Introduction to Deep Learning for Computer Vision Assignment 1: Preliminaries

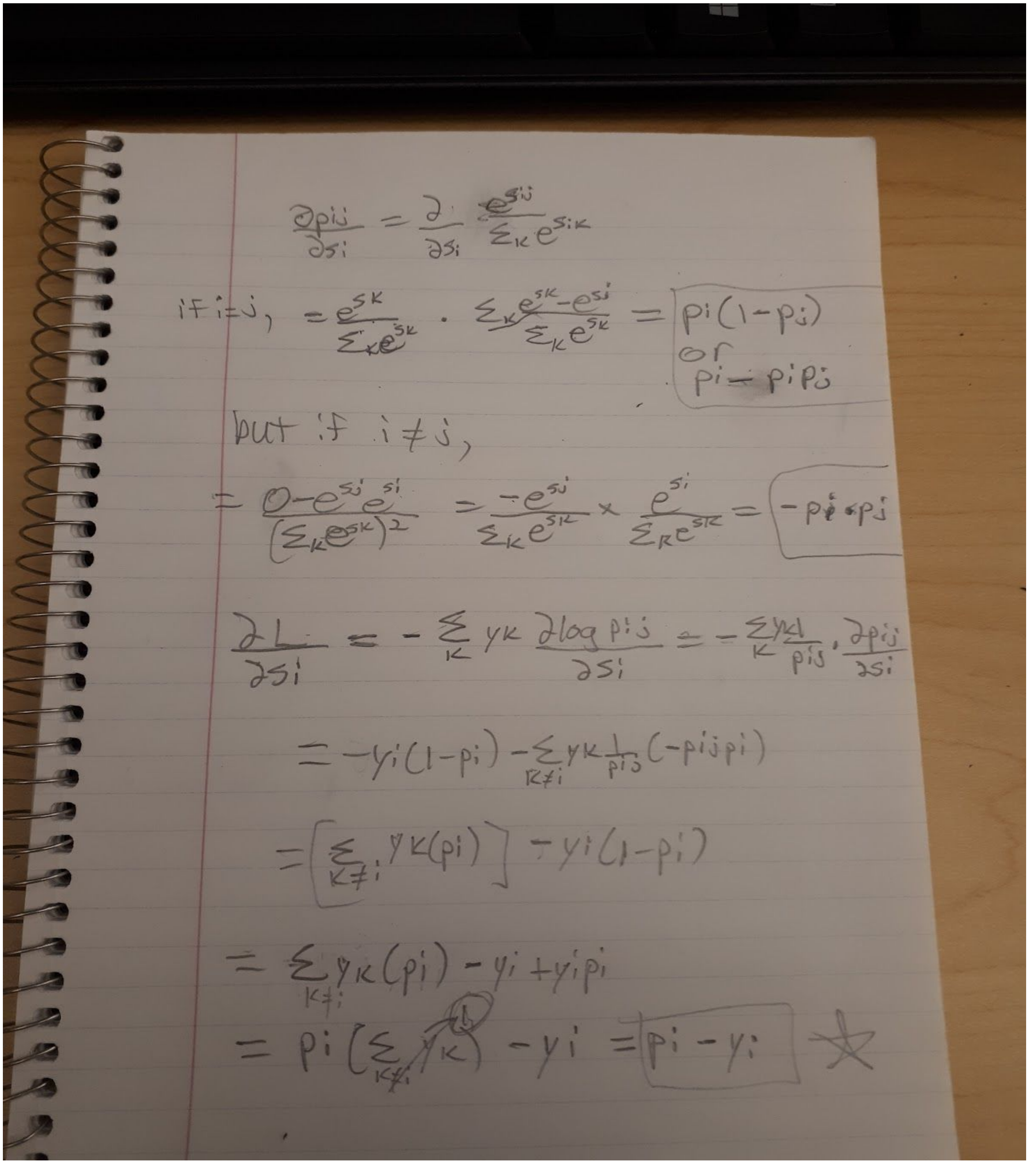
Parm Johal
V00787710

February 8, 2019

Abstract

This report focuses on the linear classifiers which were implemented in Python and executed on the CIFAR 10 data set. The results showed some interesting insights not only between cross-validated and non cross-validated models, but also between cross-validated models with various hyperparameters. The 2 hyperparameters that were varied in the models were the learning rate and regularization strength. The results showed that a learning rate of about 0.0001 and a regularization strength of about 1.0 would give optimal results, relatively speaking. A derivation of the cross entropy was applied to testing the data set. The final section of this report will contain the results of testing the various features and models.

