CMPE 462 - Project 1
Binary Classification with Logistic Regression

Student ID1: 2019700099

Student ID2: 2019705069

Student ID3: 2019700057

Task 1: Feature Extraction

The grayscaled version of the first and 1500th image of the training set is illustrated in Figure 1.

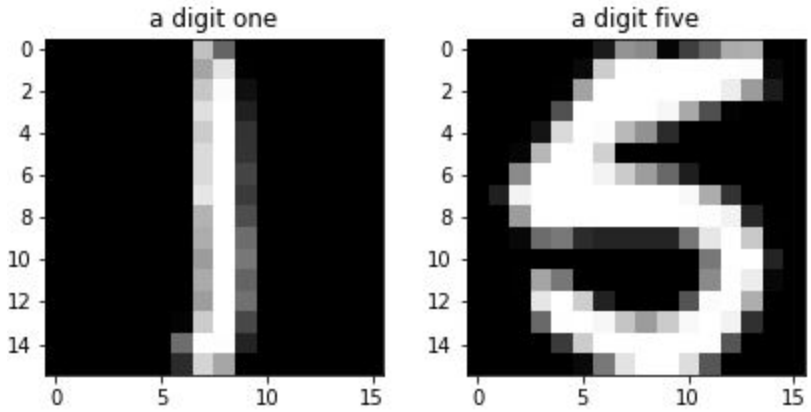

Figure 1: Example of 1 and 5

- **Representation 1:**

  The Figure 2 shows the distribution of training and test dataset with respect to the symmetry and normalized average intensity of the images.

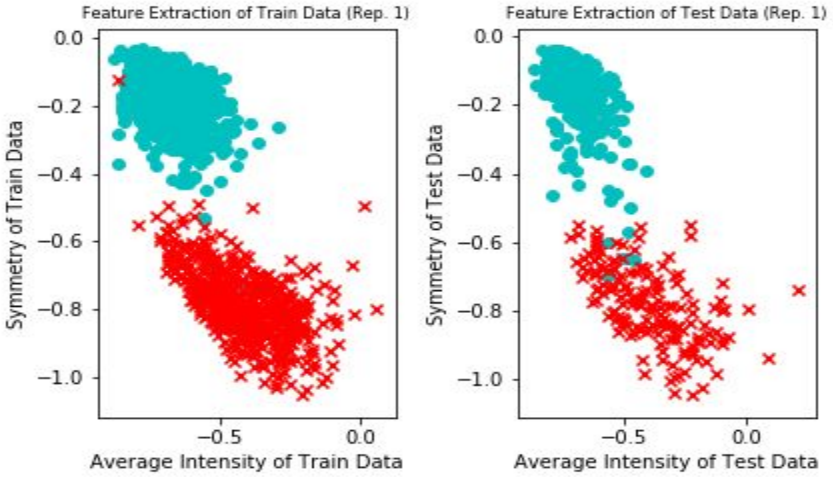

Figure 2: Feature Extraction of Train and Test Data for Representation 1

- **Representation 2:**

Since there exists only two classes for this classification problem, we decided to fetch features that specifically distinguish digits 1 and 5 rather than using more generalized ways. As it can be seen from their ideal shapes, digit 1 has less width than digit 5; furthermore, digit 5 has a wavy shape rather than digit 1's straighter shape. Therefore, we have a 2D feature mapping similar to Representation 1.

For the difference of the widths, we obtained the x-axis positions of the leftmost and the rightmost non-black pixels of the digit. Then, we created our first feature as the difference between them. However, there is a fact that digits are not in ideal shapes. For example, a skewed digit 1 could be wider than a proper small digit 5. In order to overcome this problem, we sliced the digit frame into 4 horizontal pieces on top of each other and took the average width difference of these pieces. By doing that, we performed a kind of normalization on the widths of the digit images.

