ANALYSIS OF ROAD ACCIDENTS IN INDIA USING DATA MINING TECHNIQUES

Submitted by

Sanyam Jain

1613101626/16SCSE101831

in partial fulfilment for the award of the degree
of

Bachelor of Technology
In
Computer Science and Engineering

School of Computer Science and Engineering

Under the supervision of

Mr. Lalit Sharma

Professor

TABLE OF CONTENTS

1. ABSTRACT

“**ANALYSIS OF ROAD ACCIDENTS USING CLUSTERING AND CLASSIFICATION**” is pivoted on the very idea of drawing a rather neat estimate of the road accidents grazing the states in India. The project is focused on finding out the dominant factor behind the accidents while keeping a clear track off the area most vulnerable to accidents in order to make constructive predictions so as to lower the accident rate. The data set that we are analysing gives a state-wise**--**one year span**--** excerpt on the road accidents with an insight into the various factors associated with the accident: cause, casualties, fatalities, time of occurrence, population of the region, vehicular population, driver’s educational background. The data set is first broken down into clusters via **k-means** clustering method that are further analysed for outliers only to be classified with the help of Naïve-Bayes classifier.

2. INTRODUCTION

(i) Purpose

Our strategy is to create clusters to group the homogeneous objects from this heterogeneous lot. These clusters are then checked for the presence of outliers. These outliers are then analysed. The dataset is further subjected to classification, which gives us a rather neat opportunity to examine the causes that played a major role in the road accidents.

(ii) Motivation and scope

A lot of studies have been carried out in this field so far, but none of them fairly succeeded in highlighting the different causes feeding the grotesque surrounding the accidents spooning the various regions of India. This project brings to light not just the dominant cause behind the road accidents but also gives an elaborate report on the contribution of each of the causes listed in the dataset.

Data mining techniques have seen quite an uproar in the usage these days. This is attributed to the ever increasing need to find hidden patterns in the data. Classification and clustering happen to be one of the many techniques that are used to serve the purpose right. Many a research has been done so far using the various data mining techniques in order to examine the various dimensions associated with the road accidents happening all over the world. We studied how a grand dataset can be analyzed to draw such conclusive and concrete results. From the dataset overlooking the road accidents of Erzuram[1], the districs bearing the highest risk of injury are determined. This is done as a result of cluster analysis. Clusters are made first via fuzzy c-means clustering technique and then using standard k-means clustering technique. Difference between the two techniques is that in fuzzy c-means, the clusters so formed can accommodate an object belonging to another cluster. It was observed that the clusters formed as a result of fuzzy c-means were more stable than the ones formed via k-means. The cluster analysis stated that in developed districts, the rate of accidents was rather high.

In order to analyze pedestrial fatal accidents[2], neural networks are applied. This dataset gives an elaborate pedestrian accident detail of Israel. Kohonen neural networks treat a large number of variables in order to obtain a relationship, pattern without any interruption in the form of predefined assumption. They are unsupervised and are preferred over supervised methods of learning and thus are easily interpretable as opposed to k-means clustering. They can compute large amount of data without ever needing to apply Principal Component Analysis to remove the unwanted attributes. Five clusters are formed as result that are analyzed to find out the causes.

One of the various data mining techniques that are used to determine the hidden pattern in data in order to mine it brilliantly is classification. Different classification techniques are suited for different sorts of data[3]. The various classification techniques are compared using diverse datasets from University of California, Irvine(UCI). The performance of J48, MultilayerPerceptron, NaiveBayesUpdatable, and BayesNet is evaluated in terms of accuracy and time. The efficiency of J48 is found out to be better than that of Naïve Bayes.

In addition to clustering techniques such as k-means and the basic classification techniques such as decision tree, naïve bayes, k-nearest neighbors, neural networks etc., there is yet another classification technique that can be used to analyze a dataset of road accidents[4]. ID3 and J48 are the basic decision tree algorithms that are used to classify data. Enhanced Decision tree that is used here to analyze traffic accidents here uses an enhanced version of C4.5 decision tree algorithm.

In a yet another such refreshing example of crash analysis[5], the pedestrian accidents taking place in urban areas are taken into account in order to find out the hotspots in the Novi-Sad area of Serbia. The clusters formed via k-means are:

i. gender and age

ii. crossing the road or street with respect to the direction of travel of the vehicle

iii. environment (weather when the accident occurred)

The results were then visualized with the help of KDE (Kernel Density Estimator).

The paper [6] suggests a prediction model to classify injury severity associated with the various road accidents in order to analyze the data better. Both Naïve

Bayesian classifier and J48 decision tree are used to classify the dataset. It was found out that J48 outperformed Naïve Bayesian.

This paper [7] proposed a framework that uses K-modes clustering technique to segment the data of accidents on the roads of Dehradun between 2009 and 2014. To identify the various circumstances that are associated with the occurrence of an accident, association rule mining is used. First the rules are extracted off the entire dataset and then off the clusters so formed. The results of cluster analysis and entire dataset analysis were then compared.

Various studies-pivoting on the conventional statistical methods-have been carried out in order to analyze road accidents. The statistical approach used to create a crash prediction model fails to consider the uncertainty factor associated with it. The regression candidate models (Negative Binomial Model and Poisson Model) used in reaching a concrete conclusion, go into shaping the model space. Of these candidate models, any one is selected to predict the frequency of accidents. Bayesian Information Criterion (BIC), Deviance Information Criterion (DIC) and Akaike Information Criterion (AIC) are some of the criteria used to select the most suited model that efficiently takes into consideration the posterior model probabilities. Explanatory (independent) variables' selection is the most vital part of constructing a crash analysis and prediction model. The BMA approach is used using the leaps and bounds algorithm [1] to implement the regression model. Artificial Neural Networks (ANN) have been successful in overcoming the uncertainty clause posed by the conventional statistical approach, but lead to overfit of the data.[2] Data Mining

