Author's Name : **Ali Chemkhi** (M2 BIOINFORMATICS DLAD Aix-Marseille Université)

INTRODUCTION

Here we present a small project where we tried to conduct an exploratory descriptive analysis on cancer RNA-seq data and implemented classical machine learning approaches in order to classify cancer tumors based on their expression profile .

Aims:

- Visualize the data through dimensionality reduction algorithms like PCA , UMAP and T-SNE .
- Implement different machine learning classification algorithms .
- Perform a Benchmarking analysis : Assess the performances of these approaches

DATA

This collection of data is part of the RNA-Seq (HiSeq) PANCAN data set [1], it is a random extraction of gene expressions from patients having different types of tumor:

BRCA : Breast Invasive Carcinoma
KIRC: Kidney Renal Clear Cell Carcinoma
COAD: Colon Adenocarcinoma
LUAD: Lung Adenocarcinoma
PRAD: Prostate Adenocarcinoma

Samples are stored rows. attributes of each sample are RNA-Seq gene expression levels measured by illumina HiSeq platform. (Number of samples : **801** ; Number of attributes (features): **20531)**

MATERIALS AND METHODS

- **Materials :**

Source code is available here (https://github.com/Alich13/ML_RNA-seq_analysis) .The project organisation follows the structure suggested by cookiecutter[3]. The main packages used in this project are Sklearn[4],Keras[5] and Tensorflow[6] .The conda environment utilized here can easily be cloned using either conda or a docker image available in the lutter link .

- **Data exploration and Visualizations**

House-made scripts were used to analyse the RNA-seq expression level data and generate visualizations summarizing the statistics and the distribution of the underlying data. Dimensionality reduction algorithms like (**PCA :**principal component analysis **,UMAP :** Uniform Manifold Approximation and Projection for Dimension Reduction,**T-SNE** : t-distributed stochastic neighbor embedding) were implemented later in order to transform data into much lower dimension space . Afterward we filtered the genes to keep only the highly variable genes (HVG) which are our features . Hence, only the top 1000 HVG were kept for the downstream analysis .

- **Data processing**

We tried two preprocessing methods .The first is the classic standardization which transforms the data to center it by removing the mean value of each feature, then scales the data by dividing non-constant features by their standard deviation. The second method is Robust rescaling (*rescaled* =

(gene_expression - median(gene_expression)) / IQR(gene_expression) where IQR stands for Inter Quartile Range.)

- **Classification algorithms Implementation**

Four classic ML methods were deployed in the aim of classifying tumors based on their RNA expression profile : Support Vector Machine (SVM) ,Decision Tree (DT) , k-nearest neighbors algorithm (KNN) and a Neural Network (NN) . The Neural Network model implemented (**Fig 1.B**) was fitted using the efficient Adam version of stochastic gradient descent minimizing the mean squared error (MSE). Since we have a tremendous number of features (genes), we opted for the transformed data we generated in the preprocessing step (HVG) . Then we implemented an autoencoder (**Fig 1.A**) to learn a compressed representation of the input features for a classification predictive modeling problem (Feature extraction) . We chose a configuration of the model so that the bottleneck layer has 1/10 the number of input nodes (**Fig 1.A**).

