DNA Classification using Decision Tree Classifier

Project Report

Manideep Potluru
mpotluru@unm.edu
101879531

Neelesh Angaluri
neeleshangaluri@unm.edu
101877491

We`re given a dataset consisting of DNA sequences with their splice junctions. The problem was to identify a given sequence of DNA along with the boundaries between exons and introns. This can be done by identifying the EI sites or the acceptors and the IE boundaries or the donors. A given sequence of DNA consists of 60 characters and when randomly a position of one of them is being referred, our classifier should be able to detect which class the given character belongs to. Our approach to this problem is to use an Iterative Dichotomizer variant of the Decision Tree also known as the ID3. It generates a tree by considering few techniques like Entropy, Information gain and Misclassification error. We then use the training data set to train the tree by using these techniques one at a time thereby finding which technique would provide us with the best accuracy. To compute the values of Entropy, Information Gain and Misclassification Error, we proceeded with the established mathematical notations.

Methods to calculate impurity:

Entropy

Entropy is the measure of the randomness in data. It gives a value based on the outcome of an event, if the event has a probability of 100% then the entropy would be 0, if the outcome is 50% then the entropy would be 1 as it is precisely random. Since ID3 would stop traversal when it reaches a node that has 0 entropy, it wouldn't split further. If the entropy is above 0, then it would fit further. We were able to calculate entropy with the following mathematical notation

$$\mathrm{H}(X) = -\sum_{i=1}^{n} \mathrm{P}(x_i) \log_b \mathrm{P}(x_i)$$

where b=2

Gini Index

It gives us the level of impurity of an attribute from the dataset. A low-level value of the Gini index is used as an attribute for splitting the data. This means that, we`re splitting the tree at a point where the impurity is lower compared to the other nodes.

The tree works recursively where a node calculates the Gini Index for each fresh attribute thereby generating new nodes.

$$G = 1-\sum(Pi)^2$$

Misclassification Error: It refers to the number of events that we know must belong to a certain category, but they have been classified into a different category. So, we primarily focus on attaining a value that`s as least as possible. As it is totally dependent on our dataset and the kind of problem we deal with, the best we can do is to try and minimize the error.

$$\text{Misclassification Error}() = 1-\text{Max}(Pi/P)$$

Information Gain: Entropy is used to calculate information gain, it`s the effective change in entropy after deciding on a particular attribute. It measures the decrease in the value of entropy with respect to independent variables.

$$Gain(S, A) = Entropy(S) - \sum v \in Values(A)\ |Sv|/|S|\ \ Entropy(Sv)$$

Chi Square:

The chi Square test focuses on deciding whether or not to split a node based on the input value. Our assumption is that the variables could be independent, and the split is relevant, and the other assumption is that the split is irrelevant the variables are dependent.

$$\chi_c^2 = \sum \frac{(O_i - E_i)^2}{E_i}$$

If the ChiSquare value > critical value then we can conclude that the split is noteworthy. Else, it would not split any further.

The critical value is based on the mapping between confidence and the degrees of freedom.

Approach to the Problem

Initially, we preprocess the given dataset in way such that it can be easily accessed further. The dataset is put into a NumPy array and then the DNA sequence of 60 characters is being divided in such a way that the list of 60 individual characters is appended to a 61st character which is the class that the DNA sequence belongs to. Appending this way, the list containing 60 characters turns into a list of 61 characters. Now, the whole data is in the form of a matrix such that each character within the sequence is referenced through a distinct row and column. We then had to write functions for Gini index, Misclassification error and Entropy to determine their values. We then start building an ID3 tree with the parameter being the current_dataset, complete_dataset, characteristics, labels_column and the upper_node_label. Here, we consider each column`s position as a characteristic feature. The labels_column refers to the 60th column in our dataset and that sets up the base label for our tree which is to return the label, it`s the only one in the 60th column. The tree traversal continues in such a way that when at a node if there are no new features then the label of the parent node shall be returned. The best characteristic is then obtained through Information Gain using one of the three methods to calculate impurity and this best characteristic is then found from the list through its index and this characteristic is then added into the tree. For a certain characteristic we create sub data of its column and for that subdata we recursively call the ID3 algorithm. This keeps on going until the chi square condition turns out false or one of the prior conditions are satisfied.

Results

Initially, without implementing chi-square, we had a prediction accuracy of 0.86344, 0.88445, and 0.87394 while using entropy, misclassification-error and Gini-index respectively. After the implementation of the chi-square, we have had accuracies increased as the level of confidence increased. With a higher confidence level of 95, we have accuracies of 0.89285, 0.91386 and 0.89075 and with a 99 level of confidence, we have even higher accuracies of 0.89915, 0.91176 and 0.89285. We have provided a table with all the obtained accuracies below:

Gain/Confidence level	0.00	0.95	0.99
Entropy	0.86344	0.89285	0.89915
Misc. error	0.88445	0.91386	0.91176
Gini index	0.87394	0.89075	0.89285

In our case, we had higher accuracies in the case where we used misclassification for our information gain. Although in theory, in most cases of classification entropy and Gini-index are considered better methods for classification, we have a dataset which is peculiar and makes us recall how data specific machine learning is.

We can also notice that implementing the chi-square resulted in building our tree is such a way that our tree does not overfit our training data. This led to the tree being more generic and thus has resulted in us getting improved accuracies while predicting the testing dataset.

Changes made to improve accuracies

We had very low accuracy scores for all of the methods while we have initially implemented chi-square, after careful analysis, we have noticed that whenever the chi-square triggers stopping of the tree, we missed the labels of those nodes and set those to "N". Later we had them replaced with the labels of the upper feature where the tree stopped. We also found a few cases where, we were not able to classify few DNA, which we have handled by replacing them with the highest available labels in the dataset.

References: *https://en.wikipedia.org/wiki/ID3_algorithm*
Machine Learning -Tom Mitchell