Comparative Study of ID3, C4.5, CART and CHAID Decision Tree Algorithms

Razan Alshehri, Xinyu Cai, Zihan Xie, and Ningyuan Xiong

***Abstract*— Data mining is the useful tool to discover the knowledge from big data. Various data mining algorithms available for classification based on Artificial Neural Network, Nearest Neighbour Rule & Baysen classifiers but decision tree mining is simple one. The objective of this paper is to compare ID3, C4.5, CART and CHAID decision tree algorithms. At first, the project presents the literature review, methods and theories behind each method. Then, the four algorithms are applied to three real-world datasets and two artificial datasets. Lastly, the project analyzes the advantages and disadvantages of each algorithm and makes a comparison between these four algorithms.**

I. INTRODUCTION

Decision trees are very common and are widely used in the field of data mining. They are simple, yet very effective classification methods. There are several reasons that motivated us to choose this topic. Firstly, the decision tree requires less effort for data preparation during pre-processing. Secondly, the Decision tree model is very intuitive and easy to explain to audience that even don't know about technical terms. Even a naive person can understand logic. The underlying concept of decision trees is recursive partitioning where the decision tree algorithm recursively partition data into different subsets, or nodes until all the partitioned data have the same target variable. Sometimes, partitioning stops when further splitting does not add further value, or information, to the predictions of the target variable. The four algorithms the project chose build on each other. The C4.5 is a successor of ID3, the CART is very similar to C4.5 but have more flexibility, and the CHAID has a different stopping methods from the CART. Overall, C4.5 gives the highest accuracy on datasets with both continuous and categorical variables since it has built-in pruning strategy. CART gives the highest accuracy on datasets with continuous variables as CART could deal with continuous variables and has pruning strategy with manually tuning the hyper-parameters. All four methods give a 100% accuracy on easily classifiable artificial dataset. CART performs best on artificial dataset with many noises by using the optimal pruning strategy.

II. LITERATURE REVIEW

ID3

The ID3 algorithm constructs the decision tree by employing a top-down, greedy search through the given sets to test each attribute at every tree node. The disadvantage of this algorithm is that it does not handle numeric attributes or missing values, and only one attribute is tested at a time for decision-making [1]. The concept of pruning is not present in the ID3 algorithm [2]. Besides, ID3 works fairly well on classification problems having datasets with nominal attribute values. It also works well in the case of missing attribute values but the way missing attributes are handled actually governs the performance of the algorithm [3]. In the social science domain, the ID3 algorithm has been successfully used to predict student placement and generate decision rules based on students' profiles [4]. In the natural science domain, the ID3 algorithm is used to build a forestry resource classification rule decision tree and analyze the correlation information among various factors in a forest [5].

C4.5

C4.5 algorithm was proposed in 1993, again by Ross Quinlan, to overcome the limitations of the ID3 algorithm discussed earlier [11]. During the construction of the decision tree, it is possible to manage data for which some attributes have an unknown value by evaluating the gain or the gain ratio for such an attribute considering only the records for which this attribute is defined [12]. The induced decision tree is pruned using pessimistic error estimation (Quinlan, 1992) [13]. C4.5 also manages the cases of attributes with values in continuous intervals instead of only discrete variables [11]. In the natural language processing domain, C4.5 is used to generate a decision tree to classify semantics (positive, negative, neutral) for the English documents [14]. In universities, every year there are new students who retire that do not register, therefore, it takes an application that can process a lot of data to find out the possible retirement for new students. To find out the prediction retirement prospective students, C4.5 is used, which helps change many facts into rules that can be easily visualized [15].

CART

Since the 90s, there has been increasing interest in the use of classification and regression tree (CART) analysis. The CART methodology was developed in the 80s by Breiman, Friedman, Olshen, Stone in 1984. Unlike the traditional methods, which are cumbersome to use, classification and regression trees are ideally suited for the analysis of complex data. For such data, we require flexible and robust analytical methods, which can deal with nonlinear relationships, high-order interactions, and missing values [17]. In the research of GIS-based groundwater potential mapping using boosted regression tree, classification and

regression tree, and random forest machine learning models in Iran, the CART algorithm gives the relatively high accuracy at around 0.7870 [18]. It is also used to predicting business failure for government officials, stock holders, managers, employees, investors and researchers [19].

CHAID

Chi-square Automatic Interaction Detector (CHAID) is the oldest algorithm among the four. It was invented by Gordon V. Kass in 1980 [21]. The most common application of the CHAID algorithm is found in marketing. It is used to predict customers' responses to campaigns and assess some of the influencing factors to their purchase decisions [22]. CHAID is also a very powerful and widely used tool to conduct customer segmentation and split them into subgroups of similar characteristics [23]. For example, the algorithm can be used for direct marketing purposes by understanding the customer segments and targeting them more effectively.

