Deep Learning Assignment

CS985/CS987: Advanced Machine Learning

Bethany Thompson 201865137
Bethany.thompson@strath.ac.uk

Danaya Lorpattanakasem 201880434
danaya.lorpattanakasem.2018@uni.strath.ac.uk

Wei-An Chen 201865113
weian.chen.20108@uni.strath.ac.uk

Dataset 1: MNIST

The first dataset chosen for this assignment was the MNIST dataset which is a database of handwritten numbers ranging from 0 to 9. It contains 60,000 training examples and 10,000 test examples. The digits are presented in greyscale and have been centered in fixed size images of size 28x28 pixels.

Combination 1

The network architecture with the highest test and training accuracy was found to be a convolutional neural network (CNN) with two convolutional layers, each followed by a down-sampling pooling layer, two fully connected hidden layers and an output layer. This was expected as CNNs preserve the spatial structure of the image. This architecture is shown in figure 1.

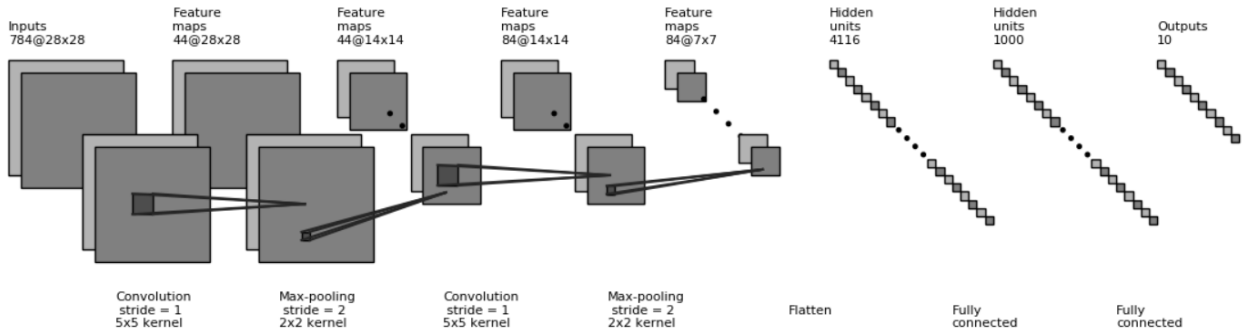

Figure 1: MNIST Combination 1 CNN Architecture

The features of combination 1 are as follows:

- Conv 1: 44 filters, LRF = 5x5, stride = 1, pool shape = 2x2, stride =2, max pooling
- Conv 2: 84 filters, LRF = 5x5, stride = 1, pool shape = 2x2, stride = 2, max pooling
- FC1: 4116 neurons (7x7x84)
- FC2 layer: 1000 neurons
- Output layer: 10 neurons
- ReLU activation
- Adam optimizer
- Cross-entropy cost function with softmax

CNN architectures with different numbers of layers were investigated. However, less layers resulted in a poorer accuracy score and more layers than the proposed architecture resulted in marginal improvement to accuracy given the increase training time. The architecture in figure 1 achieves both a high training and testing accuracy score, as well as efficient training time. The sigmoid activation function was found to achieve very poor accuracy with the CNN architecture so ReLU was chosen instead. The Adam optimizer was consistently found to perform better than standard stochastic gradient descent. This was expected as the Adam optimizer finds adaptive learning rates for different parameters by using estimates of the first and second gradient moments, whereas in standard gradient descent maintains a constant learning rate throughout training.

The weights for each layer were initialized with mean = 0 and a standard deviation of $\sigma = 2/sqrt(n_{inputs} + n_{outputs})$. This sharper gaussian for random weight initialization helps to avoid neuron saturation. Altering the LRF to 2x2 or 6x6 was found to significantly reduce the training and test accuracy so an LRF = 5x5 was deemed the best option. Different amounts of dropout were applied to FC2 to reduce overfitting. The CNN model was only tested on part of the test to avoid the testing time taking too long. However, on inspection, testing on this subset was found to produce representative training accuracies.

Combination 2

Combination 2 is a classic fully connected neural network with three hidden layers with 300, 100 and 50 neurons respectively. Three hidden layers was deemed most appropriate to avoid extensive training time while still achieving high accuracies. To avoid the weights becoming very large regularization was applied to the cost function to give a higher cost to large weights and reduce overfitting. L2 regularization was found to perform better than L1 regularization.