Moreover, for the wavy shape difference, we imagined there were vertical lines cutting through the digit images. As digit 1 is straighter than digit 5, we thought that the vertical lines would cut through digit 1 less than 5. We calculated this by counting the contrast difference through the vertical lines. For example, an ideal digit 1 would have less vertical contrast change than an ideal digit 5. Nonetheless, as told before, the digits are not in ideal shapes. If there is a wider digit 1, its vertical contrast change would be larger than a tighter digit 5. Again, we calculated the average of this value as our second feature.

To prevent potential data overflow problems, we fit our two features into [0,1] interval. Below, one can see the 2D mappings of Representation 2 for training and test data sets. In the mappings below, red crosses represent digit 5 and blue circles represent digit 1.

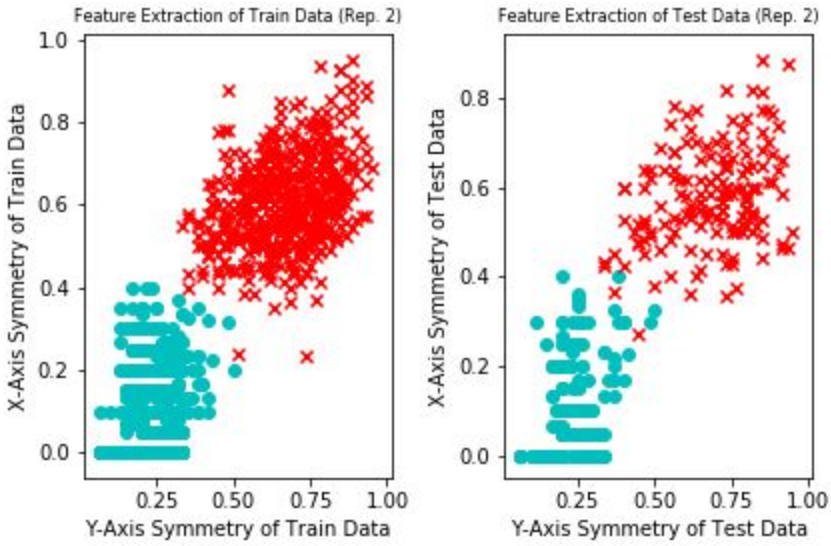

Figure 3: Feature Extraction of Train and Test Data for Representation 2

Task 2: Logistic Regression

We derived the gradient of the logistic loss with respect to w and implemented the logistic regression classifier with gradient descent.

$$E(w) = \frac{1}{N}\sum_{n=1}^{N} ln(1 + e^{-y_n w^T x_n})$$

$$\frac{dE}{dw} = \frac{1}{N}\sum_{n=1}^{N} \frac{\frac{d}{dw}(1+e^{-y_n w^T x_n})}{1+e^{-y_n w^T x_n}}$$

$$\frac{dE}{dw} = \frac{1}{N}\sum_{n=1}^{N} \frac{-y_n x_n e^{-y_n w^T x_n}}{1+e^{-y_n w^T x_n}}$$

We run the classifier for 100 iterations and the logistic loss is decreased for both representations as expected. Below, one can see the graphs of logistic losses converge to 0.

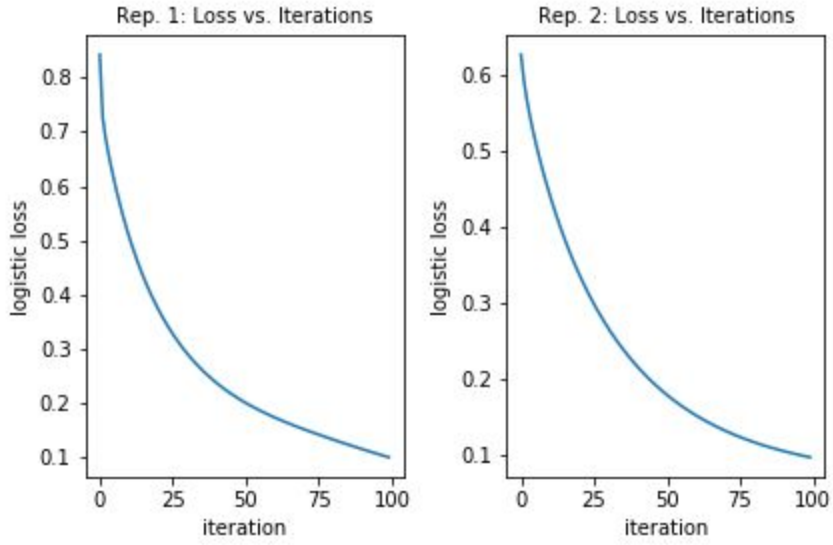

Figure 4: Logistic Loss vs Iteration Both Representation 1 and Representation 2 with iteration=100

