Finding the Causes of Drug Addiction Tendency among Bangladeshi Students using Naïve Bayes and Decision Tree

Reja, T.
BSc CSE
American International University, Bangladesh
Dhaka, Bangladesh
19-40151-1@student.aiub.edu

Hossain, S. M. A
BSc CSE
American International University, Bangladesh
Dhaka, Bangladesh
21-44421-1@student.aiub.edu

Shayoni, R. K.
BSc CSE
American International University, Bangladesh
Dhaka, Bangladesh
20-41922-1@student.aiub.edu

Jisha, S. S.
BSc CSE
American International University, Bangladesh
Dhaka, Bangladesh
20-42526-1@student.aiub.com

Abstract— **Drug addiction is a serious issue now-a-days in our society. As students are the future of a nation, so this issue of drug addiction among the student is very alarming. The purpose of this research is to identify the reasons among the students to get drug addicted and also the future prediction of a student regarding the tendency to get drug addicted by building models using two machine learning algorithms that are Naïve Bayes and Decision Tree. Naïve Bayes is a statistical method to get the prominent classification result and Decision tree is a recursive method which helps to predict the class of an instance effectively. To conduct this research, supervised and nominal data were collected from Kaggle having 211 instances, 26 attributes and 4 classes. Final dataset was generated by manually preparing raw data by eliminating outliers, discarding repeated attributes, editing values etc., and creating Attribute-Relation File Format of the dataset. Both the models were built by using WEKA version 3.8.6. Data analysis helped to know the optimum algorithm for our chosen dataset by identifying the significant reasons of drug addiction among the students. To compare the performance of the Naïve Bayes and Decision tree algorithms, different variant of statistical error and performance evaluation metrics were chosen e.g.- Kappa, Mean Absolute Error, Relative Absolute Error etc. The comparison was done between unprocessed and pre-processed dataset. Also, the comparison was done between different ratios of training and testing dataset respectively 80:20, 50:50 and 95:5. So, these models would help us to know the more compatible algorithm which has more correctly classified the data. Some difficulties were faces as the accuracy our result was not very high. A dataset with large number of instances would be compatible to analyze the performance metrices more accurately.**

Keywords—instances, attributes, Naïve Bayes, Decision Tree, preprocessing, compare, accuracy, improved, result.

I. Introduction

Being a developing country Bangladesh has to combat with immense problems. Among them "Drug addiction" is one of the most controversial problems, which is turning into a national problem from a social problem. Especially, when it comes to the younger generation, then the price for a nation is so high. The students are the core contributor to the development of a nation. Drug Addiction is alarming for our country as it influences both economic and social aspects. In our study, we tried to find out the causes why Bangladeshi students have a tendency to drug addiction. In the future, we will try to reduce this issue using our prediction based on this study. We have identified some factors of drug addiction tendency, such as experimental curiosity, peer pressure, poor socioeconomic condition etc. And there could be more factors. We are less aware of the side effect of taking drugs. Nowadays drug addiction is turning into the culture. Therefore, as a solution, we will apply Naïve Bayes and Decision Tree algorithm to the dataset, collected from Kaggle. We will compare these two models and also evaluate them on basic of their accuracy and other performance metrics.

1. Model 1-Naïve Bayes

Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem. It's a statistical classifier that performs probabilistic prediction, such as it predicts class membership probabilities. Naive Bayes learners and classifiers can be extremely fast compared to more sophisticated methods. This model is easy to implement but there has an assumption of class conditional independence and therefore happens loss of accuracy in the result. Practically, this model's dependencies exist among variables.

2. Model 2-Decision Tree

Decision tree algorithm is a data-mining induction techniques that recursively partitions a dataset of records using depth-first greedy approach or breadth-first approach until all the data items belong to a particular class. Decision trees are an approach used in supervised machine learning, a technique that uses labeled input and output datasets to train models. The approach is used mainly to solve classification problems, which is the use of a model to categorize or classify an object. This is mainly a predictive model based on a branching series of Boolean tests, that are less complex than a one-stage classifier. It uses particular facts to make more generalized results.

Applying the decision tree to the dataset, we have to make a list of discrete attributes first that we can measure. Then we need to choose a target attribute that we want to predict. And finally, we will create an experience table that lists what we have seen in the past. There could be more than one tree that fits the same data.

Decision Tree is comprehensive, specific, easy to use and versatile.

II. Methodology

A. *Flowchart of the Proposed Solution*

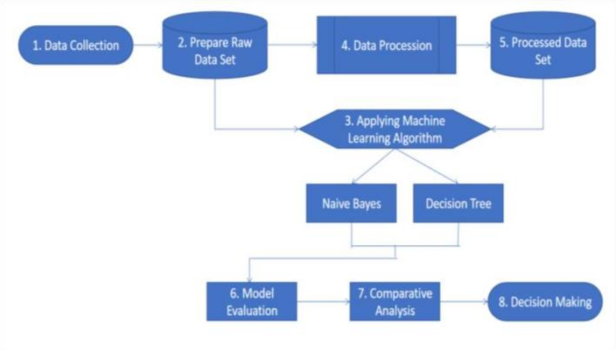


Fig: Flowchart of the solution

After the collection of the dataset, the raw dataset will be prepared manually and then the two algorithms: Naïve Bayes and Decision Tree will be applied without preprocessing & evaluation. After the dataset is preprocessed, both the algorithms will be applied again after preprocessing & evaluation. Then, these two models will be evaluated. Based on the comparative analysis, our final decision will be made.

B. *Methods Used*

In this research, we have used secondary dataset collected from Kaggle. Initial dataset is a supervised dataset, has a nominal data type. Supervised learning is a machine learning approach that's defined by its use of labeled datasets. These datasets are designed to train or supervise algorithms into classifying data or predicting outcomes accurately. Supervised learning maps an input to an output based on example input-output pairs. Using labeled inputs and outputs, the model can measure its accuracy and learn over time. It infers a function from labeled training data consisting of a set of training examples. Nominal data are used to label variables without any quantitative value.

