4	11	17	126	93	642	7	98	0	6	0
5	11	13	17	21	9	845	1	34	33	16
6	57	17	103	131	108	17	548	3	15	1
7	6	6	13	5	0	47	3	862	15	43
8	5	10	6	42	54	37	70	16	753	7
9	6	12	9	3	3	44	3	48	6	865

Figure 1: Confusion Matrix