logo.jpg

Università degli Studi di Torino - M.Sc. in Stochastic and Data Science - A.Y. 2021/2022

Final project of Statistical Machine Learning (MAT0043)

Gene selection for cancer type classification

The purpose of our project is to work on a high-dimensional genomics data and find a relatively small number of genes to predict the cancer type of a given tumorous cells. This is known as Genes Selection for Cancer Classification and it is in line with many up-to-date problems of applied medicine.

Our dataset contained 1.032 cancerous cells and their knock-out probabilities, i.e. the probabilities of stopping the growth of tumour by inhibiting one of the $\sim$ 17.000 genes. Each cell line was characterised by one of 10 possible cancer labels: Eye, Gastrointestinal, Gynecologic, Musculoskeletal, Neurological, Breast, Head-Neck, Hematologic, Genitourinary and, finally, Lung. We explored in details three Features Selection algorithms: Random Forests (RF) combined with Feature Importance, Lasso-SVM and Neural Networks (NN) combined with Permutation Importance. We studied two binary classifications (Blood-cancer vs. All, Lung-cancer vs. All) and a multiclass classification. Besides lung models, we achieved satisfying classification accuracies and we were able to select about 100-200 genes from the initial $\sim$ 17.000 ones. Models fitted on such genes obtained classification accuracies ranging from 67% to 98%.

Therefore, it seems that classifying cancer type from an extremely small set of genes depends on the cancer type itself. These methodologies worked incredibly well on Blood cancer, as we reached almost 100% accuracy with the reduced classifier, whereas failed miserably on Lung cancer.

1 Introduction

Cancer is a complex disease characterized by the uncontrolled growth of abnormal cells anywhere in the body. These abnormal cells are extremely invasive and we usually identify them with the name of their original tissue (for instance, breast cancer, lung cancer, brain cancer, etc.). In recent years, medicine has made a great step forward in finding new and efficient therapies for different diseases and cancer is one of them. In particular, thanks to numerous advances in technology, collecting huge amount of data is no longer an issue, so that one can exploit such information to define personalized treatments for patients. In this regards, the DepMap project[1] and, in particular, the Achilles project[2] aim to use genome-wide screens to collect data regarding mutations of cancerous cells, identify essential genes and report vulnerabilities across hundreds of human cancers.

Many researches are currently using DepMap datasets to identify relatively small sets of genes which are responsible of cancers growth. This procedure is often driven by medical knowledge, which we do not possess, together with some rough measures of importance. Being Maths student, we instead base our research on statistical models and on the hypothesis that, if a given classifier is able to distinguish different types of cancer, then the most relevant genes are the most important features for that given classifier (the meaning of "important features" will be clarified later). Of course, selecting few truly significant genes has outstanding implications in the medical field: building faster diagnosis tools and synthesising less toxic drugs are only two examples.

[1] DepMap Portal: https://depmap.org/portal/

[2] Achilles Project: https://depmap.org/portal/achilles/

2 Dataset

We used two public datasets from the DepMap Portal website[3] :

D1 *CRISPR_gene_dependency.csv*, which contains 1.032 cancer cell lines characterised by 17.393 gene scoring results

D2 *sample_info.csv*, which contains cell lines information, such as primary disease and sample collection site

Data were collected from real patients and successively processed, so that element (i, j) of this (1.032×17.393)-data frame is the probability that knocking out gene j has a real depletion effect on the i-th cell. Before proceeding with our analysis, we removed missing values, which affected only 10 rows coming from different tumours, and we looked for weird observation. In particular, we found 2 "Non-Cancerous" and 6 "Engineered" cells. The first can be reasonably discarded, whereas the latter requires a little care. Engineered cells are synthetically modified samples in lab and, here, they are manly associated to the Eye sample collection site. We decided to keep them and associate them to the cancer corresponding to their site.
Side note: in general, looking also at the sample collection site do not ease tumour classification. Indeed, metastasis of the original compromised tissue can be found all over the body.

