MSc Machine Learning

DD2424 Deep Learning in Data Science

Assignment 3 Bonus: Image classification with a k-layer network

---

Author:
Alexandros Ferles
ferles@kth.se

Professor:
Josephine Sullivan

Contents

1 Exercise 1: Optimize the performance of the network

Try at least 3 improvements (beyond using the all the training data) to help bump up performance and report your results.

The following optimizations have been performed:

1.1 Applying batch normalization after the activation function

In this setting the 3 best found pairs for the learning rate η and the regularization amount λ for a 3-layer network with batch normalization are applied:

- $(\eta, \lambda) = (0.034875895633392565, 10^{-5})$

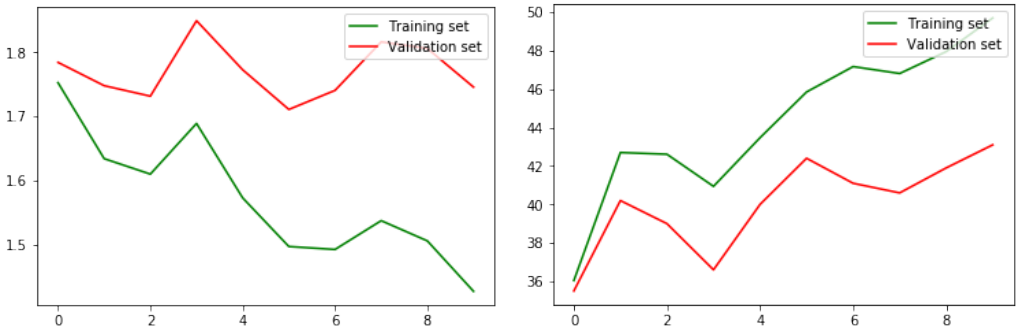

Figure 1: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set for batch normalization applied after the activation function with $(\eta, \lambda) = (0.034875895633392565, 10^{-5})$

Test set accuracy achieved: 43.34%

- $(\eta, \lambda) = (0.007986719995840757, 10^{-6})$

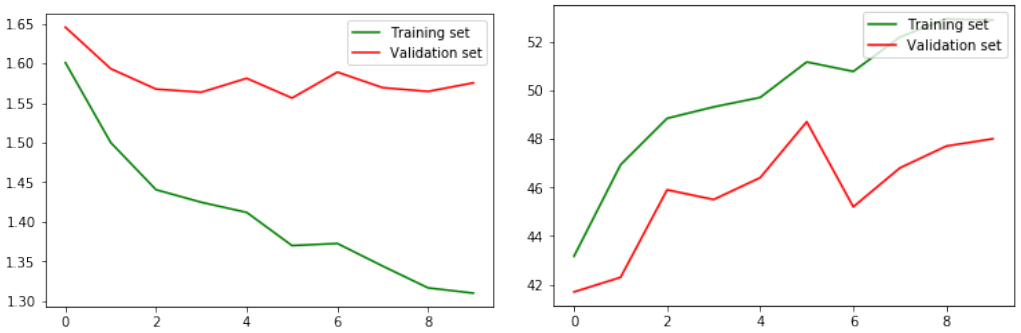

Figure 2: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set for batch normalization applied after the activation function with $(\eta, \lambda) = (0.007986719995840757, 10^{-6})$

Test set accuracy achieved: 39.23%

- $(\eta, \lambda) = (0.012913581489067944, 10^{-5})$

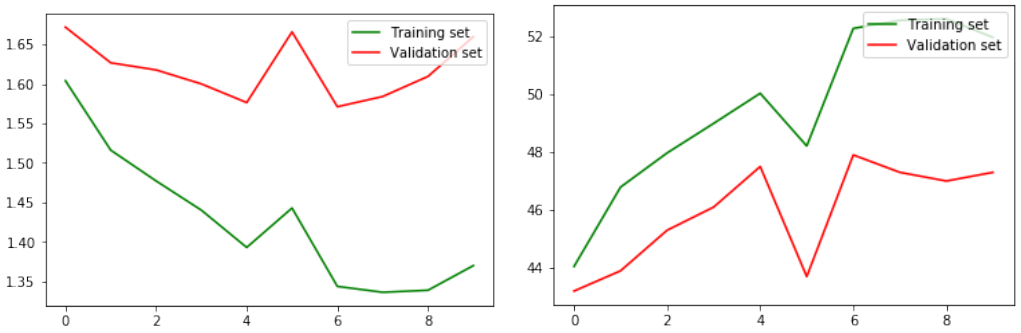

Figure 3: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set for batch normalization applied after the activation function with $(\eta, \lambda) = (0.012913581489067944, 10^{-5})$

Test set accuracy achieved: 43.83%

Learning becomes more unstable, and the test set accuracy performance achieved is lower in comparison with the setting where batch normalization is applied before the activation function.

