Deep Learning – ANN Assignment - Report

Auxiliary functions

1. **train_validation_split(X, Y, train_size)** – split the train set into train and test sets. First it creates list of indices, shuffles the indices, and finally chooses 80% of the indices as the train set and the rest as the validation set.

   X – train set examples

   Y – train set labels

   train_size – percentage of the train set

   return – tuples of (x_train, y_train), (x_validation, y_validation)

2. **plot_model_history(costs_list, type)** – plot a graph of cost per 100 iterations.

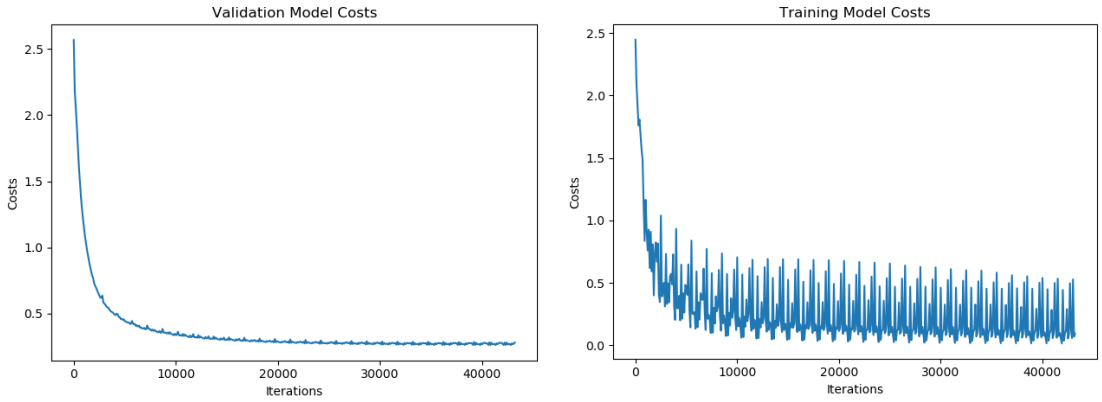
   costs_list - cost per 100 iterations

   type – string, "validation" or "training" – for the graph's title.

Stopping Criterion

If for 200 iterations the validation loss of the last iterations is higher by 0.005 than the previous 100 iterations, then stop the training.

Results without batch normalization:



Graph explanation: The training model costs graph shows the cost after 100 iterations for the 100's batch at each round of 100 iterations. This is the reason why the graph has shape of decreasing zigzag, because the batches are very small (32 samples) and very different.

On the other hand, the validation model costs graph was computed on the entire validation set and therefore the graph is smoother.

Number of iterations: 50,000

Number of iterations until convergence: 43,200

Epochs: 28

Batch size: 32

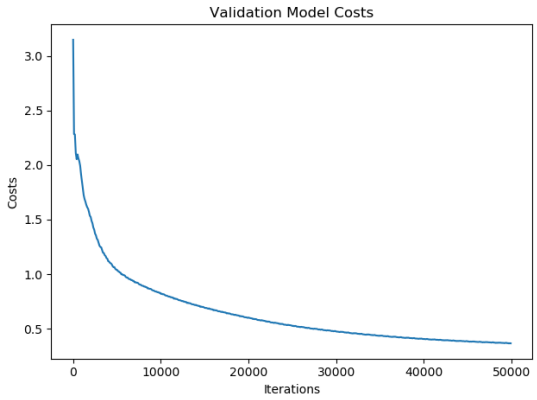
Validation accuracy: 0.926

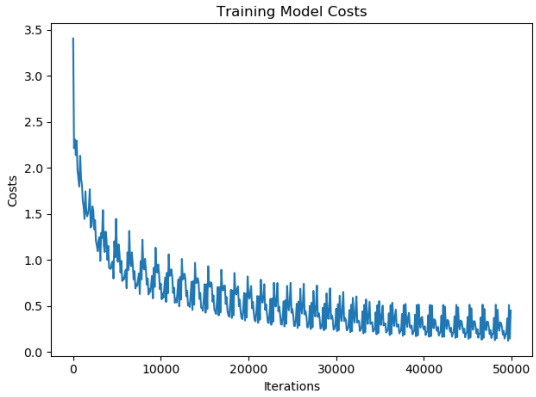
Train accuracy: 0.9485833333333333

Test Accuracy: 0.931

Total time: 141.21866154670715 seconds

Results with batch normalization:





Graph explanation: The training model costs graph shows the cost after 100 iterations for the 100's batch at each round of 100 iterations. This is the reason why the graph has shape of decreasing zigzag, because the batches are very small (32 samples) and very different.

On the other hand, the validation model costs graph was computed on the entire validation set (12,000 samples) and therefore the graph is smoother.

Number of iterations: 50,000

Number of iterations until convergence: 50,000

Epochs: 33

Batch size: 32

Validation accuracy: 0.9091666666666667

Train accuracy: 0.9165208333333333

Test Accuracy: 0.9124

Total time: 163.06582188606262 seconds.

Comparison: As we can see from the results, the running time of the network with batch normalization was increased compared to the network without batch normalization. The

reasons are that the normalization phase was added, and the network needed more iterations to converge. In addition, the accuracies without batch normalization were higher than the accuracies with batch normalization. In our intuition, the batch normalization phase was supposed to improve the accuracy and converge faster, but because we implemented a simple version it affects the results.