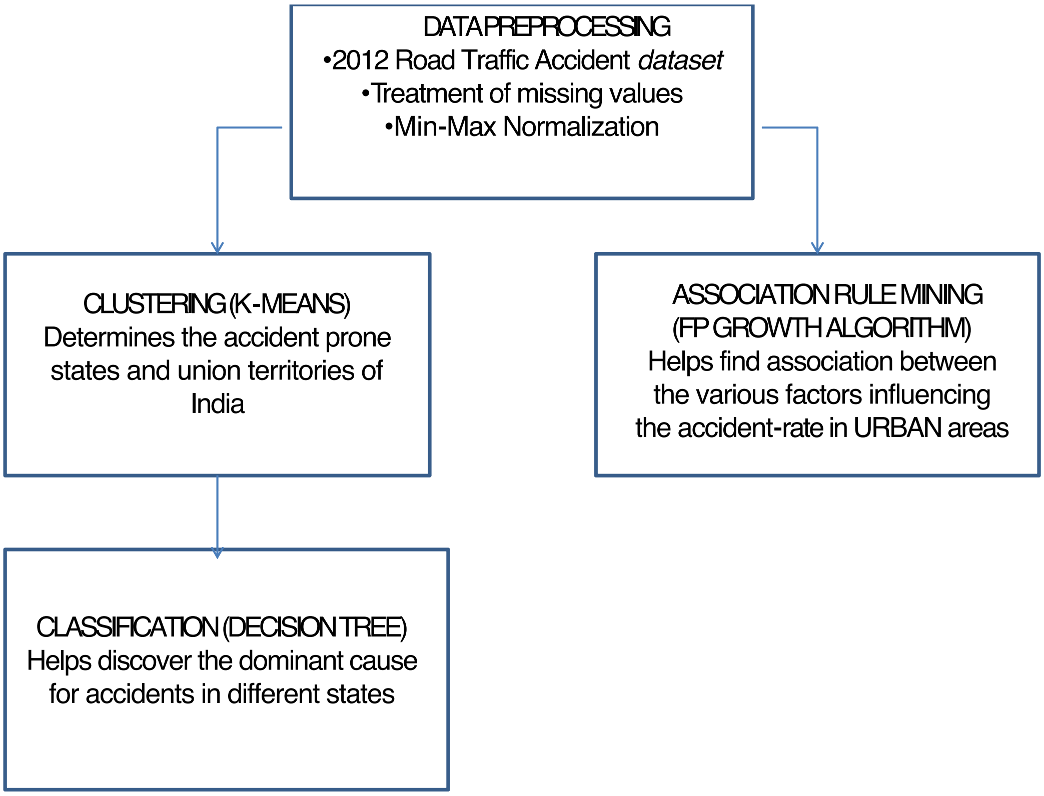
Techniques help overcome the aforementioned shortcomings. The idea is to determine the dominant factor inflicting the various states of India based on the dataset that we have; and to sketch out all the possible outcomes off the various factors that directly or indirectly had a part in shaping the dataset. The data mining techniques that we aim to use for the same are *clustering (K-Means) and classification (Decision Tree).* The data is preprocessed and made fit to be subjected to the data mining techniques.As per the estimates made by the World Health Organization (WHO) in the Global Safety Report, India accounts for more than 200,000 fatalities. Various studies have been carried out so far to determine the hike experienced in the field

The roads of India haven't abated their contribution in the traffic accident fatalities. The accident rate of India has been on an increase ever since the start of the century. According to a census taken in 2013, the number of accidents in India reached a bulging '1,37,000'. It was assessed that one death occurred every four minutes. In the year 2014, the road networks of India accounted for 63% of total road accidents and saw a 3% raise from 2013, with every hour witnessing 16 fatalities. In 2015, the rate saw a hike of 5% and the number of deaths per hour spiked to 400.

PROBLEM STATEMENT

Road safety has become quite a concern these days, considering the increase in population and vehicles. The roads nowadays are unnecessarily crowded with vehicles, welcoming a subsequent raise in the probability of a mishap happening. A lot of factors are attributed to the accidents that the roads have come to symbolise these days. Our aim is to break down the data concerned with the road accidents to the point that the factors behind these accidents can be analysed to give away some rather concrete results, in a bid to keep such mishaps from happening.

3. PROPOSED SYSTEM



DATA PREPROCESSING
•2012 Road Traffic Accident *dataset*
•Treatment of missing values
•Min-Max Normalization

CLUSTERING (K-MEANS)
Determines the accident prone states and union territories of India

ASSOCIATION RULE MINING
(FP GROWTH ALGORITHM)
Helps find association between the various factors influencing the accident-rate in URBAN areas

CLASSIFICATION (DECISION TREE)
Helps discover the dominant cause for accidents in different states

4. IMPLEMENTATION

DATA PREPROCESSING

First off, it is made sure that the dataset doesn’t have any missing values. If it happens to have any missing values, they are swapped with the integer ‘0’. In a bid to ease the entire procedure of crash analysis, the dataset is first normalized, following the min-max normalisation method.

$$V`= [V-\text{Min}(a))\div(\text{Max}(a)-\text{Min}(a)]*(\text{newMax}-\text{newMin})+\text{newMin}$$

Where **V** is the **firs**t instance of the field

Min(a) corresponds to the **minimum field value**

Max(a) corresponds to the **maximum field value**

newMax is **1** and **newMin** is **0**

PURPOSE OF USING MIN-MAX NORMALISATION

The dataset is normalised to have the field values spring anywhere between **0** and **1**

By doing so,

- The effectiveness of the mining algorithm is significantly improved
- Breaks data down, making it comprehensible
- Makes it easy to retrieve values from the dataset
- Quick retrieval improves the efficiency of the computations being carried out

The techniques that we aim to use to have every one of the dimensions of the dataset to be analyzed in the finest way possible are:

1. Clustering (K-MEANS)
2. Outlier analysis

3. Classification (NAÏVE-BAYES)
4. Association Rule Mining

CLUSTERING

The very objective of clustering is to group similar objects together and thus making it easier to identify the properties of an object belonging to one of the groups so formed.

It helps analyze the unlabelled data better.

The various clustering techniques supported by RapidMiner are:

I. K-MEANS
II. K-MEDOIDS
III. EXPECTATION MAXIMIZATION

The operator we are concerned with here is K-MEANS.

K-MEANS

- This operator makes use of the K-means Algorithm to form clusters.
- As per this algorithm, an object can be assigned to only one cluster.
- Measure of the distance between the various objects contained in a cluster forms the basis for the determination of similarity between them.
- Euclidean distance is the measure used here to define the centroid of a cluster.
- K, representing the number of clusters, is usually given a small integer value. It can be 1,2,3 and so on.
- K points are then chosen which represent the centroids of k clusters, preferably the ones without any members.

- The initial k points are chosen that are fairly distant from each other.
- These points are considered individually and placed to the cluster with the centroid nearest to it.

K—the number of clusters—is measured by either of the two indexes **(quality measures):**

1. Silhouette Index
2. Davies-Bouldin Index

Davies-Bouldin Index

- The Davies-Bouldin Index determinesthe following:
  - **Intra**-cluster **similarity**
  - **Inter**-cluster **differences**

Silhouette Index

- The Silhouette Index measures the distance between each object.
  - The centroid of the cluster the object is assigned to and the closest centroid belonging to another cluster.

Operators used in the implementation of Clustering are:

i. Read excel—this operator is used to read the dataset from the excel file

ii. K-means Clustering—this operator uses k-means algorithm to form clusters

iii. Cluster Distance Performance—the centroid based clustering methods are evaluated using this operator

The various visualization techniques used to graphically describe the behavior of data are:

- Scatter plot
- Histogram
- Box plot
- Parallel plot
- Matrix plot etc.

Algorithm

(The k-means algorithm for partitioning) The centre of each cluster is represented by the mean value of the objects in the cluster.

Input

k: the number of clusters,

D: a data set containing n objects.

Output: A set of k clusters.