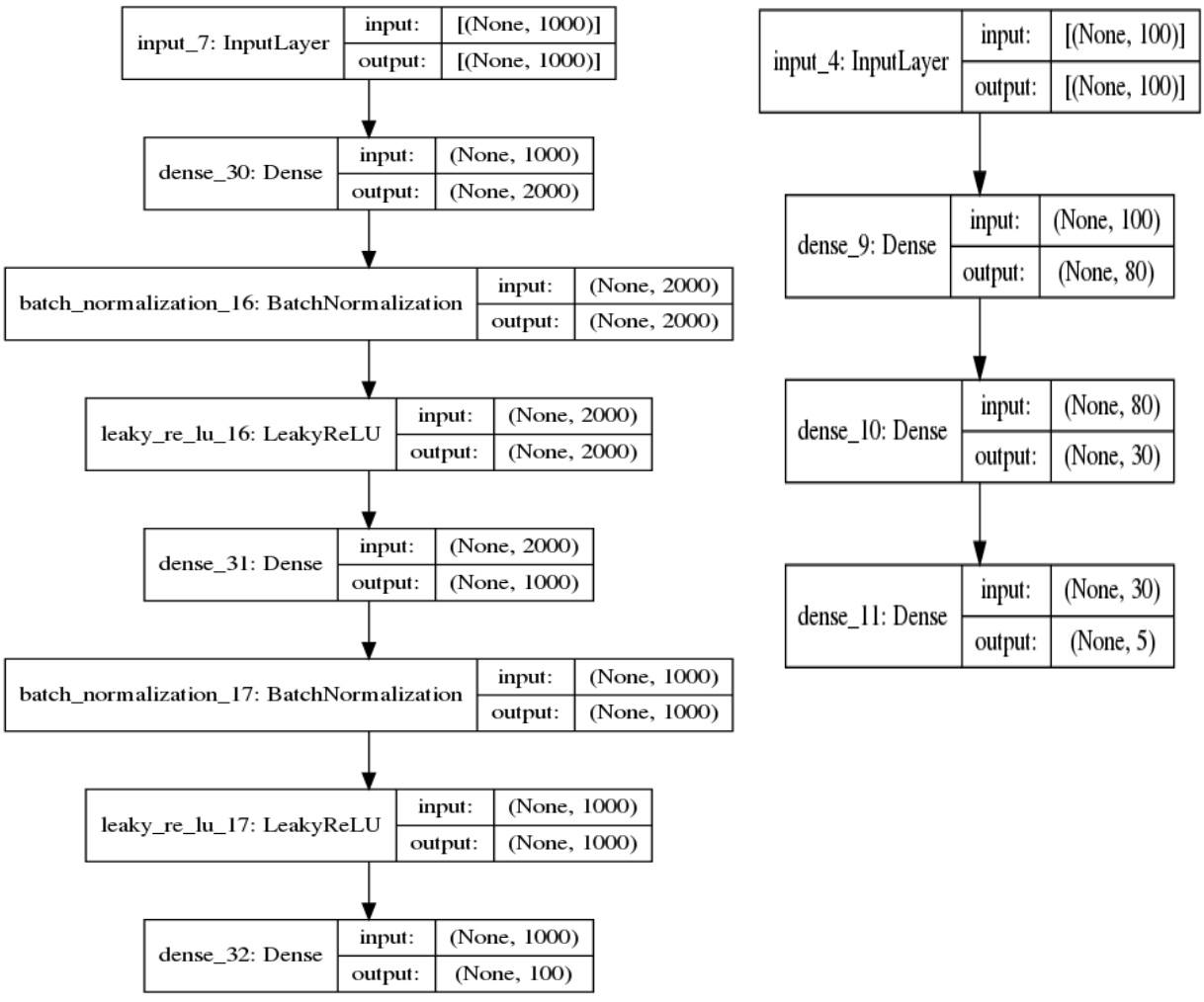


Figure1 (A) Diagram of Encoder Model for Classification used for feature extraction **(B)** Diagram summarizing the Neural network model its input, hidden and output layers

RESULTS

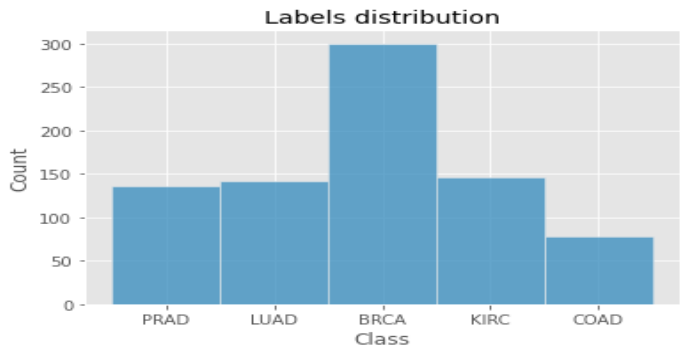
Statistics summaries and plots generated from our in-house scripts showed that the data was already preprocessed (**Fig 2 A**) . In fact expression levels were comparable and in the same range as illustrated by **Fig 2.B**.