1.2 Augmenting the input data by applying a small random noise on the fly

.

A small random noise is applied to each batch of data before the process, based on a percentage of the standard variance of the batch itself.

- $(\eta, \lambda) = (0.034875895633392565, 10^{-5})$

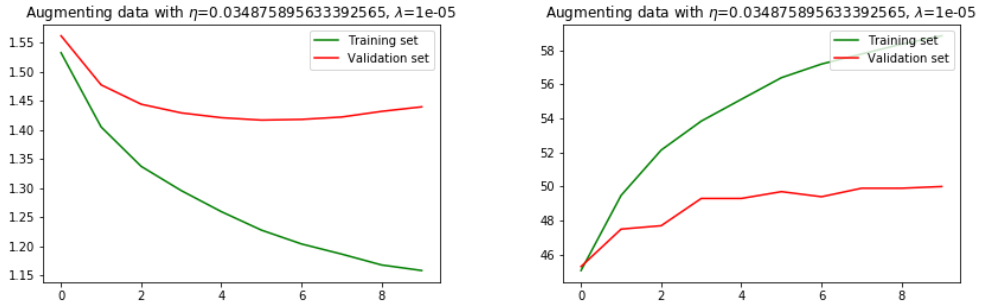

Figure 4: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set with augmenting input data with random noise on the fly for $(\eta, \lambda) = (0.034875895633392565, 10^{-5})$

Test set accuracy achieved: 49.67%

- $(\eta, \lambda) = (0.007986719995840757, 10^{-6})$

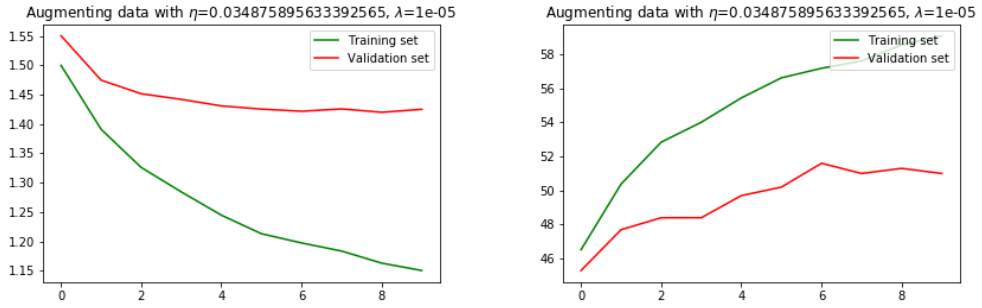

Figure 5: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set with augmenting input data with random noise on the fly for $(\eta, \lambda) = (0.007986719995840757, 10^{-6})$

Test set accuracy achieved: 51.25%

- $(\eta, \lambda) = (0.012913581489067944, 10^{-5})$

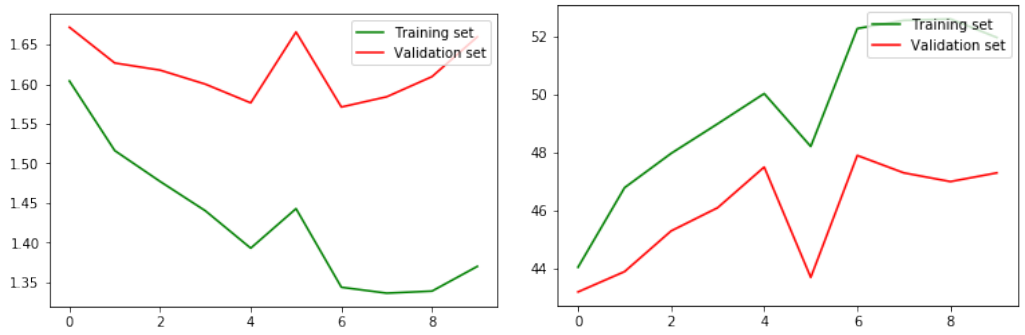

Figure 6: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set with augmenting input data with random noise on the fly for $(\eta, \lambda) = (0.012913581489067944, 10^{-5})$

Test set accuracy achieved: 44.31%

The performance is slightly decreased when random noise is applied.