1 Cross Entropy Gradient (Figure 1)



$$\frac{\partial p_{ij}}{\partial s_i} = \frac{\partial}{\partial s_i} \frac{e^{s_{ij}}}{\Sigma_k e^{s_{ik}}}$$

if $i=j$, $= \frac{e^{s_k}}{\Sigma_k e^{s_k}} \cdot \frac{\Sigma_k e^{s_k} - e^{s_i}}{\Sigma_k e^{s_k}} = \boxed{p_i(1-p_j) \text{ or } p_i - p_i p_j}$

but if $i \neq j$,

$$= \frac{0 - e^{s_j} e^{s_i}}{(\Sigma_k e^{s_k})^2} = \frac{-e^{s_j}}{\Sigma_k e^{s_k}} \times \frac{e^{s_i}}{\Sigma_k e^{s_k}} = \boxed{-p_i * p_j}$$

$$\frac{\partial L}{\partial s_i} = -\sum_k y_k \frac{\partial \log p_{ij}}{\partial s_i} = -\sum_k y_k \frac{1}{p_{ij}} \cdot \frac{\partial p_{ij}}{\partial s_i}$$

$$= -y_i(1-p_i) - \sum_{k\neq i} y_k \frac{1}{p_{ij}}(-p_{ij} p_i)$$

$$= \left[\sum_{k\neq i} y_k(p_i)\right] - y_i(1-p_i)$$

$$= \sum_{k\neq i} y_k(p_i) - y_i + y_i p_i$$

$$= p_i\left(\overset{1}{\cancel{\sum_{k\neq i} y_k}}\right) - y_i = \boxed{p_i - y_i} \quad ☆$$

Figure 1: Derivation of the gradient for cross entropy

2 Training Curve

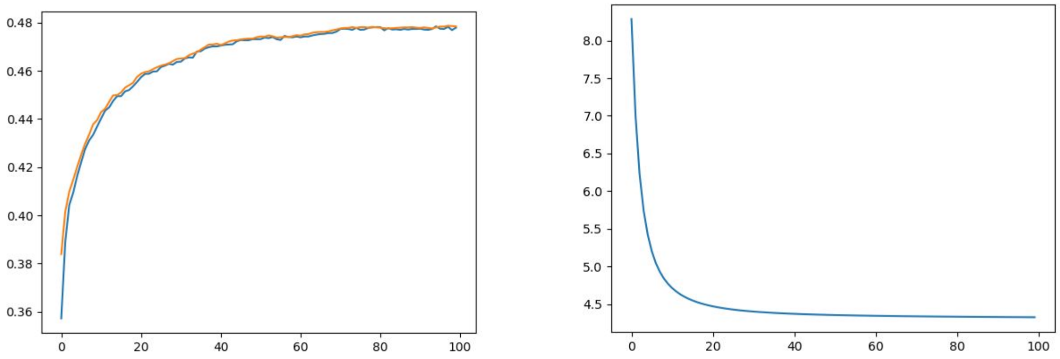


Figure 2: accuracy curve(left) and loss curve(right) for the HOG feature option

For testing the HOG feature option, the initial validation accuracy was roughly 11.8%, which was expected before training the linear classifier and can be seen in the accuracy curve in figure 2. As the training finished, the training accuracy was 47.79% with the validation accuracy being 47.84%, which was an impressive finish after 100 rounds. The training loss ended up being 4.32, which was reduced from 8.29%. The accuracy curve seems stable in this model, along with the loss curve declining smoothly.

3 Cross Validation Results

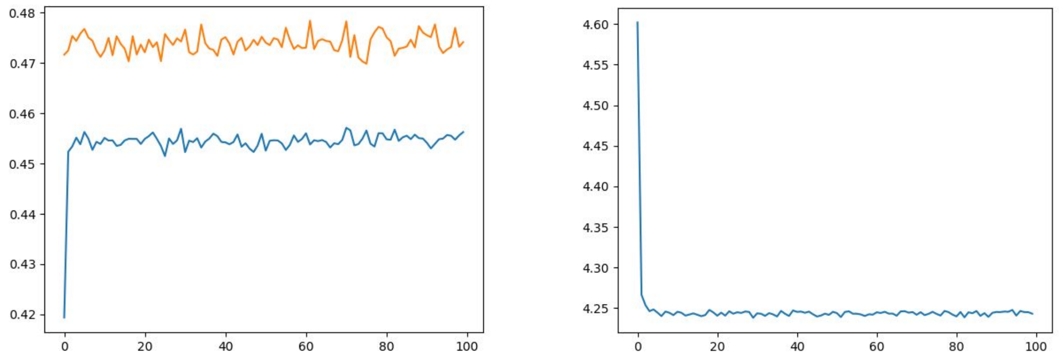


Figure 3: CV results of HOG feature with learning rate=0.01 and regularization=0.5. Accuracy(left) and loss(right)

With a learning rate of 0.01 and regularization value of 0.5, the table below summarizes the initial and final values of train loss, train accuracy, and validation accuracy. The average best validation accuracy was 48.47%. As seen in Figure 3, the train data and validation data flatten out but are quite uneven. The loss curve shows a quick decline in the very beginning. The hyperparameters chosen in this model would not be good choices for further training/validating.

