CSE343/ ECE363/ ECE563: Machine Learning W2021
Assignment-1
Linear / Logistic Regression and Naive Bayes

Programming Question 2 :Logistic Regression Report

1.

Both the training and testing dataset used for this question was imbalanced.

- Training Dataset
    - Salary <= 50K : 22654 data points
    - Salary > 50K : 7508 data points
- Testing Dataset
    - Salary <= 50K : 11360 data points
    - Salary > 50K : 3700 data points

However for the scope of this assignment sampling (undersampling/oversampling/ stratified sampling) was not required , hence it has not been done.

All the categorical features were one hot encoded and the continuous value features were standardized

For feature sub-selection, I found the correlation between different features. The correlation between different features suggests how closely two values are related, if two values have very high correlation then this suggests that they will have a similar effect on the predicted value. There were no two pairs of features which had correlation > 0.9 . So, I went forward with all the features.

The model parameters were selected by hit and trial and these are the parameters which performed best on training and validation set.

LogisticRegression(n_iter=5000,l_rate=0.1,regularization = "L2",reg_lambda = 0.01)

Accuracy table		
Data set	**L1 Regularization**	**L2 Regularization**
Training Set	0.84873	0.84873
Validation Set	0.84372	0.84369

Testing Set	0.84760	0.84754

Which regularization will work better: L1 or L2 ?. Both the regularizers help to prevent overfitting but their usage depends upon a lot of factors.

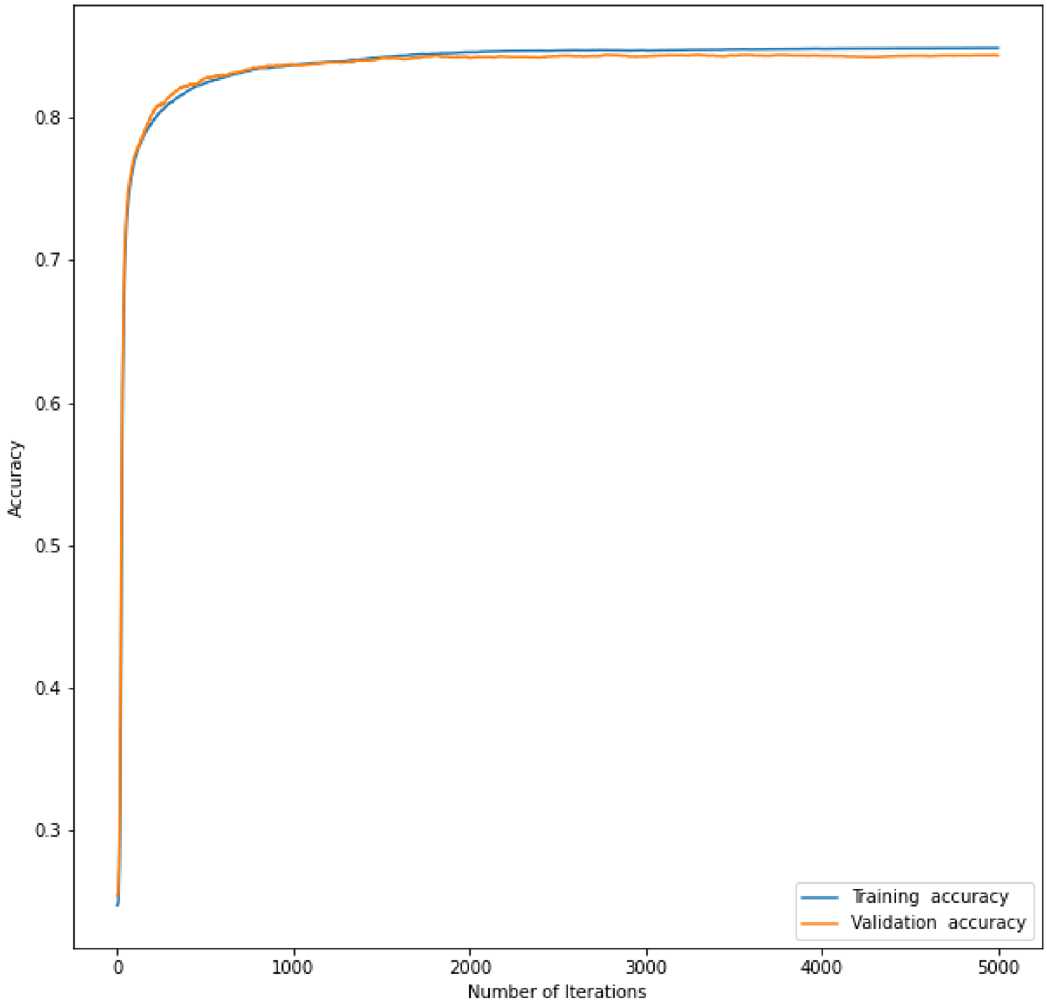
Since in L1 regularization, in the gradient update we change the weights according to the sign of the weight. If the weight is positive we subtract the regulazier from it and vice versa. This technique essentially promotes sparsity and brings the weights closer to zero. Thus L1 helps in feature subselection by bringing the weights to zero.

