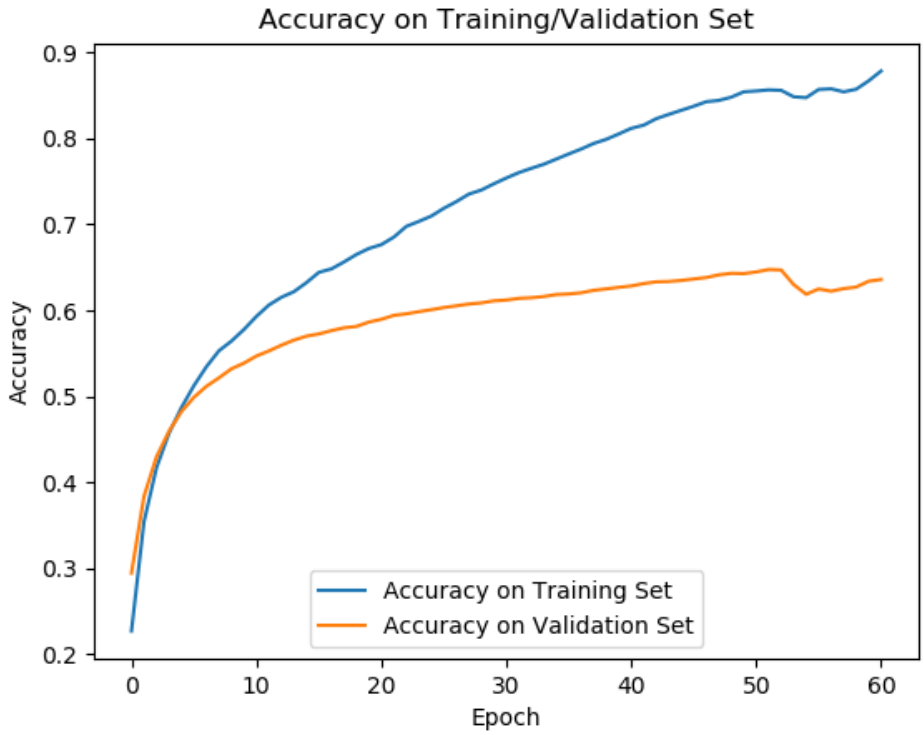
Donovan Guelde
CSCI-5922
Assignment 4

1.a The 50,000 training examples were split into a training and validation set in a ratio of 80:20, giving 40,000 training examples and 10,000 examples in the validation set. Training/validation sets were drawn via many-fold validation. The original training set (the full 50,000 examples) was shuffled, then split into training and validation sets. Code was based on the tutorial found at http://speech.ee.ntu.edu.tw/~tlkagk/courses/MLDS_2017/Lecture/Tensorflow_CNN.pdf, which was in turn based on the tensorflow tutorial.

1.b. My first network was a simple cnn consisting of input;c1(32 5x5 kernel)max pooling (2x2), c2(32 5x5 kernel), maxPool2 (2x2), and a fully connected layer of 1024 nodes.

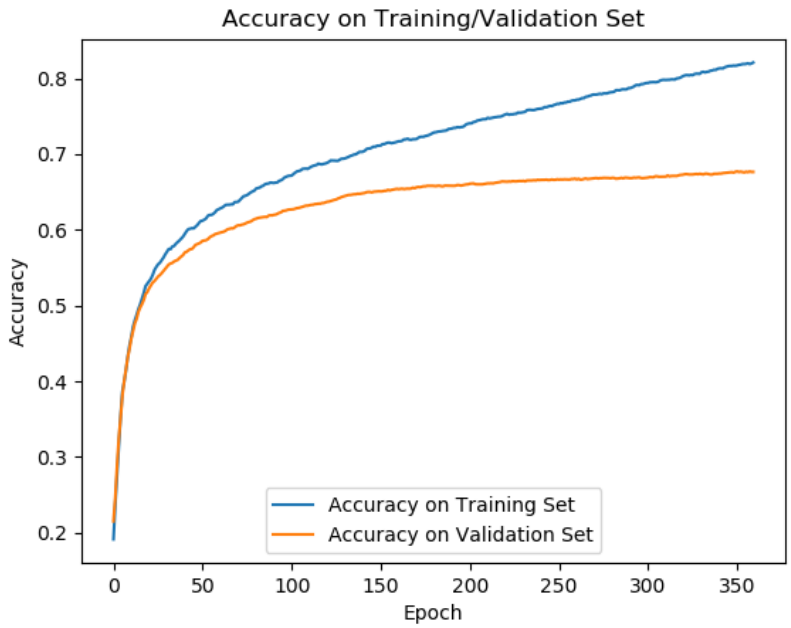


This network gave accuracy on the validation set of 0.6475, but with a large degree of overfitting. Taking this network as a starting point, I made efforts to reduce the degree of overfitting. I attempted to do this by continually reducing the number of neurons in the fully connected layer. With all other layers remaining the same, the following results were achieved:

# connected neurons	512	256	128	64	32	16	8	4
final accuracy on validation set	.6389	.6553	.6566	.6640	.6690	.6772	.6469	.4051

Donovan Guelde
CSCI-5922
Assignment 4

As the number of fully connected neurons in the final layer decreased, accuracy increased slightly while overfitting seemed to decrease. With 16 nodes in the final layer, the following results were obtained:



Overfitting appears to be present nut to a lesser degree, while accuracy increased slightly, to 0.6772.

Donovan Guelde
CSCI-5922
Assignment 4

With only 4 nodes in the fully connected layer, both accuracy and overfitting decreased by a large degree. 4 nodes, it appears, creates a bottleneck that is too narrow for a reliable classification.

Performance of net with 4 nodes in fully connected layer

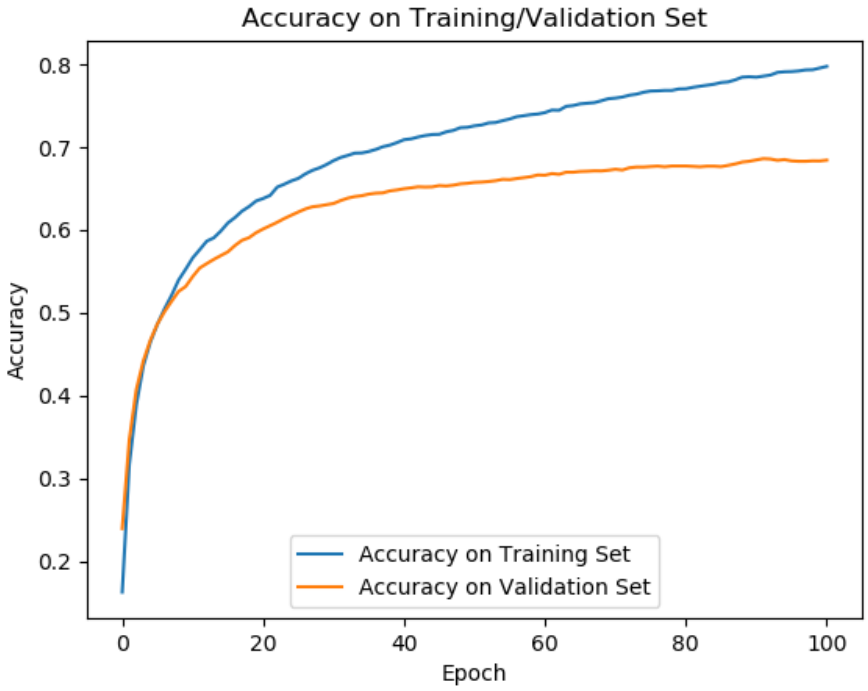
To remove the bottleneck, and hopefully allow for better classification, an additional fully-connected layer was added, and changing the location of the pooling layers was modified. Best results were obtained with the following configuration:

c1 = 5x5 kernel, 32 filters
c2 = 5x5 kernel, 64 filters
maxPool1 = 2x2
maxPool2 = 2x2
connected layer 1 = 16 nodes
connected layer 2 = 64 nodes
output layer = 10 nodes
AdamOptimizer
validation accuracy .6747

Donovan Guelde
CSCI-5922
Assignment 4

Results were similar to those obtained from the original configuration. In a final attempt to improve accuracy, additional convolution layers were added. The best results were obtained from the following network:

c1 = 3x3 kernel, 32 filters
c2 = 3x3 kernel, 64 filters
c3 = 3x3 kernel, 64 filters
c4 = 3x3 kernel, 64 filters
c5 = 3x3 kernel, 64 filters
maxPool1 = 2x2
maxPool2 = 2x2
maxPool3 = 2x2
connected layer 1 = 64 nodes
output layer = 10 nodes
no augmentation, dropout, etc. Simple cnn
AdamOptimizer
validation accuracy .6861



2.a. Final test set results on this network gave an accuracy of .6764