Machine Learning
Homework 5

Group 33

Ophelia Prillard
Espen Marstein Sandtveit
Daniel Tavakoli

May 2019

TÉCNICO LISBOA

I. Description of the problem

The presented homework concerned the implementation and development of a neural network that had the ability to recognize objects in an image. The task of the network was to take in a sequence of images and return if there are any fruits present. If that is the case, the network should return which fruits.

For this assignment, a dataset containing various fruits was downloaded from Kaggle [8]. From this dataset the following fruits were to be used in the homework:

- Apples
  - Crimson Snow
  - Golden
  - Golden-Red
  - Granny Smith
  - Pink Lady
  - Red
  - Red Delicious
- Lemons
  - Normal
  - Meyer
- Oranges
- Pears
  - Abate
  - Kaiser
  - Monster
  - Red
  - Williams

The idea was that the network implemented would not update the weights of the initial layers because they were assumed to be sufficiently accurate for the problem at hand. On the other hand, the layers close to the output would be retrained in order to ensure that the model classified well for the given dataset.

Subsequently, the initialization of the weights and parameters for regularization would be altered in order to analyze the impact of these quantities for the classification of the network.

II. Choice of Model

Multiple neural networks were presented in the assignment text as possible options to use when implementing the solution. The chosen model for the assignment was **Inception v3**. The reason for choosing this model was that it performed quite well and yields robust image classification [9], which fitted the assigned work well.

A. Architecture of Inception v3

The Inception v3 was an image recognition model with a 42-layer deep neural network. It consisted of two parts: a feature extraction convolutional neural network part and a classification part with fully-connected layers with softmax activations.

III. Results

The implemented code is given at `https://github.com/danieta/ml_hw5`. The images within a given fruit category all illustrated the same physical fruit, only rotated and/or translated. A problem could therefore arise if an image in the test data included a fruit with small deviations from the trained fruit, e.g. a bruised pear or an apple with stem. Even with small changes, the neural network could have problems identifying it due to the structure of the training data.

Altering the initial weights was deemed counter-intuitive given the model chosen in Section II. The weigths were already trained before downloading the model and hence initializing the weights did not make sense.

Testing the model using the test data, the performance of the trained model could be discussed.

A. Default initialization without regularization

The default configurations had 4000 epochs and a learning rate of 0.01. Running the script gave an accuracy of 99.4%. The accuracy was given by Figure 1 and the loss by Figure 2. The blue and red plot illustrates the performance for the test set and validation set, respectively.

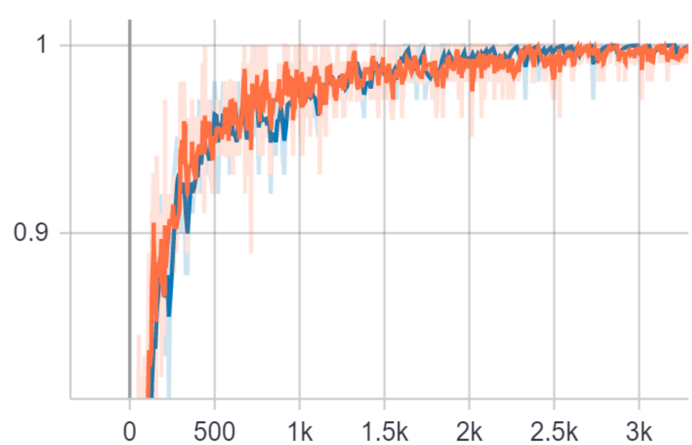

Fig. 1. Accuracy plot with default configurations.

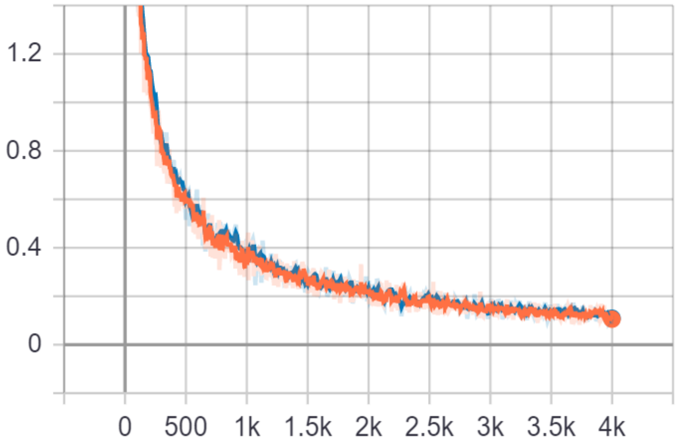

Fig. 2. Cross-entropy with default configurations.

B. Epochs reduced to 1000

When decreasing the number of epochs to 25% of the previous value the performance of the model was analyzed. The most significant change was the run time: the code would previously use approximately 5-6 minutes, whereas now the model finished training after less than a minute. The accuracy was now valued at 96.4%, and is illustrated in Figure 3. The loss is given by Figure 4. Though the accuracy was not as good as for the 4000 epochs the model still performed quite well, and the trade-off between run time and accuracy seemed to be suitable.