In our initial dataset, some problems were detected like, outliers, similar attributes and missing values. Initial dataset has only one ‘above 48’ value in age attributes, that is the outlier. Similar attributes like ‘conflict with the law’ and ‘case in the court’ was there in the initial dataset.

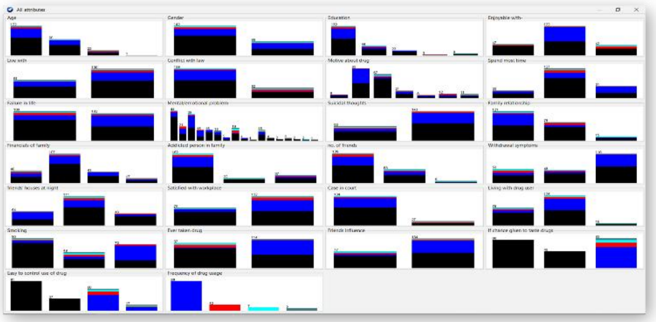
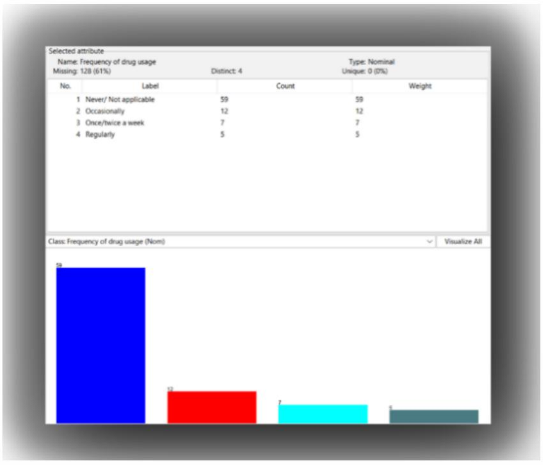


Fig: Initial Dataset Problems

Here, the black boxes represent the problems in the dataset.



Selected attribute
Name: Frequency of drug usage
Missing: 128 (61%)
Distinct: 4
Type: Nominal
Unique: 0 (0%)

No.	Label	Count	Weight
1	Never/ Not applicable	59	59
2	Occasionally	12	12
3	Once/twice a week	7	7
4	Regularly	5	5

Class: Frequency of drug usage (Nom)

Fig: Visualization of Initial Dataset

After detecting the issues, the raw data is prepared manually by discarding outliers, editing values so that it is easily readable by models, discarding repeated attributes, making sure that there is no missing values. Finished dataset is got, that has 211 Instances and 22 Attributes, which was previously 211 Instances and 26 Attributes in initial dataset. Instance is the single row of data and Attribute is a characteristic or feature of dataset, that is measured for each observation and can vary from one observation to another. After that, Attribute-Relation File Format flies are created for model creation.

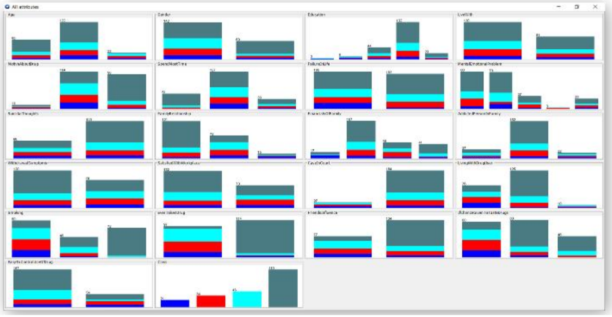
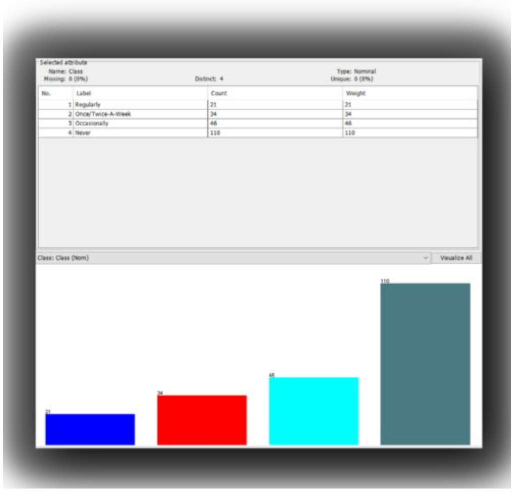


Fig: Final Dataset Representation

After manual preparing, there is no black boxes or outliers in finished dataset.

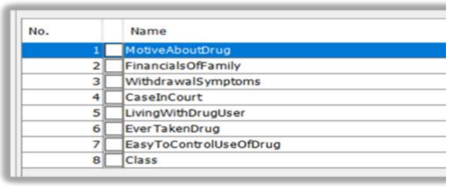


Selected attribute
Name: Class   Missing: 0 (0%)   Distinct: 4   Type: Nominal   Unique: 0 (0%)

No.	Label	Count	Weight
1	Regularly	21	21
2	Once/Twice-A-Week	34	34
3	Occasionally	46	46
4	Never	110	110

Fig: Visualization of Final Dataset

To create predictive models in tool, Weka version 3.8.6 is used. For preprocessed the data, Attribute Selection Method is applied. It also allows various search & evaluating methods to be combined. After preprocessing, Attribute No is 8.