Method

(1) k objects are randomly chosen from 'D' as the initial cluster centers

(2) Repeat 1

(3) Each object is assigned or reassigned to the cluster to which the object resembles the most. This is done on the basis of the mean value of the objects in the cluster

(4) The mean value of the objects in each cluster is updated

(5) This goes on until no change is encountered

The algorithm was applied for different values of k=2,3…7. The quality of clusters is then evaluated using Davies-Bouldin Index.

CLASSIFICATION

In a bid to target categories and classes, items are assigned in a specific collection. This is referred to as Classification. Its aim is to predict the class for each and every object in the dataset. The nature of the classification that is carried out is rather discrete.

The various techniques that could be used by to classify the data are:

1. Naïve bayes classifier
2. Decision trees
3. K-nearest neighbours

NAÏVE BAYES CLASSIFIER

Naïve bayes classifier is a probabilistic classifier that works on the principal of the Bayesian theorem. A rather strong, naïve assumption is made regarding the independence of the various features of the data, in order to carry out classification.

The **probabilistic model** the classifier is based on

It is referred to as the ‘conditional probability model’

Object to be classified is described by a vector

$$\mathbf{x} = (x_1, \ldots, x_n)$$

Where, n is the number of independent variables

This vector is assigned to instance probabilities

$$p(C_k | x_1, \ldots, x_n)$$

Where, k is the number of possible outcomes

Using Bayesian theorem, the conditional probability can be broken down as,

$$\text{posterior} \quad \frac{\text{prior} \times \text{likelihood}}{\text{evidence}}$$

Naïve bayes Classifier is the modified version of the model stated above. It is blends the theorem with the decision rule, hence making it edgier.

DECISION TREE

Decision tree follows the tree structure and consists of a root node, intermediate nodes and lead nodes. Decision is contained by each node and based on this decision, the tree progresses. Label, value and action define the mutually exclusive spaces created in a tree. These trees are generated via recursive partitioning. The performance of J48, MultilayerPerceptron, NaiveBayesUpdatable, and BayesNet was evaluated to analyze road accidents in terms of accuracy and time. The efficiency of J48 is found out to be better than that of Naïve Bayes[5].

DECISION TREE

The attributes in our dataset are numeric. Using the decision tree operator on the already formed clusters, a decision tree is generated.

A decision tree can be defined as a tree-like structure. It tends to grow downwards with root at the top, which makes it look like an inverted tree. The classification model that it helps generate is easy to understand and interpret.
The aim is to facilitate the creation of a classification model that helps determine the target attribute's value. This target attribute is usually referred to as class or label.

The decision tree operator in RapidMiner helps predict the attribute that is to be taken as label. The input attributes are represented by the nodes of the tree. Each node of the tree is associated with one of the attributes.

The number of edges branching out of node closely resembles the number of possible values of the attribute. These branches are labeled with their respective ranges. The value of the label attribute is represented by the leaf node, effectively giving the path followed with from the root to the leaf.

Generation of decision trees is attributed to recursive partitioning which helps split on the values of attributes repeatedly.

- **An attribute ‘a’ is chosen to split on**

In order to generate an effective decision tree, this attribute is carefully selected. The ‘selection Criterion’ parameter helps make this choice.

- **Instances in the Dataset are sorted into subsets** that give the range depending on the value of attributes.
- Recursive application of this algorithm results in each branch having a subtree. For each subset, the tree is returned with an appropriate branch.

A halt in recursion is encountered when all the instances have the same label value. In other words, recursion stops when the subset turns out to be pure.

Factors that determine the generation of a decision tree are:

- Split parameter can be used to define a minimal size which determines whether the number of instances in the tree are sufficient or not.
- It is made sure that the decided threshold is reached by the attributes.Minimum gain parameter is used for this.
- Maximal depth parameter that goes into defining the maximal depth of the tree makes sure that the recursion doesn’t make it past this point.
- It is defined as a technique that removes the leaf nodes that lack relevant discriminative power.

- This ensures that the model so formed efficiently predicts outcomes from the unspecified datasets and also keeps over-fitting at bay.

Pre-pruning

- This type of pruning runs parallel with the creation of the tree.

Post-pruning

This comes alive once the tree is generated.

Criterion

Selects the criterion on which attributes will be selected for splitting. It can have one of the following values:

1. Information gain:
    - It is an impurity-based criterion
    - It uses the 'entropy' measure
    - The entropy of each and every attribute is calculated.
    - The attribute associated with minimum entropy is taken..
    - This crierion usually selects attributes with a fairly large number of instances.

2. Gain ratio:
    - It can be defined as a variant of information gain.
    - It helps adjust the information gain of the attributes in order to enhance the uniformity of the attributes.
    - It is used to normalize the information gain

    GainRatio(ai , S) = [InformationGain(ai , S)]/[Entropy(ai , S)]

3. Accuracy:
    - The attribute selected as a result of this measure maximizes the accuracy of the tree so formed.
4. Gini Index

- Gini index is an impurity-based criterion
- It evaluates the divergences seen between the probability distributions of the values of the target attribute.

Following are the parameters used to define the various aspects of the decision tree

1. **Minimal size for split**
    - Nodes with value greater than or equal to the set minimal size are chosen for split
2. **Minimal leaf size**
    - A leaf node's size equals the number of instances in the subset.
    - It is made sure that every leaf node subset has the 'minimum leaf size' number of instances.

3. **Minimal gain**

    - The gain associated with a node is evaluated before splitting it.
    - If the Gain turns out to be greater than the minimal gain, the node is split. Higher the value of minimal gain, smaller would be the tree.

4. **Maximal depth**

    - It is used to restrict the size of the tree

5. **Confidence**

    - This parameter is used to specify the confidence level used to calculate the pessimistic error associated with pruning.

6. **Number of prepruning alternatives**

- Since prepruning performs parallel to the tree generation, splitting may be prevented at certain nodes. In this case, other nodes are operated on for splitting. This parameter is used to adjust the number of such alternative nodes whenever the splitting is prevented at some node.

7. **No prepruning**

   - The Decision Tree is generated with prepruning, by default.

8. **No pruning**

   - This parameter is used when the tree is to be generated without pruning.

ASSOCIATION RULE MINING

Association rule mining is used to derive intriguing relations between the various items that comprise the database. This method is usually applied while mining enormous amount of data.

There are a certain rules that need to be justified in order for the mining to take place on the data set. A threshold is set on the following aspects:

1. Support
2. confidence

Association rules are defined as ‘if-then’ statements that help discover underlying associations between the various elements of a data.

- An association rule is comprised of two parts:

  i. Antecedent (if)

Item found in the data is called an antecedent. It can be an itemset as well.

ii. Consequent (then)

An item or itemset that is found along with the antecedent is a consequent.

Association rules are created by mining data for hidden frequent patterns. In order to find out the most important relationships, ***support*** and ***confidence*** criteria are used. **Support** signifies just how frequently an item/itemset appears in the dataset. **Confidence** states the number of times the if/then relationship has turned out to be true.