1.3 Network architecture search

During this setting, an extended search has been conducted looking for good 3-layer architectures. The nodes in each layer tried, vary between 50 and 100 for the first layer, 35 and 50 for the second layer, with a step of 5 layers.

During this search the following results were noticed:

Test set accuracy performance for (55, 35): 51.0
Test set accuracy performance for (55, 45): 51.38
Test set accuracy performance for (60, 35): 51.55
Test set accuracy performance for (65, 35): 51.41
Test set accuracy performance for (65, 40): 51.79
Test set accuracy performance for (70, 35): 51.11
Test set accuracy performance for (70, 40): 51.89
Test set accuracy performance for (70, 45): 51.5
Test set accuracy performance for (75, 35): 51.14
Test set accuracy performance for (75, 40): 51.7
Test set accuracy performance for (75, 45): 51.86
Test set accuracy performance for (80, 35): 51.42
Test set accuracy performance for (80, 40): 52.25
Test set accuracy performance for (80, 45): 51.83
Test set accuracy performance for (85, 35): 52.43
Test set accuracy performance for (85, 40): 51.3
Test set accuracy performance for (90, 35): 51.5
Test set accuracy performance for (90, 40): 51.9

Test set accuracy performance for (90, 45): 51.76
Test set accuracy performance for (95, 35): 51.29
Test set accuracy performance for (95, 40): 51.94
Test set accuracy performance for (95, 45): 52.36

Based on these results, some extra experiments were conducted to define a good value for the number of nodes in the third hidden layer, for a 4-layer network.

Additionally, a more thorough search for a good 4-layer architecture was also conducted, with nodes in the first hidden layer ranging from 50 to 100, nodes in the second hidden layer ranging from 30 to 95 and nodes in the third hidden layer ranging from 20 to 90, with a step of 5.

- 85 nodes in the first layer, 35 nodes in the second layer and experimenting between 20 and 30 nodes in the third layer. Best test set performance is 51.36% achieved by 25 nodes in this layer, with the following loss and accuracy evolution:

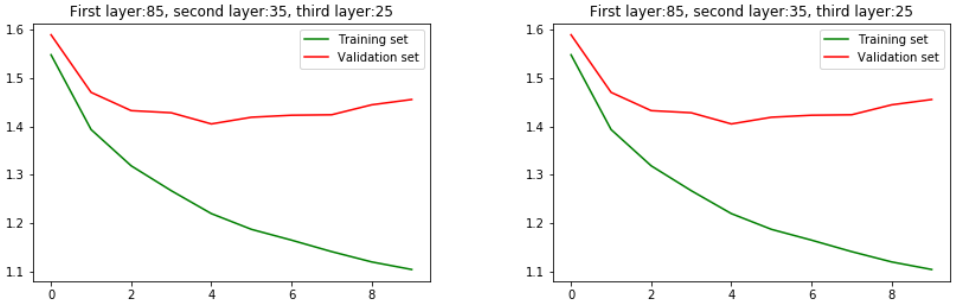

Figure 7: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set with augmenting input data with random noise on the fly for $(\eta, \lambda) = 0.012913581489067944, 1e-04)$ and 85, 35, 25 nodes in the hidden layers respectively

- 95 nodes in the first layer, 45 nodes in the second layer and experimenting between 20 and 40 nodes in the third layer. Best test set performance is 52.17% achieved by 40 nodes in this layer, with the following loss and accuracy evolution:

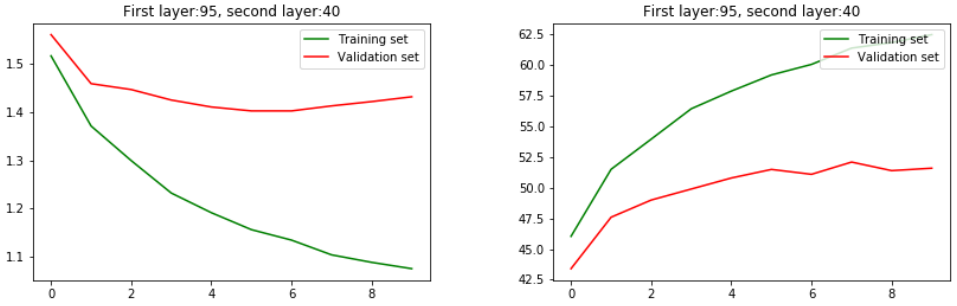

Figure 8: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set with augmenting input data with random noise on the fly for $(\eta, \lambda) = 0.012913581489067944, 1e-04)$ and 95, 45, 40 nodes in the hidden layers respectively