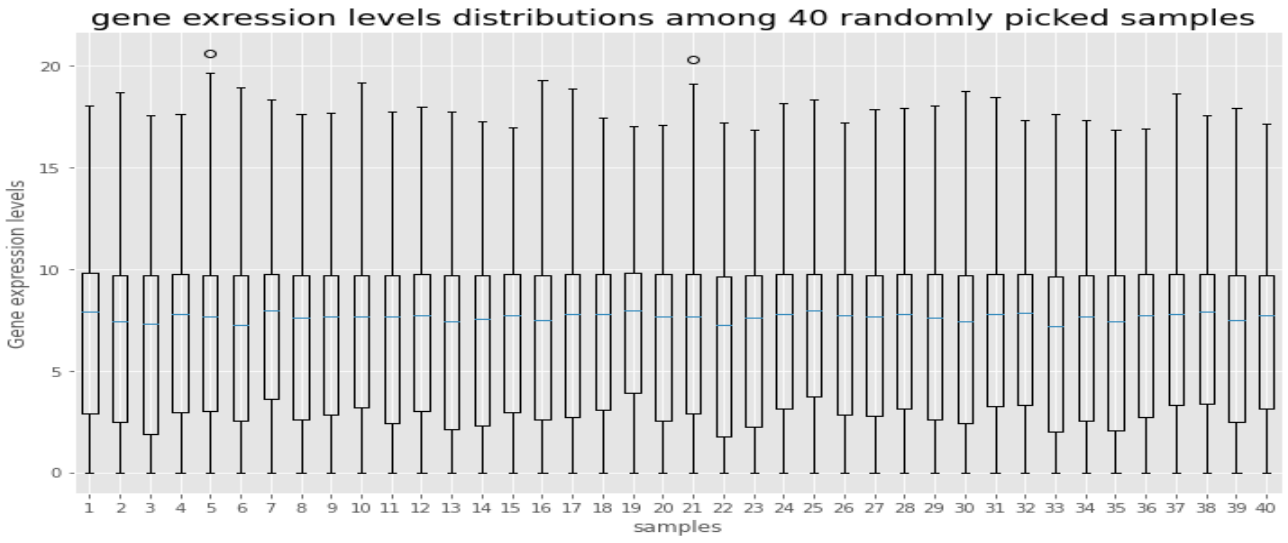
The visualization of the given data through the reduction methods cited above showed a very good separation of the samples with respect to the phenotype tumor type (**Fig 3.A** and **Fig 3.B**).Besides the hierarchical ascending clustering (**Fig 3.D**) applied on unlabeled data as well as the sum of squared distance between each point and the centroid in a cluster (WCSS) as illustrated in the figure Fig 3.C confirmed the presence of 5 clusters corresponding to our 5 labels (tumor types).

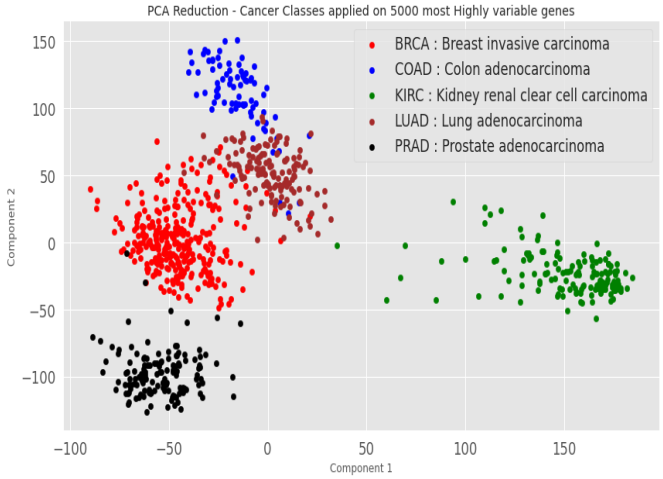
Other figures as well as confusion matrices and accuracy scores can be found in the notebooks in the underlying github repository[2] . Plus, a summary of the obtained results will be presented in the discussion section below . In summary all methods gave excellent accuracy scores (superior to 0.96) .