We picked delta as 10^{-5} and experimented with 5 different learning rates (eta): 0.25, 0.4, 0.6, 0.7, 0.99. We observed that increasing the learning rate will make the classifier to converge in less number of iterations. Below, the plots of logistic losses for different learning rates are illustrated.

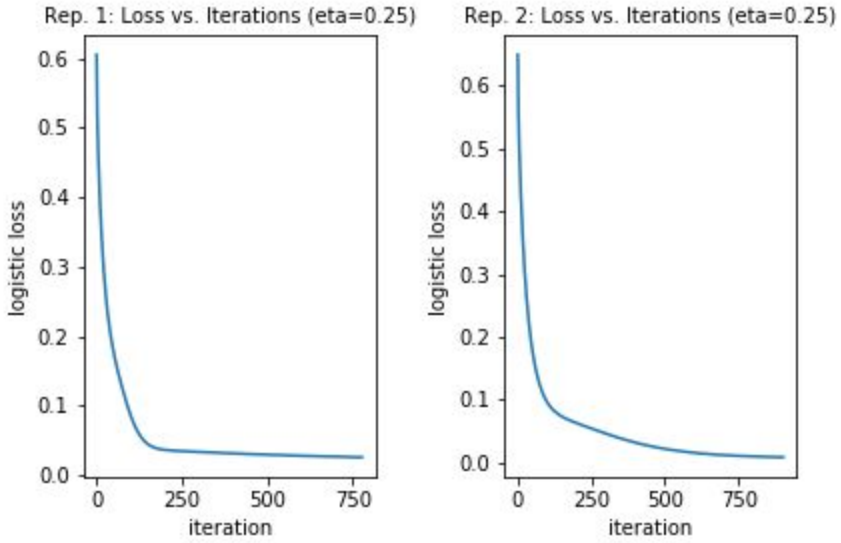

Figure 5: Logistic Loss vs Iteration Both Representation 1 and Representation 2 with eta=0.25

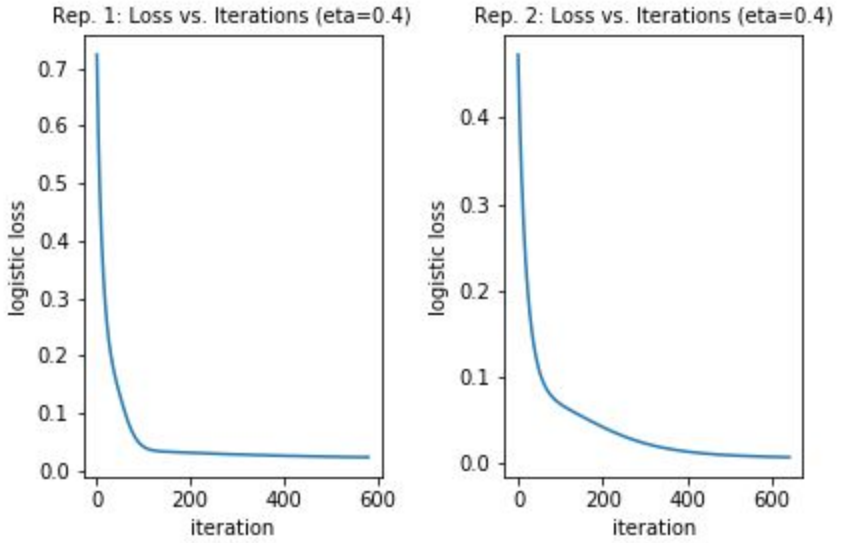

Figure 6: Logistic Loss vs Iteration Both Representation 1 and Representation 2 with eta=0.4