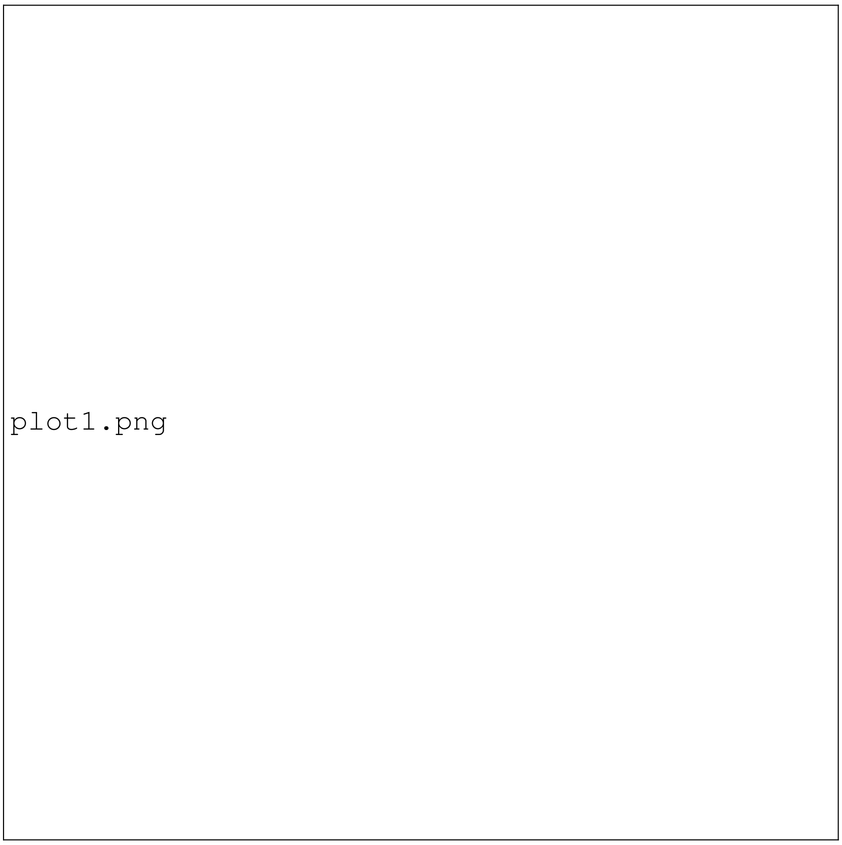

Figure 1: Cancer classes

We hence grouped the various cancer types in 10 classes according to common medical knowledge[4] and we obtained classes as reported in Figure 1. "Eye" is the smallest one as there are only 16 observations, 5 of which labelled as "Enginereed". On the other hand, "Gastrointestinal" is the largest group and it comprehends 7 types of cancer, making this group quite heterogeneous. We decide to investigate two binary classification problems, Blood vs Rest and Lung vs Rest, and the multiclass problem. We chose Lung because of the nature of such class: it is the most numerous group composed only by Lung cancer samples. The choice of Blood was instead driven by some underlying biological knowledge. In fact, Blood cancer is quite different from other tumours because

- blood is in the whole body

[3] Download dataset from DepMap: https://depmap.org/portal/download/:
[4] Cancer types grouped by body location: https://www.cancer.gov/types/by-body-location

- Leukemia, Lymphom and Myeloma are the main kinds of cancer but they all affect white blood cells
- not all blood cancers require a treatment, just periodical monitoring

3 Methods

3.1 Random Forest

Whenever we are only interested in model performance and not in interpretability, Random Forest (RF) is a valid starting point. It frequently performs well on imbalanced data and it is a good compromise when working with correlated high-dimensional data. Furthermore, we can use Variable Importance of RFs to simply detect the most important features. This measure is calculated in three steps. First, prediction accuracy are measured on the out-of-bag samples. Then the values of the variable are randomly shuffled, keeping all other variables the same. Finally, the decrease in prediction accuracy on the shuffled data is measured and the mean decrease in accuracy across all trees is reported. Intuitively, the random shuffling means that, on average, the shuffled variable has no predictive power. This importance is a measure of by how much removing a variable decreases accuracy, and vice versa.
We used this Variable Importance measure to select the most important feature in two different ways:

- *Cross-validation.* We perfomed a 5-fold cross-validation on the model, selected the top most important features of each model and finally averaged.
- *Boruta algorithm*[5]: Boruta repeatedly measures feature importance then carries out statistical tests to screen out the features which are irrelevant. Here the steps:
    1. Create copies of all the original features present in your dataset.
    2. Add randomness by shuffling these new feature. This is done to ensure that these new features show no correlation with the response variable.
    3. Run random forest classifier on the extended dataset and generate feature importance score based on mean accuracy decrease estimate for all the shadow variables.
    4. Compare Z-score of original variable with the maximum Z-score of shadow variables. Real features that have low score compared to the best of shadow features are deemed unimportant.
    5. Remove shadow features and repeat the process until an importance score is generated for all the variables.

3.2 SVM-Lasso

Support vector machines (SVM) is based on the idea of finding a hyperplane that best separate classes. Here, we combine this method with a classical Lasso penalty, so that the objective function to be minimised is:

$$\frac{1}{n}\sum_{i=1}^{n} hingeLoss(y_i(x_i w + t)) + \lambda \sum_{j=1}^{p} |w_j| \qquad \text{where} \qquad hingeLoss(z) = max\{0, 1 - z\}$$

Thus, we obtain the usual sparsity in predictors: some w are shrunken all the way to zero and the remaining are the few relevant features.

3.3 Neural Networks

Nowadays, Neural Networks are a very attractive approach to obtain excellent performances.

[5] https://www.researchgate.net/publication/220443685_Boruta_-_A_System_for_Feature_Selection

4 Results

4.1 Binary classifications

Before starting our Binary classifications on Blood and Lung cancer, we ran a Principal Component Analysis to gain some valuable insights. We noticed a surprising result: even if observations formed a cloud of points, Blood cancer cells were mainly concentrated in just one part of the 3-dimensional plot. The same did not happen for Lung cancer observations.

We split the dataset into two parts, one used to train the model with all the features and select a small subset (~ 200) of them, the other to validate our model with only the selected features. The split was made with a ratio $(60 : 40)$ between first and second split and $(80 : 20)$ between training and test set in each split. The method used for selecting the features was different for RF and NN.

We then moved to RF classifiers.

Random Forests

SVM-Lasso

Neural Networks

4.2 Multiclass classification

When working with the multiclass problem we decided to remove the group "Eye" because it was a small heterogeneous group that even after tuning for RF and NN held the worst accuracy scores. "Eye" was a cluster of different types of cancer: this didn't create problems when we were using a One vs All aproach but with multiclass the effects of doing the cluster ourselves was mor evident. So after removing the group "Eye", we did the same as before and split the dataset in two.

Random Forests

We considered two different random forests, the first with the same parametes as the one used in Blood vs All random forest. Looking at the confusion matrix we can see hat most of the missclassification errors are data points classified as Gastrointestinal. Gastrointestinal was the biggest cluster we consider with a lot of different cancers so the genes(i.e. the features) important can be different within the cluster. To limit the weight of Gastrointestinal we decided to tune by hand the class weights. In this way we solve two problems: we reduced the importance of the group Gastrointestinal and we achieved at least one correct classification in each class. At this point we used cross validation to select five models and we extracted the most important features by looking at their relative importance in every model subjected to their presence in every model. We set a threshold to obtain (~ 200) features and with this we created a reduced model that we fitted using the validation set we had created at the beginning.

SVM-Lasso

Neural Networks

We first fitted a toy NN with only one hidden layer to see how the net worked. We then fitted three different NN: the first increasing by one the number of hidden layers, the second changing the loss function with the focal loss and the third using again the focal loss, but changing the alpha parameter w.r.t. the class percentage in the set. The three models held similar result with the second one being slightly more accurate. We the used the AWS console to perform feature importance because our computers were not sufficiently powerful to do that. (questa non so se metterla)

5 Overall Results

6 Conclusion and future works