DECISION TREES

ASSIGNMENT 3

Further Artificial Intelligence

Task 3 (20%): *How well motivated is the choice of dataset?*

- Is the choice of dataset and feature predicted with the decision tree well justified and explained? (10%)

The dataset was chosen from UC Irvines' Machine learning repository: https://archive.ics.uci.edu/ml/datasets/car+evaluation In the machine learning realm, this dataset was utilized for the assessment of HINT (Hierarchy INduction Tool), which was shown to be capable of totally recreating the original hierarchical model. The dataset consists of 1728 number of Instances, 6 attributes, attribute values are the following: {buying v-high, high, med, low}, {maint v-high, high, med, low}, {doors 2, 3, 4, 5-more}, {persons 2, 4, more}, {lug_boot small, med, big}, {safety low, med, high}, and there are no missing attribute values. This dataset is also found in multiple Kaggle entries used to explore decision trees. As such, considering the reputation of UCI and Kaggle as public datasets, as well as the attribute values shown, then the dataset is an appropriate choice (Tan, 2017).

As for the features predicted, the information does bring back value: for the weather dataset: the output was used to decide whether to play tennis. And as for the car evaluation dataset: it is to decide whether a car is acceptable.

- Is the dataset chosen suitable to a DT approach? Beyond accuracy, are the benefits of using a DT approach explained? (10%)

Decision trees are employed for managing non-linear datasets effectively. As in the cases at hand, a categorical variable decision tree contains categorical output variables that are split into classifications. For instance, if the categories are binary such as: "yes" or "no", then the leaf at the end of the decision tree will classify the prediction as one of these 2 categories. The Decision Tree is suitable for the following reasons: 1. An instance is embodied as pairs of attribute-value. For instance, in the weather dataset, "Wind" can be classified as "Weak" or "Strong". 2. The output function has discrete targets. It should deal with Boolean values such as "True", or "False". 3. The training data might include errors that will be handled with pruning. It is also suitable because of its inductive learning appeal such as: 1. Decision tree is an adequate abstraction for unseen instances, only if the instances are expressed in terms of features that are associated with the target class. 2. The methods are effective in computation that are proportionate to the number of detected training instances. 3. The output of the decision tree offers a depiction of the notion in a tree format that is desirable to humans since it makes the Classification procedure self-evident. (Tan, 2017)

Task 4 (20%): *How well are the results explained and contextualized?*

- Are clear hypotheses about the data made and tested using the DT produced? (10%)

There are several concepts mentioned, used, and tested in the notebook. Starting with entropy which sheds light on the extent that a random variable value contains uncertainty or in an output of a random method. The entropy value shown below is considered as a high value, meaning there is a low amount of purity or a high level of uncertainty. This is where information gain comes into play to inform about the decrease in uncertainty of the output variable (Nowozin, 2012).

$$E = \sum_{i=1}^{c} -p_i log_2 p_i$$

Figure 2: Entropy formula

$$IG(N, a) = E(N) - E\left(\frac{N}{a}\right)$$

Figure 1: Information Gain Formula

```
print("Entropy is equal to: ", entropy_calculation(dataset[y_target]))
for i in range(len(feats)):
    print("Class: ", feats[i], "\t IF: ", info_gain(dataset, feats[i], y_t
```

```
Entropy is equal to:  0.9402859586706311
Class:  Outlook          IF:  0.24674981977443933
Class:  Temperature      IF:  0.02922256565895487
Class:  Humidity         IF:  0.15183550136234159
Class:  Wind     IF:  0.04812703040826949
```

Figure 3: Entropy and IG in code

As an extension, what is also considered is the maximum depth of tree. If max depth isn't set, then the nodes will keep expanding until all the leaves are explored. So, it is set to reduce overfitting and allowing better generalization of predictions. But also, need to be considerate of not setting lower values or else there will be underfitting (Dietterich, 1995).

```
prune='depth', max_dpth=4,
```