In L2 regularization apart from the penalty imposed due to the sign, we also impose an additional penalty of the magnitude of a particular weight. This also brings the weights close to zero but not equal to zero. The L2 regression mostly helps when the dataset is small and has high dimensions. It gives out stable results even with few samples in comparison to L1.

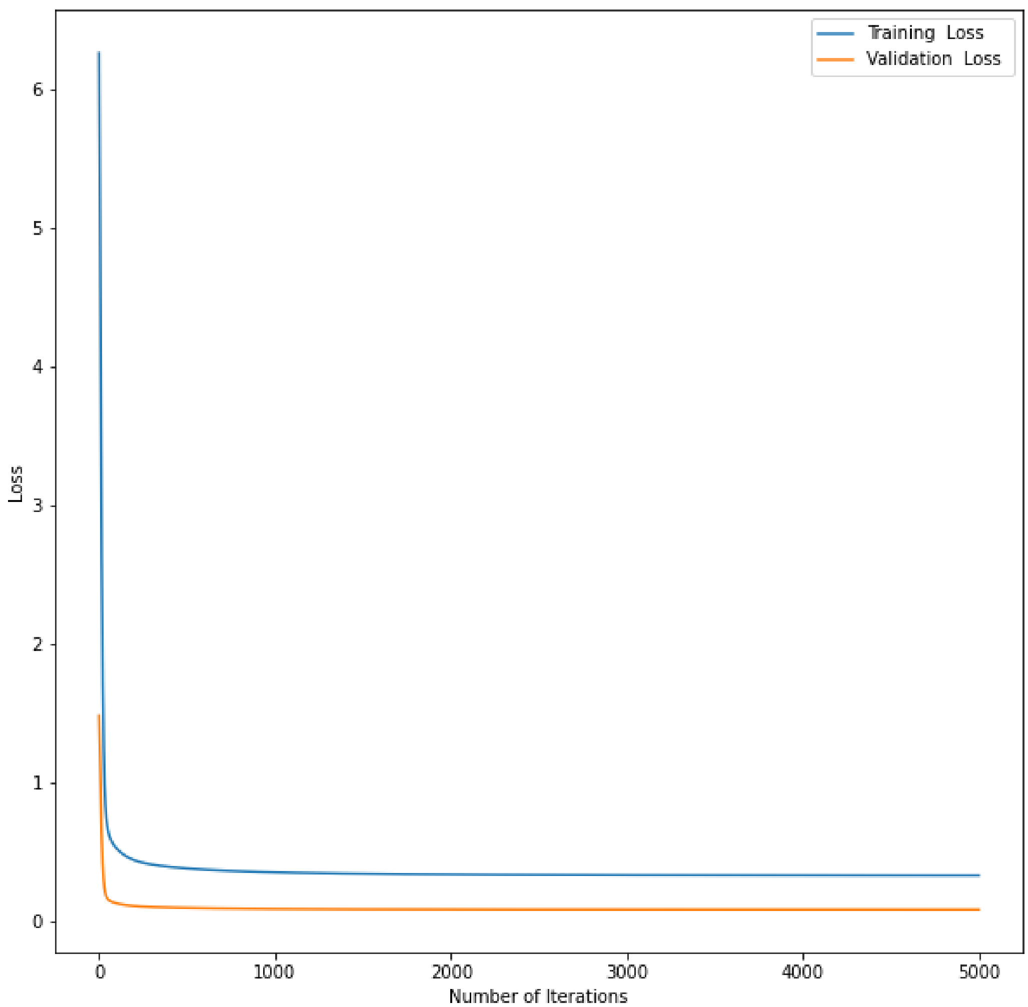
In our case , L1 and L2 both are having comparable performances, however the L1 regularizer had a slight edge over the L2 (0.01 % better). This may be due to the sparsity promotion nature of L2 which helps in feature subselection.