No.	Name
1	MotiveAboutDrug
2	FinancialsOfFamily
3	WithdrawalSymptoms
4	CaseInCourt
5	LivingWithDrugUser
6	EverTakenDrug
7	EasyToControlUseOfDrug
8	Class

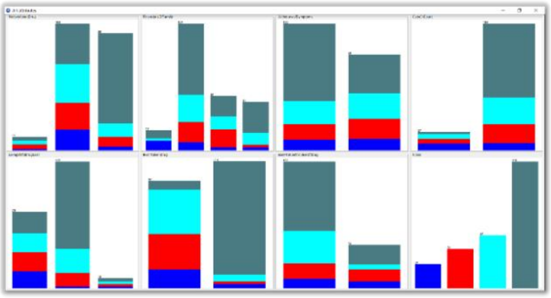


Fig: Overall Representation

III. LITERATURE REVIEW

State-of-the-Arts

In this study, information about future prediction model were explored to build a basic model to predict the future regarding the drug addiction tendency among Bangladeshi Students.

As for some of the basis of future prediction model, namely:

Research conducted by Sayali D. Jadhav and H. P. Channe with the title "Comparative Study of K-NN, Naive Bayes and Decision Tree Classification Techniques". The main objective of this research is to predict student graduation using Naïve Bayes Classifier with a data mining approach. Naïve Bayes provides accurate prediction results with a minimum error rated compared to all other data mining components. It was implemented on 3 different sizes of dataset to compare the performance of the model built using WEKA. The aim of this research was equally aiming to predict the future result using Naïve Bayes and Decision Tree. While, they also used K-NN to make a dynamic comparison between the three methods. [1]

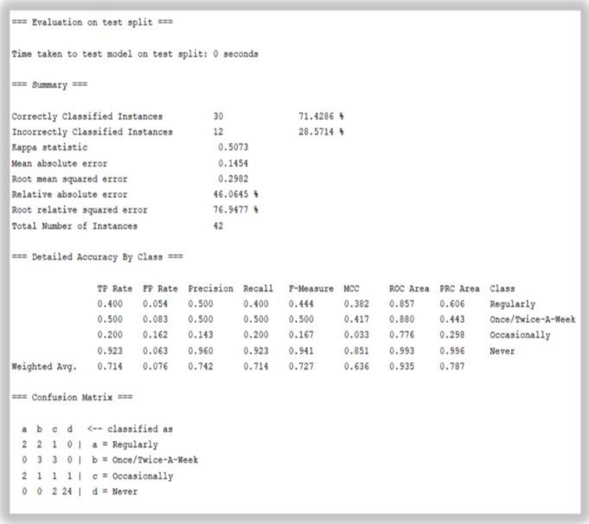
Research conducted by Md Ariful Arif, Saiful Islam Sany, Farah Sharmin, Md. Sadekur Rahman and Md. Tarek Habib under the heading “Prediction of addiction to drugs and alcohol using machine learning: A case study on Bangladeshi population”. This paper explores the technique to process a dataset and forecast the risk of becoming addicted to drugs by comparing the performance of nine different algorithms of machine learning. Our work of shows comparison of two machine learning algorithms and the data processing was done using WEKA framework to predict the drug addiction tendency of Bangladeshi students. [2]

Research with the title “Student’s performance prediction using KNN and Naïve Bayesian”, conducted by Ihsan A. Abu Amra and Ashraf Y. A. Maghari. This paper proposes a student performance prediction model by applying two classification algorithms: KNN and Naïve Bayes on educational data set of secondary schools, collected from the ministry of education in Gaza Strip for 2015 year. The main objective of such classification may help the ministry of education to improve the performance due to early prediction of student performance. Parallelly with our research this research has the same motive except for our research was based on Naïve Bayes and Decision and was applied for a different future prediction model. [3]

IV. FINDINGS AND ANALYSIS

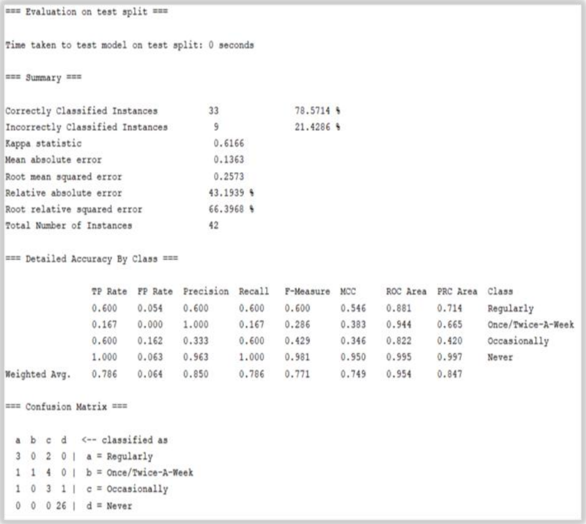
A. Training Set – 80% & Testing Set – 20%

Before preprocessing, Naïve Bayes takes 0 sec to build the model. It has correctly classified 30 out of 42 instances. Accuracy is 71.4286%. Kappa statistics is 0.5073.



```
=== Evaluation on test split ===

Time taken to test model on test split: 0 seconds

=== Summary ===

Correctly Classified Instances          30               71.4286 %
Incorrectly Classified Instances        12               28.5714 %
Kappa statistic                          0.5073
Mean absolute error                      0.1454
Root mean squared error                  0.2982
Relative absolute error                 46.0645 %
Root relative squared error             76.9477 %
Total Number of Instances               42

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.400    0.054    0.500      0.400    0.444      0.382    0.857     0.606     Regularly
                 0.500    0.083    0.500      0.500    0.500      0.417    0.880     0.443     Once/Twice-A-Week
                 0.200    0.162    0.143      0.200    0.167      0.033    0.776     0.298     Occasionally
                 0.923    0.063    0.960      0.923    0.941      0.851    0.993     0.996     Never
Weighted Avg.    0.714    0.076    0.742      0.714    0.727      0.636    0.935     0.787

=== Confusion Matrix ===

  a  b  c  d   <-- classified as
  2  2  1  0 |  a = Regularly
  0  3  3  0 |  b = Once/Twice-A-Week
  2  1  1  1 |  c = Occasionally
  0  0  2 24 |  d = Never
```

