

Deep Learning Lab WSE 2018 Exercise 1

Assignment: Feed-forward Neural Network

Implementation: Implement a feed-forward neural network by completing the provided stub this includes:

- Possibility to use 2-4 layers
- Sigmoid/tanh and ReLU for the hidden layer
- Softmax output layer
- Optimization via gradient descent (gd)
- Optimization via stochastic gradient descent (sgd)
- Weight initialization with random noise (normal distribution with std. deviation)

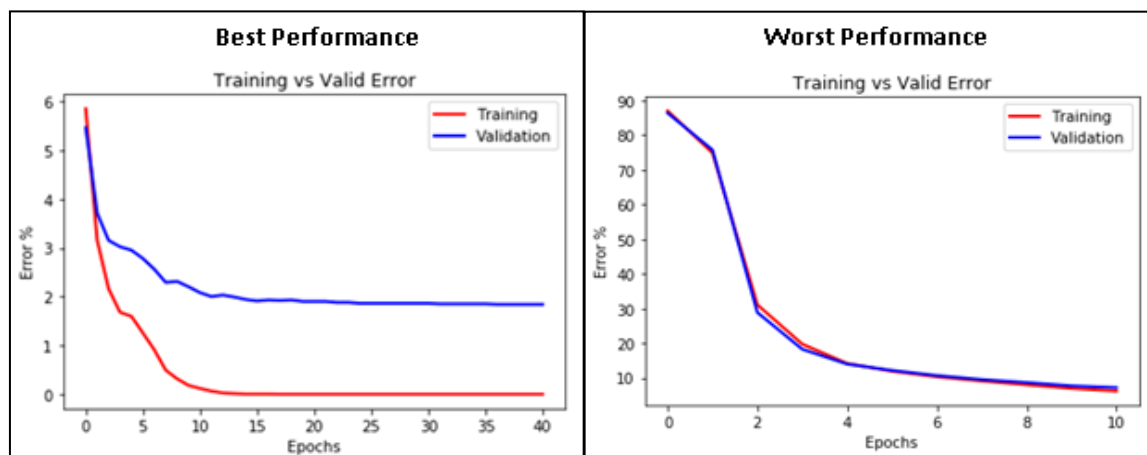
Results: The following results were obtained with the iteration of the parameters listed in the table.

Fixed conditions: Neural network of two hidden layers

ReLu as activation function, Stochastic Gradient Descent, Softmax

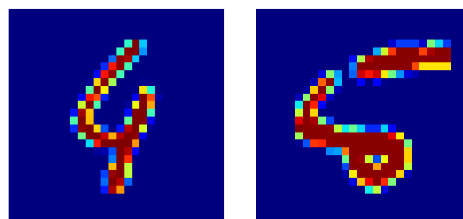
Batch size 64 units

| Neural Combination | Neurons Combination | #Samples Train_Set | EPOCH | Learning Rate | Training Error | Validation Error | Loss |
|--------------------|---------------------|--------------------|-------|---------------|----------------|------------------|--------|
| Worst | 100, 100 | 10,000 | 10 | 0,1 | 6,2% | 7,18% | 0,0002 |
| Best | 500, 300 | 40,000 | 40 | 0,23 | ~0% | 1,84% | 0,2060 |



Conclusion: It is shown that the best performance is achieved by a combination of several parameters, which we only “guessed” by “trial and error”. Better performance could be reach by the optimization of these parameters.

The images show below were part of the incorrect labeled batch, predicted as “9” and “6” respectively, been “4” and “5” the correct numbers. In my opinion is hard to differentiate between these values even by a human eye/brain.



Feedback: Extremely complex to figure it out how to implement the wanted sections of the code. Implementation/Solution of the code was over 35 hours.