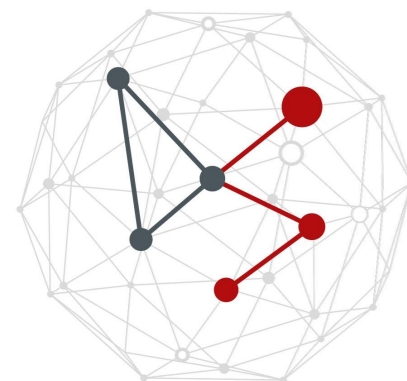


HUMAN DATA ANALYTICS: LAB_1

Lab. classes

Francesca Meneghello

meneghello@dei.unipd.it



Lab_1

- FFNN based classifier
 - Step-by-step implementation
 - The challenge:
 - classification of cat images
 - You will learn to:
 - implement a feed forward neural network classifier, defining the model, the forward and backward propagation steps and the update rule
 - train the classifier and test its performance

References from theory

- **FEED FORWARD NEURAL NETWORKS (FFNN)**
- Specifically:
 - slides 12-16: classification problem
 - slide 25: the artificial neuron
 - slide 36: sigmoid function
 - slides 42-46: ReLU function
 - slides 52-56: FFNN structure
 - slides 62-63: two classes classification error
 - slides 72-85: gradient computation (appendix slides 103-105)
 - slide 87: update rule – batch-mode learning
 - slides 93-94: learning rate
- **NOTE:** the notation in the lab is slightly different from the one in the theory slides (different letters but obviously same structure). The notation is explained in the text parts of the notebook.

Objective

- You will design two networks
 - 2-layer FFNN, (7, 1) neurons → 72% accuracy on the test set
 - L-layer FFNN → 80% accuracy on the test set with $L=4$, (20, 7, 5, 1) neurons

Structure of the lab session

- Two notebooks
- **Lab_1a_FFNN_step_by_step**
 - implement all the needed functions to:
 - **initialize** the parameters – 3.1, 3.2
 - **forward** the input through the network – 4.1, 4.2, 4.3
 - compute the **cost** function – 5
 - **backpropagate** the error through the network – 6.1, 6.2, 6.3
 - **update** the parameters – 6.4
- **Lab_1b_Cat_classifier_FFNN**
 - combine the functions implemented in Lab_1_a to build the FFNN classifiers and test their performance using the **cat vs non-cat images** dataset – 4 (`two_layer_model`), 5 (`L_layer_model`)

HUMAN DATA ANALYTICS: LAB_1

Lab. classes

Francesca Meneghello

meneghello@dei.unipd.it