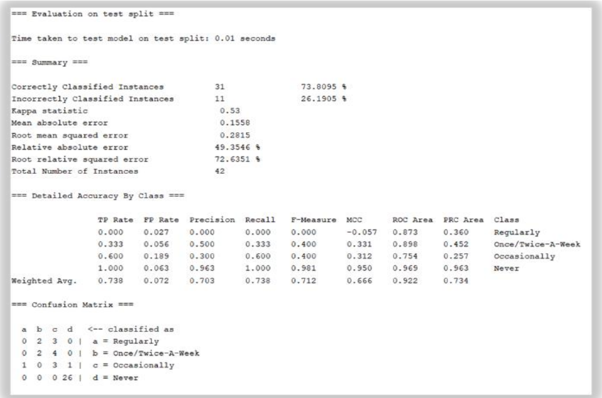
After preprocessing, Naïve Bayes Classifier Algorithm will be applied again. It takes 0 sec to build the model. It has correctly classified 33 out of 42 instances. Accuracy is 78.5714%. Kappa statistics is 0.6166.



```
=== Evaluation on test split ===

Time taken to test model on test split: 0 seconds

=== Summary ===

Correctly Classified Instances          33               78.5714 %
Incorrectly Classified Instances         9               21.4286 %
Kappa statistic                          0.6166
Mean absolute error                      0.1363
Root mean squared error                  0.2573
Relative absolute error                 43.1939 %
Root relative squared error             66.3968 %
Total Number of Instances               42

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.600    0.054    0.600      0.600    0.600      0.546    0.881     0.714     Regularly
                 0.167    0.000    1.000      0.167    0.286      0.383    0.944     0.665     Once/Twice-A-Week
                 0.600    0.162    0.333      0.600    0.429      0.346    0.822     0.420     Occasionally
                 1.000    0.063    0.963      1.000    0.981      0.950    0.995     0.997     Never
Weighted Avg.    0.786    0.064    0.850      0.786    0.771      0.749    0.954     0.847

=== Confusion Matrix ===

  a  b  c  d   <-- classified as
  3  0  2  0 |  a = Regularly
  1  1  4  0 |  b = Once/Twice-A-Week
  1  0  3  1 |  c = Occasionally
  0  0  0 26 |  d = Never
```

It is noticed that the time has no change, before and after preprocessing. Accuracy has increased 7.1428% and Kappa statistics has increased 0.1093. There are four errors. Mean Absolute Error is the measure of how far the predictions are from the actual output. Root Mean Squared Error penalizes the error terms more. Relative Absolute Error measures the performance of a predictive model. Root Relative Squared Error is a a basic metric that gives a first indication of how well model performance. The more the values will closer to 0, the better the result is. Before preprocessing the values are 0.1454, 0.2982, 46.0645%, 76.9477% that decreased after preprocessing into 0.1363, 0.2573, 43.1939%, 66.3968%. That indicates a better and improved result. Recall indicates how many times the model was able to detect a specific category and Precision indicates how good the model is at predicting a specific category. Values of recall and precision is closer to 1 is better. Recall increased into 0.786 from 0.714 and precision increased into 0.850 from 0.742. F-measure, the harmonic mean of precision and recall increased 0.771 from 0.727. Overall, Naïve Bayes model gives a improved result in all evaluation metrics.

On the other hand, before preprocessing, Decision Tree takes 0.1 sec to build the model. No of leaf nodes is 34 and size of tree is 55, is determined. It has correctly classified 31 out of 42 instances. Accuracy is 73.8095%. Kappa statistics is 0.53.



```
=== Evaluation on test split ===

Time taken to test model on test split: 0.01 seconds

=== Summary ===

Correctly Classified Instances          31               73.8095 %
Incorrectly Classified Instances        11               26.1905 %
Kappa statistic                          0.53
Mean absolute error                      0.1558
Root mean squared error                  0.2815
Relative absolute error                 49.3546 %
Root relative squared error             72.6351 %
Total Number of Instances               42

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.000    0.027    0.000      0.000    0.000      -0.057   0.873     0.360     Regularly
                 0.333    0.056    0.500      0.333    0.400      0.331    0.898     0.452     Once/Twice-A-Week
                 0.600    0.189    0.300      0.600    0.400      0.312    0.754     0.257     Occasionally
                 1.000    0.063    0.963      1.000    0.981      0.950    0.969     0.963     Never
Weighted Avg.    0.738    0.072    0.703      0.738    0.712      0.666    0.922     0.734

=== Confusion Matrix ===

  a  b  c  d   <-- classified as
  0  2  3  0 |  a = Regularly
  0  2  4  0 |  b = Once/Twice-A-Week
  1  0  3  1 |  c = Occasionally
  0  0  0 26 |  d = Never
```