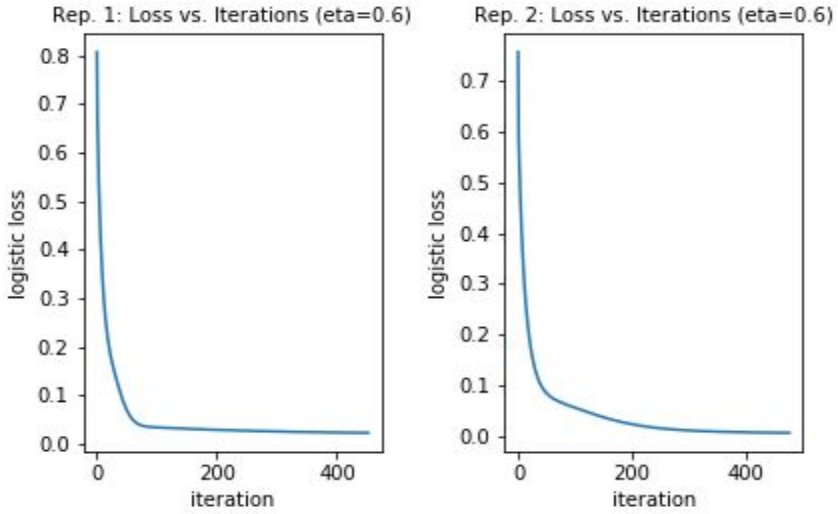

Figure 7: Logistic Loss vs Iteration Both Representation 1 and Representation 2 with eta=0.6

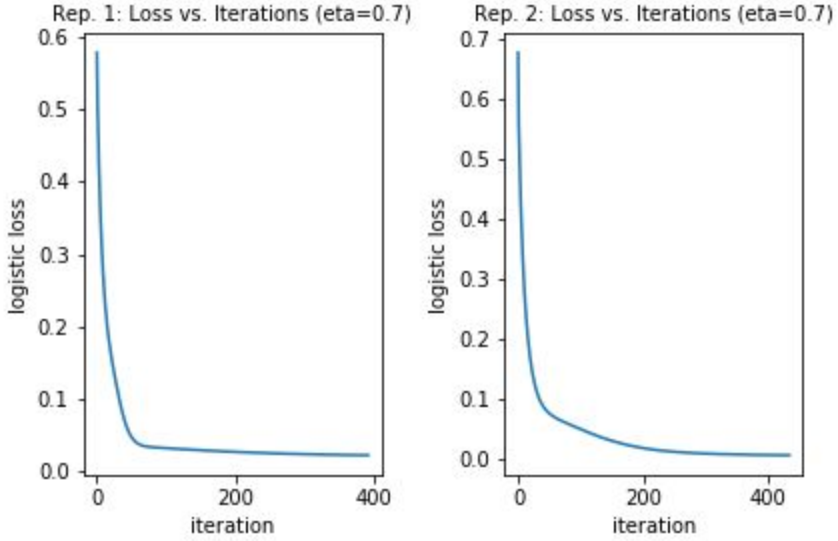

Figure 8: Logistic Loss vs Iteration Both Representation 1 and Representation 2 with eta=0.7

