



DEEP LEARNING PROJECT № 1

DM873 / DS809, Fall 2022

12/10/2022

To do these exercises, you will use Python 3 and the following packages:

- [TensorFlow](#). The used backend used for Keras.
- [Keras](#). Good front-end package for easier integration of neural networks.
- [Matplotlib](#). This package allows you to graph your data and data transformations.

These can all be installed using pip, the python package manager. You are not strictly forced to use these packages, but it is highly recommended. Feel free to use other packages you think are necessary.

Installations

Run the following commands on terminal if not installed;

Listing 1: Installations of required Python packages

```
1 pip install tensorflow
2 pip install matplotlib
```

Project Details

When to submit

You need to submit before Friday 11-11-2022 at 12.00.

What to submit

You must hand-in: A report, your code, your models, and all visualizations saved as images. Your report must reflect who wrote and worked on each part.

How to submit

You should create a directory at [this](#) drive folder, and submit your project here. Name the directory according to the members in the group.

Notes

You must do this assignment in groups, and only one submission is needed per group. The data is both available on ItsLearning and same google drive that you will upload your submissions, named catdog_data.zip

Training the networks can be a slow process, keep this in mind. You're NOT allowed to use pre-trained networks, or data not found in the catdog_data.zip file! You are allowed to use any networks made during the exercises. Also, remember to use `model.save()` and NOT `model.save_weights()`.

Please read through the entire project description, as there will be two Q&A sessions on Wednesday 26th of Oct. and Wednesday 2nd of Nov. For questions or comments send an email to ugo@sdu.dk. Bear in mind, this should not be used for assistance as in the exercises, as you should try to complete this project on your own.

Problem: Cat & Dog Classification



We want to distinguish between images of cats and dogs.

Task 1 - The Dataset

The dataset consists of 3600 images of cats and dogs, 1800 images of each class. Also keep in mind, that the image format is jpeg, with 3 color channels.

You need to organize the data into directories as shown on Figure 1. You need to determine the training, validation and test splits yourselves, but need to justify your split choice.

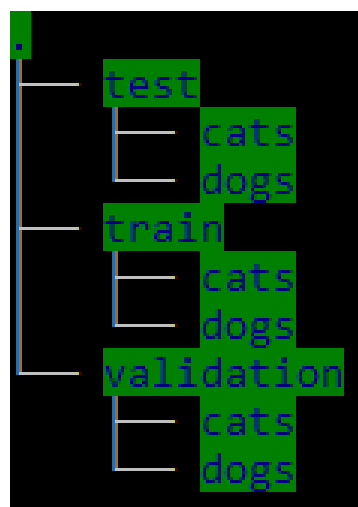


Figure 1: The structure of the dataset.

Task 2 - Create the datagenerators

Due to the high amount of data, you need to write a datagenerator to load the images. This requires you to use the [ImageDataGenerator](#) (as shown in Keras Intro - 2) in which you must apply at least one on-the-fly data augmentation. Which type of data augmentation is up to you, but you need to justify your choice in your report. On top of the ImageDataGenerator you need to apply the [.flow_from_directory](#) function to make it work with the directory tree in Task 1

Task 3 - Constructing the network

After successfully crafting the data generators, you need to craft a neural network, and use the data generators to feed the network. The architecture, complexity and regularizers are all up to you, but you need to justify your choices in the report. You are more than welcome to replicate already known architectures or architectures we made during the course, but you are NOT allowed to use any pretrained networks. You are also not allowed to use any training data that isn't included on ItsLearning.

Carefully consider which hyperparameters to test and strategically try to find the optimal architecture for the task. In the comments, please describe your method for the optimization and your choice of hyperparameters.

Task 4 - Visualizing your results

Finally, you must visualize some aspects of your model. It can be a graph of the training/validation performance, visualization of the filters or feature maps, misclassified samples, or anything you can think of. This has to be saved as an image file, and uploaded along with your report and code.