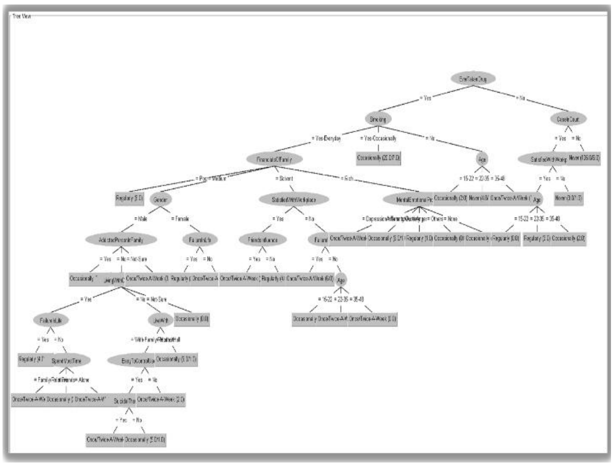
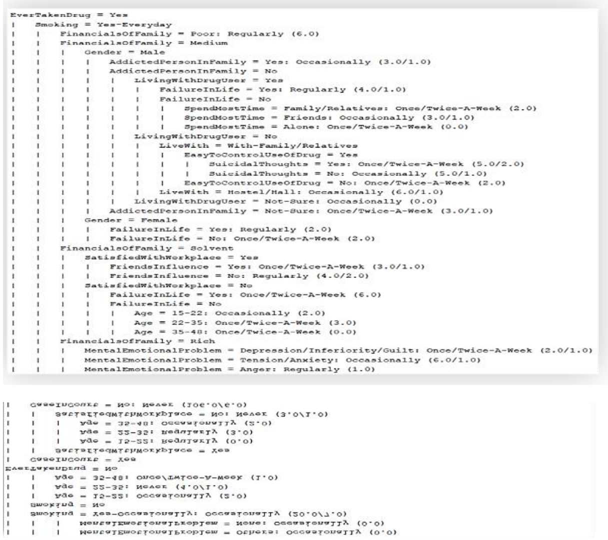


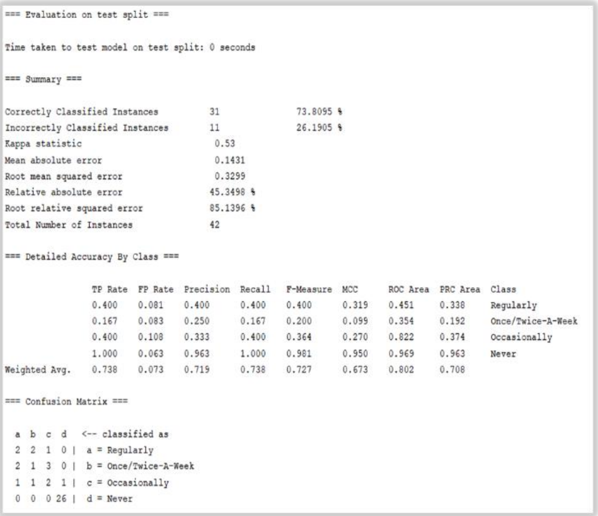
Fig: Decision Tree created by model (before preprocessing)



```
EverTakenDrug = Yes
|   Smoking = Yes-Everyday
|   |   FinancialsOfFamily = Poor: Regularly (6.0)
|   |   FinancialsOfFamily = Medium
|   |   |   Gender = Male
|   |   |   |   AddictedPersonInFamily = Yes: Occasionally (3.0/1.0)
|   |   |   |   AddictedPersonInFamily = No
|   |   |   |   |   LivingWithDrugUser = Yes
|   |   |   |   |   |   FailureInLife = Yes: Regularly (4.0/1.0)
|   |   |   |   |   |   FailureInLife = No
|   |   |   |   |   |   |   SpendMostTime = Family/Relatives: Once/Twice-A-Week (2.0)
|   |   |   |   |   |   |   SpendMostTime = Friends: Occasionally (3.0/1.0)
|   |   |   |   |   |   |   SpendMostTime = Alone: Once/Twice-A-Week (0.0)
|   |   |   |   |   LivingWithDrugUser = No
|   |   |   |   |   |   LiveWith = With-Family/Relatives
|   |   |   |   |   |   |   EasyToControlUseOfDrug = Yes
|   |   |   |   |   |   |   |   SuicidalThoughts = Yes: Once/Twice-A-Week (5.0/2.0)
|   |   |   |   |   |   |   |   SuicidalThoughts = No: Occasionally (5.0/1.0)
|   |   |   |   |   |   |   EasyToControlUseOfDrug = No: Once/Twice-A-Week (2.0)
|   |   |   |   |   |   LiveWith = Hostel/Hall: Occasionally (6.0/1.0)
|   |   |   |   |   LivingWithDrugUser = Not-Sure: Occasionally (0.0)
|   |   |   |   AddictedPersonInFamily = Not-Sure: Once/Twice-A-Week (3.0/1.0)
|   |   |   Gender = Female
|   |   |   |   FailureInLife = Yes: Regularly (2.0)
|   |   |   |   FailureInLife = No: Once/Twice-A-Week (2.0)
|   |   FinancialsOfFamily = Solvent
|   |   |   SatisfiedWithWorkplace = Yes
|   |   |   |   FriendsInfluence = Yes: Once/Twice-A-Week (3.0/1.0)
|   |   |   |   FriendsInfluence = No: Regularly (4.0/2.0)
|   |   |   SatisfiedWithWorkplace = No
|   |   |   |   FailureInLife = Yes: Once/Twice-A-Week (6.0)
|   |   |   |   FailureInLife = No
|   |   |   |   |   Age = 15-22: Occasionally (2.0)
|   |   |   |   |   Age = 22-35: Once/Twice-A-Week (3.0)
|   |   |   |   |   Age = 35-48: Once/Twice-A-Week (0.0)
|   |   FinancialsOfFamily = Rich
|   |   |   MentalEmotionalProblem = Depression/Inferiority/Guilt: Once/Twice-A-Week (2.0/1.0)
|   |   |   MentalEmotionalProblem = Tension/Anxiety: Occasionally (6.0/1.0)
|   |   |   MentalEmotionalProblem = Anger: Regularly (1.0)
```

