



Aprendizagem Automática Avançada (2º Ciclo de Informática)

Deep Learning Exercises

Take the tutorial on image classification (<https://www.tensorflow.org/tutorials/images/classification>) either in Google Colab or on your own PC.

Consider the reference model presented in the tutorial for solving the following problems.

Problem 1 *

Create 4 different models, one at a time, by modifying the reference model in the following manner:

- i) Using the mean squared error as a loss function.
- ii) Using a dropout probability of 40%.
- iii) Using a batch size of 8.
- iv) Using batch normalization (see this link for more information).
- v) Combine modifications ii, iii and iv.

Compare the models (including the reference model) in terms of accuracy and loss values, and verify the learning curves for train and validation sets. Also, record the number of epochs until overfit and the average time per epoch.

Problem 2 *

Implement an autoencoder to learn to encode images from the dataset given in the image classification tutorial, based on the reference model used in the previous exercise.

You may find it helpful to look at this introduction to Autoencoders in Keras: <https://www.tensorflow.org/tutorials/generative/autoencoder>.

- a) Train an autoencoder on 3500 images from the image classification dataset, leaving 170 aside for validation.
- b) Sample 3 examples from each dataset split, the train and validation set, and use the autoencoder to encode and decode the images, comparing them with the original.
- c) Sample 3 examples from each dataset split, the train and validation set, encoding the images using the Encoder, obtaining the intermediate representations. Using the **GaussianNoise** layer, introduce 5%, 10% and 20% noise to the representations. Finally, decode the representations using the Decoder and analyze the obtained images.

- d) Sample 3 examples from each dataset split, the train and validation set, and introduce 5%, 10% and 20% noise (**GaussianNoise** Layer) to the image pixels (input level). Run the autoencoder with these inputs and analyze the obtained images.