III. Methods

A. Pseudocode of General Decision Tree Algorithm

Input: an attribute-value dataset D
1: Tree =
2: If D is "pure" OR other stopping criteria met then
3: terminate
4: end if
5: for all attribute a belongs to D do
6: Compute information-theoretic criteria if we spit on a
7: end for
8: a_{best} = Best attribute according to above criteria
9: Tree = create a decision node that tests a_{best}
10:D_v = Induced sub-datasets from D based on a_{best}
11: for all D_v do
12: $Tree_v$ = C4.5(D_v)
13: Attach $Tree_v$ to the corresponding branch of Tree
14: end for
15: return Tree

The idea behind the 4 algorithms are similar. They are all based on the general decision tree, so the project first presents the general idea of decision tree algorithm.

ID3

ID3 constructs a wide decision tree for the given data in a top-down manner, beginning with a set of instances and a feature specification. One feature is tested at each node of the tree based on maximizing information gain and minimizing entropy, and this feature is used to split the data. This process is repeated until the elements of a given sub-tree are homogeneous (i.e. it contains objects belonging to the same category). This is added to the decision tree as a leaf node.

C4.5

C4.5 is a successor of ID3. It builds the decision tree the same way as ID3 does but with several major improvements: auto pruning, deal with continuous variables and missing values, and more reasonable splitting criteria.

CART

The CART decision tree is represented by a set of questions that splits the learning sample into smaller and smaller parts and it asks only yes or no questions. The CART algorithm will search for all possible variables and all possible values in order to find the best split. CART can easily handle both numerical and categorical variables. Usually, the splitting algorithm will isolate outliers in individual nodes.

CHAID

The Chi-square Automatic Interaction Detector (CHAID) uses the chi-square test to evaluate the relationship between the dependent (target) variable and independent variables. It calculates the chi-square value and splits the decision tree on the feature with the highest statistical significance. The algorithm repeats that until it achieves homogeneous observations at the leaf nodes or has no further features to split on.

IV. Theoretical Topic

ID3

ID3 stands for Iterative Dichotomiser 3. It is named such because the algorithm iteratively dichotomizes features into two or more groups at each step. The ID3 algorithm is considered the very first decision tree algorithm developed by Ross Quinlan in 1983 at the University of Sydney. ID3 was first presented in 1975 in a book [6], Machine Learning, vol.1, no.1. ID3 is based on the Concept Learning System (CLS) algorithm which describes the process by which experience allows us to partition objects in the world into classes for the purpose of generalization, discrimination, and inference. [7]

The original set is used as the root node in the ID3 algorithm. The algorithm iterates through every unused attribute in the set and calculates the entropy (information gain) of that attribute on each iteration. It then chooses the attribute with the lowest entropy (or highest information gain). The set is then subdivided by the selected attribute to produce data subsets. The algorithm continues to recurse on each subset, considering only attributes that have never been selected before.

Entropy is a measure in information theory that characterizes the impurity of a random sample of examples. If the target attribute has c different values, the entropy S with respect to this c-wise classification is defined as

$$\text{Entropy}(S) = \sum_{i=1}^{c} -p_i \log_2 p_i \tag{1}$$

where p_i represents the proportion/probability that S belongs to class i. The logarithm is base 2 because entropy

is a measure of expected encoding length measured in bits.

The information gain is based on the decrease in entropy after a dataset is split on an attribute. The information gain, Gain(S, A) of an attribute A, relative to the collection of examples S, is defined as

$$\text{Gain}(S, A) = \text{ Entropy } (S) - \sum_{v \in \text{Values}(A)} \frac{|S_v|}{|S|} \text{Entropy}(S_v) \quad (2)$$

where $Values(A)$ is the set of all possible values for attribute A, and S_v is the subset of S for which the attribute A has value v. ID3 employs this metric to rank attributes and construct a decision tree in which the attribute with the highest information gain among the attributes not yet considered in the tree from the root is located at each node.

The ID3 algorithm uses a greedy search. To be specific, it chooses an attribute to test based on the information gain criterion and then never considers alternative options. In terms of processing time, this makes it a very efficient algorithm. The time complexity of ID3 is $O(dmlog(m))$ where d is the number of the attributes and m is the number of training instances. Besides, the depth of the ID3 tree is $O(log(m))$.

