**Tumor Classification Proposal**

**Intro**

The classification of a different tumor types is very important in today’s cancer diagnosis and drug discoveries. If we look back in the history earlier studies of cancer classification were clinical-based and have limited diagnostic ability. Cancer classification using gene expression data is knows to contain the keys of addressing the fundamental problems relating to cancel diagnosis. The discovery of DNA microarray technique has made simultaneous monitoring of gene expressions possible. With that the scientists have the ability to analyze the data and make accurate predictions of different type of tumors which helps identify necessary treatment for the patient.

**Objective**

In this project I will apply three different classifications algorithms to the gene expression dataset with 5 different types of tumors and about 20.5k attributes. The goal is to identify the algorithm that is best suited for current gene expression dataset.

**Data**

The dataset used in this project comes from University of California, Irvine\*. The dataset consists of 801 instances, 20.5k of attributes and 5 different classes.

**Data Cleaning and Preparation**

The data cleaning and preparation will consist of termination of attributes with few to no values and identifying the attributes that represent the most value in classification determination. Dimensionality Reductio (PCA) might be necessary as we have more attributes than instances.

**Methods**

1. **Linear Discriminant Analysis (LDA).**

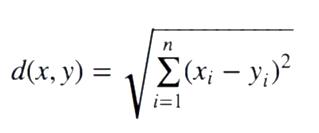
Based on my research LDA is one of the most popular methodologies used when working with gene expressions. This should be specifically good fit in this case as it is able to distinguish and predict a dataset that has more than 2 classes. “The linear Discriminant analysis estimates the probability that a new set of inputs belongs to every class. The output class is the one that has the highest probability. That is how the LDA makes its prediction.

LDA uses Bayes’ Theorem to estimate the probabilities. If the output class is (k) and the input is (x), here is how Bayes’ theorem works to estimate the probability that the data belongs to each class.

P(Y=x|X=x) = (PIk \* fk(x)) / sum(PIl \* fl(x))” [1]

1. **K-nearest neighbors**

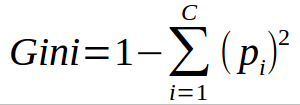
I chose K-nearest neighbors as a second classification method, it is also a great algorithm to be used in this case at it is able to work with multiple class dataset. For the KNN to perform properly the data has to be linearly separable to use the Logistic regression algorithm. As the KNN can perform the multiclass classification it does not require any specific assumptions. It works on all kinds of data on which the classification is to be performed. The preliminary concepts that one needs to be aware of are: nearest neighbors, distance metrics — Euclidian distance, Manhattan distance, Majority vote. All these concepts are the basic mathematical concepts on which the classification in the KNN algorithm is done. “As the classification is done by taking into consideration the K-nearest neighbors, we need to decide the optimum value of ‘K’ or the number of nearest neighbors”[3], which we have in this case as it was provided with the dataset. In this specific case we will be using Euclidean Distance as it is the most widely used and set up as the default metric in SKlearn library of Python for KNN. It is measure of the true straight line distance between two pint in Euclidean space. Formula displayed below:



1. **Random Forest**

Random forest is a flexible, easy to use machine learning algorithm that produces, even without hyper-parameter tuning, a great result most of the time. It is also one of the most used algorithms, because of its simplicity and diversity. The Random Forest Algorithm is composed of different decision trees, each with the same nodes, but using different data that leads to different leaves. It merges the decisions of multiple decision trees in order to find an answer, which represents the average of all these decision trees. “Random Forest uses labeled data to “learn” how to classify unlabeled data. This is the opposite of the K-means Cluster algorithm, which we learned in a past article was an unsupervised learning model. The Random Forest Algorithm is used to solve both regression and classification problems, making it a diverse model that is widely used by engineers.”[2]

Since we have a classification problem on our hands, we will most likely be using Gini Index. This formula uses the class and probability to determine which of the branches is more likely to occur. The formula is displayed below:



**Sources:**

Data: [https://archive.ics.uci.edu/ml/datasets/gene+expression+cancer+RNA-Seq#](https://archive.ics.uci.edu/ml/datasets/gene+expression+cancer+RNA-Seq)

1. (<https://www.digitalvidya.com/blog/linear-discriminant-analysis/#:~:text=The%20linear%20Discriminant%20analysis%20estimates,Theorem%20to%20estimate%20the%20probabilities>.)

2. <https://medium.com/capital-one-tech/random-forest-algorithm-for-machine-learning-c4b2c8cc9feb>

3. https://medium.com/analytics-vidhya/summary-of-knn-algorithm-when-used-for-classification-4934a1040983#:~:text=The%20main%20advantage%20of%20KNN,can%20be%20a%20suitable%20algorithm.