FP-Growth operator is used to mine the data for frequent patterns via FP data structure.

The **Create Association Rules** operator takes into account the so formed frequent-itemsets and thus, generates association rules.

PARAMETERS associated with Association Rule Mining

1. **Criterion**
2. The criterion used to generate rules is specified using this.
3. **Confidence**

   For, $X \Rightarrow Y$ (rule) and **T**(transactions),

   Confidence is defined as the number of times both(X and Y) occur together

   $$\mathrm{conf}(X \Rightarrow Y) = \mathrm{supp}(X \cup Y)/\mathrm{supp}(X).$$

4. **Lift**
   - The lift of a rule is defined as

     lift(X implies Y) = supp(X ∪ Y)/((supp(Y) x supp(X))
   - The ratio of the observed support to that expected provided X and Y were independent is referred to as lift.

- Its being close to 1 implies that X and Y are independent and the rule is not interesting.

5. **Conviction**
   - It's sensitive to rule direction
     conv(X implies Y) differs from conv(Y implies X)
     conv(X implies Y) =(1 - supp(Y))/(1 - conf(X implies Y))
6. **Gai**n
   - On this option's selection, the gain is estimated using the gain theta parameter.
7. **Laplace**
   - On this option's selection, the Laplace is estimated using the laplace k parameter.
8. **Ps**
   - On this option's selection, the ps criterion is used for the selection of rule.
9. **Min confidence**
   - The minimum confidence of the rules is specified using this parameter.
10. **Min criterion value**
    - The minimum value of the rules for the selected criterion is specified using this parameter.
11. **Gain theta**
    - The parameter is used to calculate Gain.
12. **Laplace k**
    - The parameter *k* which is used in the Laplace function is specified using this parameter.

Association rules are extracted off the dataset using FP Growth Algorithm. This operator efficiently calculates all frequent itemsets from the dataset using the FP-tree data structure and efficiently elaborates many-to-many relationship between two kinds of objects.

1. A **minimum support threshold** is used to find all *frequent itemsets* in a dataset.
2. **A minimum confidence constraint** is applied to the frequent itemsets in order to derive rules.

FP-GROWTH ALGORITHM

- It was introduced by Han in 2000
- It follows the pattern growth approach.
- It uses specific data structure; FP Tree
- The frequent itemsets are molded into a pattern based on node-structure of the FP tree.
- FP-tree provides a avoids a way to mine data without having to scan it repeatedly, hence saving time, space and resources.
- Strays from the idea of the generation of candidate sets.

CONSTRUCTION OF A FP TREE

1. The **root** is set to *null*, a set of **item-prefix subtrees-**children of the root; **frequent-item**-header table.
2. Each and every node that goes into forming the item-prefix subtree is comprised of three fields:

- Item_name
- count
- node_link

item-name identifies the item the node represents

count identifies the transactions in the form of the edges reaching this node

Node-link links the target node to the next node that bearing the same name

Node link can be null if there is no such node in succession to the current node

3. Frequent-item-header table’s entries are comprised of two fields:
    - **item-name**
    - **head of node-link**; a pointer that points to the node representing the same item and bearing the same item-name

DATA COLLECTION

The dataset gives an elaborate account on the road accidents that transpired through the year 2012 in the many states of India, and is subjected to the various data mining techniques in order to draw rather conclusive information.

The idea is to determine the dominant factor inflicting the various states of India based on the dataset that we have; and to sketch out all the possible outcomes off the various factors that directly or indirectly had a part in shaping the dataset; namely, the educational background of the drivers involved in the accidents, the type of vehicle that bore cause, the various causes that intermittently contributed in the proceedings over the course of twelve months and the many lives that fell short of their time.
The various attributes that go into adding to the skeletal of the dataset, against the States and the Union Territories are:

1. Total number of accidents involving alcohol
2. Number of people killed due to alcohol’s involvement
3. Number of people injured due to alcohol’s involvement
4. Total number of accidents due to exceeding speed limit

5. Number of people killed due to speeding

6. Number of people injured due to speeding

7. Total Number of accidents due to driver’s fault

8. Number of people killed due to driver’s fault

9. Number of people injured due to driver’s fault

10. Total number of accidents involving cyclists

11. Number of people killed in accidents involving cyclists

12. Number of people injured in accidents involving cyclists

13. Total number of accidents involving pedestrians

14. Number of people killed in accidents involving pedestrians

15. Number of people injured in accidents involving pedestrians

16. Total number of accidents due to vehicular defect

17. Number of people killed due to vehicular defect

18. Number of people injured due to vehicular defect

19. Total number of accidents due to road defect

20. Number of people killed in accidents due to road defect

21. Number of people injured in accidents due to road defect

22. Total number of accidents due to bad weather

23. Number of people killed in accidents involving weather

24. Number of people injured in accidents involving weather

25. Total number of accidents due to overloading

26. Number of people killed in accidents involving overloading

27. Number of people injured in accidents involving overloading

28. Total number of accidents due to load protruding

29. Number of people killed in accidents due to load protruding

30. Number of people injured in accidents due to load protruding

31. Total number of accidents involving two wheelers

32. Number of people killed in accidents involving two wheelers

33. Number of people injured in accidents involving two wheelers

34. Total number of accidents involving auto-rickshaws

35. Number of people killed in accidents involving auto-rickshaws

36. Number of people injured in accidents involving auto-rickshaws

37. Total number of accidents involving cars

38. Number of people killed in accidents involving cars

39. Number of people injured in accidents involving cars

40. Total number of accidents due to bus

41. Number of people killed in accidents due to bus

42. Number of people injured in accidents due to bus

43. Total number of accidents due to trucks

44. Number of people killed in accidents due to trucks

45. Number of people injured in accidents due to trucks

46. Total number of registered drivers having passed the 8th grade

47. Total number of registered drivers having passed the 10th grade

48. Total number of registered drivers having attended school beyond 10th grade

49. Total number of registered drivers with an unknown educational status

50. Total number of accidents that occurred in urban areas

51. Number of people killed in accidents that occurred in urban areas

52. Number of people injured in accidents that occurred in urban areas

53. Total number of accidents that occurred in rural areas

54. Number of people killed in accidents that occurred in rural areas

55. Number of people injured in accidents that occurred in rural areas

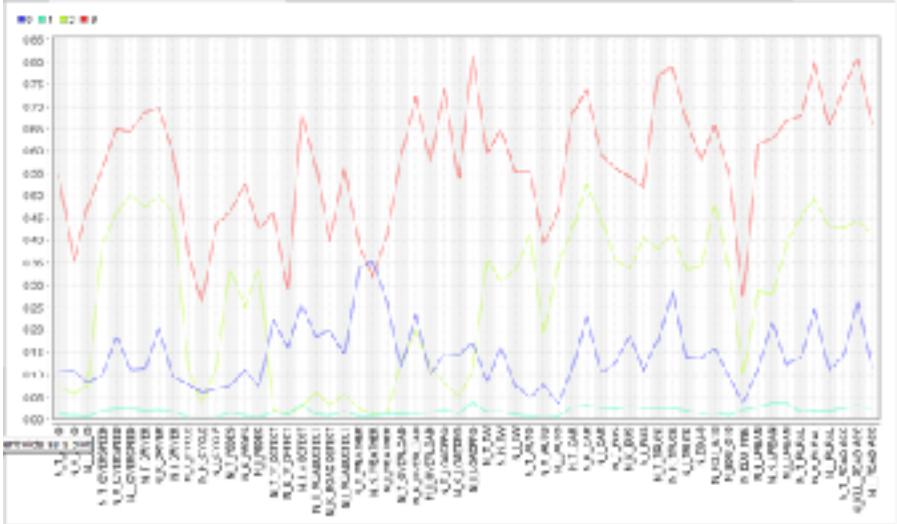
56. Total number of road accidents that inflicted the country in 2012