C4.5

The C4.5 decision tree algorithm, a classification, and prediction algorithm was invented in 1979 by JR Quinlan who also proposed the ID3 algorithm for discrete attribute data. In other words, C4.5 is a successor of ID3 with some improvements. Splitting criteria gain ratio is defined as

$$\text{SplitInfo } = -\sum_{j=1}^{n} \frac{|D_j|}{|D|} \log_2 \left(\frac{|D_j|}{|D|} \right) \quad (3)$$

$$\text{Gain Ratio } = \frac{\text{Information Gain}}{\text{SplitInfo}} \quad (4)$$

Parent node is splitting into n partitions and D_j is the number of records in partition j

From the equation above, gain ratio penalizes a large amount of small partitions by dividing Information Gain by SplitInfo. The information gain used in ID3 favors choosing attributes with large distinct values because these attributes result in large information gain. However, the gain ratio eliminates this drawback.

Computation Complexity of C4.5 is O(n) + O(mnlog_2n) + O(log_2n). The first part is the computation complexity on calculating entropy or other information-theoretic indexes, which are bounded by the dataset size n. The computation performed on one input attribute requires O(log_2n) and since all attributes are considered then the total cost for this operation will be O(mnlog_2n). Similarly, to analyze the recursive call of the algorithm on the subset of the training set, the estimated complexity for such operation is O(log_2n) since at each partition, the algorithm considers the instances and their respective target values. Since O(mnlog_2n) is the dominant feature, the computation complexity can be simplified as O(mnlog_2n).

CART

CART constructs the maximum tree using the Gini splitting rule, which is the most broadly used rule. The function to calculate the impurity is

$$Gini = 1 - \sum_{i=1}^{C} (p_\text{i})^2 \quad (5)$$

where pi is the probability of an object being classified to a particular class. Gini impurity is a measurement of the likelihood of incorrect classification of a new instance of a random variable if that new instance were randomly classified according to the distribution of class labels from the dataset. The range of Gini impurity is from 0 to 1. It can be computed by summing the probability of item x being chosen times the probability $(1 - P(x))$ of a mistake in categorizing that item. If the dataset contains only one class, then the Gini impurity is 0.

The CART algorithm has several pruning strategies, and the project focused on the early stopping and minimal cost-complexity pruning. The early stopping method is to set the value for the maximum tree depth, and the algorithm will stop generating the tree once it reaches the value. The minimal cost-complexity pruning sets the value of ccp_alpha parameter to find the best number of nodes being pruned.

CART uses Gini index splitting criteria. Hence, the process of constructing a decision tree using the CART algorithm can be estimated as O(mnlog_2n) where m is the attributes and n is the observations.

CHAID

One of CHAID's biggest advantages over other models is that it is non-parametric since it uses chi-square statistics, which does not care about the distribution and accepts any form of distribution [24]. The main goal of the CHAID algorithm is to evaluate the relationship between the dependent (target) variable and independent variables by calculating the chi-square value using the Chi-square test, defined as

$$\text{Chi-square test } = \sqrt{\frac{(y - y')^2}{y'}} \quad (6)$$

where y is the observed value and y' is the expected value. The algorithm performs the calculation on each observation in a tree node and chooses the feature with the highest value, i.e. highest statistical significance, to split on. The final goal is to achieve homogeneous subgroups with least variation in order to be able to perform classification using the resulted tree [25].

CHAID accepts nominal, continuous, and interval variables [24]. However, a disadvantage of CHAID is that it requires a large number of observations to produce meaningful results [4]. Another disadvantage of CHAID is that it does not handle missing values well as it treats missing values for an independent variable as a single category [26]. This could negatively affect the accuracy of classification as the algorithm could possibly be splitting

on the missing value category early in the tree, causing the tree to proceed with further splits based on inaccurate information.

V. Dataset

A. German Credit Dataset

The German Credit dataset has 1000 observations and 10 independent variables. The target variable "Risk" is a binary variable with two categories good and bad. The explanatory variables are Age (continuous), Sex(categorical), Job(categorical), Housing (categorical), Saving accounts(categorical), Checking account (categorical), Credit amount(continuous), Duration(continuous), Purpose(categorical). One thing to notice is that Credit Amount and Duration has a high correlation coefficient.

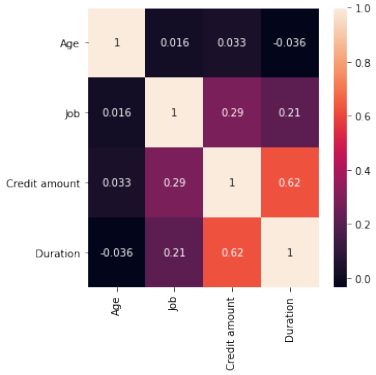

Fig. 1. German Credit Dataset Correlation

B. Red-Wine Quality Dataset

Red-Wine Quality data set consists of all continuous variables. It has 1599 observations and 11 independent variables: fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, and alcohols. The target variable, "quality", is a discrete variable, and it has a range from 1 to 10. This data set is very unbalanced, most of the target value is in 5 and 6. The implementation of the CART algorithm with sklearn library could be easily applied to this data set without encoding since the dependent variables are all continuous.

