Statistical Learning & Data Mining

Final Assignment: Classification Challenge on Alzheimer's Disease using MRIs and Gene Expression data

Alexandru-Petru Vasile

alexandrupetru.vasile@studentmail.unicas.it

Introduction

This challenge consisted of 3 binary classification problems in the field of Alzheimer's Disease diagnosis: AD vs CN, AD vs MCI, MCI vs CN.

Dataset

For this project, we were presented with 3 datasets, one for each problem consisting of a split between training and test for each problem. The main issue regarding data was *class imbalance*. The solution is proposed in the following section.

An interesting process in this project was data exploration. Coming from a Medical Engineering background, the first thing I did was contact acquaintances from the medical field to discuss the features in the dataset to assess their relevance. Apart from this, I crosschecked the whole gene expression part of the data with a gene expression database provided by genecards.org.

After this documentation stage I gathered this information:

Relevant Brain Regions for AD classification:

- Hippocampus
- Frontal Lobe
- Temporal Lobe
- Parietal Lobe
- Amygdala
- Corpus Callosum

Relevant genes (present in the dataset):

- CYP4A (group)
- EN5G (group)
- SLC6A (group)
- SRP19 (group)

This step proved very important, as consistently better results were obtained if only the above mentioned data was used.

Preprocessing

As a first exploratory the correlation matrix has been computed:

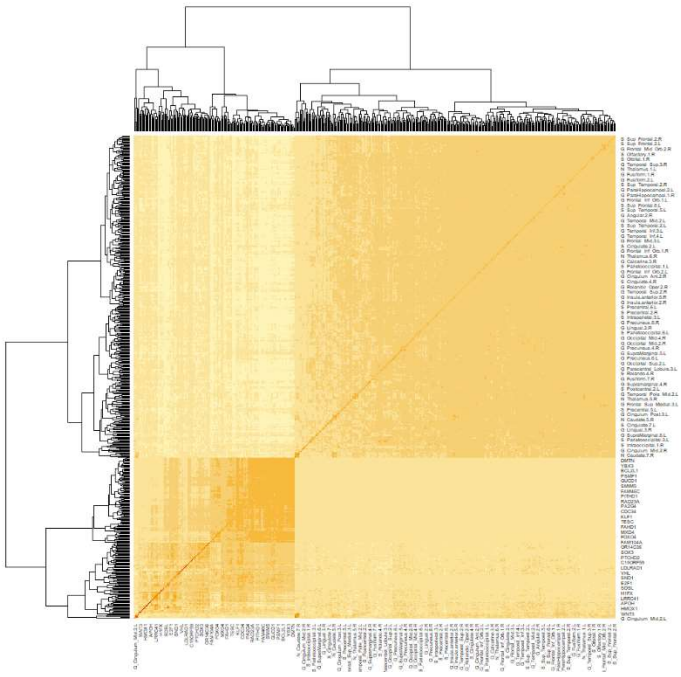

Figure 1 - Correlation Matrix - ADCN

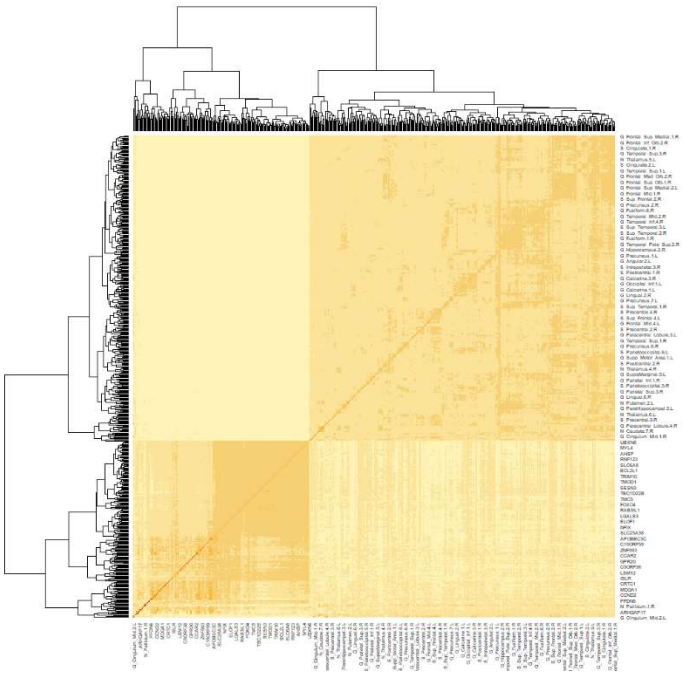

Figure 2 - Correlation Matrix ADMCI

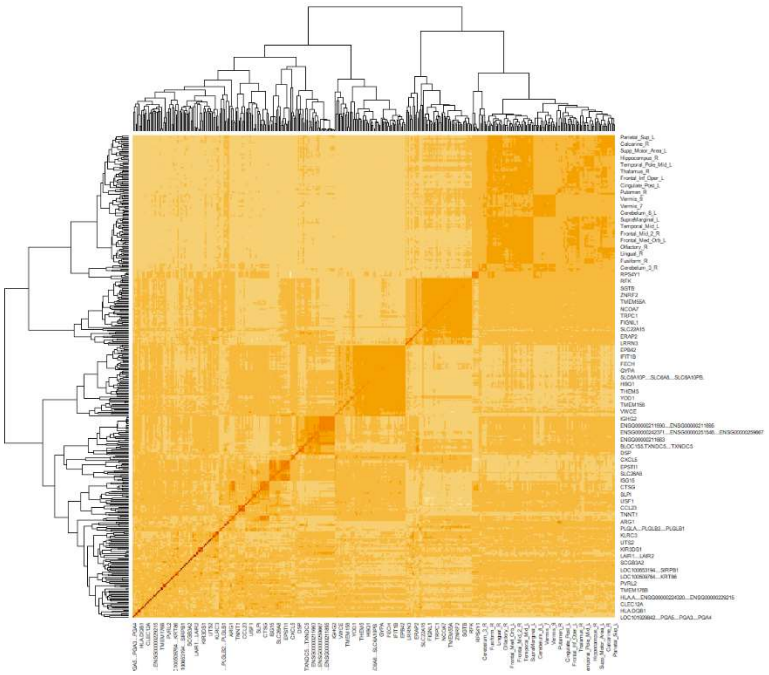

Figure 3 - Correlation Matrix MCICN

In these figures the two kinds of data (MRI and Gene Expression) can clearly be observed. It is also interesting to see that the gene expression data and MRI features become less differentiated as we compare more similar brain conditions.

Next, the issue of class imbalanced has been explored. I've computed the imbalance ratio for every class:

- ADCN ~ 0.17
- ADMCI ~ 0.01
- MCICN ~ 0.58

(Over)Sampling

Then, after computing the imbalance ratio, I used this information to adjust it accordingly using the Synthetic Minority Over-sampling Technique (SMOTE). For ADCN I've used the following ratios for the oversampling:

- ADCN ~ 0.43
- ADMCI ~ 0.39
- MCICN ~ 0.58 (as is)

The last decision was taken out of a wish to stay true to the nature of the dataset.

Model Tuning & Selection

After documentation I've settled on a number of candidate classifiers for these 3 binary problems:

- Knn
- Adaboost
- Node Harvest
- Extreme Gradient Boosting
- SVM (3 flavors)

For tuning each classifier I've implemented an automatic routine by using the *train.control* function in combination with the *tuneGrid* concept in which I iterate through relevant values of the tunable parameters of each classifier.

As indicated in the Challenge statement, (5-fold) cross validation has been used for the training of the models.

The selection criterion for each algorithm was *accuracy*. The metrics used were the ones presented in the Challenge statement: *accuracy, sensitivity, specificity, precision, f1, auc and mcc*.

Final Results

AD vs CN

	acc	auc	mcc
Knn	0.84	0.86	0.71
Adaboost (of)	0.94	1	1
nodeHarvest	0.86	0.93	0.90
Xgb (of)	0.94	1	1
Radial SVM	0.95	0.99	0.99
Poly. SVM	0.97	0.99	0.98
Linear SVM	0.96	1	1

AD vs MCI

	acc	auc	mcc
Knn	0.78	0.87	0.67
Adaboost (of)	0.96	1	1
nodeHarvest	0.86	0.90	0.85
Xgb (of)	0.94	1	1
Radial SVM	0.95	0.93	0.97
Poly. SVM	0.97	1	1
Linear SVM	0.90	0.95	0.89

MCI vs CN

	acc	auc	mcc
Knn	0.63	0.64	0.32
Adaboost (of)	1	1	1
nodeHarvest	0.63		
Xgb (of)	0.94	1	1
Radial SVM	0.64	0.53	0.18
Poly. SVM	0.63	0.57	0.24
Linear SVM	0.62	0.51	0.04

of = "overfit" – for these cases, there is a strong possibility the models overfit

Observation:

The string vector a that is generated in ***# Feature Selection based on medical insight*** *contains the name of the columns that I used for training. In the saved file, I have only saved the index of the column as instructed.*