Contents

Project 2: Decision Tree

Jaudat Raza JAUDAT.RAZA@JHU.EDU
Master of Computer Science
John Hopkins University
Baltimore, MD 21218

Editor: Jaudat Raza

Abstract

This document will cover the implementation process of decision tree for classification and regression. The resource used will be linked below in the resource section of the document. The document will follow a simple pattern for which a less than 5-minute video will be submitted as well. The Jupyter Notebook starts with getting the data into Pandas Data from the link provided in the data source. After that, first the first three data set will be processed through decision tree for the classification and the other 3 will be processed through decision tree for regression. In classification tree, implement the gain ration criteria.

1 Problem Statement

The purpose of this assignment is to give you a chance to get some hands-on experience implementing Decision Tree for classification and regression. Breast Cancer, Car Evaluation and Image Segmentation will run through Decision Tree for classification. Abalone, Computer Hardware, and Forest Fire will run through the Decision Tree for regression. I expect that after pruning the data using the reduced error method for the classification tree, the results will improve. Also, with the regression tree, by adding a early cutoff in making the tree will hurt the results by increasing our error.

2 Algorithm Studied

In this project we are covering Decision Tree, which is a non-parametric supervised learning method used for classification and regression.

2.1 Decision Trees

Decision Tree is a tree like structure, which uses decision (internal Nodes), to finds its way to a leaf Node, which are the result we get after making some decisions. Internal Nodes have arrows pointing toward them and they point toward either another Internal Node or the Leaf. Leaf Node on the other hand only have Internal Nodes pointing toward the Leaf. Leaf Nodes do not point toward anything.

The first step toward building a decision tree is to find the root of the tree. The way to find the use all the attribute and make see how well they perform making decision. The top root is best predictor our of all the attributes in the data. To find the root note, use the Gini impurity formula below to calculate which attribute has the lowest impurity. Place the attribute with the lowest impurity in the root. Then do same method to find the right and the left child of the root by finding the attribute with the lowest impurity for left and right and traverse down till you get to a point where the Gini impurity does not decrease.

2.1.1 Handle Numeric Data

1) Sort the Numeric attribute
2) Calculate the average for all adjacent patient

3) Calculate the Gain Information (Entropy) for each average value calculated in the previous step.

2.1.2 Handling Ranking Data

1) Calculate the Gain Information (Entropy) for each rank in the attribute.

2.2 Decision Tree for Classification

We will use Information Gain as our criterion for selection measure for partitioning the data. To build the tree. We implement a function which finds attribute that returns us the highest information gain. We calculate the information gain for all the attributes and the one with the best result is selected for node.

Entropy Equation:

$$H(X) = -\sum_{X} p(x) * log_2 p(x)$$

Information Gain:

$$\text{Information} = \text{entropy(parent)} - \text{Weights Average} * \text{entropy(Children)}$$

entropy(parent):

$$Info(D) = -\sum_{1}^{m} p_i \, log_2(p_i)$$

weightsAverage * entropy(Childeren):

$$InfoA(D) = -\sum_{j=1}^{v} \frac{abs(Dj)}{abs(D)} * Info(D)$$

Find the most information Gain and make that the decision node and do that till you get to a leaf or the information

2.3 Decision Tree for Regression

A decision tree is built top-down from a root node and involves partitioning the data into subsets that contain instances with similar values (homogenous).

Hours Played
25
30
46
45
52
23
43
35
38
46
48
52
44
30

$$\textbf{Count} = n = 14$$

$$\textbf{Average} = \bar{x} = \frac{\sum x}{n} = 39.8$$

$$\textbf{Standard Deviation} = S = \sqrt{\frac{\sum (x - \bar{x})^2}{n}} = 9.32$$

$$\textbf{Coeffeicient of Variation} = CV = \frac{S}{\bar{x}} * 100\% = 23\%$$

The std of the target variable is calculated. The data is split on different attribute. The std for each branch is calculated, the resulting std is subtracted from std before the split, the attribute with the largest std is chosen for the decision node. This process is run recursively until all data is processed. During the creation of the tree, we will see that if let go without a termination it could go for a while, which leads us to a termination criterion. In our case if the tree becomes larger then given depth it would not go any further then that.

2.4 Prediction

Once the tree is created. The prediction is very simple. Just follow the node from the start and return the leaf class that is the result of the decision made by the test data.

3 Data Split

- For Classification tree, the training set was 90% of the whole data set, which was then split into 5 Folds
- For Regression tree, the training set was 90%.

4 Data Clean Up

This is the section that describes all the data and how it was updated to make it work with the Algorithm. This will be described in the video as well.

4.1 Abalone

1) The only change made in the data was the "Sex" Column. The infants were changed to 1, Female were changed to 2, and Male were changed to 3.

4.2 Breast Cancer

1) Dropped missing data from the set
2) Change the class, where 2 is 1 and 4 is 0 now
3) Converted BareNuclei to numeric type so the math could be done on it.

4.3 Forest Fires

1) The Month column was updated from 1 to 12, where 1 is January and 12 was December.
2) The Day column was updated from 1 to 7, where 1 was Monday and then 7 was Sunday.

4.4 Segmentation

1) The column name was updated
2) Dead elements were dropped out of the data

4.5 Car

1) All the attributes were converted to integers.

4.6 Machine

1) Model column in the data is serving the unique identifier code purpose, so all of them were updated from their name to integer from 1 to 208.
2) Vendor column had 29 models in it, so they were changed to 1 to 29.
3) Vendor column was set as the last column of the data.

5 Results

The Trees for the classification are stored in their given folder. Each dataset has five trees for each fold and then one with the pruned set of Fold 5 as an example in the src folder of the submission.

5.1 Abalone

Stopping Criteria	Depth	MSE
No	2251	0.86
Yes	450	2.64

5.2 Forest Fires

Stopping Criteria	Depth	MSE
No	237	11.3
Yes	100	42.6

5.3 Machine

Stopping Criteria	Depth	MSE
No	94	2.27
Yes	70	4.71

5.4 Breast Cancer

	Original	Pruned
Leaves Count	7	8
Accuracy	91.20%	92.75%

5.5 Image Segmentation

	Original	Pruned
Leaves Count	37	38
Accuracy	00.00%	00.00%

5.6 Car Evaluation

	Original	Pruned
Leaves Count	48	24
Accuracy	89.60%	93.06%

6 Conclusion

As we had expected for the regression tree. When the criteria to stop the tree at certain depth was given, the MSE incremented. Testing different depths on Jupyter Notebook showed that smaller the tree higher the MSE. For Forest Fire data the results provided above had the Max Depth of 100, which resulted in 42.6 of MSE, which is obviously not great. The results change dramatically when it could go to the depth of 200 from 42.6 to

For classification Tree, Pruning did not show as much of the improvement as I was expecting in my hypothesis. The biggest improvement was in the Car Evaluation. Image Segmentation was not able to do much because it needed some data filtering. To enhance those decimal values to make the leaves a little less complicated. 38 Leaves with only 188 training data set. The results were not going to be great as expected. There are other ways of pruning like Top Down, Bottom

Up and others which could be implemented and then the result could be compared for further research.

7 Resources

- “Decision Tree Pruning.” Wikipedia, Wikimedia Foundation, 26 Aug. 2020, en.wikipedia.org/wiki/Decision_tree_pruning.
- Dectrees, Mitchell. *Chapter 3 Decision Tree Learning*. Princeton Education , 2020, www.cs.princeton.edu/courses/archive/spr07/cos424/papers/mitchell-dectrees.pdf.
- Sayad, Saed. “Decision Tree - Regression.” *Decision Tree Regression*, www.saedsayad.com/decision_tree_reg.htm.
- Tyagi, Neelam. “Understanding the Gini Index and Information Gain in Decision Trees.” *Medium*, Analytics Steps, 30 Sept. 2020, medium.com/analytics-steps/understanding-the-gini-index-and-information-gain-in-decision-trees-ab4720518ba8.