C. Adult Income Dataset

The adult income data set has 48842 observations and 14 independent variables [7]. These variables are used as explanatory variables to predict the target variable, income. The target variable is binary with two outcomes: income <= 50K or income > 50K. The independent variables are age (numerical), workclass (categorical), fnlwgt (numerical), education (categorical), education-num (numerical), marital-status (categorical), occupation (categorical), relationship (categorical), race (categorical), sex (categorical/binary), capital-gain (numerical), capital-loss (numerical), hours-per-week (numerical), and native-country (categorical).

The data set had 7% of the observations with a missing value across either workclass, occupation, or native-country. These observations were not dropped because the goal is to examine how each algorithm handles missing values.

There is an imbalance in the distribution of the target variable, income, with the label >50K representing 24% of the observations, and <=50K representing the majority of observations with 76%. After grouping the observations by income, it follows our expectations that adults with higher income are older, had more years of education, and worked more hours on average. However, it was surprising to see that, as shown in figure 2, that there was almost no correlation between all the variables in the data set. For example, higher years of education had no strong correlation with capital gain or hours worked per week.

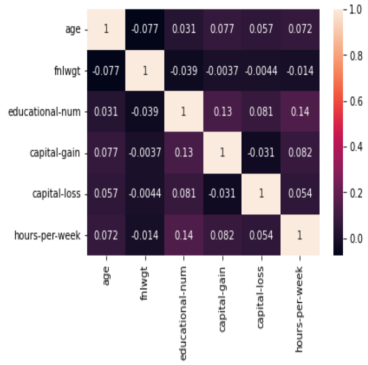

Fig. 2. Adult Income Dataset Correlation

D. Artificial Dataset

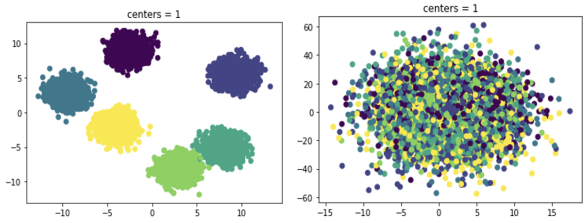

Fig. 3. First Artificial Dataset & Second Artificial Dataset

There are two artificial datasets. The first artificial dataset is made by using the make_blobs function. It has 10,000 observations and 50 features. This dataset is easily classifiable because data are extracted from completely different distributions. The second artificial dataset has the same number of features and observations but is nosier. Function make_classification (n_samples = 10000, n_features = 50, n_informative = 45, n_redundant = 5, n_repeated = 0, n_classes = 6, n_clusters_per_class = 2, flip_y = 0.05, shift = 1, random_state = 42) is used. Five features are redundant which is a linear combination of other variables. Five percent of the data is randomly assigned with class labels.

VI. Data Application

A. ID3

The project first applied ID3 to the German Credit dataset. Since ID3 can't handle any missing values, the first step is to change null values to the "no_inf" category which ID3 would recognize as a separate new category. The project used the Chefboost Python library to build a tree. The Credit Amount variable in this dataset is continuous. Figure 4 indicates that ID3 treats each value in continuous variables as a category, so it builds a wide decision tree. Similar to the Wine Quality Dataset, since all variables in this dataset are continuous, the test accuracy is much lower. Table 1 indicates that there is an overfitting issue for both datasets.

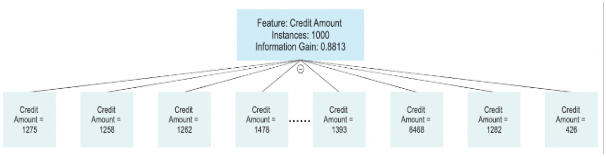

Fig. 4. ID3 Tree Example

In order to solve the overfitting issue, the project tried to bin all continuous variables. If there are fewer categories in a category, the overfitting issue may be improved. The project used quantile binning methods and label encoding with each category. The test accuracy does increase, and the leaf nodes decrease a little for both datasets. The overfitting issue improved but was not solved. The project also applied ID3 on the first artificial dataset which is an easily classifiable dataset. The test accuracy is 100% in this case. The project then applied ID3 on the second artificial dataset. With noises included, the overfitting issue leads to much lower test accuracy of 26.25%.