57. Total number of people killed in the road accidents in 2012

58. Total number of people injured in the road accidents in 2012

5. RESULTS

Before the dataset is subjected to k-means clustering, in order to group the homogeneous factors contained in it, it is normalized using Min-Max normalization. Of the two quality measures (Davies-Bouldin Index and Silhouette Index), Davies-Bouldin Index is used to determine the strength and quality of the clusters so created.



Cluster 0: This cluster turned out to be consisted of 7 states namely Bihar, Chhattisgarh, Haryana, Jharkhand, Odisha, Punjab and West Bengal. An investigation of the cluster led us to set the "medium" label to these states as these were the states with an average population situated in the northern plains, as per the dataset.

Cluster 1: This cluster turned out to be consisted of a total of 19 items including 12 states and 7 Union Territories of India, namely Arunachal Pradesh, Assam, Goa, Himachal Pradesh, Jammu & Kashmir, Manipur, Meghalaya, Mizoram, Nagaland, Sikkim, Tripura and Uttrakhand , Andaman & Nicobar Islands, Chandigarh, Dadar & Nagar Haveli, Daman & Diu, Delhi, Lakshadweep and Puducherry respectively. A study of this cluster based on geography and population revealed that all the states grouped together were the least populated in the country and the most of these states had a hilly terrain. These two factors resulted in low accident rate in these states. Therefore, the states in this cluster were labeled "low".

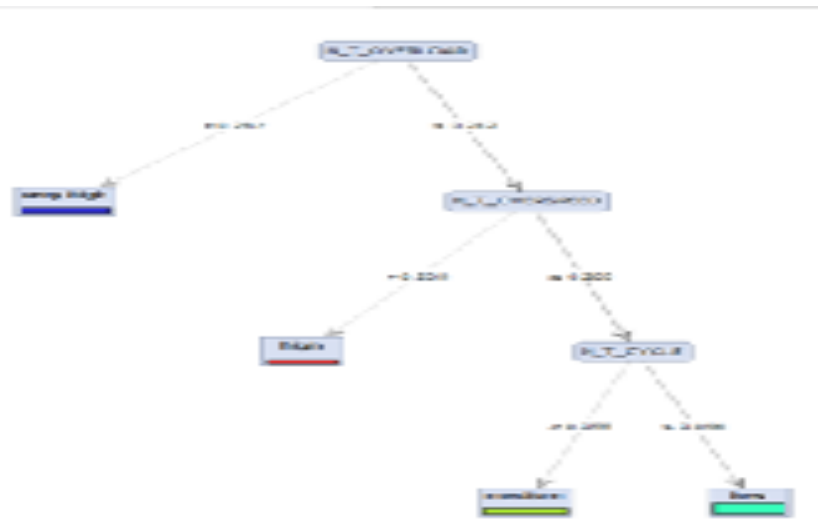
Cluster 2: This cluster was found to contain 3 states namely Gujarat, Rajasthan and Kerala. Gujarat and Rajasthan are desert states whereas Kerala is a coastal state. Both these states have a high population despite the less favorable geography. Also, the connectivity via the roads is good. So, the states in this cluster were assigned a "High" label.

Cluster 3: This cluster was found to be comprised of 6 states: Andhra Pradesh, Karnataka, Madhya Pradesh, Maharashtra, Tamil Nadu and Uttar Pradesh. They are flourishing states with highest population rate in the country. The terrain in some regions is plain and others, plateau with a not-so-well constructed network of roads. This causes the accident-rate to see a swell. Thus, 'Very high' label is assigned to the states present in this cluster.

Davis Bouldin	Avg. within centroid distance	Avg. within centroid distance_cluster_0	Avg. within centroid distance_cluster_1	Avg. within centroid distance_cluster_2	Avg. within centroid distance_cluster_3
1.071	0.918	0.619	0.072	1.327	3.741

CLASSIFICATION MODEL

Decision tree follows the tree structure and consists of a root node, intermediate nodes and lead nodes. Decision is contained by each node and based on this decision, the tree progresses. Label, value and action define the mutually exclusive spaces created in a tree. These trees are generated via recursive partitioning. The performance of J48, MultilayerPerceptron, NaiveBayesUpdatable, and BayesNet was evaluated to analyze road accidents in terms of accuracy and time. The efficiency of J48 is found out to be better than that of Naïve Bayes[5].



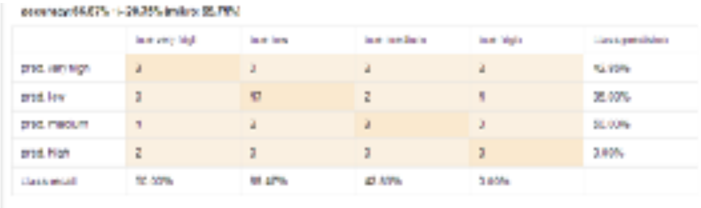
Classification methods are used to identify the main cause of accidents. The data set is classified based on the labels assigned to it following cluster analysis. Gain Ratio criteria is set to build the decision tree.

The gain ratio is defined as

GainRatio(*A*) = *Gain*(*A*)/*SplitInfo*(*A*) :

In the decision tree generated, the most important variable to split on is the total number of accidents due to overloading/overcrowding of the vehicle.