Fig: Alternative Representation of Decision Tree

After preprocessing, Decision Tree Algorithm will be applied again. It takes 0 sec to build the model this time. No of leaf nodes is reduced to 13 and size of tree is reduced to 20. It has correctly classified 31 out of 42 instances. Accuracy is 73.8095%. Kappa statistics is 0.53. Tree has been pruned 348 times.



```
=== Evaluation on test split ===

Time taken to test model on test split: 0 seconds

=== Summary ===

Correctly Classified Instances          31               73.8095 %
Incorrectly Classified Instances        11               26.1905 %
Kappa statistic                          0.53
Mean absolute error                      0.1431
Root mean squared error                  0.3299
Relative absolute error                 45.3498 %
Root relative squared error             85.1396 %
Total Number of Instances               42

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.400    0.081    0.400      0.400    0.400      0.319    0.451     0.338     Regularly
                 0.167    0.083    0.250      0.167    0.200      0.099    0.354     0.192     Once/Twice-A-Week
                 0.400    0.108    0.333      0.400    0.364      0.270    0.822     0.374     Occasionally
                 1.000    0.063    0.963      1.000    0.981      0.950    0.969     0.963     Never
Weighted Avg.    0.738    0.073    0.719      0.738    0.727      0.673    0.802     0.708

=== Confusion Matrix ===

  a  b  c  d   <-- classified as
  2  2  1  0 |  a = Regularly
  2  1  3  0 |  b = Once/Twice-A-Week
  1  1  2  1 |  c = Occasionally
  0  0  0 26 |  d = Never
```

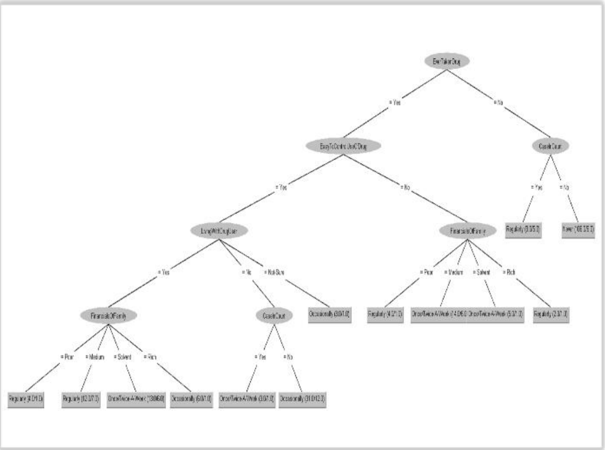
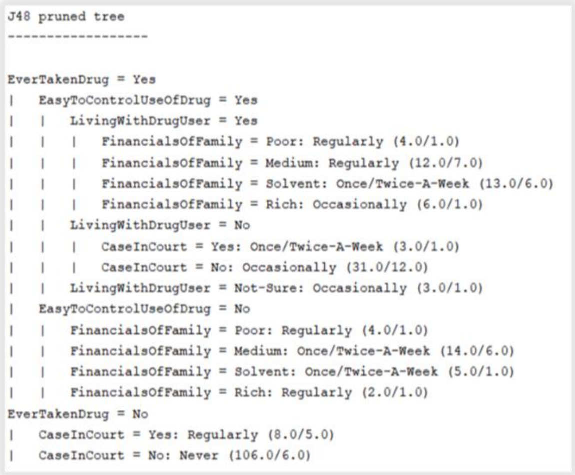


Fig: Decision Tree created by model (after preprocessing)



```
J48 pruned tree
------------------

EverTakenDrug = Yes
|   EasyToControlUseOfDrug = Yes
|   |   LivingWithDrugUser = Yes
|   |   |   FinancialsOfFamily = Poor: Regularly (4.0/1.0)
|   |   |   FinancialsOfFamily = Medium: Regularly (12.0/7.0)
|   |   |   FinancialsOfFamily = Solvent: Once/Twice-A-Week (13.0/6.0)
|   |   |   FinancialsOfFamily = Rich: Occasionally (6.0/1.0)
|   |   LivingWithDrugUser = No
|   |   |   CaseInCourt = Yes: Once/Twice-A-Week (3.0/1.0)
|   |   |   CaseInCourt = No: Occasionally (31.0/12.0)
|   |   LivingWithDrugUser = Not-Sure: Occasionally (3.0/1.0)
|   EasyToControlUseOfDrug = No
|   |   FinancialsOfFamily = Poor: Regularly (4.0/1.0)
|   |   FinancialsOfFamily = Medium: Once/Twice-A-Week (14.0/6.0)
|   |   FinancialsOfFamily = Solvent: Once/Twice-A-Week (5.0/1.0)
|   |   FinancialsOfFamily = Rich: Regularly (2.0/1.0)
EverTakenDrug = No
|   CaseInCourt = Yes: Regularly (8.0/5.0)
|   CaseInCourt = No: Never (106.0/6.0)
```

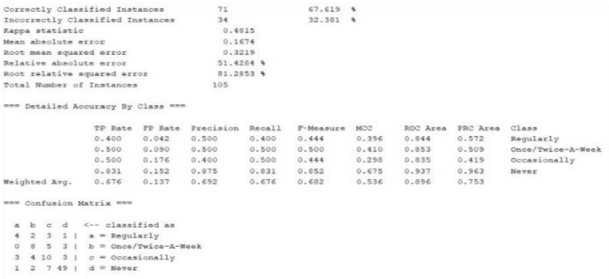
Fig: Alternative Representation of Decision Tree

It is noticed that the time has been reduced after preprocessing. Accuracy and Kappa statistics are same as before. Kappa statistics basically tells how much better classifier is performing. Before preprocessing, the values of Mean Absolute Error and Relative Absolute Error are 0.1558 and 49.3546% that decreased after preprocessing into 0.1431 and 45.3498%, that indicates improved result. But the values of Root Mean Squared Error and Root Relative Squared Error are 0.2815 and 72.6351% that increased after preprocessing into 0.3299 and 85.1396%, which indicates result isn't improved. Recall value is 0.738, that is same as before. But precision value increased into 0.719 from 0.703. Besides, F-measure increased 0.727 from 0.712. Here, values of precision and F-measure indicates improved result. Overall, Decision Tree model gives partially improved result.

Therefore, comparing these two models, in case of 80%-20% data, Naïve Bayes gives the better performance.