Figure 4: Max_depth

As for the choice of the algorithm, ID3 was first selected, but pruning is a downside and not done, then a CART algorithm is implemented to show its advantages. ID3 constructs a decision tree for the presented data in a top-down approach. The object set is then split based on increasing IG and decreasing entropy. This process is done recursively. It uses a greedy search that selects a test using the IG criteria and doesn't explore alternative choices which makes it an efficient algorithm. Advantages: -Comprehensible prediction rules are generated from the training data. -Creates the tree the quickest. -Creates a short tree. -Only requires testing sufficient attributes till all data is categorized. -Obtaining leaf nodes allows test data to be pruned, decreasing number of tests. -All the dataset is explored to generate the tree. Disadvantages: -Data might be overfitted if a short sample is used. -Only one attribute is used at a time to decide. As for the CART algorithm: it builds decision tree with evert internal node having only two outgoing edges. The divisions are chosen by comparing the Gini indices, that are calculated by combining category and feature noises by calculating the weighted sum of the Gini impurity, and the created tree is pruned. It also allows users to offer previous probability distribution (Lerman, 1984). Although not used in our case, an important feature of CART is its capability to create regression trees. Advantages: -Can manage categorical and numerical variables. -Will detect the adequate variables to consider and remove unimportant ones. -Can manage outliers. Disadvantages: -Might have unstable DT. -Splits are done only by one variable (Tan, 2017).

Characteristic(→) / Algorithm(↓)	Splitting Criteria	Attribute type	Missing values	Pruning Strategy	Outlier Detection
ID3	Information Gain	Handles only Categorical value	Do not handle missing values.	No pruning is done	Susceptible on outliers
CART	Towing Criteria	Handles both Categorical and Numeric value	Handle missing values.	Cost-Complexity pruning is used	Can handle Outliers
C4.5	Gain Ratio	Handles both Categorical and Numeric value	Handle missing values.	Error Based pruning is used	Susceptible on outliers

Figure 5: Basic characteristics of decision tree algorithms

- Are the results intelligently reflected upon? Are the results used to make observations and conclusion about the dataset and its source? (10%)

Started by comparing the graph output with that from the course, then the accuracy of the custom algorithm is calculated by splitting the dataset into test and train, predicting the outcomes, and determining the accuracy. If the sample dataset was short like the weather one, then the accuracy could be compared to that of sklearn. Once those comparisons are done then the conclusion can be drawn that the algorithm is implemented correctly, and the dataset and source are chosen adequately. Other metrics were also assessed for the acceptability of the data, as previously mentioned: Entropy, Gini index, and generalization as the accuracy is tested on a sample test set rather than the original.

```
Accuracy from sklearn
0.6666666666666666
Count of Correct Predictions = 2
Count of Total items = 3
Accuracy = 0.6666666666666666
Comparison column:
   Decision   0 compare
0        No  No    True
5        No  No    True
10      Yes  No   False
```

Figure 6: Weather dataset: Accuracy comparison between custom DT algorithm implemented vs sklearn

```
Accuracy from sklearn:
0.9479768786127167
Count of Correct Predictions = 317
Count of Total items = 346
Accuracy = 0.9161849710982659
Comaprison column:
     decision      0 compare
986       acc    acc    True
1707    unacc  unacc    True
1539    unacc  unacc    True
729     unacc  unacc    True
1026    unacc  unacc    True
```

Figure 7: Car-eval dataset: Accuracy comparison between custom DT algorithm implemented vs sklearn

References

Tan, L., 2017. Comparative study Id3, cart and C4.5 decision tree algorithm: A survey. [Online]. Available from: https://www.academia.edu/34100170/Comparative_Study_Id3_Cart_And_C4_5_Decision_Tree_Algorithm_A_Survey [Accessed 1 January 2022].

Lerman, R.I. and Yitzhaki, S., 1984. A note on the calculation and interpretation of the Gini index. *Economics letters* [Online], 15(3–4), pp.363–368. Available from: https://doi.org/10.1016/0165-1765(84)90126-5.

Dietterich, T., 1995. Overfitting and undercomputing in machine learning. *ACM computing surveys* [Online], 27(3), pp.326–327. Available from: https://doi.org/10.1145/212094.212114.

Nowozin, S., 2012. Improved information gain estimates for decision tree induction. *arXiv [cs.LG]* [Online]. Available from: http://arxiv.org/abs/1206.4620 [Accessed 5 January 2022].