*Bioinfo\_Project User Manual*

**1) creazione\_manifest.py**

This script is responsible for processing the json and manifest files related to mRNA and miRNA, in order to produce the final manifest files that are used to download the data from the server.

The following global variables are used: pay attention to the names of the files used in the script and present in the folder.

mrnaJson = '2\_mRNA.json'

mrnaManifest = '2\_mRNA\_manifest.txt'

mirnaJson = '1\_miRNA.json'

mirnaManifest = '1\_miRNA\_manifest.txt'

Output:

nuovo\_manifest\_m.txt

nuovo\_manifest\_mi.txt

new\_label\_mi.txt

new\_label\_m.txt

**2) Data download using the client gdc\_client.exe**

With the following commands (run as administrator) the files contained in **new\_manifest\_m.txt** and **new\_manifest\_mi.txt** are downloaded.

gdc-client.exe download -m <absolute\_path\_to\_project\_folder>\n1\_unione\_maninfest\nuovo\_manifest\_m.txt --dir .\mRNA\

gdc-client.exe download -m <absolute\_path\_to\_project\_folder>\n1\_unione\_maninfest\nuovo\_manifest\_mi.txt --dir .\miRNA\

N.B.: Check the confirmation message of "Successfully downloaded: 1160".

**3) Dataset creation**

* Move the files ”new\_label\_mi.txt” and ”new\_label\_m.txt” (which contain the labels) created in step(1) to the folder ./n2\_creazione\_dataset.
* Run scripts ”move\_files\_miRNA.py” and ”move\_files\_mRNA.py”.
* Run script ”creazione\_dataset.py”.

Output:

dataset\_miRNA.csv

dataset\_mRNA.csv

**4) Dataset preprocessing**

Run the script dataset\_Preprocessing.py twice, changing from time to time the parameter (the dataset filename) given to the script.

Input commands:

* dataset\_Preprocessing.py dataset\_miRNA.csv
* dataset\_Preprocessing.py dataset\_mRNA.csv

Output:

scaled\_dataset\_miRNA.csv

scaled\_dataset\_mRNA.csv

**5****) Feature Selection Methods**

**5a) copyDatasetfromPath.py**

Every time you need to move multiple dataset files into the *n4\_featureSelectionMet*hods folder in order to use them for feature selection methods, run *copyDatasetfromPath.py* to move every “.csv” file from the source path given as input to the function.

**5b) Running FS methods**

Run the script related to the desired FS method and answer the questions asked on the command line: the answers given by the user perform the method with different parameters.

a) Complete or reduced list of labels? (R = reduced/C = complete)

b) Which dataset do you want to process?

c) Do you want to save the dataset containing only the most relevant features? (Y/N)

d) How many features do you want to select?

* 1. Reduced = implements the FS on the dataset containing a smaller number of labels without considering all subcategories of tumors.

Complete = implements the FS on the dataset containing all types of labels.

* 1. Name of the dataset to use.
  2. Y = saves a dataset that contains only the most relevant features obtained from the FS method used.

N = executes the FS method without saving the dataset.

* 1. Number of features to select (1-1000).

Output:

<FS>\_<nFeatures>\_<labelType>\_dataset.csv

Result\_FS.txt

**5c) CCA**

CCA is used for

a) Dataset#1

b) Dataset#2

c) Number of features

d) Supervised version?

e) Save CCA dataset?

Output:

CCA\_result\_dataset.csv

**6) Convolutional Neural Network**

**7) Validation**