Cross validation of the model so formed gives the accuracy at 66.67%.



accuracy: 66.67% +/- [illegible] (mikro: [illegible])

	true very high	true low	true medium	true high	class precision
pred. very high	4	0	0	4	[illegible]
pred. low	3	57	2	5	[illegible]
pred. medium	1	4	3	0	[illegible]
pred. high	2	0	0	0	0.00%
class recall	[illegible]	[illegible]	[illegible]	0.00%	

Algorithm

Input:

- Data-partition *D*, which is a set of 'training tuples' and their corresponding class-labels
- *List of attributes*-set of candidate attributes

- *‘Attribute selection method’*, it is used to determine the ‘splitting criterion’

  Output: Decision tree

Method:

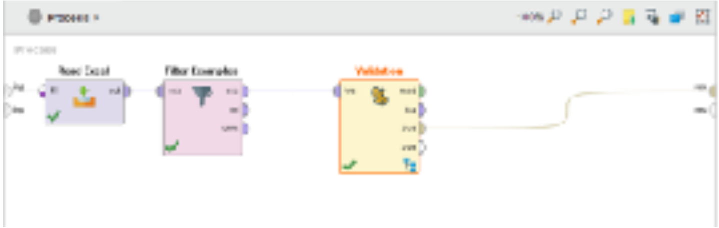
```
(1) A node N is created
(2) if tuples in D belong to the same class, C then
(3) N is returned as a leaf-node labeled C
(4) if attribute-list is empty, then
(5)  N is returned as a leaf node labeled with the class in D that holds majority
(6) Attribute selection method is applied to find the most suitable splitting criterion
(7) Node N is labeled with the obtained splitting criterion
(8) if splitting-attribute is discrete-valued and multiway splits are allowed, then
(9) attribute-list, splitting-attribute
(10) for every outcome j of splitting criterion
 (11) let Dj represent the set of data tuples in D satisfying outcome j
(12) if Dj is found to be empty, then
(13) leaf labeled with the majority-class in D  is attached to node N;
(14) else the node returned by Generate decision tree is attached to node N;
endfor
(15)  N is returned
```

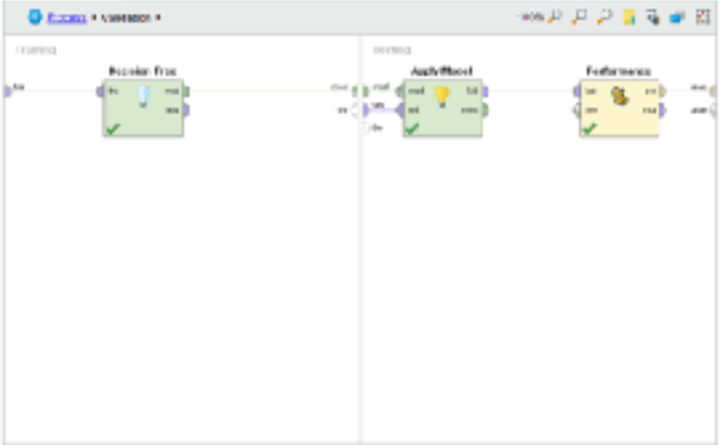
Cross Validation for a decision tree

Cross validation is a standard tool used to analyse the performance of a predictive model. Cross- Validation is used to estimate how accurately a model will perform. It consists of two sub-processes: a training sub-process and a testing sub-process.

The training sub-process consists the predictive model which requires to train the model. This model is then applied to the sub-process called testing. It is during this phase that the performance of the model is evaluated.

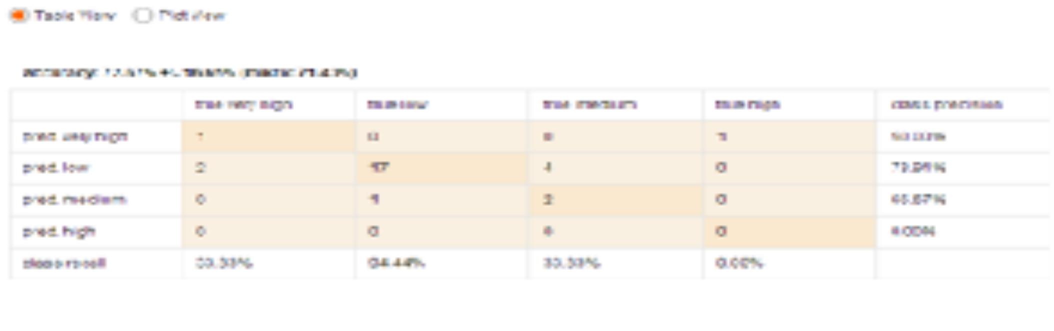
The dataset is partitioned into k subsets of equal size. The number of validations parameter is used to set the value of k. Out of all the k subsets, one subset is detained as the testing data set and the remaining k-1 subsets are used as training dataset. The cross validation is repeated k times, where each subset is used as a testing data exactly once. The result of each k iteration is then averaged to produce a single estimation.





Cross validation of the model so formed gives the accuracy at 72.67% .From the confusion matrix generated we can conclude that

- the “low” label is predicted correctly 94.44% times
- the “very high” label is predicted correctly 33.33% times
- the “medium” label is predicted correctly 33.33% times
- the “high” label is never predicted correctly



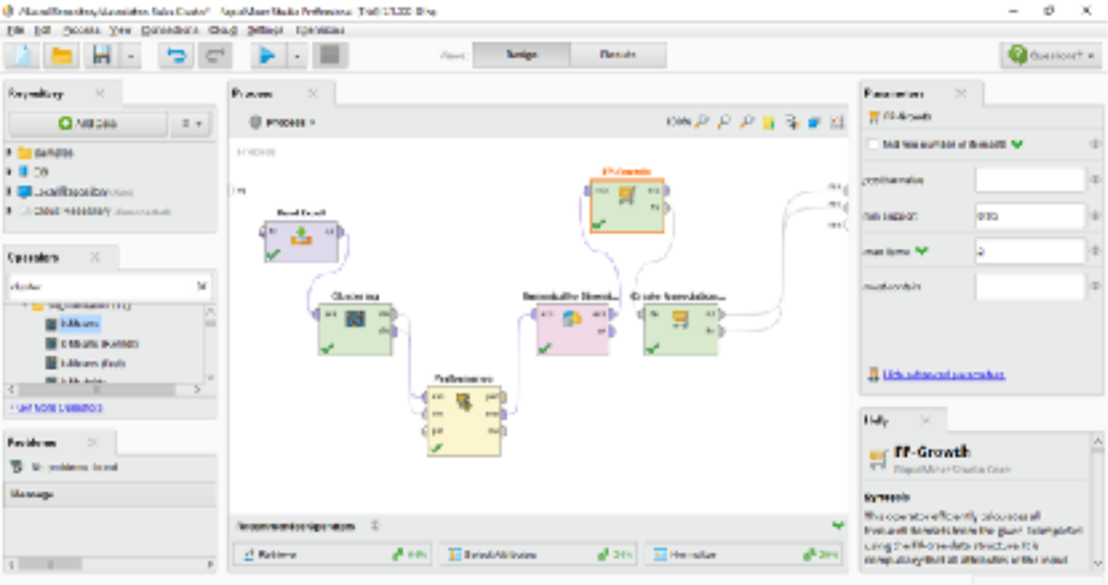
● Table View ○ Plot View

accuracy: [illegible]

	true very high	true low	true medium	true high	class precision
pred. very high	1	0	0	1	50.00%
pred. low	2	17	4	0	73.91%
pred. medium	0	1	2	0	66.67%
pred. high	0	0	0	0	0.00%
class recall	33.33%	94.44%	33.33%	0.00%	