- 80 nodes in the first layer, 40 nodes in the second layer and experimenting between 20 and 35 nodes in the third layer. Best test set performance is 51.54% achieved by 20 nodes in this layer, with the following loss and accuracy evolution:

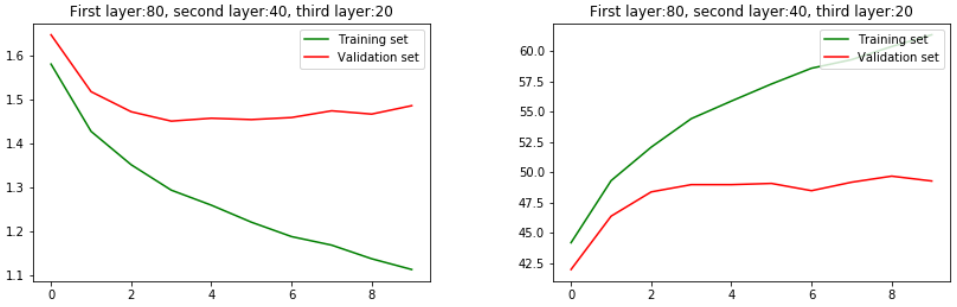

Figure 9: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set with augmenting input data with random noise on the fly for $(\eta, \lambda) = 0.012913581489067944, 1e-04)$ and 80, 40, 20 nodes in the hidden layers respectively

- 75, 65 and 55 nodes in the three hidden layers respectively:

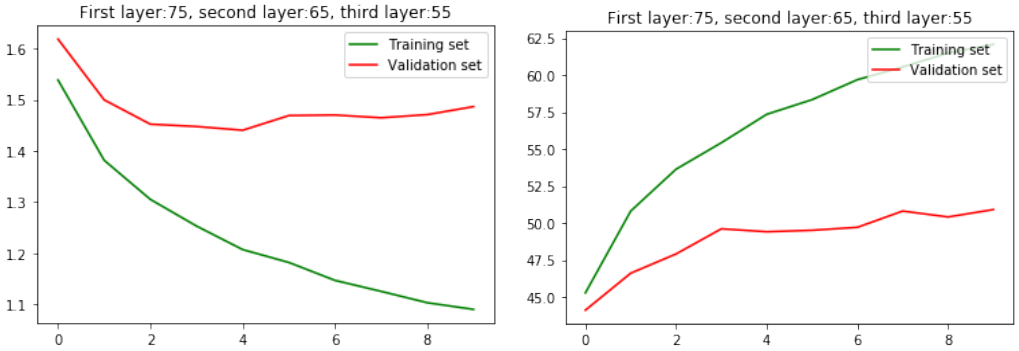

Figure 10: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set with augmenting input data with random noise on the fly for $(\eta, \lambda) = 0.012913581489067944, 1e{-}04)$ and 75, 65, 55 nodes in the hidden layers respectively

Test set accuracy performace: 51.91%

- 100, 60 and 45 nodes in the three hidden layers respectively:

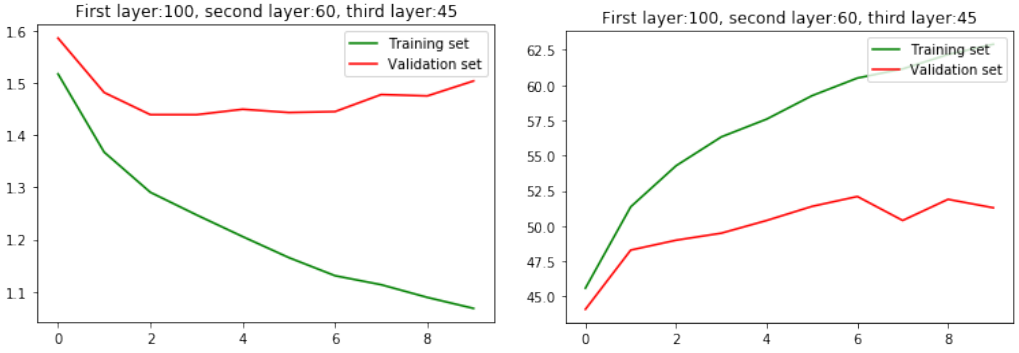

Figure 11: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set with augmenting input data with random noise on the fly for $(\eta, \lambda) = 0.012913581489067944, 1e - 04)$ and 100, 60, 45 nodes in the hidden layers respectively

