Now, here the problem arises that how to deal with the categorical variable present in our dataset. Because we cannot feed the categorical variables directly to our machine learning algorithm.

For example, in our dataset we have 4 features where 3 are independent and 1 is dependent. Class is dependent and gene, variation and text are independent. Here gene column is categorical so we need to assign some value to this column to feed this data to ML algorithm.

How do we transform the categorical variables?

There are two methods to do that transformation would be as followed:

* One-hot encoding method
* Response encoding

What is one-hot encoding and how do we implement that method in our dataset?

To answer this question first we need to understand what is one hot encoding method. It is a method in which we create extra columns with the name of unique values is our categorical variables. And we create sparse matrix from that.

Like here in our dataset we have 229 unique genes which means we need to create a sparse matrix like structure adding the 229 new columns.

There lies problem with the one-hot encoding method i.e., it increases the dimensionality of the dataset. Few algorithms have big trouble with this type of high dimensionality of the data, specially trees kind of algorithms like Random forest and decision tree (CART) etc.

What is response encoding and how do we implement that method in our dataset?

It is very interesting stuff where we assign some values just like we do with the NaN missing values. For that assigning of some logical values to categorical feature some domain knowledge about the topic plays plus point.

Now, I am going to explain how this works in contrast of this gene feature in our dataset.

As we know we have 9 classes in our dataset and unique values of genes associated with the classes as we have shown in the code.

For each class we create a column and assign probabilities to the classes. Like probability of gene being in a class 3 given that the gene is BRCA1(name of the gene).

Similarly, we deal like this with all the gene values and assign some kind of posterior probabilities to them.