Artificial Neural Networks and Deep Learning

Prof. Matteo Matteucci and Prof. Giacomo Boracchi
Date: 15/01/2020

Name:

Surname:

Student id (matricola):

Notes: Write ***clearly*** your answers in the provided sheets we need to read in order to grade! Write your name, surname and id, **on any** sheet of paper you turn in. You **can** use pencil, although we do not care if you just erase with a strikethrough. You **cannot** use books, notes, slides, cheatsheets, and electronic devices of any sort. For each answer you provide, please state clearly the number of the exercise and the number of the question, e.g., *Ex.1.2*, which you are currently answering.

Exercise 1 (4 points):

With reference to a Feed Forward Neural Network with 3 input, 1 hidden layer having 5 hidden neurons, and an output layer with 2 neurons.

1. Draw the network, and provide its output characteristics, i.e., the mathematical formula, for the output of the previous network as a function of its input and weights
2. Consider the previous network to be used for classification. Define its activation functions and error/loss function providing motivations for your choices
3. What is backpropagation and how does it work?

Exercise 2 (5 points):

When training neural networks, we need to use some form of regularization. Answer in detail the two following questions about regularization

1. What is regularization and why do we need regularization when training an artificial neural network
2. Describe TWO (2) regularization techniques used in training artificial neural networks providing their rationale/motivation, i.e., what do they do and why they do that.

Exercise 3 (12 points):

Consider the following Python snippet

```
import keras
from keras.utils  import np_utils
from keras.models import Sequential, Model
from keras.layers import Dense, Activation, Flatten, GlobalAveragePooling2D, Dropout
from keras.layers import Conv2D, MaxPooling2D, AveragePooling2D, Dropout
from keras.applications import MobileNet, VGG16

patchsize          = 64
pool_size          = (2,2)
kernel_size        = (3,3)
nb_filters         = 8

model = Sequential()

model.add(Conv2D(nb_filters, (3,3) , input_shape=(patchsize, patchsize, 3), padding
= "same"))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size = pool_size))
model.add(Conv2D(nb_filters*2, (3,3), padding = "same"))
model.add(Activation('relu'))
model.add(Conv2D(nb_filters*2, (3,3), padding = "same"))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size = pool_size))
model.add(Conv2D(nb_filters*3, (5,5), padding = "same"))
model.add(Activation('relu'))
model.add(Conv2D(nb_filters*3, (5,5), padding = "same"))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size = pool_size))
model.add(GlobalAveragePooling2D())
model.add(Dense(64))
model.add(Activation('relu'))
model.add(Dropout(0.5))
model.add(Dense(16))
model.add(Activation('relu'))
model.add(Dropout(0.5))
model.add(Dense(5))
model.add(Activation('softmax'))
```

Answer the following questions providing a short explanation for each

1. Is this a linear model?
2. Is this a classification or a regression network?
3. What is the input size? What is the output size?
4. Is this a fully convolutional neural network?
5. Once trained, can we feed to the network a larger image? Can we feed an image with more channels? Why?
6. Is there any form of regularization?

7. The following is the output of the model.summary() command. A few numbers have been replaced by dots. Please, fill all the numbers in, together with the computation you are performing. Note: the first dimension is the batch size, and as such this is set to None.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None,...,...,...)        .......
_________________________________________________________________
activation_1 (Activation)    (None,...,...,...)        .......
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None,...,...,...)        .......
_________________________________________________________________
conv2d_2 (Conv2D)            (None,...,...,...)        .......
_________________________________________________________________
activation_2 (Activation)    (None,...,...,...)        .......
_________________________________________________________________
conv2d_3 (Conv2D)            (None,...,...,...)        .......
_________________________________________________________________
activation_3 (Activation)    (None,...,...,...)        .......
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None,...,...,...)        .......
_________________________________________________________________
conv2d_4 (Conv2D)            (None,...,...,...)        .......
_________________________________________________________________
activation_4 (Activation)    (None,...,...,...)        .......
_________________________________________________________________
conv2d_5 (Conv2D)            (None,...,...,...)        .......
_________________________________________________________________
activation_5 (Activation)    (None,...,...,...)        .......
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None,...,...,...)        .......
_________________________________________________________________
global_average_pooling2d_1 ( (None,...)                .......
_________________________________________________________________
dense_1 (Dense)              (None,...)                .......
_________________________________________________________________
activation_6 (Activation)    (None,...)                .......
_________________________________________________________________
dropout_1 (Dropout)          (None,...)                .......
_________________________________________________________________
dense_2 (Dense)              (None,...)                .......
_________________________________________________________________
activation_7 (Activation)    (None,...)                .......
_________________________________________________________________
dropout_2 (Dropout)          (None,...)                .......
_________________________________________________________________
dense_3 (Dense)              (None,...)                .......
_________________________________________________________________
activation_8 (Activation)    (None,...)                .......
=================================================================

Total params: .....................
_________________________________________________________________
```

Exercise 4 (6 points):

Consider the problem of sequence modeling and the classical Seq2Seq model used for machine translation. Answer the following questions:

1. What is “sequence to sequence modeling”?
2. Describe the Long Short-Term Memory cell and its use in sequence modeling
3. Describe the classical seq2seq model based on LSTM cells for language translation, its training procedure and its run-time inference
4. What is an attention mechanism? How can it be used in a Seq2Seq model?