**CS697A – Topic in Computer Science – Machine Learning**

Assignment 4

**Submitted by: Prachi Panse**

**NetID: ve3568**

Q1: Implement the logistic regression (i.e. 1-layer neural network with a single sigmoidal output) algorithm yourself in Python, use adaptive learning rate and momentum for training. Train and test 10 times, each time, start from different random initial weights and use a random subset of 80% of the training data and also start with a different initial learning rate (e.g. 0.0001, 0.005, 0.001, 0.01 etc.) and momentum (0.9, 0.95, 0.99). Report the total training and test errors for each of the 10 runs. Report also the initial learning rate and the momentum you used.

***Answer:***

Program file: <Assignment4Q1.ipynb>

* Output for the given 80% of training and testing data:

mean training error : 0.0067

mean training error : 0.0075

mean training error : 0.0055

mean training error : 0.0068

mean training error : 0.0070

mean training error : 0.0052

mean training error : 0.0064

mean training error : 0.0069

mean training error : 0.0059

mean training error : 0.0060

mean testing error : 0.0065

mean testing error : 0.0073

mean testing error : 0.0070

mean testing error : 0.0059

mean testing error : 0.0078

mean testing error : 0.0060

mean testing error : 0.0045

mean testing error : 0.0058

mean testing error : 0.0082

mean testing error : 0.0072

* Learning rates used are 0.00005, 0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1.0

**Q2:** Using the 10 runs in Q1, for each feature compute feature importance

as

a) Fa: the average (over 10 runs) increase in the test error when the weight for that

feature is set to 0.

For each feature, compute also,

b) Fb: the variance of each feature on the test set and

c) Fc: the mutual information between the feature column and the label column.

Plot x=Fa and y=Fb, Fc

Comment on how your feature importance Fa the other measurements Fb and Fc

are related.

***Answer:***

Program file:<Assignment4Q2.ipynb>

* Output For the given training and random testing data:

Fa: The average increase in the total test error :

4.9999999999999826e-05

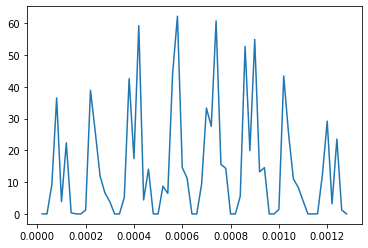
Fb: the variance of each feature on the test set :

[0.0, 0.0, 9.222222222222221, 36.5, 3.888888888888889, 22.38888888888889, 0.3541666666666667, 0.0, 0.0, 1.2222222222222219, 38.90972222222222, 25.6875, 11.805555555555555, 6.576388888888889, 3.8541666666666665, 0.0, 0.0, 5.1875, 42.57638888888889, 17.409722222222218, 59.24305555555555, 4.409722222222222, 14.076388888888891, 0.0, 0.0, 8.722222222222221, 6.472222222222222, 44.1875, 62.22222222222222, 14.576388888888888, 11.25, 0.0, 0.0, 9.354166666666666, 33.30555555555555, 27.576388888888886, 60.743055555555564, 15.576388888888888, 14.354166666666666, 0.0, 0.0, 5.576388888888888, 52.72222222222222, 19.909722222222225, 54.90972222222221, 13.25, 14.576388888888886, 0.0, 0.0, 1.4166666666666667, 43.40972222222223, 25.25, 11.020833333333334, 8.38888888888889, 4.138888888888888, 0.0, 0.0, 0.0, 12.243055555555557, 29.1875, 3.1875, 23.555555555555554, 1.2430555555555554, 0.0]

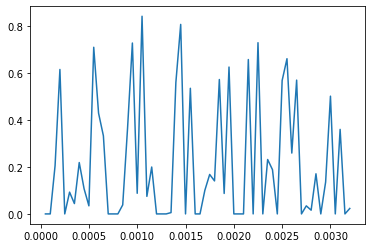
Fc: the mutual information between the feature column and the label column :

[0.0, 0.0, 0.20552248677248697, 0.61432178932179, 0.0, 0.0926286676286685, 0.04401455026455103, 0.21848845598845656, 0.10731120731120813, 0.03439153439153486, 0.7087662337662348, 0.4285744348244356, 0.33217893217893324, 0.0, 0.0, 0.0, 0.038293650793651324, 0.36412337662337757, 0.7268217893217903, 0.08773448773448878, 0.8407106782106792, 0.0746392496392505, 0.19944083694083803, 0.0, 0.0, 0.0, 0.005591630591631258, 0.5586670274170282, 0.8059884559884567, 0.0, 0.5343614718614731, 0.0, 0.0, 0.10033369408369519, 0.16782708032708116, 0.14037698412698463, 0.5716299903799915, 0.08704004329004467, 0.6246392496392501, 0.0, 0.0, 0.0, 0.6565836940836955, 0.0, 0.7282106782106788, 0.0, 0.23138528138528258, 0.18823051948052005, 0.0, 0.5674272486772491, 0.6601551226551237, 0.2587662337662351, 0.5690836940836947, 0.0, 0.034262265512266854, 0.015942159692160063, 0.17067099567099642, 0.0, 0.1355519480519487, 0.500796657046658, 0.0, 0.35949374699374825, 0.0, 0.023250360750361132]

* Plotting x=Fa and y=Fb



* Plotting x=Fa and y=Fc



**Q3:** Use the scikit-learn neural network implementation to train a neural network and test it using the same instances as in Q1. Decrease the test error as much as you can through selection of:

-different number of hidden layers and units,

-L1 or L2 regularization/weight decay,

-different optimization algorithms,

-feature selection

***Answer:***

Program file: <Assignment4Q3.ipynb>

* Output For the given training and testing data:

Different no. of hidden layers and units:

mean\_score for ~100% instances: 0.5046

mean\_score for ~78% instances: 0.4236

mean\_score for ~39% instances: 0.3711

Feature Selection:

mean\_score for ~100% instances: 0.4626

mean\_score for ~62% instances: 0.2690

mean\_score for ~47% instances: 0.2979