TABLE I
Test Accuracy of ID3

ID3	German Credit	Red Wine	Adult Income	1st Artificial	2nd Artificial
Without Binning	66.5%	50.62%	/	100%	/
With Binning	67%	57.2%	76.25%	100%	26.25%

In conclusion, one of ID3's biggest disadvantages is that ID3 doesn't handle any missing values or numeric values. This could negatively affect the accuracy if the tree splits on the missing category early in the tree which leads to the further split based on incorrect information. Another disadvantage is that data may be overfitted and no built-in pruning strategy is done. The overfitting issue leads to a high training accuracy but a relatively lower test accuracy which would negatively impact the prediction on other instances. However, the advantage is that ID3 builds a short tree in a relatively small time since it only needs to test enough attributes until all data is classified.

B. C4.5

The German credit dataset contains null values, so the project made another version of the German credit dataset in which null values are replaced by " ? ". After running the dataset with "?", figure 5 shows part of the decision tree. Obviously, the "?" is treated as a special attribute value and the result accuracy is 82.5%. However, applying C4.5 on the dataset with null values, the result accuracy is about 81.2%. Therefore, C4.5 does not simply treat null values as a special attribute value. Indeed, how C4.5 deals with missing values are illustrated above(in the theoretical part).

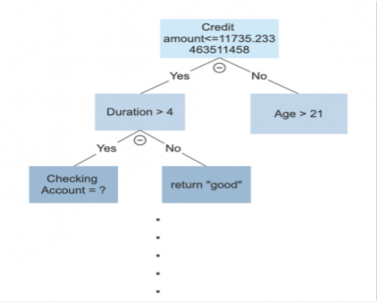

Fig. 5.

Because the Red-Wine Quality dataset is composed of continuous and numerical variables, it is a great dataset to illustrate that C4.5 can handle both discrete and continuous variables. Figure 6 is the generated decision tree using C4.5 on the Red-Wine Quality dataset. Noticeably, the decision tree is binary. The theoretical part before illustrates how C4.5 deals with continuous variables and generates binary decisions. In addition, I binned the Red-Wine Quality dataset the same way as ID3 does. Importantly, the running time of C4.5 is much shorter than ID3, which means the computational complexity of C4.5 is less than ID3. The adult Income dataset is

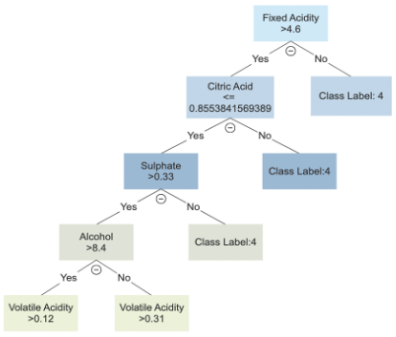

Fig. 6.

another dataset with a combination of continuous and

categorical variables with missing data. Applying both C4.5 and ID3 on the raw dataset without any binning. The resulting accuracy is similar but with different tree sizes. The decision tree generated by ID3 has 5400 leaf nodes. But C4.5 generated decision tree only has 4700 leaf nodes, which corresponds to the theoretical part that C4.5 does a simple auto pruning strategy.

Finally, is C4.5 good at handling noisy data? Two artificial datasets are used for exploration. The graphs show the relationship between number of features used and the accuracy. The left graph of figure 7 is the easily classifiable

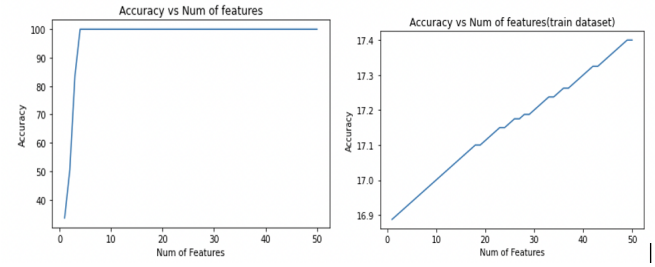

Fig. 7.

dataset, and the right one is the noisy dataset. Similarly, when the number of features increases, the accuracy of both increases. Easily classifiable dataset quickly converge and stabilize at 100%, which shows perfect classification. Although the accuracy increases for the noisy dataset, it increases at an extremely small amount, which is around 0.1%. And the final accuracy it achieved is around 17.5%, which shows that C4.5 does not perform well on the noisy dataset with outliers.

C. CART

German Credit dataset and Adult Income dataset consists of categorical variables. So encoding the variables is necessary before applying the CART algorithm. Sklearn classifier method does not contain encoding method when applying CART, so the project used a ordinal encoder to all categorical variables in these two datasets. Red-wine quality and the artificial datasets contain only numerical variables. Thus, the process of applying to these three datasets is simpler.