The accuracy and loss vs iteration graphs are attached below :

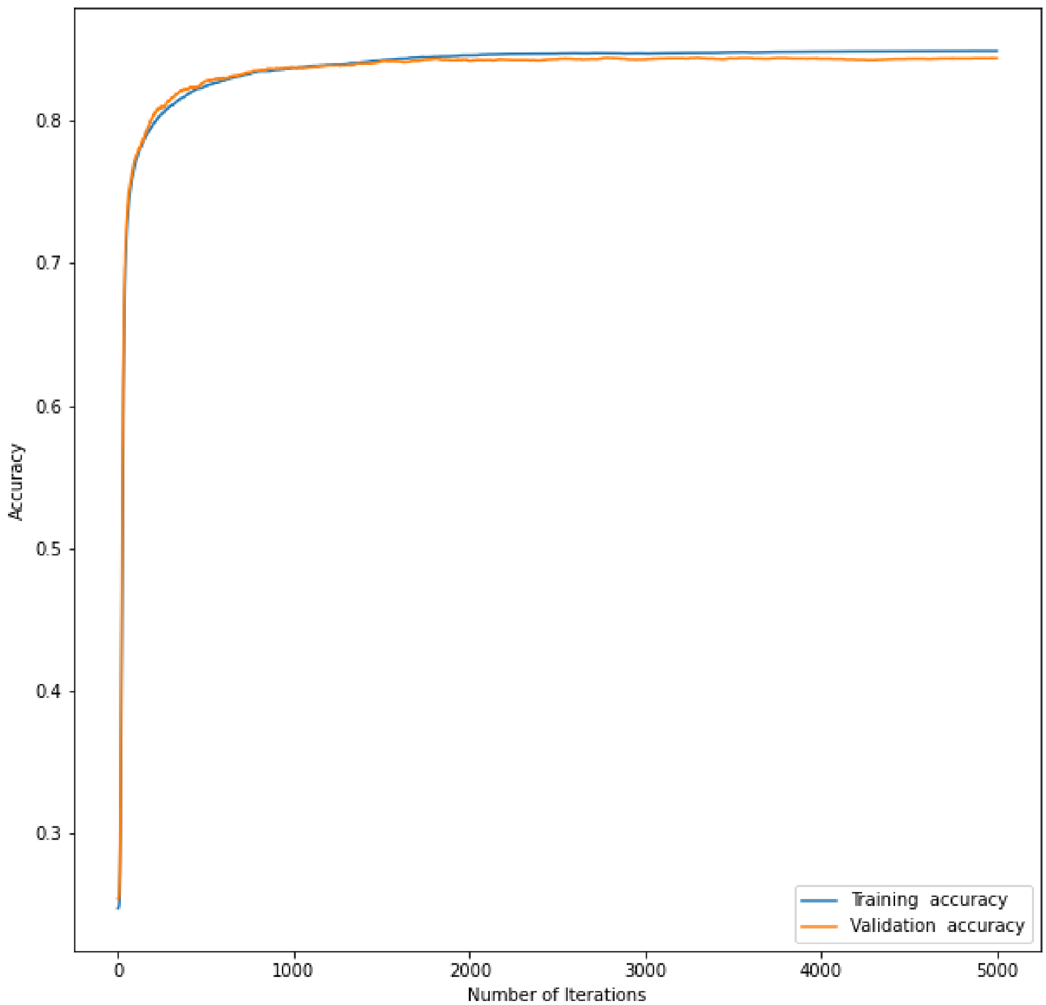
For L1 :



