THE UNIVERSITY OF MELBOURNE

Assignment 2
COMP90051-Statistical Machine Learning

Muhammad Ali Qadar - 727121

Department of Electrical and Electronic Engineering
The University of Melbourne

October 11, 2017

Multiclass Classifier using Multilayer Perceptron

1 Motivation

Multilayer perceptron (MLP) is a type of feed forward artificial neural network (ANN) that consists of three layers which are backpropagated to compute error gradients. In general MLP is powerful in sense that it can approximate any bounded continuous function with arbitrary precision and can perform multiclass classification via one-hot encoding scheme. Larger hidden layers incorporate more representation capacity. However, there is a tradeoff between reasonable learning(underfitting) and not just memorizing training data (overfitting).

2 Data analysis

In this implementation we have developed MLP with one input, one hidden and one output layer to perform hand written digit classification for the given dataset.

2.1 Parameter Initialization

We initialized MLP function with small weights and with different sizes of hidden units (380) and min-batches (in each epoch split the training data into 130 minibatches followed by computation of gradient of minibatch). Learning rate η is set to a small number 0.002 in order avoid divergence. L2 regularization value of 0.03 is used to avoid overfitting and control the decay of weights. The parameter α along with previous weight is added to current weight($\Delta \boldsymbol{w}_t$) for faster learning $\Delta \boldsymbol{w}_t = \eta \bigtriangledown J(\boldsymbol{w}_t) + \alpha \Delta \boldsymbol{w}_{t-1}$. Where decrease constant d is used to decrease learning rate η over time for better convergence i.e learning rate $= \eta / 1 + t \times d$, t is the epoch number.

2.2 Convergence analysis

We trained 35000 samples using parameters defined earlier, the execution of the MLP took approximately 3 hour and 25 minutes on a quadcore i7 desktop computer, this time can be reduced to minimum 30 min if we reduce the number of hidden units and minibatches however, this effects accuracy greatly. During each epoch we saved the cost to visualize the convergence. We have shown the plot of cost in Fig.1 as we can see it looks very noisy this is because of process of minibatch optimization. Therefore, we averaged cost function over each minibach to obtain a smoother representation

shown in Fig.2. It is evident that algorithm has converged after almost 461 epoch (60000/130 = 461), to further evaluate the performance, we calculated prediction accuracy of training that is 99.47% and accuracy with the test set determined after submission in Kaggle was 94.8%. This shows that MLP model has correctly classified training digits with good accuracy. Fig.3 shows correctly classified images by MLP, we could be able to show misclassified examples in case original labels are available.

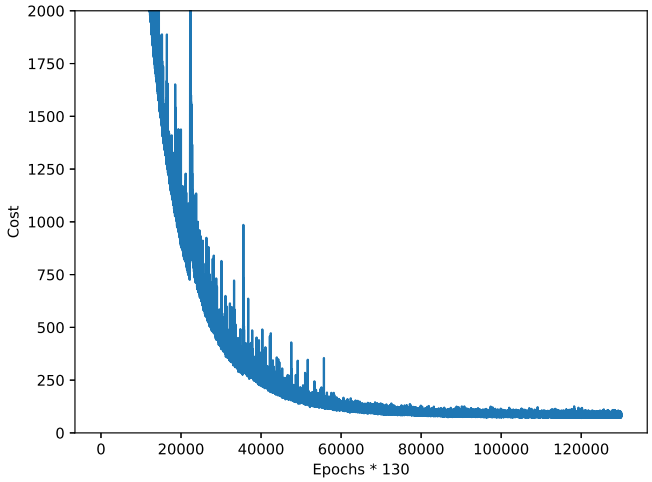

Figure 1: Cost of MLP algorithm using 130 minibatches over each epoch

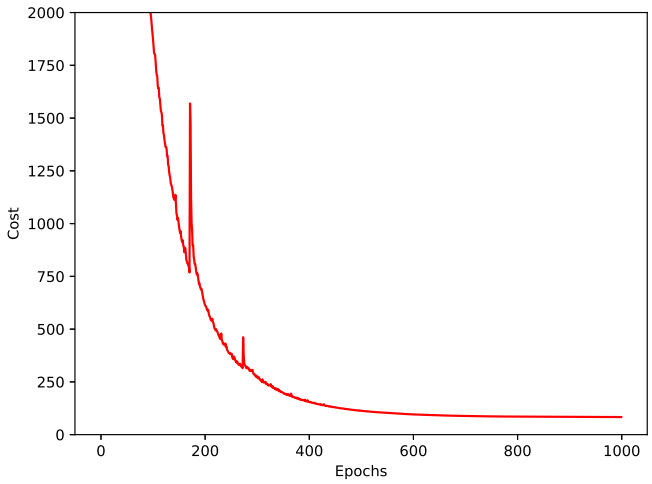

Figure 2: Averaged cost of MLP algorithm using 130 minibatches over each epoch

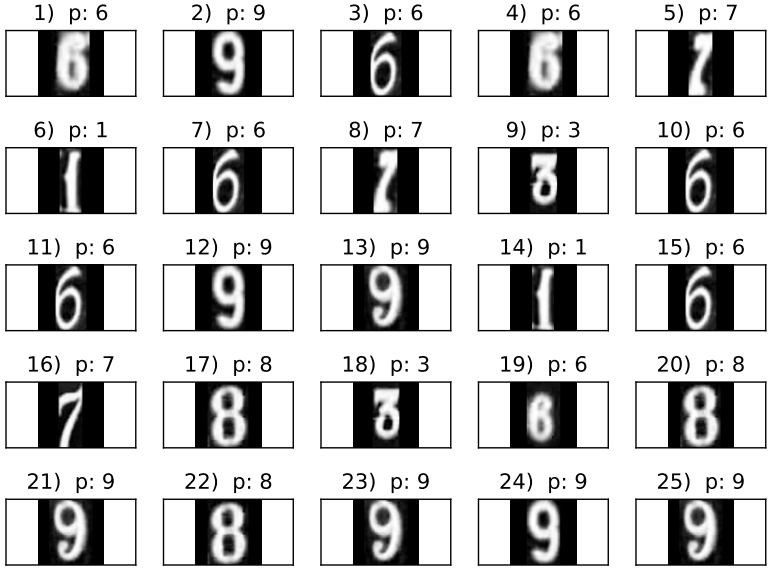

Figure 3: MLP Predicted digits labeled at the top of each image denote by p