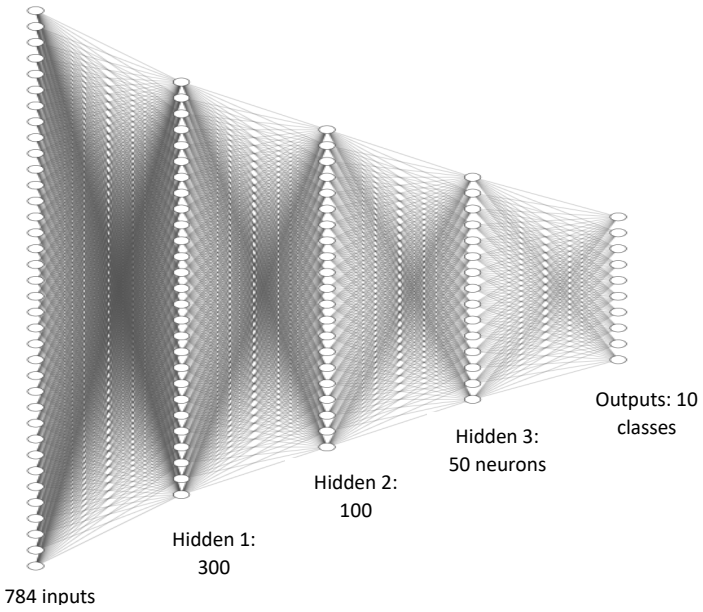

Figure 2: MNIST Combination 2 FC Architecture

The features of combination 2 are as follows:

- Layer 1 = 300 neurons
- Layer 2 = 100 neurons
- Layer 3 = 50 neurons
- Sigmoid activation
- Gradient descent optimizer
- Cross-entropy cost function with softmax

Table 1 shows the training, validation and testing accuracies for different variations of combinations 1 and 2.

Example	Combination	Parameters and Configuration	Training Accuracy	Validation Accuracy	Testing Accuracy
A	**1**	**LR :0.001, epochs:600, batches:100, dropout:0.5 (applied to FC2)** **Network described above.** **Dropout applied to FC2.**	**99.0%**	**98.2%**	**99.2%**
B	-	Combination 1 but conv1 has 32 filters, conv2 has 64 filters and FC1 has 3136 (7x7x64) neurons	99.0%	98.1%	98.8%
C	-	LR :0.01, epochs:300, batches:140, dropout:1 (none) Network described in B.	97.9%	95.8%	95.7%
D	2	LR :0.01, epochs:300, batches:140, L2 regularisation:0.4 Network described above.	96.0%	95.0%	94.8%
E	-	Combination 2. LR :0.01, epochs:100, batches:50, L2 regularisation:0.4	94%	93.8%	93.5%
F	-	Combination 1 with alterations described in B. LR :0.001, epochs:600, batches:100, dropout:0.8 Dropout applied to FC2. Gradient Descent Optimiser.	91.0%	90.5%	92.2%
G	-	Combination 2 but with L1 regularisation instead.	90.0%	91.1%	90.8%

		LR :0.01, epochs:300, batches:100, L1 regularisation:0.4			
H	-	Combination 2. LR :0.01, epochs:300, batches:50, L2 regularisation:0.5	96.0%	90.3%	89.8%

Table 1: Comparison of Architectures for Dataset 1: MNIST

Dataset 2: Forests

The second dataset chosen for this assignment was the Forest Cover Types dataset. This is a database consisting of many 30×30m^2 patches of forest from the US Forest Service (USFS) Region 2 Resource Information System. The task was to forecast each patch's cover type according to 7 different classes. It contains 522,911 training examples and 58,101 testing examples.

Combination 1

Combination 1 is a fully connected neural network with four hidden layers. It has 54 inputs, four hidden layers with 50, 40, 30 and 20 neurons respectively, and the output layer has 7 neurons. This architecture achieves the highest test (94.785%) and training (95.960%) accuracy with early stopping and training for either 280 or 300 epochs. This combination uses the Adam Optimizer, ReLU activation function applied to all layers, a learning rate of 0.001, a batch of 100 and a dropout of 0.5. This architecture is shown in figure 3.

The weights for each fully connected layer were initialised with mean = 0 and a standard deviation of $\sigma = 2/sqrt(n_{inputs} + n_{outputs})$. This sharper gaussian was chosen for initial weight sampling to prevent the unstable gradient problem. To avoid overfitting and encourage better model generalisation to new data, two regularisation methods were used during training: early stopping and dropout. Early stopping interrupts training when the validation set performance begins to decrease. Dropout was applied to all layers and the amount was varied. In each training step, every neuron has the possibility of being temporarily ignored or "dropped" by the network and then reinstated again in the next training step.