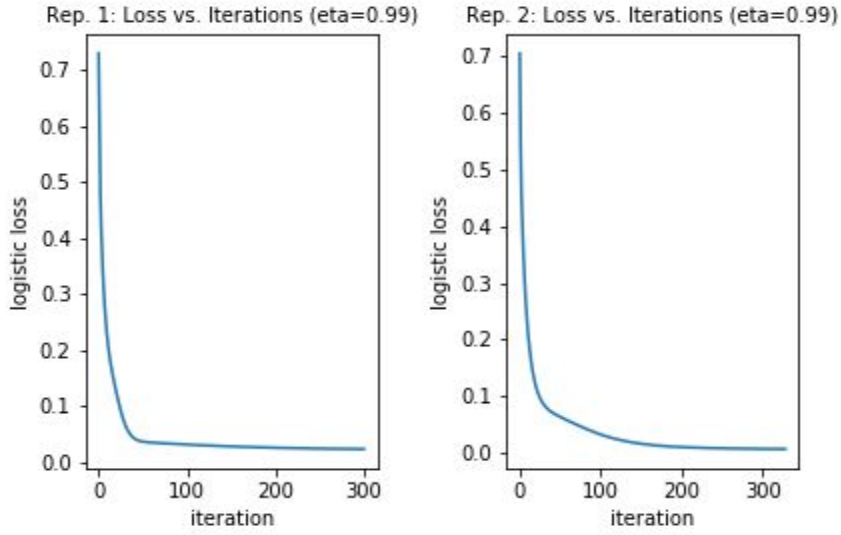

Figure 9: Logistic Loss vs Iteration Both Representation 1 and Representation 2 with eta=0.99

- **Logistic Regression with Regularization:**

Below, one can see the loss function and its derivation with regularization added.

$$E(w) = \frac{1}{N}\sum_{n=1}^{N} ln(1 + e^{-y_n w^T x_n}) + \lambda||w||_2^2$$

$$\frac{dE}{dw} = \frac{1}{N}\sum_{n=1}^{N} \frac{-y_n x_n e^{-y_n w^T x_n}}{1+e^{-y_n w^T x_n}} + 2\lambda w$$

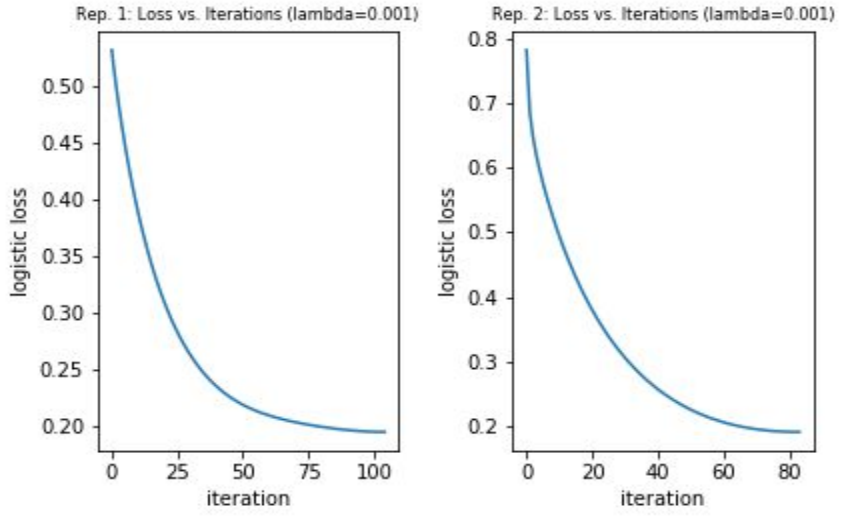

Figure 10: Logistic Loss vs Iteration Both Representation 1 and Representation 2 with regularization (λ=0.001 eta=0.25)

Lambda / Representation	*0.05*	*0.1*	*0.5*
Representation 1 mean	***93.54***	***89.39***	***67.34***
Representation 2 mean	***97.69***	***96.54***	***78.21***
Representation 1 std	***1.83***	***2.66***	***3.54***
Representation 2 std	***1.37***	***0.86***	***7.54***

Table 1 : Representations vs different λ values

Task 3: Evaluation

We picked the learning rate as 0.99 and λ as 0.05. The accuracy of the resulting model on training and test set are listed in Table 2. The decision boundaries of the resulting model on both representations.

Data Set / Representation	Training Set	Test Set
Representation 1 accuracy(%)	93.52	91.50
Representation 2 accuracy(%)	97.82	96.46

Table 2 : Accuracy vs Data Sets, λ = 0.05

As an interesting result, when we picked learning rate as 0.99 and λ as 0, we obtained the best accuracies. Below the result for these values can be seen.