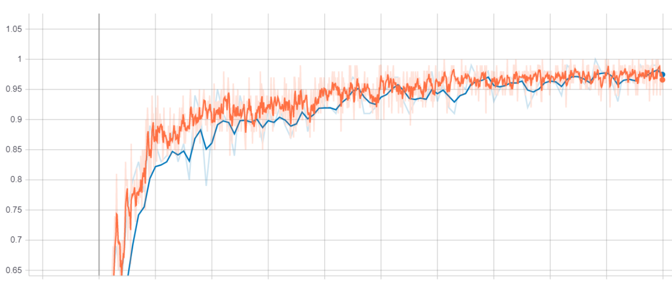

Fig. 3. The accuracy with 1000 epochs.

C. Learning rate 0.1 and 1000 epochs

In this case the accuracy became 99.9%, as shown in Figure 5. The loss is shown in Figure 6. By multiplying the learning rate by 10 the model seemed to perform significantly better, as shown by the accuracy and loss.

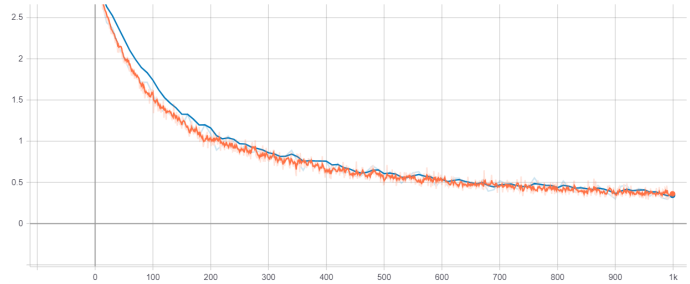

Fig. 4. The cross entropy with 1000 epochs.

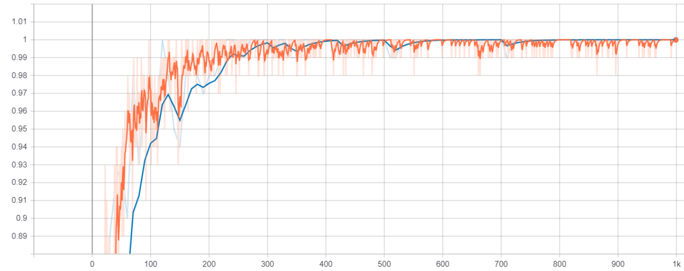

Fig. 5. Plot of the accuracy with a learning rate 0.1.

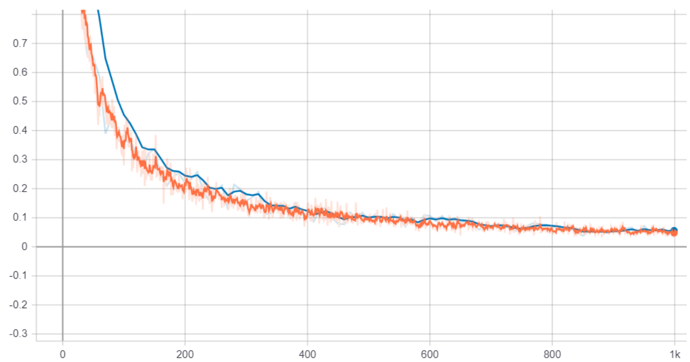

Fig. 6. Plot of cross-entropy with learning rate 0.1.

IV. Conclusion

By changing the regularization in terms of tweaking e.g. epochs and learning rate the performance changed, as expected. These type of parameter estimation exercises are a central part in the quest of finding the optimal solution for the problem that the model is supposed to solve. For future analysis, the activations of the layers could also have been changed, in addition to trying to using the technique of dropout regularization to see how this affects the accuracy and loss of the model.

References

[1] Andreas Wichert, *Lecture 5: Perceptron and Logistic Regression*. Department of Computer Science and Engineering, IST, May 2019.

[2] Andreas Wichert, *Lecture 9: Multilayer Perceptrons*. Department of Computer Science and Engineering, IST, May 2019.

[3] Chris Piech, *Logistic Regression*. https://web.stanford.edu/class/archive/cs/cs109/cs109.1166/pdfs/40%20LogisticRegression.pdf Stanford University, May 2019.

[4] Jamie Browniee, *A Gentle Introduction to Dropout for Regularizing Deep Neural Networks* https://machinelearningmastery.com/dropout-for-regularizing-deep-neural-networks/ May 2019.

[5] Imad Dabbura, *Coding Neural Network - Regularization* https://towardsdatascience.com/coding-neural-network-regularization-43d26655982d May 2019.

[6] Shubham Jain, *Overview of Regularization Techniques in Deep Learning* https://www.analyticsvidhya.com/blog/2018/04/fundamentals-deep-learning-regularization-techniques/ Analytics Vidhya, May 2019.

[7] GitHub Keras, http://faroit.com/keras-docs/1.2.0/getting-started/sequential-model-guide/ Keras, May 2019.

[8] Kaggle Dataset, https://www.kaggle.com/moltean/fruits. Kaggle Inc, May 2019

[9] Sik-Ho Tsang, *Review: Inception-v3* https://medium.com/@sh.tsang/review-inception-v3-1st-runner-up-image-classification-in-ilsvrc-2015-17915421f77c Medium Inc, May 2019.

[10] TensorFlow, *How to Retrain an Image Classifier for New Categories* https://www.tensorflow.org/hub/tutorials/image_retraining TensorFlow, May 2019.