The algorithm without pruning strategy works best on the first artificial dataset since it is easily classified. Adult income dataset although is the largest, and it takes the longest to process, the testing accuracy is the highest. It is because the Adult accuracy is very unbalanced. The second artificial dataset gives the worst accuracy because it contains noise and it is the most difficult one to classify.

To use the first pruning accuracy, manually deciding which parameter gives the best results is necessary. When choosing the second pruning accuracy, the project constructed a plot of which ccp_alpha parameter gives the best testing accuracy. The result shows that the two pruning strategies give the same tree depth on German Credit dataser, while the testing accuracy from cost-complexity pruning is slightly higher. The cost-complexity pruning method not only stop the tree earlier, but also gives a parameter that regularizes the tree and decides on which part to prune (Figure 8). This pruning strategy solce the overfitting problems a little, but with decreased training accuracy.

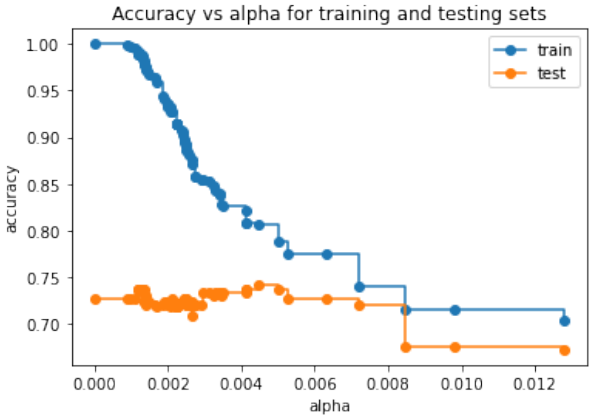

Fig. 8. Accuracy vs. alpha for training and testing sets

TABLE II
TEST ACCURACY OF CART

CART	German Credit	Red Wine	Adult Income	1st Artificial	2nd Artificial
No Pruning					
Tree Depth	15	18	42	6	21
Test Accuracy	72.73%	56.63%	80.13%	100%	32.97%
Early Stopping					
Tree Depth	8	10	8	6	9
Test Accuracy	72.12%	57.95%	85.43%	100%	35.36%
Cost Complexity					
Tree Depth	8	11	17	6	14
Test Accuracy	73.33%	55.49%	85.88%	100%	35.48%

D. CHAID

Table III shows the accuracy score of CHAID's decision tree performance on all the previously presented data sets. The training was done on 80% of the data, whereas testing was performed on the remaining 20%. Since CHAID accepts nominal and continuous variables, no binning for continuous data was needed. CHAID's algorithm does not include pruning as one of its steps, hence, all the results are from fully grown trees.

TABLE III
TEST ACCURACY OF CHAID

CHAID	German Credit	Red Wine	Adult Income	1st Artificial	2nd Artificial
Accuracy Score	62%	49.4%	80.4%	100%	31.55%

VII. RESULTS

ID3

Based on the application in 5 different datasets, the project concluded that ID3 tends to build a short and wide tree which is relatively efficient compared to other decision tree algorithms. However, ID3 has serious overfitting issues. With an easily classifiable dataset, the ID3 could make a good prediction, but it performs poorly with noisy datasets or real-world datasets. In order to achieve better accuracy for noisy or real-world data, the pruning strategy could be done manually. Since ID3 has a relatively shallow tree, pruning could focus on reducing the number of leaf nodes instead of tree depth.

C4.5

The accuracy for German Credit Dataset with null value is 81.2%, but the accuracy for replacing null value with Question mark is 82.5%. Therefore, C4.5 does not treat null value as another special value. For two artificial dataset, the accuracy for easily classifiable one is 100% and accuracy for noisy one is only 17.4%. C4.5 does not work well with noisy data. In addition, C4.5 creates binary decision when dealing with continuous variables. It loops over all attribute value and split it above or below the value, and chooses the one with highest information gain. Finally, C4.5 does a simple and basic auto pruning using reduced error pruning.

CART

CART algorithm is very flexible. It could be easily apply to dataset consists of numerical and categorical variables. It also supports classification and regression on numerical target variables. The pruning method although can not solve overfitting completely, but it improve the issue on most dataset. However, CART is very unstable when there are changes in the dataset because CART splits only by one variable to construct binary tree.

CHAID

As presented in table III, CHAID not perform well on second artificial data sets due to the high noise introduced in the data and the multi-class nature of the target variable. The same observation was consistent when CHAID was tested on the Red wine data set, which also had multi-class target variable. On the other hand, the algorithm performed extremely well on first artificial data set, which is what was expected since it was created to be easily classifiable. As for the two data sets with a binary target variable, German Credit and Adult Income, CHAID performed significantly better on the adult income data set, as well as all other algorithms. The reason might be that the adult data set had 10x more observations to train the tree and less to no correlations among its independent variables.