Data Set / Representation	Training Set	Test Set
Representation 1 accuracy(%)	99.67	97.87
Representation 2 accuracy(%)	99.80	98.82

Table 3: Accuracy vs Data Sets, λ = 0

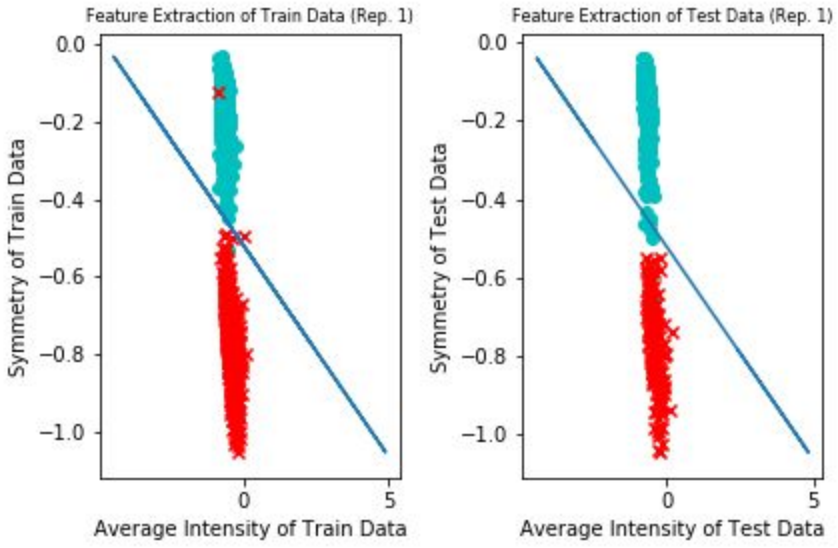

Figure 11: Decision boundary of the final model on Representation 1 (eta=0.99)

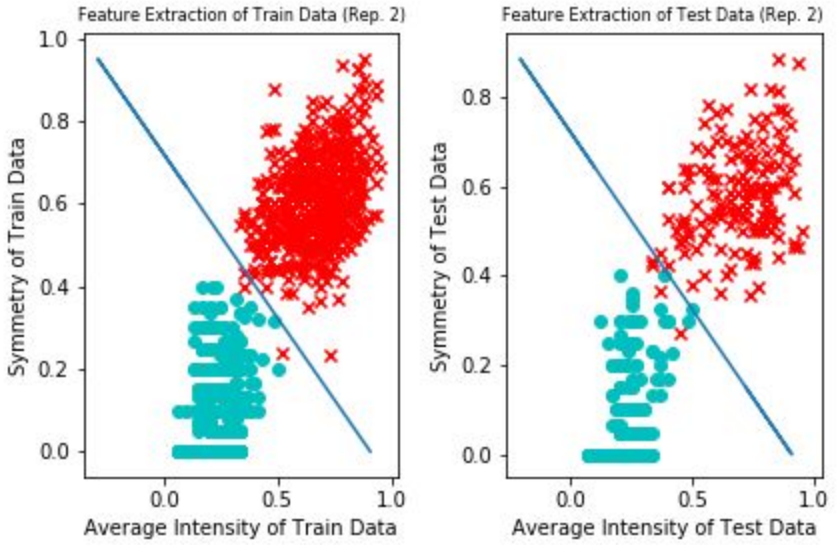

Figure 12: Decision boundary of the final model on Representation 2 (eta=0.99)

Comments:

The reason why regularization is applied is to prevent overfitting on the training set where it has not a similar distribution with the test set in most cases. Nonetheless, in this problem, the distributions of the both sets seem to have noticeable similarities; hence, when the model overfits the training set, it does not cause a big problem. Adding an external factor, i.e regularization factor, to loss function causes disturbance that affects the test accuracy as well due to its similar distribution to the training set. Therefore, in this problem, regularization does not affect beneficially for accuracy on both sets. The same situation applies for both representations.

The feature set that we have extracted for Representation 2 gives the best results. It can be observed from the accuracy values and 2D mappings of the features of both of the representations.

We could construct an algorithm to detect the outliers in the dataset and do not include these data points in the training and evaluation.