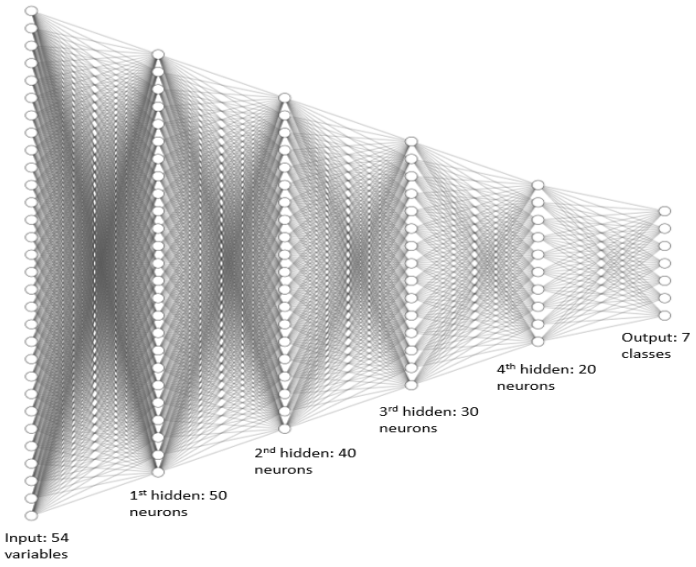

Figure 3: Forests Combination 1 FC Architecture

A cross entropy cost function was chosen instead of a quadratic MSE to encourage faster learning during training. It is more appropriate for classification tasks such as this because it measures the probability error in tasks where classes are mutually exclusive. In this way, cross entropy punishes models that have a low probability of estimating the target category. A SoftMax activation function was chosen for the output layer as the classes are mutually exclusive in this dataset.

Combination 2

Combination 2 is a classic fully connected neural network with similar architecture to combination 1. It has 54 inputs with three hidden layers with 50, 50 and 50 neurons respectively, and an output layer with 7 neurons. It uses a Gradient Descent Optimizer, ReLU activation function on layers, and its optimal hyperparameters were found to be: learning rate = 0.05, epochs = 300, batches = 500 and dropout = 0.5. Combination 2 achieves the second highest test and training accuracies of 94.140% and 96.393% repsectively.

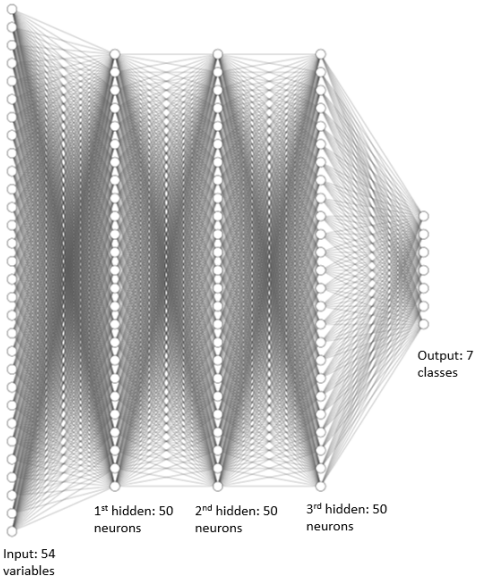

Figure 4: Forests Combination 2 FC Architecture

Alternative Architectures

The table 2 show six different alternative combinations of Forest Cover Types dataset. Among them, different types of hyperparameters, optimizers, regularization and activation functions have been tried. The hyperparameters and the number of layers had a great effect on the training time and results, so the challenge was to find the balance between training time and accuracy score.

Example	Combination	Parameters and Configuration	Training Accuracy	Validation Accuracy	Testing Accuracy
A	**1**	**LR :0.001, epochs:300 and 280, batches:100 Network described above.**	**95.96%**	**94.263%**	**94.785%**
B	2	LR :0.05, epochs:300, batches:500 Network described above.	96.393%	94.196%	94.140%
C	3 layers	LR :0.01, epochs:300 and 230, batches:500, Optimizer: Adam Optimizer, AF: ReLU	93.988%	92.762%	93.004%
D	4 layers	LR : 0.1, epochs:300 and 230, batches:1000, Optimizer: Gradient Descent Optimizer, AF: Leaky ReLU	93.694%	92.496%	92.658%
E	2 layers	LR : 0.0005, epochs:500 and 310, batches:300, Optimizer: Adam Optimizer, AF: Leaky ReLU	93.311%	90.937%	90.878%
F	2 layers	LR : 0.01, epochs: 1000, batches:1000, Optimizer: Gradient Descent Optimizer, AF: Sigmoid	84.985%	84.410%	84.499%

Table 2: Comparison of Architectures for Dataset 2: Forests