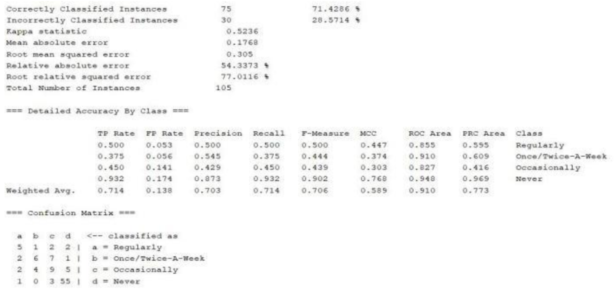
B. Training Set – 50% & Testing Set – 50%

Before preprocessing, Naïve Bayes has correctly classified 71 out of 105 instances. Accuracy is 67.619%.



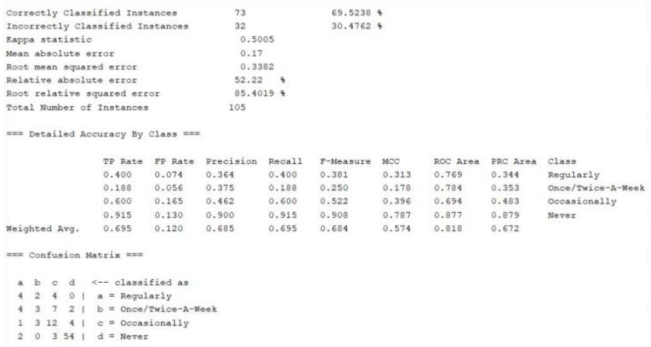
```
Correctly Classified Instances          71               67.619  %
Incorrectly Classified Instances        34               32.381  %
Kappa statistic                          0.4815
Mean absolute error                      0.1674
Root mean squared error                  0.3219
Relative absolute error                 51.4284 %
Root relative squared error             81.2853 %
Total Number of Instances              105

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.400    0.042    0.500      0.400    0.444      0.396    0.844     0.572     Regularly
                 0.500    0.090    0.500      0.500    0.500      0.410    0.853     0.509     Once/Twice-A-Week
                 0.500    0.176    0.400      0.500    0.444      0.298    0.835     0.419     Occasionally
                 0.831    0.152    0.875      0.831    0.852      0.675    0.937     0.963     Never
Weighted Avg.    0.676    0.137    0.692      0.676    0.682      0.536    0.896     0.753

=== Confusion Matrix ===

  a  b  c  d   <-- classified as
  4  2  3  1 |  a = Regularly
  0  8  5  3 |  b = Once/Twice-A-Week
  3  4 10  3 |  c = Occasionally
  1  2  7 49 |  d = Never
```

After preprocessing, Naïve Bayes classifier algorithm has correctly classified 75 out of 105 instances. Accuracy is 71.4286%.



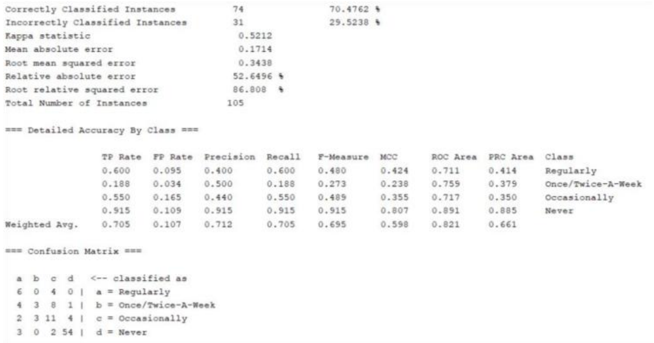
```
Correctly Classified Instances          75               71.4286 %
Incorrectly Classified Instances        30               28.5714 %
Kappa statistic                          0.5236
Mean absolute error                      0.1768
Root mean squared error                  0.305
Relative absolute error                 54.3373 %
Root relative squared error             77.0116 %
Total Number of Instances              105

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.500    0.053    0.500      0.500    0.500      0.447    0.855     0.595     Regularly
                 0.375    0.056    0.545      0.375    0.444      0.374    0.910     0.609     Once/Twice-A-Week
                 0.450    0.141    0.429      0.450    0.439      0.303    0.827     0.416     Occasionally
                 0.932    0.174    0.873      0.932    0.902      0.768    0.948     0.969     Never
Weighted Avg.    0.714    0.138    0.703      0.714    0.706      0.589    0.910     0.773

=== Confusion Matrix ===

  a  b  c  d   <-- classified as
  5  1  2  2 |  a = Regularly
  2  6  7  1 |  b = Once/Twice-A-Week
  2  4  9  5 |  c = Occasionally
  1  0  3 55 |  d = Never
```

Before preprocessing, Decision Tree has correctly classified 73 out of 105 instances. Accuracy is 69.5238%.



```
Correctly Classified Instances          73               69.5238 %
Incorrectly Classified Instances        32               30.4762 %
Kappa statistic                          0.5005
Mean absolute error                      0.17
Root mean squared error                  0.3382
Relative absolute error                 52.22   %
Root relative squared error             85.4019 %
Total Number of Instances              105

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.400    0.074    0.364      0.400    0.381      0.313    0.769     0.344     Regularly
                 0.188    0.056    0.375      0.188    0.250      0.178    0.784     0.353     Once/Twice-A-Week
                 0.600    0.165    0.462      0.600    0.522      0.396    0.694     0.483     Occasionally
                 0.915    0.130    0.900      0.915    0.908      0.787    0.877     0.879     Never
Weighted Avg.    0.695    0.120    0.685      0.695    0.684      0.574    0.818     0.672

=== Confusion Matrix ===

  a  b  c  d   <-- classified as
  4  2  4  0 |  a = Regularly
  4  3  7  2 |  b = Once/Twice-A-Week
  1  3 12  4 |  c = Occasionally
  2  0  3 54 |  d = Never
```