Test set accuracy performace: 51.9%

- 95, 90 and 75 nodes in the three hidden layers respectively:

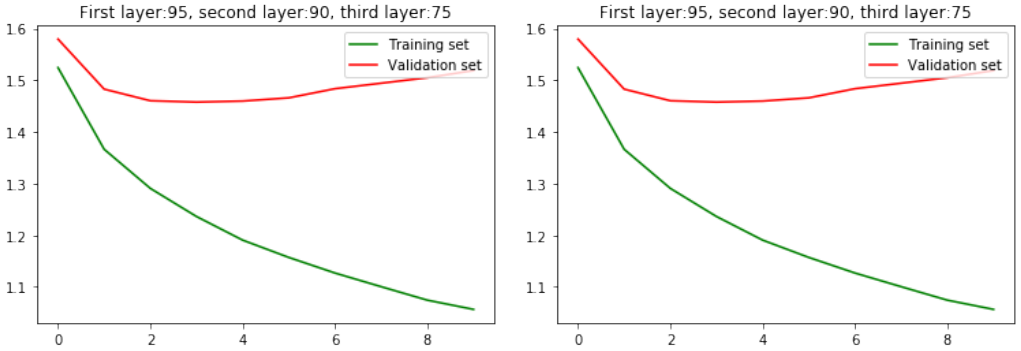

Figure 12: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set with augmenting input data with random noise on the fly for $(\eta, \lambda) = 0.012913581489067944, 1e-04)$ and 100, 60, 45 nodes in the hidden layers respectively

Test set accuracy performace: 52.54%

Hence, expanding the network's architecture to 4 layers can improve the test set accuracy performance, a lot of tweaking is needed in order to define the appropriate amount of nodes used in each hidden layer, and more than one good architectures can be found. Some of these architectures were tried to be expanded in a fourth hidden layer (5 layer-architecture in total), but the performance in the test set slightly dropped in comparison with the 4-layer architectures.

1.4 Hyperparameter search

. In this setting, more experiments are ran, to define a good value for the learning rate η. In addition to that, this time the amount of regularization applied is not searched in a fixed space like $\{10^{-6}, 10{-}5, 10{-}4, 10{-}3, 10{-}2, 10{-}1\}$, but is also search in the log space.

$$\eta \epsilon [0.005, 0.009] \text{ and } \lambda \epsilon [10^{-6}, 10^{-5}]$$
$$\eta \epsilon [0.015, 0.025] \text{ and } \lambda \epsilon [10^{-6}, 10^{-5}]$$
$$\eta \epsilon [0.05, 0.12] \text{ and } \lambda \epsilon [10^{-6}, 10^{-5}]$$
$$\eta \epsilon [0.004, 0.008] \text{ and } \lambda \epsilon [10^{-6}, 10^{-4}]$$
$$\eta \epsilon [0.01, 0.04] \text{ and } \lambda \epsilon [10^{-6}, 10^{-4}]$$
$$\eta \epsilon [0.004, 0.008] \text{ and } \lambda \epsilon [10^{-5}, 10^{-3}]$$
$$\eta \epsilon [0.02, 0.04] \text{ and } \lambda \epsilon [10^{-4}, 10^{-2}]$$

Some of the best results drawn for this search are:

```
BEST PERFORMANCE:  43.05
Best eta:  0.03755382547299599
Best lambda:  3.228604292714727e-05
---------------------------------
SECOND BEST PERFORMANCE:  42.95
Second best eta:  0.005922465774108084
Second best lambda:  2.4167542995897113e-06
---------------------------------
THIRD BEST PERFORMANCE:  42.81
Third best eta:  0.03755382547299599
Third best lambda:  3.228604292714727e-05
```