Interestingly enough, it was noticed that after dropping the observations with missing values in the German Credit and Adult Income data sets, CHAID performed slightly worse with regards to accuracy. It could be that dropping these observations caused loss of the explanatory power of the variables that had no missing values, resulting in a different decision tree and overall accuracy.

VIII. DIVISION OF LABOR

In this project, each of us was in charge of implementing one algorithm and applying the algorithm to all five datasets. Figure 9 shows the division of labor. Xinyu focused on the ID3 algorithm. Zihan focused on the C4.5 algorithm. Ningyuan focused on the CART algorithm and Razan focused on the CHAID algorithm. After finishing applying the algorithms on five datasets, we analyzed and compared the performance of each method together.

Division of Labor	**Algorithm**	**Dataset**	**Tested Datasets**
Xinyu Cai	ID3	German Credit Dataset	All datasets
Zihan Xie	C4.5	Two Artificial Datasets	All datasets
Ningyuan Xiong	CART	Red Wine Quality Dataset	All datasets
Razan Alshehri	CHAID	Adult Income Dataset	All datasets

Fig. 9. Division of Labor

IX. CONCLUSION

Table IV shows a comparison of four methods on five different datasets. Overall, C4.5 gives the highest accuracy on German Credit Dataset. C4.5 has built-in pruning strategy, so it performs good on datasets with both continuous and categorical variables. CART gives the highest accuracy on Red Wine Quality Dataset which is a dataset with all continuous variables. CART performs well on dataset with all continuous variables because CART could deal with continuous variables and has manually pruning strategy built in. CART also performs best on Adult Income Dataset. The accuracy was obtained from optimal pruning with manually tuning the hyper-parameters. Since the 1st Artificial Dataset is easily classifiable, so all methods give a 100% accuracy. The 2nd Artificial Dataset has many noises and randomly assigned classes. CART performs best on this noisy dataset by using the optimal pruning strategy.

Overall, CART has the lowest runtime among the four algorithms. Since CART uses scikit learn package and other three algorithms use Chefboost package, the runtime

are different for these methods. Besides, CART handles continuous variables in a more efficient way, so it has a relatively lower runtime. However, ID3 can't handle continuous variables, so it treats each value in a continuous variable as an individual category. Thus, the runtime is relatively high.

TABLE IV
COMPARISON OF FOUR METHODS

Accuracy Score / Runtime (secs)	German Credit	Red Wine	Adult Income	1st Artificial	2nd Artificial
ID3	67%	57.2%	76.25%	100%	26.25%
ID3 Runtime	39.2	77.65	505	3.89	1115.2
C4.5	**82.5%**	44.59%	77.8%	100%	17.25%
C4.5 Runtime	4.43	71.75	371	5.96	54.35
CART	73.33%	**57.95%**	**85.88%**	100%	**35.97%**
CART Runtime	0.09	0.05	0.25	0.15	0.29
CHAID	62%	49.4%	80.4%	100%	31.55%
CHAID Runtime	9.34	93	305.8	6.34	610.55

REFERENCES

[1] Singh, S., amp; Giri, M. (2014). Comparative Study Id3, Cart And C4.5 Decision Tree Algorithm: A Survey. International Journal of Advanced Information Science and Technology (IJAIST), 3. https://doi.org/10.15693/ijaist/2014.v3i7.47-52

[2] Chary, N., amp; R. (march. 2017). A Survey on Comparative Analysis of Decision Tree Algorithms in Data Mining. International Journal of Advanced Scientific Technologies, Engineering and Management Sciences, 3(1).

[3] Bahety, A. (n.d.). Extension and Evaluation of ID3 – Decision Tree Algorithm. University of Maryland, College Park.

[4] Kirandeep, K., amp; Madan, P. N. (2018). Deployment of id3 decision tree algorithm for placement prediction. International Journal of Trend in Scientific Research and Development, Volume-2(Issue-3), 740-744. doi:10.31142/ijtsrd11073

[5] Danwa, S., Ning, H., amp; Dandan, L. (2009). Construction of Forestry Resource Classification Rule decision tree based on Id3 algorithm. 2009 First International Workshop on Education Technology and Computer Science. doi:10.1109/etcs.2009.730

[6] J.R. QUINLAN, Induction of Decision Trees, 1986, Machine Learning 1:81-106

[7] Concept learning. Concept Learning - an overview | ScienceDirect Topics. (n.d.). Retrieved December 5, 2021, from https://www.sciencedirect.com/topics/psychology/concept-learning.

