

# Intro to Computer Vision: HW 4

Alvin Lin

August 2018 - December 2018

## Logistic Regression Classifier

The logistic regressor classifier uses a single layer of weights and biases with the same dimensions as the features in the input to determine the output layer. Each epoch, the weights were adjusted to:

$$W_{t+1} = W_t - \eta X^T \cdot (pY - Y)$$

Overall, the logistic regressor took far less training time to arrive at the same classification rate as the neural network.

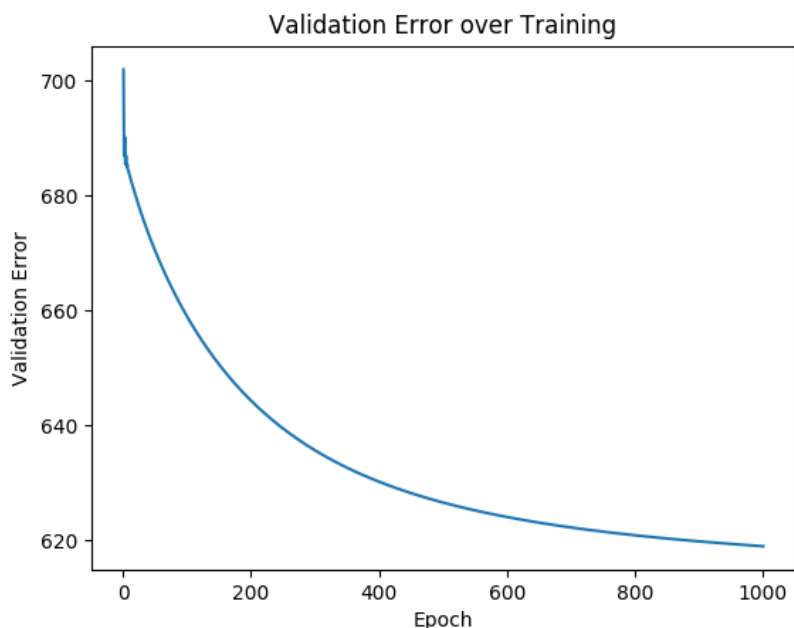


Figure 1: Validation Error over 1000 Epochs

The best validation error obtained by the linear regression classifier was 31.2% after 1000 epochs. On the testing data, this classifier performed with a 70.3% accuracy.

## Neural Network Classifier

For the neural network, one hidden layer was used with 16 nodes in the hidden layer. During each epoch, the weights were adjusted to:

$$\begin{aligned} W_{(1)t+1} &= W_{(1)t} - \eta Z^T \cdot (pY - Y) \\ dZ &= (pY - Y) \cdot (W_2)^T \cdot (1 - Z^2) \\ W_{(2)t+1} &= W_{(2)t} - \eta X^T \cdot dZ \end{aligned}$$



Figure 2: Training and Validation Error over 3000 Epochs

The lowest validation error encountered by the neural network was 31.1% after 3000 epochs of training. Without regularization, this neural network performed with 67.1% accuracy.

### Regularization

I used a regularization term equal to the following:

$$\frac{\lambda}{2} \left( \sum_i \|W_{1(i)}\|^2 + \sum_j \|W_{2(j)}\|^2 \right)$$



Figure 3: Training and Validation Error over 3000 Epochs

After 3000 epochs, the training and validation error began to diverge, and the lowest validation error obtained was 31.1%. On the testing data, this neural network performed with a 70.1% accuracy.

## Scikit-Learn Classifiers

Using the support vector machine and AdaBoost ensemble classifiers provided by `scikit-learn` gave the following results.

Classifier	Accuracy
SVM	70.3%
Adaboost Ensemble	68.5%

If you have any questions, comments, or concerns, please contact me at [alvin@omgimanerd.tech](mailto:alvin@omgimanerd.tech)