Statistik un Data Science für die Informatik

Humboldt-Universität zu Berlin, SS 22, Prof. Dr. Alan Akbik

Übungsblatt 4

Task 3

a)

The output of running the neural net program with the arguments
`python neural_net.py mnist.npz 60000 50 256`:

```
Starte die Evaluation des Modells
              precision    recall  f1-score   support

           0       0.952     0.980     0.966       980
           1       0.969     0.977     0.973      1135
           2       0.937     0.889     0.912      1032
           3       0.905     0.907     0.906      1010
           4       0.931     0.928     0.930       982
           5       0.889     0.880     0.885       892
           6       0.936     0.948     0.942       958
           7       0.930     0.912     0.921      1028
           8       0.873     0.880     0.876       974
           9       0.889     0.911     0.900      1009

    accuracy                          0.922     10000
   macro avg       0.921     0.921     0.921     10000
weighted avg       0.922     0.922     0.922     10000
```

b)

The most optimally recognized number is 1 with the F1-score of 0.973. In contrast, the most misclassified number is 8 (F1-score = 0.876).

c)

The output of the program given the following number of epochs:

1 epoch:

```
accuracy (aka micro f1) 0.823
macro f1 0.821
```

5 epochs:

```
accuracy (aka micro f1) 0.892
macro f1 0.891
```

10 epochs:

```
accuracy (aka micro f1) 0.907
macro f1 0.906
```

15 epochs:

```
accuracy (aka micro f1) 0.912
macro f1 0.911
```

20 epochs:

```
accuracy (aka micro f1) 0.915
macro f1 0.914
```

25 epochs:

```
accuracy (aka micro f1) 0.917
macro f1 0.916
```

30 epochs:

```
accuracy (aka micro f1) 0.918
macro f1 0.917
```

40 epochs:

```
accuracy (aka micro f1) 0.920
macro f1 0.919
```

50 epochs:

```
accuracy (aka micro f1) 0.922
macro f1 0.921
```

100 epochs:

```
accuracy (aka micro f1) 0.925
macro f1 0.924
```

Interpretation: with the increasing number of epochs (iterations) increase the values of accuracy (i.e. micro-F1) and that of macro-F1. This means that with a bigger number of epochs the model makes increasingly better adjustments by calculating a linear transformation matrix that fits the training data more closely. We can see that also by the mean loss function output in the 100th epoch (0.2595), which is the smallest across all iterations. This, however, might potentially lead to overfitting as the model is trained to fit the training dataset perfectly while it is most likely that it displays a rather poor fit when run on the test data.

d)

The output of the program given the following number of batches:

200 training samples:

```
accuracy (aka micro f1) 0.535
macro f1 0.523
```

1000 training samples:

```
accuracy (aka micro f1) 0.769
macro f1 0.769
```

5000 training samples:

```
accuracy (aka micro f1) 0.881
macro f1 0.879
```

10000 batches:

```
accuracy (aka micro f1) 0.898
macro f1 0.897
```

20000 batches:

```
accuracy (aka micro f1) 0.911
macro f1 0.910
```

Interpretation: the tendency is that the bigger the training dataset to train the model on, the better the model performs, i.e. the better the fit.