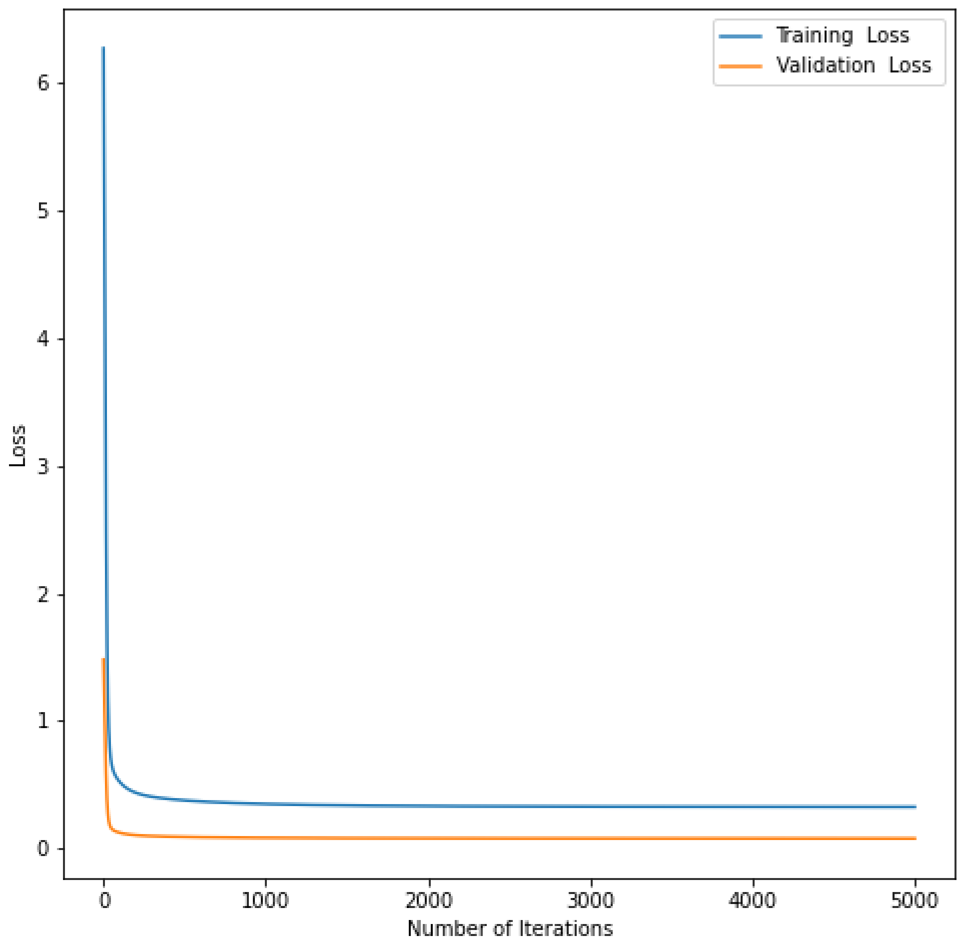
Training accuracy
Validation accuracy
Accuracy
Number of Iterations
0.3
0.4
0.5
0.6
0.7
0.8
0
1000
2000
3000
4000
5000


Training Loss
Validation Loss
Loss
0
1
2
3
4
5
6
0
1000
2000
3000
4000
5000
Number of Iterations

For L2 :



0.8
0.7
0.6
0.5
0.4
0.3
Accuracy
0
1000
2000
3000
4000
5000
Number of Iterations
Training accuracy
Validation accuracy


Training Loss
Validation Loss
Loss
Number of Iterations
0
1
2
3
4
5
6
0
1000
2000
3000
4000
5000

2.

Accuracy for each class was defined as follows while using the one vs rest approach :

- Take the class for which accuracy has to be reported as a positive class.
- Every other class is combined to become a negative class.
- Calculate true positives and true negatives considering the classes defined above.
- Accuracy is give by : (True positive + True Negative)/Total No. of data points in data
- This accuracy is for every one-v-rest approach.

The model accuracy is calculated by adding all the true positives for each of the 10 classes and dividing it by total number of data points.

L1 regularization		
Class	**Training Accuracy**	**Test Accuracy**
0	0.9994	0.9855
1	0.9994	0.9865
2	0.9942	0.9695
3	0.9917	0.966
4	0.9986	0.971
5	0.993	0.959
6	0.9991	0.98
7	0.9983	0.97
8	0.9909	0.954
9	0.9936	0.9625
Average	0.9958	0.9704