This search was re-conducted to account for more experiments, with results such as:

```
BEST PERFORMANCE:  42.88
Best eta:  0.06934983494884617
Best lambda:  0.0003205783956857385
---------------------------------
SECOND BEST PERFORMANCE:  42.81
Second best eta:  0.06803047691314905
Second best lambda:  0.00333146870374239
---------------------------------
THIRD BEST PERFORMANCE:  42.54
Third best eta:  0.06565038298774621
Third best lambda:  0.0003886238279263038
```

Another search was also done for high value for etas, but the results were not that promising:

```
BEST PERFORMANCE:  39.94
Best eta:  0.7699879169215156
Best lambda:  0.00014496822822119055
---------------------------------
SECOND BEST PERFORMANCE:  39.6
Second best eta:  0.6276744728118182
Second best lambda:  0.0002627109830322336
---------------------------------
THIRD BEST PERFORMANCE:  39.58
Third best eta:  0.6276744728118182
Third best lambda:  0.0002627109830322336
```

Finally, a last extended search provides us with some more encouraging results:

```
BEST PERFORMANCE:  43.64
Best eta:  0.03159661490765174
Best lambda:  0.00015494946823958237
---------------------------------
SECOND BEST PERFORMANCE:  43.54
Second best eta:  0.013917284924906376
Second best lambda:  1.0632777281477375e-06
---------------------------------
THIRD BEST PERFORMANCE:  43.36
Third best eta:  0.006537886088207429
Third best lambda:  0.0006432091840869645
```

Then, the three top results were selected and tested in the whole dataset for 10 epochs:

- First best setting:

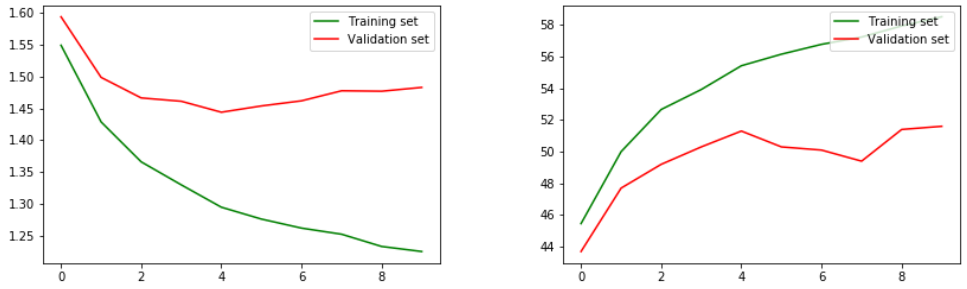

Figure 13: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set for $(\eta, \lambda) = 0.03159661490765174, 0.00015494946823958237)$

Test set accuracy performance: 50.86%

- Second best settinng:

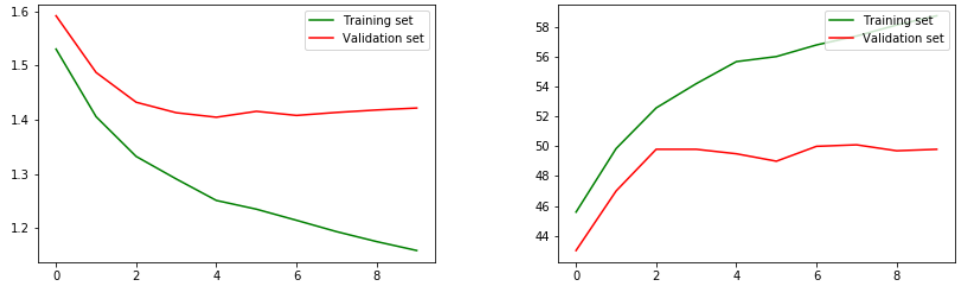

Figure 14: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set for $(\eta, \lambda) = 0.013917284924906376, 1.0632777281477375 * 10^{-6})$

Test set accuracy performance: 50.69%

- Third best settinng:

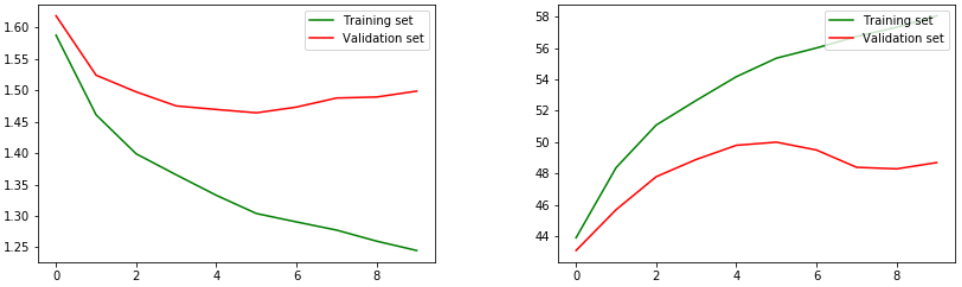

Figure 15: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set for $(\eta, \lambda) = 0.006537886088207429, 0.0006432091840869645)$

Test set accuracy performance: 50.86%

2 Exercise 2: Train network using different activation to ReLU

Use one of the other activation functions described in the lecture notes, for several sensible parameter settings.

In this part, ReLu is replaced by the Leaky ReLU and eLU activation functions:

$$Leaky\_ReLU(x) = \begin{cases} a * x, & x < 0 \\ x, & x > 0 \end{cases}$$

and

$$eLU(x) = \begin{cases} a * (e^x - 1) & x < 0 \\ x, & x > 0 \end{cases}$$

where in both cases $a = 0.01$. Additionally, in both cases we make use fo the best results for (η, λ) from the compulsory part of this setting.

2.1 Leaky ReLU results

1. First setting:

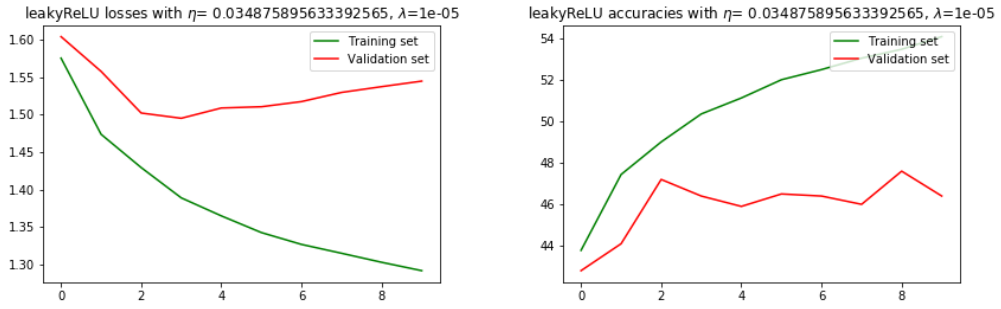

Figure 16: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set for $(\eta, \lambda) = 0.034875895633392565, 10^{-5})$

Test set accuracy performance: 48.33%

2. Second setting:

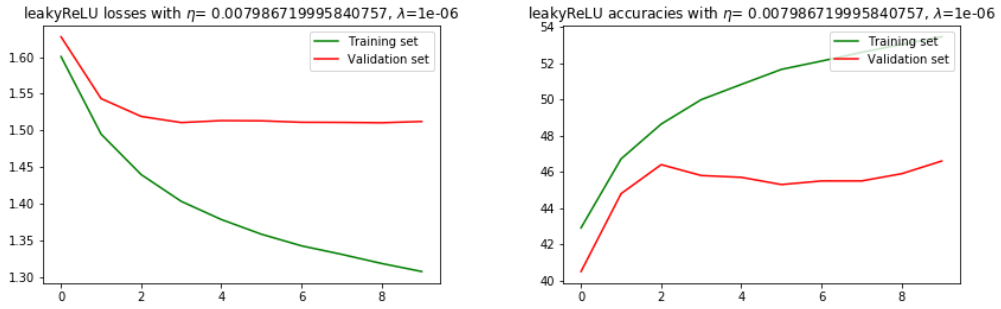

Figure 17: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set for $(\eta, \lambda) = 0.007986719995840757, 10^{-6})$