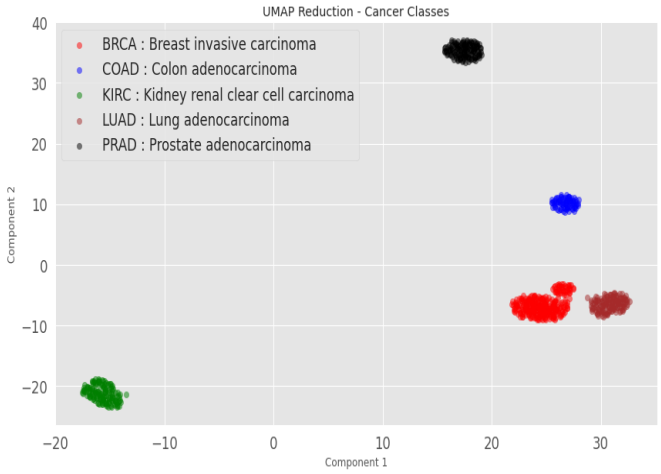
Description :

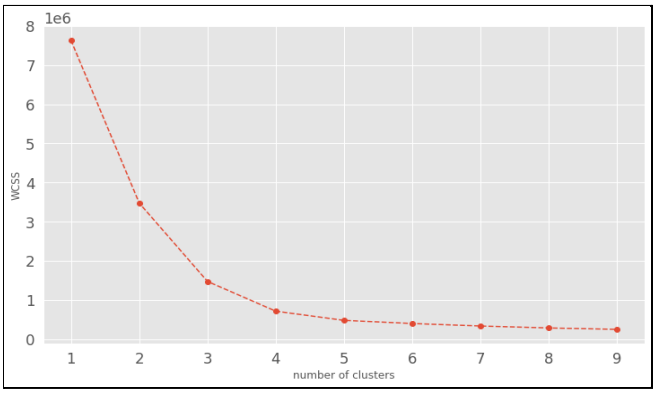
Figure 1: (A) Labels distribution histogram **(B)** Boxplot showing the expression levels distribution for 40 randomly picked sample

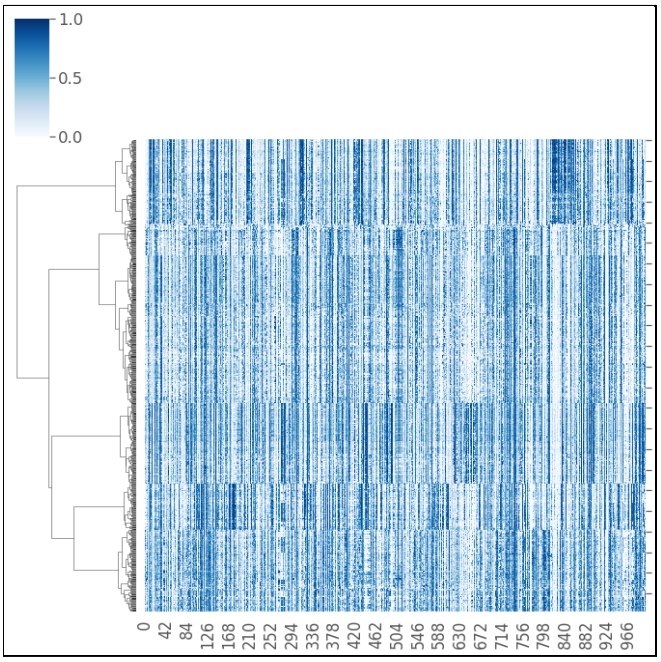






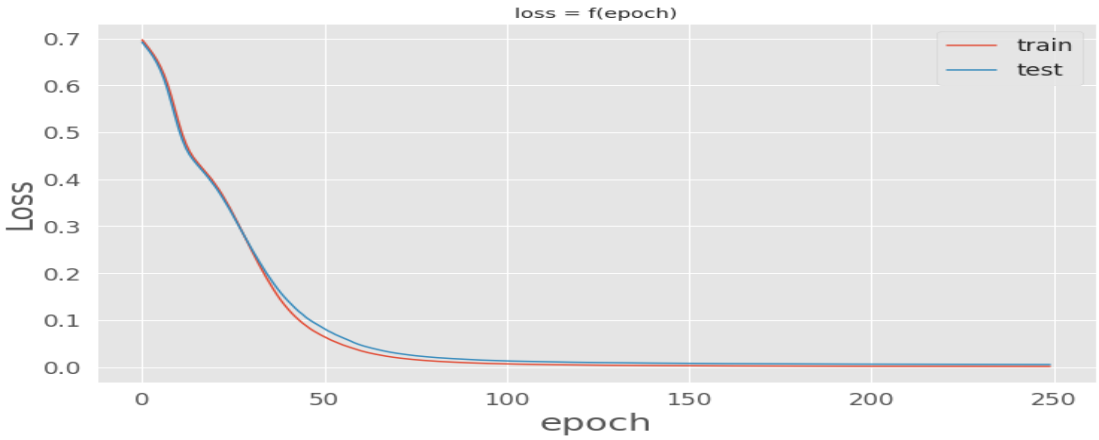






Description :

Figure 3: (A) PCA : 2D projection of the 2 principle components **(B)** UMAP projection **(C)**WCSS graph :We can find the optimum value for K using an Elbow point graph. We randomly initialise the K-Means algorithm for a range of K values and will plot it against the WCSS for each K value. **(D)** expression level heatmap with a dendrogram illustrating the hierarchical ascending clustering



Description: Loss with respect to epoch number of the NN model (Learning curve)

DISCUSSIONS & BIOLOGICAL INTERPRETATIONS

As mentioned earlier one of the main aims of this work is to perform a benchmark analysis (**Tab 1**) .Even Though accuracy scores were very close , small differences were observed , In general processed data gives slightly better results . Also in our case, predictions from models trained with transformed reduced data (UMAP,PCA or TSNE) outperform models trained with original data and this can be explained by the very nature of data .In fact , we think that because biologically speaking the transcriptomic differences between samples from different tumor are too significant and the number of features (genes) is extremely high , the ML methods we applied were able to easily learn the patterns within the data and successfully classify data with unknown labels .

Accuracy	**KNN**	**SVM**	**DT**	**NN**
Original data	Training :0.9982 Test :0.9958	Training :1 Test :0.9958	Training :1 Test :0.9709	An autoencoder Was used to extract features from the original dataset
Processed data	Training :0.9964 Test :1.0	Training :1 Test :1	Training :1 Test :0.9889	
Transformed data (UMAP or PCA) or AutoENCODER	Training :1 Test :1	Training 1: Test :1	Training :1 Test :1	Training :0.9962 Test :1

Tab 1 :Benchmarking analysis : Table summarizing all the accuracy scores

CONCLUSIONS

In summary , data was very easy to learn by all the underlying classifiers we used . In fact , the features describe very well the biological variance between the different classes .These suggestions are comforted by 2-dimension projection of PCA and UMAP reductions . However the number of features is extremely high compared to the number of samples , and despite the absence of overfitting in our learning curves and the fact that excellent scores were obtained from cross validation , We think that complementary tests on data from different sources would be very appreciated in order to be more confident about our models.

REFERENCES

1. UCI Machine Learning Repository: Data Set. https://archive.ics.uci.edu/ml/datasets/gene+expression+cancer+RNA-Seq%20UCI%20-%20Machine%20Learning.
2. Alich13. *Cancer Tumor classification based on RNA-seq_analysis data*. (2022).
3. drivendata. *Cookiecutter Data Science*. (2022).
4. scikit-learn: machine learning in Python — scikit-learn 1.0.2 documentation. https://scikit-learn.org/stable/.
5. Keras: the Python deep learning API. https://keras.io/.
6. TensorFlow. https://www.tensorflow.org