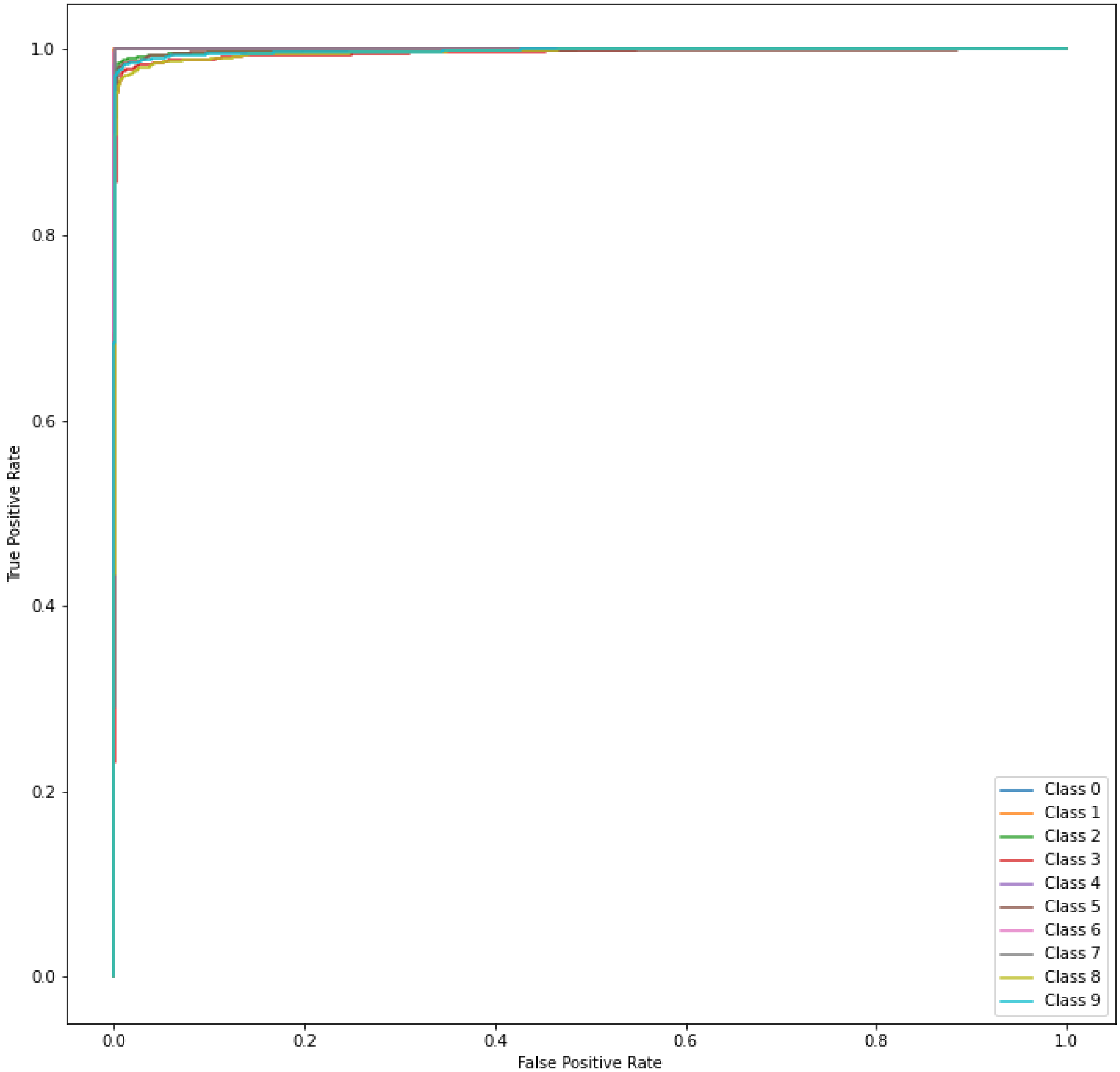
In L1 regularization : Accuracy for
Training : 0.9791
Testing : 0.852

L2 regularization		
Class	**Training Accuracy**	**Test Accuracy**
0	0.9998	0.9845
1	1.0	0.9855
2	0.9946	0.962
3	0.9919	0.96
4	0.9999	0.9665
5	0.9941	0.956
6	1.0	0.9755
7	0.9998	0.965
8	0.9919	0.9475
9	0.9956	0.9555
Average	0.99675	0.9658

In L2 regularisation : Accuracy for
Training : 0.9838
Testing : 0.829

In both the cases (L1 and L2 regularization), we observe that the training accuracy is very high as compared to the test accuracy. We have a very high variance (approx 15%) and low bias(approx 1.4%) which clearly indicates that the model is overfitted. The model performs very good on the training set but fails to perform for the testing set.

3. The Roc curve for training data is as follows:



The Roc curve for testing data is as follows: