Final Report
01. Decision Tree
02. K Nearest Neighbor

CSE-0408 Summer 2021

Md. Evan Khan Emon
UG02-47-18-009
Department of Computer Science and Engineering
State University of Bangladesh (SUB)
Dhaka ,Bangladesh
emonevankhan@gmail.com

***Abstract*—In this report we discuss about Decision Trees**

Decision Trees usually implement exactly the human thinking ability while making a decision, so it is easy to understand.A Decision Tree is a supervised Machine learning algorithm. It is used in both classification and regression algorithms. The decision tree is like a tree with nodes. The branches depend on a number of factors. It splits data into branches like these till it achieves a threshold value. A decision tree consists of the root nodes, children nodes, and leaf nodes.

I. INTRODUCTION :

Decision trees are assigned to the information based learning algorithms which use different measures of information gain for learning. We can use decision trees for issues where we have continuous but also categorical input and target features. The main idea of decision trees is to find those descriptive features which contain the most ”information” regarding the target feature and then split the dataset along the values of these features such that the target feature values for the resulting sub datasets are as pure as possible The descriptive feature which leaves the target feature most purely is said to be the most informative one. This process of finding the ”most informative” feature is done until we accomplish a stopping criteria where we then finally end up in so called leaf nodes. The leaf nodes contain the predictions we will make for new query instances presented to our trained model. This is possible since the model has kind of learned the underlying structure of the training data and hence can, given some assumptions, make predictions about the target feature value (class) of unseen query instances. A decision tree mainly contains of a root node, interior nodes, and leaf nodes which are then connected by branches.

II. LITERATURE REVIEW:

Decision Tree induction is a classification pattern recognition approach that is applied in both CART and C5.0 algorithms. A decision tree is an inductive inference method widely used in a supervised classification learning technique for “[classifying] instances by sorting them down the tree from the root to some leaf node, which provides the classification of the instance. Each node in the tree specifies a test of some attribute of the instance, and each branch descending from that node corresponds to one of the possible values for this attribute. An instance is classified by starting at the root node of the tree, testing the attribute specified by this node, then moving down the tree branch corresponding to the value of the attribute. This process is then repeated for the subtree rooted at the new node (Bahety, 2014). The construction of a decision tree required two conceptual phases: growing and pruning. By its nature, the decision tree classifier is a greedy algorithm because the tree is grown by “recursively splitting the training set based on local optimal criteria until all or most of the records belonging to each of the partitions bearing the same class label”. With that respect, the tree usually contains overfitting data. Overfitting the data occurs when a predictive model would classify perfectly the known data, but fail to classify anything useful on the yet-unseen data.This would mean that the decision trees have fulfilled their objectives and have indeed discovered some underlying property of the data (Quinlan, 1986). Thus the attempt to make a tree not too closely taking on inaccurate data that can infect itself with substantial errors and reduce its predictive power is called pruning. The pruning phase handles the problem of over-fitting the data by removing the noise and outliers, which eventually increases the accuracy of the classification

III. ADVANTAGE OF DECISION TREE

i.Considered a white box type of ML algorithm, decision tree uses an internal decision-making logic; this means that the acquired knowledge from a data set can be easily extracted in a readable form which is not a feature of black box algorithms such as Neural Network. This makes the training time of decision tree faster compared to the latter.

ii.Due to its simplicty, anyone can code, visualize, interpret, and manipulate simple decision trees, such as the naive binary type. Even for beginners, the decision tree classifier is easy

to learn and understand. It requires its users minimal effort for data preparation and analysis.

iii.The decision tree follows a non-parametric method; meaning, it is distribution-free and does not depend on probability distribution assumptions. It can work on high-dimensional data with excellent accuracy.

iv.Decision trees can perform feature selection or variable screening completely. They can work on both caregorical and numerical data. Furthermore, they can handle problems with multiple results or outputs.

IV. DISADVANTAGE OF DECISION TREE

i.A small change in the data can cause a large change in the structure of the decision tree causing instability.

ii.For a Decision tree sometimes calculation can go far more complex compared to other algorithms.

iii.Decision tree often involves higher time to train the model.

iv.The Decision Tree algorithm is inadequate for applying regression and predicting continuous values.

V. DECISION TREE ALGORITHM:

01. Select the best attribute using Attribute Selection Measures(ASM) to split the records.

02. Make that attribute a decision node and breaks the dataset into smaller subsets.

03. Starts tree building by repeating this process recursively for each child until one of the condition will match:

*All the tuples belong to the same attribute value.

*There are no more remaining attributes.

*There are no more instances.

2. Transform that attribute to become a decision node and divide the dataset to create smaller subsets.

3.Start to create trees by repeating the process recursively for each child. When it matches one of the conditions below, the process shall end: All the tuples are contained in the same attribute value.

* No more attributes remain.

* No more instances.

VI. CONCLUSION :

Decision Trees are easy to interpret, don't require any normalization, and can be applied to both regression and classification problems. Unfortunately, Decision Trees are seldom used in practice because they don't generalize well. Stay tuned for the next article where we'll cover Random Forest, a method of combining multiple Decision Trees to achieve better accuracy.

***Abstract*—In this report we discuss about K Nearest Neighbor Algorithm In Python:**

KNN (k-nearest neighbor) is an extensively used classification algorithm owing to its simplicity, ease of implementation and effectiveness. It is one of the top ten data mining algorithms, has been widely applied in various fields. KNN has few shortcomings affecting its accuracy of classification. It has large memory requirements as well as high time complexity K-Nearest Neighbours (KNN) is an effortless but productive machine learning algorithm. It is effective for classification as well as regression.

VII. INTRODUCTION :

The k-Nearest-Neighbors (kNN) method of classification is one of the simplest methods in machine learning, and is a great way to introduce yourself to machine learning and classification in general. At its most basic level, it is essentially classification by finding the most similar data points in the training data, and making an educated guess based on their classifications. Although very simple to understand and implement, this method has seen wide application in many domains, such as in recommendation systems, semantic searching, and anomaly detection. K-Nearest Neighbors, or KNN for short, is one of the simplest machine learning algorithms and is used in a wide array of institutions. KNN is a non-parametric, lazy learning algorithm. When we say a technique is non-parametric, it means that it does not make any assumptions about the underlying data. In other words, it makes its selection based off of the proximity to other data points regardless of what feature the numerical values represent. Being a lazy learning algorithm implies that there is little to no training phase. Therefore, we can immediately classify new data points as they present themselves.

VIII. PROPOSED METHODOLOGY:

A data set with lots of different points and labelled data is the ideal to use. The best languages to use with KNN are R and python. To find the most accurate results from your data set, you need to learn the correct practices for using this algorithm.

IX. KNN ALGORITHM

To understand better the working KNN algorithm applies the following steps when using it:

Step 1 – When implementing an algorithm, you will always need a data set. So, you start by loading the training and the test data.

Step 2 – Choose the nearest data points (the value of K). K can be any integer.

Step 3 – Do the following, for each test data –

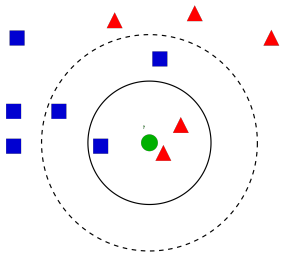
1 – Use Euclidean distance, Hamming, or Manhattan to calculate the distance between test data and each row of training. The Euclidean method is the most used when calculating distance.

2 – Sort data set in ascending order based on the distance value.

3 – From the sorted array, choose the top K rows.

4 – Based on the most appearing class of these rows, it will assign a class to the test point.

Step 4 – End



X. ADVANTAGES OF KNN:

1. Quick calculation time.

2. Simple algorithm – to interpret.

3.Versatile – useful for regression and classification.

4.High accuracy – you do not need to compare with bettersupervised learning models.

XI. DISADVANTAGES OF KNN

1.Accuracy depends on the quality of the data.

2. With large data, the prediction stage might be slow.

3.Sensitive to the scale of the data and irrelevant features.

XII. SUMMARY

K is a positive integer,

With a new sample, you have to specify K K is selected from database closest to the new sample KNN doesn't learn any model KNN makes predictions using the similarity between an input sample and each training instance. This blog has given you the fundamentals of one of the most basic machine learning algorithms. KNN is a great place to start when first learning to build models based on different data sets. Data set with a lot of different points and accurate information is your best place, to begin with KNN.

XIII. CONCLUSION :

The K Nearest Neighbors algorithm doesn't require any additional training when new data becomes available. Rather it determines the K closest points according to some distance metric (the samples must reside in memory). Then, it looks at the target label for each of the neighbors and places the new found data point into the same category as the majority. Given that KNN computes distance, it's imperative that we scale our data. In addition, since KNN disregards the underlying features, it's our responsibility to filter out any features that are deemed irrelevant.

ACKNOWLEDGMENT

I would like to thank my honourable **Khan Md. Hasib Sir** for his time, generosity and critical insights into this course.