Practical work 09 – Conv Neural Networks with Keras (part 2)

Banfi Gregory (banfigre@students.zhaw.ch)
Benjamin Kühnis (bkuehnis@hsr.ch)

Exercice 1 **Data Augmentation**

(a) Without data augmentation
Train accuracy: 0.769, Test accuracy: 0.7129

Data augmentation values:
rotation_range=8
width_shift_range=0.08
shear_range=0.3
height_shift_range=0.08
zoom_range=0.08

390 train samples, 78 test samples
Train accuracy: 0.713, Test accuracy: 0.695

781 train samples, 156 test samples
Train accuracy: 0.739, Test accuracy: 0.702

1562 train samples, 312 test samples
Train accuracy: 0.7884, Test accuracy: 0.7441

Data augmentation values:
rotation_range=15
width_shift_range=0.1
height_shift_range=0.1
horizontal_flip=True

1562 train samples, 312 test samples
Train accuracy: 0.7811, Test accuracy: 0.7583

3125 train samples, 625 test samples
Train accuracy: 0.77624, Test accuracy: 0.7504

(b) Comparing a model using data augmentation and without we notice that when we add data augmentation the difference between train and test loss decreases. Data augmentation is used to prevent over-fitting, the model gets transformations of the image and learns to treat them similarly. As example take an image of a cat looking at the right, if we train the model with that image we will have a good accuracy in recognizing cats that are looking at right. When applying data augmentation the image will be, for example flipped and the cat will now look at left and the model will also be trained to recognize left-looking cats. The model will have a more general interpretation of a concept/class.

Exercice 2 **Visualisation of activations**

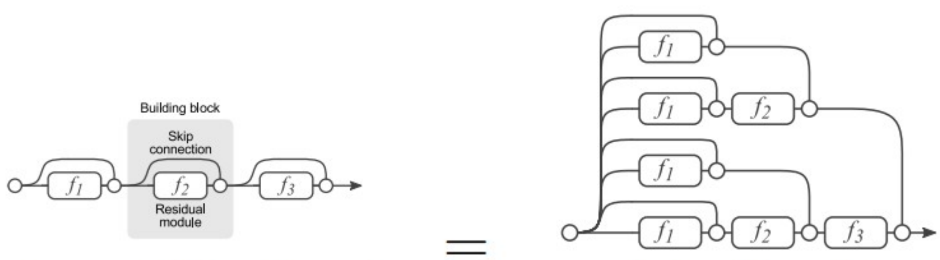
The 7 Layers were visualized. The first layers the truck is still visible, while the last layer has higher level features. We can say, that the deeper the layer, the more abstract or higher the level of features it gets.

Exercice 3 **Deep architectures**

The idea is to use very deep convolutional networks like the *inception* architecture with *residual connections*. The use of residual connections does not improve the accuracy but rather the training speed of deep networks.

The idea of inception architecture is to increase the width of the network while still go deep. The problem we want to solve is the uncertainty about the size and where a pattern is present in an image. Filters match patterns on images (pixel values), but sometimes the pattern is big so we need a big filter and sometimes are small so it would be better to have a small filter. The same work with the position of the pattern, some filters are good to match a pattern on the center of the image and some in the left bottom corner. We want to have a variety of filters applied to the same pixel values so that we get the best result from the best matching filter, or an average of the results. This is achieved with an inception architecture in which the output of a layer goes through different filters size and positions before going to the next layer. That's why it is sometime described as a network in the network.

Residual connections use the fact that while going through the layers of the network an higher abstraction is created. In a typical architecture we only use information provided by the previous layers to feed the following layers. With residual networks we want to pass to the next level as much information as we gained until there, not only the last layer's output but also the second last's output and so on. This architecture aims to reduce to the minimum the loss information between not direct-connected layers.



(a) Conventional 3-block residual network

(b) Unraveled view of (a)

Figure 1: Residual Networks are conventionally shown as (a), which is a natural representation of Equation (1). When we expand this formulation to Equation (6), we obtain an *unraveled view* of a 3-block residual network (b). Circular nodes represent additions. From this view, it is apparent that residual networks have $O(2^n)$ implicit paths connecting input and output and that adding a block doubles the number of paths.

Source: Residual Networks Behave Like Ensembles of Relatively Shallow Networks