ASSOCIATION RULE MINING

With minimum support set to 0.9 and confidence set to 1, rules were generated with the help of FP Growth operator and Create Association Rules operator are used to find associations between the various factors



URBAN

With support equals 0.971 and confidence equals 1, the association rules that stand out are as follows:

N_T_URBAN → N_K_URBAN

N_T_CAR→ N_K_URBAN

N_K_TW→ N_K_URBAN

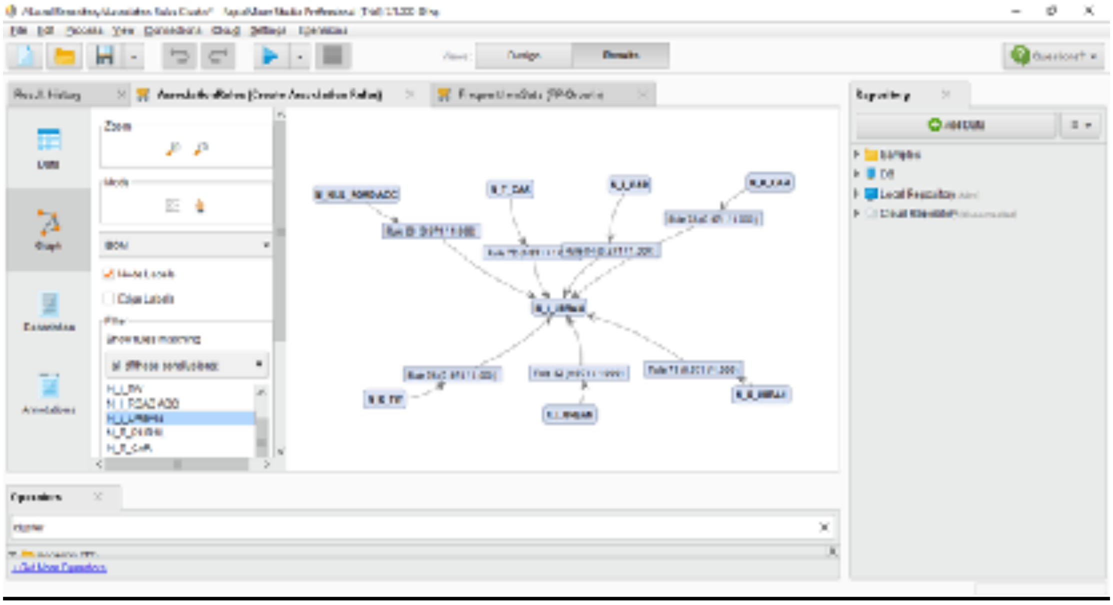
N_K_CAR→ N_K_URBAN

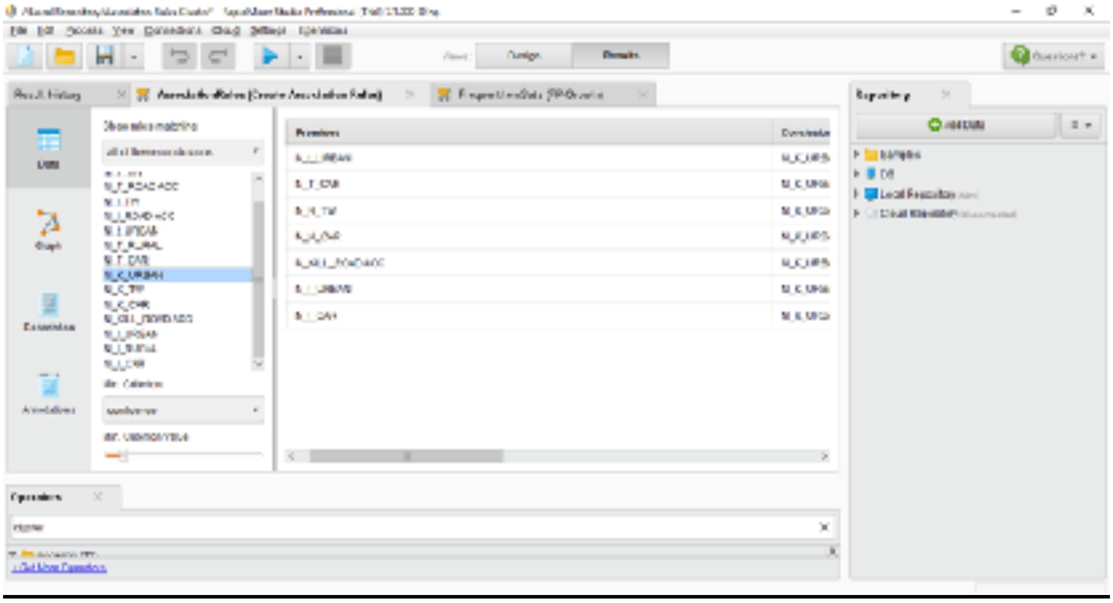
N_K_ROAD_ACC→ N_K_URBAN

N_I_URBAN→ N_K_URBAN

N_I_CAR→ N_K_URBAN

From the rules so formed, it is discovered that in urban areas, the prominent factors contributing to fatalities involve car and two wheelers.





Premises	Conclusion
N_I_URBAN	N_C_URB[illegible]
N_T_CAR	N_C_URB[illegible]
N_H_TW	N_C_URB[illegible]
N_H_CAR	N_C_URB[illegible]
N_KILL_ROAD ACC	N_C_URB[illegible]
N_I_URBAN	N_C_URB[illegible]
N_I_CAR	N_C_URB[illegible]

6. CONCLUSION

RURAL

With support equals 0.914 and confidence equals 1, the prominent rules formed are as follows:

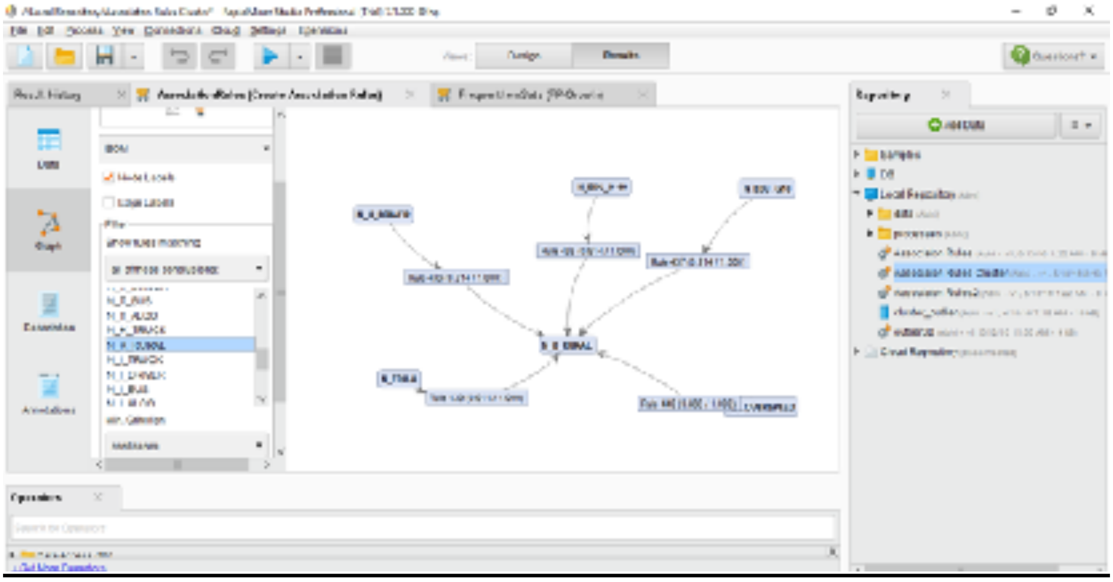
N_K_DRIVER→ N_K_RURAL

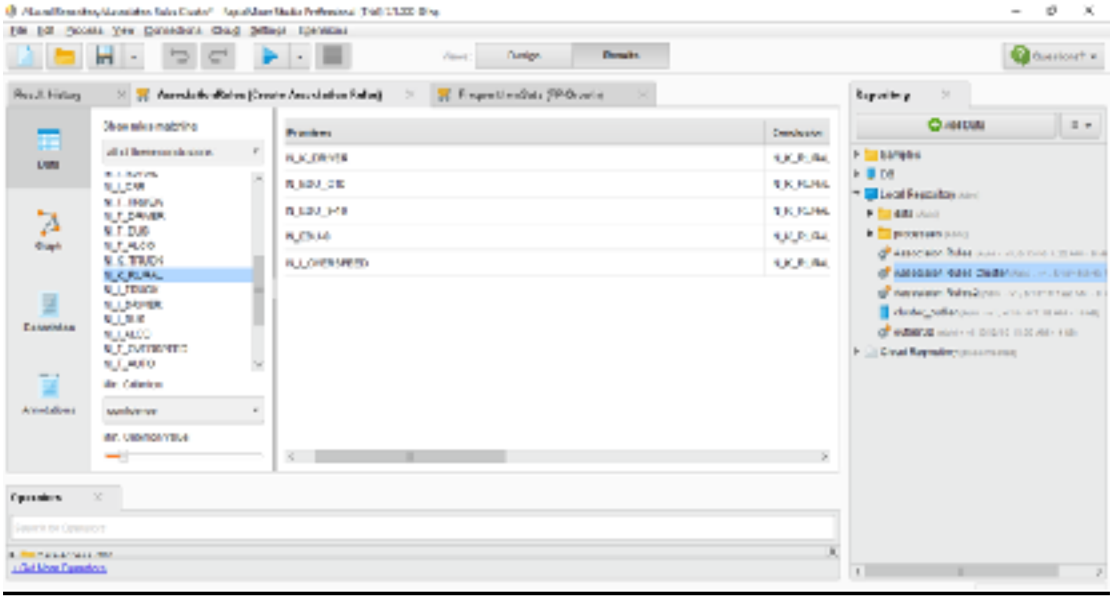
N_EDU_9-10→ N_K_RURAL

N_EDU_G10→ N_K_RURAL

N_EDU_8→ N_K_RURAL

N_I_OVERSPEED→ N_K_RURAL





From the rules so formed, it is discovered that the number of fatalities in rural areas are hugely influenced by driver’s fault. In addition to this, it was discovered that no driver driving the accident-inducing vehicles at the time of the accident was educated beyond grade 10.

7. REFERENCES

1. Yajie Zou, Dominique Lord(PHD), Yunlong Zhang (PHD), Yichuan Peng, 'Application of the Bayesian Model Averaging in Predicting Motor Vehicle Crashes', Texas A&M University, Journal of Transportation and statistics, vol10
2. Mohamed A Abdel-Aty, A Essam Radwan, 'Modeling traffic accident occurrence and involvement' , Accident Analysis & Prevention, vol32, Issue5
3. Hümeyra Bolakar , Ahmet Tortum , 'Clustering of Districts in Erzurum by Number of Injury', Journal of Traffic and Logistics Engineering Vol. 3, No. 2, December 2015 Department of Civil Engineering, Engineering Faculty, Aksaray University, Aksaray, Turkey

4. Carlo Giacomo Prato, Victoria Gitelman, Shlomo Bekhor, 'Mapping patterns of pedestrian fatal accidents in Israel', University of Queensland; article in accident; analysis and prevention · january 2012, (Research Gate), Israel Institute of Technology

5. V. Vaithiyanathan, K. Rajeswari, Kapil Tajane, Rahul Pitale, 'Comparison of different classification techniques using different datasets, International Journal of Advances in Engineering & Technology, May 2013, ISSN: 2231-1963

6. Naina Mahajan, Bikram Pal Kaur (PhD), 'Analysis of Factors of Road Traffic Accidents using Enhanced Decision Tree Algorithm', International Journal of Computer Applications (0975 – 8887) ,Volume 135 – No.6, February 2016 1 , Punjab Technical University, Punjab, India

7. ; svetlana bačkalić, boško matović, dragan jovanović , 'Identification of hotspots road locations of traffic accidents with pedestrian in urban areas', faculty of technical sciences, serbia

8. S.Vigneswaran, A.Arun Joseph, E.Rajamanickam, ‘Efficient Analysis of Traffic Accident using Mining Techniques’, International Journal of Software and Hardware research in engineering, vol2, issue3, march 2014, K.S.Rangasamy College of Arts & Science, Tiruchengodu, Namakkal District, Tamilnadu

9. Sachin Kumar and Durga Toshniwal, Kumar and Toshniwal, ‘A data mining framework to analyze road accident data’, Journal of Big Data (2015) 2:26 DOI 10.1186/s40537-015-0035-y, Journal of BigData, a Springeropen Journal

10. Amira A. El Tayeb, Vikas Pareek, Abdelaziz Araar, ‘Applying association rules mining algorithms for traffic accidents in dubai’, International Journal of Soft Computing and Engineering (IJSCE), ISSN: 2231-2307, Volume-5 Issue-4, September 2015

11. Tibebe beshah, Shawndra hill, ‘Mining road traffic accident data to improve safety; role of road-related factors on accident severity in Ethiopia’, Addis Ababa university, Ethiopi, university of Pennsylvania, Philadelphia PA

12. A. Priyanka, K. Sathiyakumari, GR Govindarajulu, ‘A comparative study of classification algorithm using accident data’, International Journal of Computer Science & Engineering Technology (IJCSET)

13. http://timesofindia.indiatimes.com/india/16-deaths-every-hour-Indian-roads-claim-the-maximum-number-of-lives-in-2014/articleshow/48128946.cms

14. http://sites.ndtv.com/roadsafety/important-feature-to-you-in-your-car/

15. http://www.indiaspend.com/cover-story/india-100-more-traffic-deaths-than-china-19692