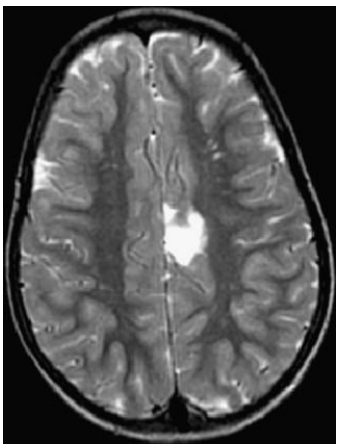
Practical session 3: Virtual biopsy of brain tumors combining magnetic resonance spectroscopy with artificial neural networks

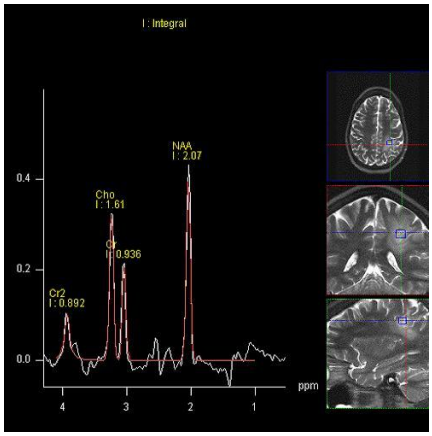
Biomedical Data Science

Pablo Ferri Borredà

Biomedical Data Science Lab (BDSLab)

ITACA, UPV





Contents

1. Objective

To develop a clinical decision support system to classify brain tumors based on metabolite concentrations estimated by means of magnetic resonance spectroscopy (MRS).

2. Material

- Practical session 3: Introduction (Virtual biopsy of brain tumors combining MRI spectroscopy with artificial neural networks.pdf).
- Data file (Metabolite_concentrations_diagnosis.csv).

3. Evaluation

It will be evaluated the practical session report along with code files. The data exploration report (html file) should also be submitted.

The report must include:

- Page 1: cover page, title, authors and professors.
- Page 2: contents.
- Page 3 and following pages: answers to the questions and exercises raised in each block.
- Last page: references.

4. Tasks

4.1. Block I. Data preparation

4.1.1. Objective

Understand your working data and prepare it to posterior model training and selection.

4.1.2. Notes

- Use pandas profiling package for data exploration and save the html file generated.
- Numpy, pandas and sklearn libraries provide functions to solve each of the tasks posed.

4.1.3. Questions & exercises

1. Load your data and extract basic data descriptors using the pandas profiling package. Save the html report generated, since you will have to attach it in your Poliformat task. Include a brief comment about report results.
2. Create an auxiliary one hot encoded variable for each of the three types of brain tumors you have to classify (meningioma, astrocytoma and glioblastoma).

3. Split your data using a holdout methodology considering odds of 70:30 while fixing a random seed to ensure reproducibility.
4. Perform robust scaling on each of the datasets generated in 3) over the numerical variables, considering a symmetric percentile range of 95%. Take into account which is your available information in a prediction environment to apply your transformations properly.

4.2. Block II. Model training and selection

4.2.1. Objective

To select the *optimal* hyperparameters and train an artificial neural network model based on that *optimal* configuration. These hyperparameters are intrinsic to model architecture and model training.

4.2.2. Notes

- Check the evaluation metrics functions implemented in sklearn.
- Check the MLPClassifier class from sklearn.

4.2.3. Questions & exercises

1. Choose a proper evaluation metric suitable for your problem, i.e., this metric must describe model performance without being biased to a specific diagnosis. You may have to review the data exploration report generated in 1.1) and check the metrics available in sklearn.metrics.
2. Run some experiments to study the influence of:
   2.1. Learning rate.
   2.2. Batch size.
   2.3. Architecture.
   2.4. Number iterations.
3. Select the best hyperparams configuration, according to the experimental results obtained in 2) and the evaluation metric chosen in 1). Estimate model performance and retrain it with the whole data.
   3.1. Attach figures regarding subsection 2), including a brief discussion of your experimental procedure, along with the results obtained. Choose a hyperparam configuration as the *best* hyperparam set and justify your answer.
   3.2. Estimate model performance with the validation set, reporting area under curve, precision, recall and f1-score for each diagnosis.
   3.3. Retrain your model with the whole training data, considering the *best* hyperparams set. Attach a figure including the training loss across iterations and include a brief comment about this graph.

4.3. Block III. Clinical decision support system deployment

4.3.1. Objective

Deploy the clinical decision support system developed in blocks I and II.

4.3.2. Notes

- Reuse code from prior exercises.
- You do not have to implement a user interface.

4.3.3. Questions & exercises

1. Create a new script (or a new function but in a separated code file) able to make predictions in a new dataset. This dataset would be provided in .csv format, with the same delimiters as your data files but just with metabolite concentrations data. Likewise, it has to be a black box for the physician. He/she will have to just click the run button and get a csv file where for each case, probabilities for each diagnosis as well as the recommended tumor diagnosis are shown.