[8] Xiaohu, W., Lele, W., amp; Nianfeng, L. (2012). An application of decision tree based on ID3. Physics Procedia, 25, 1017–1021. https://doi.org/10.1016/j.phpro.2012.03.193 Magn.Jpn., vol. 2, Aug. 1987, pp. 740–741 [Dig. 9th Annu. Conf. Magnetics Japan, 1982, p. 301].

[9] Yan Ke-wu, Zhu Jin-fu, amp; Sun Qiang. (2007). The application of ID3 algorithm in aviation marketing. 2007 IEEE International Conference on Grey Systems and Intelligent Services. https://doi.org/10.1109/gsis.2007.4443479

[10] Zhang, Q., You, K., amp; Ma, G. (2011). Application of ID3 algorithm in exercise prescription. Lecture Notes in Electrical Engineering, 669–675. https://doi.org/10.1007/978-3-642-21747-0_85

[11] HSSINA, B., MERBOUHA, A., EZZIKOURI, H., ERRITALI, M. (n.d.). A comparative study of Decision Tree Id3 and C4. Retrieved December 5, 2021, from https://saiconference.com/Downloads/SpecialIssueNo10/Paper_3-A_comparative_study_of_decision_tree_ID3_and_C4.5.pdf.

[12] Benjamin Devéze Matthieu Fouquin, DATAMINING C4.5 – DBSCAN, PROMOTION 2005, SCIA Ecole pour l'informatique et techniques avancées.

[13] Turney , P. D. (n.d.). Cost-sensitive classification: Empirical ... - arxiv.org. Retrieved December 5, 2021, from https://arxiv.org/pdf/cs/9503102.pdf.

[14] Ngoc, P.V., Ngoc, C.V.T., Ngoc, T.V.T. et al. A C4.5 algorithm for english emotional classification. Evolving Systems 10, 425–451 (2019). https://doi.org/10.1007/s12530-017-9180-1

[15] Darmawan , E. (n.d.). C4.5 algorithm application for prediction of self ... - core. Retrieved December 5, 2021, from https://core.ac.uk/download/pdf/295600592.pdf.

[16] Lewis, Roger. (2000). An Introduction to Classification and Regression Tree (CART) Analysis.

[17] De'ath, G., amp; Fabricius, K. E. (2000). Classification and regression trees: A powerful yet simple technique for ecological data analysis. Ecology, 81(11), 3178–3192. https://doi.org/10.1890/0012-9658(2000)081[3178:cartap]2.0.co;2

[18] Li, H., Sun, J., amp; Wu, J. (2010). Predicting business failure using classification and regression tree: An empirical comparison with popular classical statistical methods and top classification mining methods. Expert Systems with Applications, 37(8), 5895–5904. https://doi.org/10.1016/j.eswa.2010.02.016

[19] Naghibi, S. A., Pourghasemi, H. R., amp; Dixon, B. (2015). GIS-based groundwater potential mapping using boosted regression tree, classification and regression tree, and random forest machine learning models in Iran. Environmental Monitoring and Assessment, 188(1). https://doi.org/10.1007/s10661-015-5049-6

[20] "Adult income dataset," Kaggle, 06-Oct-2016. [Online]. Available: https://www.kaggle.com/wenruliu/adult-income-dataset?select=adult.csv. [Accessed: 05-Dec-2021].

[21] G. V. Kass, "An exploratory technique for investigating large quantities of categorical data," Applied Statistics, vol. 29, no. 2, p. 119, 1980.

[22] Y. Susanti, E. Zukhronah, H. Pratiwi, Respatiwulan, and H. Sri Sulistijowati, "Analysis of Chi-Square Automatic Interaction Detection (CHAID) and classification and regression tree (CRT) for classification of corn production," Journal of Physics: Conference Series, vol. 909, p. 012041, 2017.

[23] B2B International, "Chaid Analysis: Decision tree analysis," B2B International, 24-Nov-2017. [Online]. Available: https://www.b2binternational.com/research/methods/statistical-techniques/chaid-analysis/. [Accessed: 05-Dec-2021].

[24] F. M. Díaz-Pérez, C. G. García-González, and A. Fyall, "The use of the CHAID algorithm for Determining Tourism Segmentation: A purposeful outcome," Heliyon, vol. 6, no. 7, 2020.

[25] A. Abbas, M. A. Ullah, and A. Waheed, "Body weight prediction using different data mining algorithms in Thalli Sheep: A comparative study," Veterinary World, pp. 2332–2338, 2021.

[26] "Missing Values in Tree Models," Missing values in Tree Models. [Online]. Available: https://www.ibm.com/docs/en/spss-statistics/24.0.0?topic=option-missing-values-in-tree-models. [Accessed: 05-Dec-2021].