Test set accuracy performance: 47.43%

3. Third setting

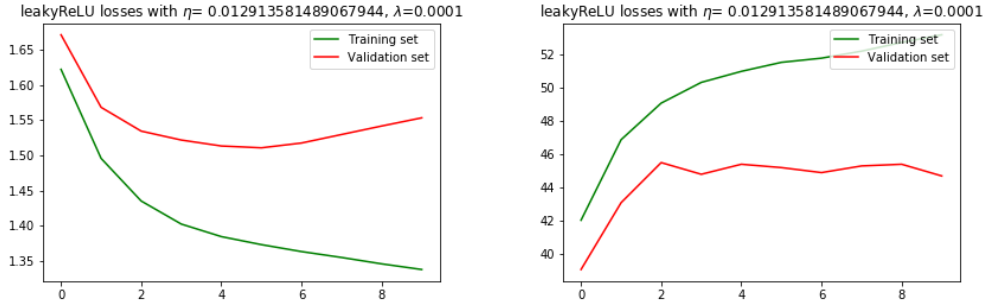

Figure 18: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set for $(\eta, \lambda) = 0.012913581489067944, 10^{-4})$

Test set accuracy performance: 42.32%

2.2 Exponential LU results

1. First setting:

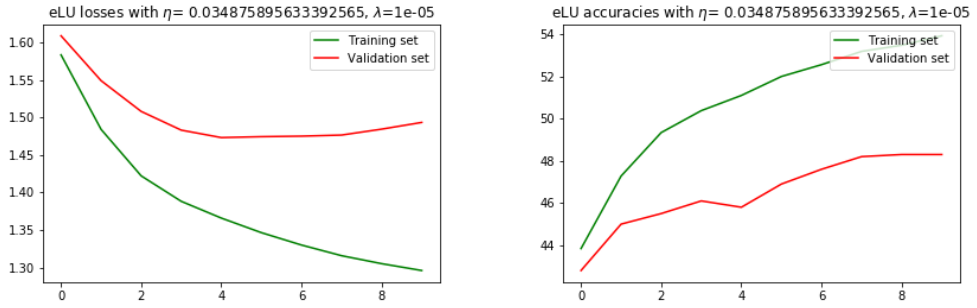

Figure 19: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set for $(\eta, \lambda) = 0.034875895633392565, 10^{-5})$

Test set accuracy performance: 48.39%

2. Second setting:

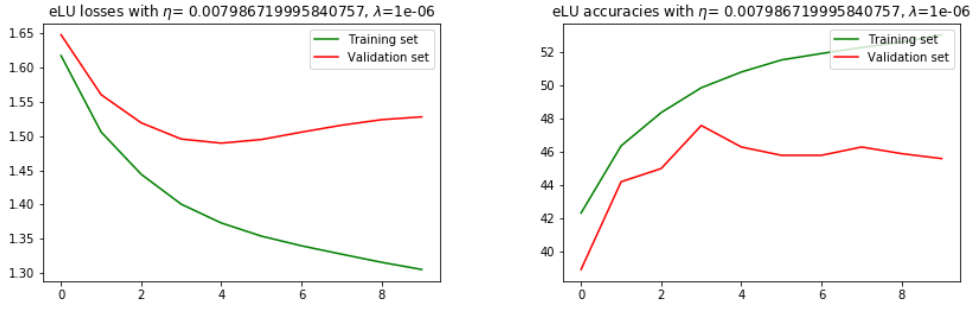

Figure 20: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set for $(\eta, \lambda) = 0.007986719995840757, 10^{-6})$

Test set accuracy performance: 43.35%

3. Third setting

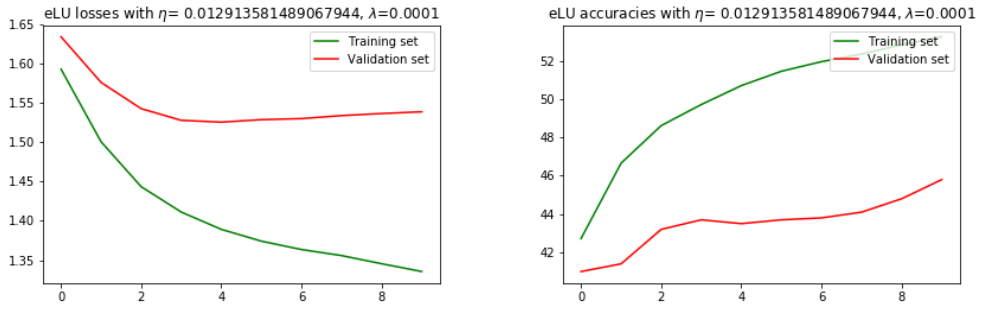

Figure 21: Cross-entropy loss (left) and accuracy performance (right) evolution on the training and validation set for $(\eta, \lambda) = 0.012913581489067944, 10^{-4})$

Test set accuracy performance: 47.79%