Fold 0	Epoch 0	Epoch 99
Train loss	4.56%	4.19%
Train Accuracy	41.90%	45.62%
Validation Accuracy (Initial=9.47%)	47.07%	47.19%
Fold 1		
Train loss	4.57%	4.23%
Train Accuracy	42.00%	46.10%

Validation Accuracy (Initial=9.71%)	47.80%	47.03%
Fold 2		
Train loss	4.64%	4.28%
Train Accuracy	41.51%	44.64%
Validation Accuracy (Initial=10.56%)	46.14%	46.80
Fold 3		
Train loss	4.62%	4.28%
Train Accuracy	42.14%	45.98%
Validation Accuracy (Initial=13.40%)	47.63%	47.67%
Fold 4		
Train loss	4.62%	4.24%
Train Accuracy	42.13%	45.79%
Validation Accuracy (Initial=8.82%)	47.19%	48.39%

Table 1: resulting data from HOG feature HOG feature with learning rate=0.01 and regularization=0.5.

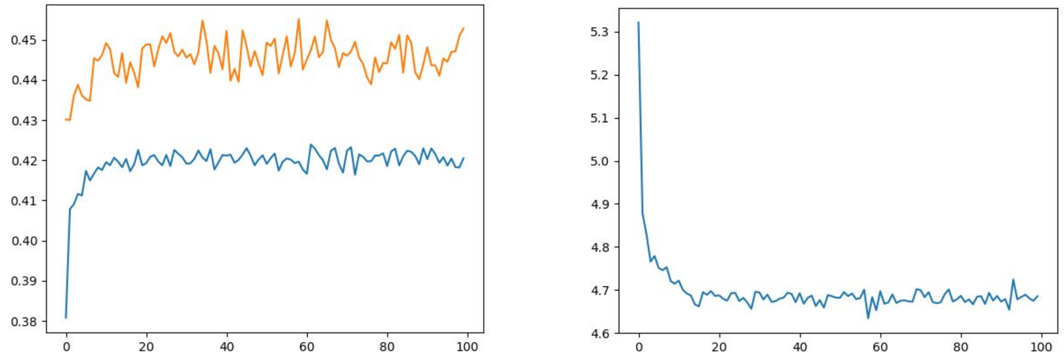


Figure 3: CV results of HOG feature with learning rate=0.1 and regularization=0.005.

Accuracy(left) and loss(right)

With a learning rate of 0.1 and regularization value of 0.005, the table below summarizes the initial and final values of train loss, train accuracy, and validation accuracy. With an average best validation accuracy of 46.39%, it is not as accurate as the previous CV models. It can even be seen in the table that the train loss is 4.68% at the very end of fold 0, whereas the previous result was 4.19%. Based off of these 2 CV models, it can be said that the regularization strength hyperparameter should be a value near 1.0 and the learning rate hyperparameter should be a value near 0.0001. These hyperparameters will give more accurate results and a smoother accuracy curve and a better decline in the loss gradient.

Fold 0	Epoch 0	Epoch 99
Train loss	5.34%	4.68%
Train Accuracy	38.53%	42.21%
Validation Accuracy (Initial=9.04%)	44.11%	46.21%
Fold 1		
Train loss	5.35%	4.78%
Train Accuracy	37.20%	41.50%
Validation Accuracy (Initial=10.36%)	43.04%	45.71%
Fold 2		
Train loss	5.25%	4.58%
Train Accuracy	38.44%	42.29%
Validation Accuracy (Initial=12.10%)	43.57%	46.18%
Fold 3		
Train loss	5.37%	4.68%

Train Accuracy	37.62%	41.92%
Validation Accuracy (Initial=12.96%)	42.02%	44.77%
Fold 4		
Train loss	5.29%	4.71%
Train Accuracy	38.60%	42.32%
Validation Accuracy (Initial=12.02%)	42.33%	43.57%

Table 2: resulting data from HOG feature HOG feature with learning rate=0.1 and regularization=0.005.