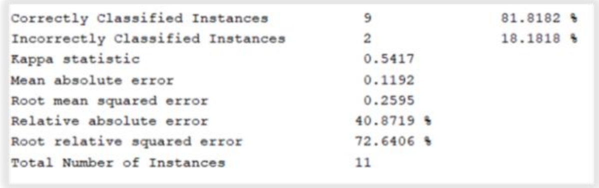
After preprocessing, Decision Tree model classifier has correctly classified 74 out of 105 instances. Accuracy is 70.4762%.



```
Correctly Classified Instances          74               70.4762 %
Incorrectly Classified Instances        31               29.5238 %
Kappa statistic                          0.5212
Mean absolute error                      0.1714
Root mean squared error                  0.3438
Relative absolute error                 52.6496 %
Root relative squared error             86.808  %
Total Number of Instances              105

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.600    0.095    0.400      0.600    0.480      0.424    0.711     0.414     Regularly
                 0.188    0.034    0.500      0.188    0.273      0.238    0.759     0.379     Once/Twice-A-Week
                 0.550    0.165    0.440      0.550    0.489      0.355    0.717     0.350     Occasionally
                 0.915    0.109    0.915      0.915    0.915      0.807    0.891     0.885     Never
Weighted Avg.    0.705    0.107    0.712      0.705    0.695      0.598    0.821     0.661

=== Confusion Matrix ===

  a  b  c  d   <-- classified as
  6  0  4  0 |  a = Regularly
  4  3  8  1 |  b = Once/Twice-A-Week
  2  3 11  4 |  c = Occasionally
  3  0  2 54 |  d = Never
```

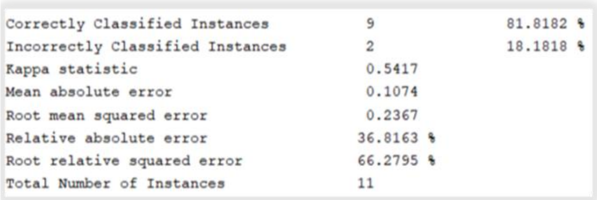
C. Training Set – 95% & Testing Set – 5%

Before preprocessing, Naïve Bayes has correctly classified 9 out of 11 instances. Accuracy is 81.8182% and Kappa statistics is 0.5417.



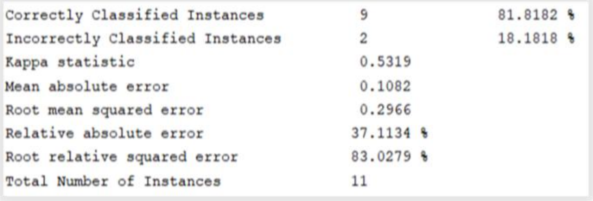
```
Correctly Classified Instances           9               81.8182 %
Incorrectly Classified Instances         2               18.1818 %
Kappa statistic                          0.5417
Mean absolute error                      0.1192
Root mean squared error                  0.2595
Relative absolute error                 40.8719 %
Root relative squared error             72.6406 %
Total Number of Instances               11
```

After preprocessing, Naïve Bayes classifier algorithm has correctly classified 9 out of 11 instances, same as before. Accuracy and Kappa statistics are also same as before.



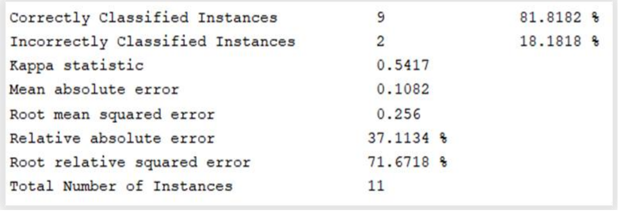
```
Correctly Classified Instances           9               81.8182 %
Incorrectly Classified Instances         2               18.1818 %
Kappa statistic                          0.5417
Mean absolute error                      0.1074
Root mean squared error                  0.2367
Relative absolute error                 36.8163 %
Root relative squared error             66.2795 %
Total Number of Instances               11
```

Before preprocessing, Decision Tree has correctly classified 9 out of 11 instances. Accuracy is 81.8182% and Kappa statistics is 0.5319.



```
Correctly Classified Instances           9               81.8182 %
Incorrectly Classified Instances         2               18.1818 %
Kappa statistic                          0.5319
Mean absolute error                      0.1082
Root mean squared error                  0.2966
Relative absolute error                 37.1134 %
Root relative squared error             83.0279 %
Total Number of Instances               11
```

After preprocessing, Decision Tree model classifier has correctly classified 9 out of 11 instances, same as before. Accuracy is 81.8182%, same as before. Kappa statistics is 0.5417, that increased 0.0098.



```
Correctly Classified Instances           9               81.8182 %
Incorrectly Classified Instances         2               18.1818 %
Kappa statistic                          0.5417
Mean absolute error                      0.1082
Root mean squared error                  0.256
Relative absolute error                 37.1134 %
Root relative squared error             71.6718 %
Total Number of Instances               11
```

REFERENCES

[1] S. D. Jadhav, H. P. Channe, "Comparative Study of K-NN, Naive Bayes and Decision Tree Classification Techniques." International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064. Paper ID: NOV153131. Volume 5 Issue 1, January 2016

[2] Arif, M. A. I., Sany, S. I., Sharmin, F., Rahman, M. S., & Habib, M. T. "Prediction of addiction to drugs and alcohol using machine learning: A case study on Bangladeshi population." International Journal of Electrical and Computer Engineering, Vol. 11, No. 5. October 2021

[3] I. A. A. Amra, A. Y. A. Maghari, "Students Performance Prediction Using KNN and Naïve Bayesian." 8th International Conference on Information Technology (ICIT). DOI: 10.1109/ICITECH.2017.8079967. May 2017