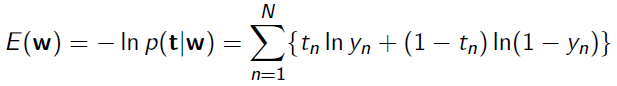
Atra Akandeh (A47212054) CSE847 HW#4

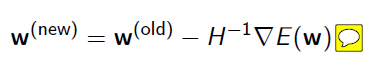
---------------------------------------------------------------------------------------------------------------------

Q1) https://github.com/atrakriv/hw4.git

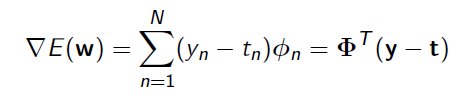
In this question logistic regression has been implemented. The error function is the negative logarithm of the likelihood, namely, Cross-entropy error function:

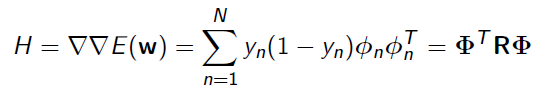


Newton-Raphson Algorithm uses a local quadratic approximation to the cross-entropy error function to update w iteratively:



For cross-entropy error function, gradient and Hessian matrix can be calculated as follows:

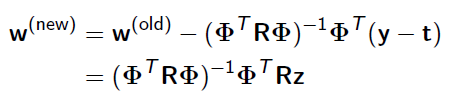




where row of Φ is ,

and R is an diagonal matrix, and .

The Newton-Raphson update for cross-entropy error function is:



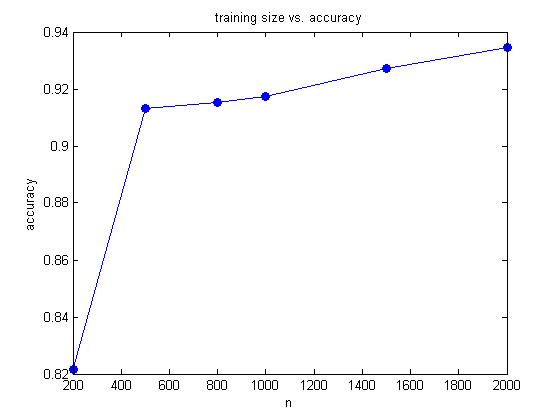
where



A data set of size 4601 has been provided. Testing size is fixed and contained 2061 samples. Training size n is variable and accuracy on the test data as a function of n is reported.

Note: To get the final result, step function to map output values to labels (probabilities) 0 and 1 has been used.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| n | 200 | 500 | 800 | 1000 | 1500 | 2000 |
| accuracy | 0.82161 | 0.91311 | 0.91542 | 0.91734 | 0.92734 | 0.93464 |



Q2) https://github.com/atrakriv/hw4.git

In this problem sparse logistic regression has been implemented. function has been used to perform l1-regularization. Bias is not included in data sent as a parameter to function . To map the labels from 0,1 to -1,1 has been used. Function has been defined to initialize the flags. In this problem both training size and testing size is fixed. Different values of regularization parameter have been tried. As z (l1-regularization parameter) increases the number of selected feature decreases. In the table below accuracy and number of selected features have been shown.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| z | 0.01 | 0.02 | 0.05 | 0.1 | 0.15 | 0.2 |
| accuracy | 0.8516 | 0.85429 | 0.86813 | 0.86236 | 0.8366 | 0.83622 |
|  | 14 | 13 | 11 | 7 | 6 | 6 |

