COMPUTING SUBJECT: Machine Learning

TYPE: WORK ASSIGNMENT

IDENTIFICATION: MLPClassificationFashion

COPYRIGHT: *Michael Claudius*

DEGREE OF DIFFICULTY: Medium

TIME CONSUMPTION: 1-3 hours

EXTENT: < 150 lines

OBJECTIVE: Basic understanding of MultiLayerPerceptron
                                MNIST data set

COMMANDS:

IDENTIFICATION: MLP Classification Fashion/MICL

The Mission
To understand the idea behind complex neural networks like the Multiple Layer Perceptrons.

Precondition
You must have read the second part of Training Models in chapter 4.

The problem
Given a dataset of 60.000 images (28x28 pixels) of fashion clothes (X) and the label (Y) (i.e. the correct value here the name of clothe), you are to train a MLP classification multinomial with 10 classes. You are to use the Fashion MINST dataset built into Sklearn.

Figure 10-11. Samples from Fashion MNIST

Useful links

https://www.tensorflow.org
https://keras.io
https://keras.io/models/sequential

Remember read the whole fu….. exercise

Assignment 1: Download data set and program
As the data set is built in we don't need to download it.
For the program it is **easiest** to download a downsized version with only the necessary code for
ANN's from your teacher's home page. The file is *"MLP Fashion Ch.10.ipynb"* encapsulated in the
"ANN Programs.zip" file.

Alternatively, You could choose to from the Github to copy the Chapter 10 Jupyter program, "*10-neural-nets-with-Keras.ipynb*", into the folder for your solutions (Machine Learning/Solutions)
folder. Rename it "*MyMLPFashion.ipynb*", before you cut away a lot of not–necessary code.

Assignment 2: Application program, adjusting the program
Start Jupyter and open the file. If you are using the downloaded *"MLP Fashion Ch.10.ipynb"* You
don't need to adjust the program, otherwise you will now have to delete and out-comment many
lines/cells like *"Perceptron","Regression"* and some other examples.

Now we can start to execute the cells.

Assignment 3: MLP Image classifier
Run all the cells
Notice and look especially at the following cells:

> [22]: Defining the model of layers, neurons, activation functions
> [34]: History of accuracy for each epoch's training and validation set
> [38]: Learning and loss curves
> [39]: Accuracy for test set

You can actually follow the history on the fly, when it is running !

NOTE: Assignment 4 and 5 can be solved parallel !

Assignment 4: Hyperparameters
In order to analyse and find the best (accuracy and speed) MLP, one need to change the values of the
hyperparameters:
- number of layers,
- activation function,
- number of neurons,
- number of layers

The model in the code is slow and complex with has 4 layers and many neurons.
When performing this analysis it is better to simplify the model and then finally make it more
complex.

a. Therefore, start to set epochs to 10, so it goes faster but with lower accuracy.

b. Test different Activation functions.

Copy cell [22] and make a model with only one inner layer with 50 neurons
Then run the program with: relu, tanh and sigmoid functions
Each time write down the accuracy and loss for training set, validation and test set in a table like:

Activation	accuracy	loss	Val_accuracy	Val_loss	Test_accuracy	Test_loss
relu						
sigmoid						
tanh						

From now on use the best one (probably relu)

IF you are smart, you let each group member do either the c or the d question to save time.

c. Test number of layers
   Fix the number of neurons in inner layers to 200
   Run the program with
   - One inner layer with 200 neurons
   - Two inner layers: first 150 neurons second 50 neurons
   - Three inner layers: 100, 50 , 50
   - Four inner layers: 90, 50, 40, 20

   Notice the learning curves
   Write down the accuracy of each model

d. Test number of neurons
   Fix number of layers to 3
   Run the program with different number of neurons
   - 100, 50, 50
   - 100, 100, 100
   - 150, 100, 50
   - 300, 200, 50

e. Choose the model
   Now set epochs to the initial value 30
   Run your best model
   How high is the accuracy

Assignment 5: Theoretical issues
Now discuss some of the topics and write down the answers to the following questions:

a. What is the idea behind MLP ?

b. What is backpropagation ?

c. It important to scale, why ?

d. What is an activation function the purpose of it and and which types do you know?

e. Go to the homepage for Keras.io, see the definition of *Sequential*. There are really many parameters.
   Compare the parameters with the one used in the program code and explain 2 of them.
   Tip: https://keras.io/models/sequential/

f. Let's analyse the number of weights for MINST Digit dataset in a MLP with the following layers:
   - Input layer: 28x28 pixels 784 features/values on a grey scale
   - First inner layer: 16 neurons
   - Second inner layer: 16 neurons
   - Output layer: 10 classes (0,1,2, ….9)

   Remember in inner layer each neuron has with a bias weight.
   Can you prove (calculate) that the total number weights including bias is 13.002 or is it 13.001 ?!

Congratulation: You are now an expert-level 1 in MLP classification ☺