Bayesian Network Modelling Workshop

Team Neo

EBAC 4 (Part Time), Institute of Systems Science, NUS

[e0146864, e0146946, e0147017, e0146771] @u.nus.edu

Abstract

This work is part of assignment to build a Naïve Bayes network + one other Bayesian network and then compare the results. We decided to use the Tree Augmented Naïve Bayes for the comparison.

1 EXECUTIVE SUMMARY

Our business problem is to build and compare Bayesian network prediction models to predict the likely injury level for vehicle occupants. Initially, the dataset was checked for missing values. Since, we found a considerable amount of missing values, they were imputed using kNN imputation method in R. Later, all the variables were analyzed for their correlation and the highly correlated variables were removed. We decided to build the **Naïve Bayes** model using R and also Genie. To accomplish the comparison, we also performed **Tree Augmented Naïve Bayes** on the dataset using Genie software. In both the models K-fold (5) cross validation was done for model validation.
Our key findings are:

1. Though we expected TAN to outperform Naïve Bayes but with the current data, the TAN probabilistic models **performed slightly better** than Naïve Bayes. [compared based on Genie output for both to avoid any margin of error due to tools. R output was for exploration only and not used in the analysis]
2. Both the models are good in predicting Minor to Moderate injuries. But have low accuracy to predict critical injuries.
3. Confusion Matrix for Naïve Bayes[1] and TAN[2]
4. If the car comes with safety Bag and it was deployed during accident, 76% chances are there that the injury will be Minor (Class 1) [3]
5. For the highest level of injury with maximum death probability, we found that majority of the vehicles involved are between 1400 – 2000 Kgs of weight, Speed limit was between 45-60.
6. For the critical injuries, even if 73% of the times bag was deployed. This implies that safety bag deployment is not the only factor in controlling the injury level.

2 DATA DESCRIPTION

The dataset was collected from Bayesian website, containing **21 variables** and approx. **20,500 observations**, **16** are **numerical variables** and **5** are **categorical variables.** However, at the later stage of data preprocessing, we binned numerical data into different categories based on range value.

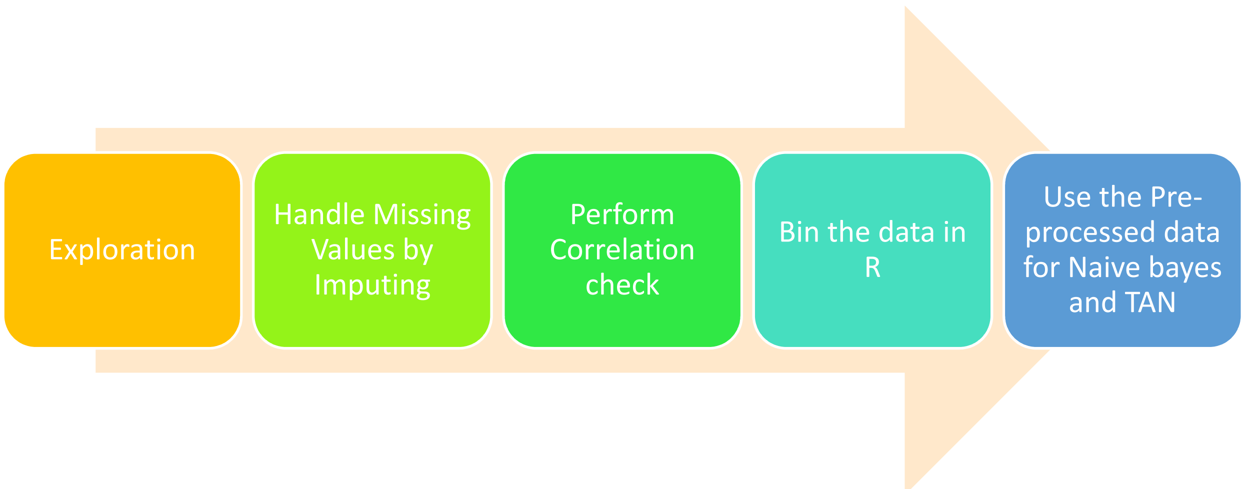
Table 1: Variable Description

S.No.	Name	Description	Variable type	Data Type
1.	GV_CURBWGT	Vehicle curb weight	Explanatory	Numerical
2.	GV_DVLAT	Lateral component of Delta V	Explanatory	Numerical
3.	GV_DVLONG	Longitudinal component of Delta V	Explanatory	Numerical
4.	GV_ENERGY	Energy absorption	Explanatory	Numerical
5.	GV_LANES	Number of Lanes	Explanatory	Numerical
6.	GV_MODELYR	Vehicle model year	Explanatory	Numerical
7.	GV_OTVEHWGT	Weight of the other vehicle	Explanatory	Numerical
8.	GV_SPLIMIT	Speed limit	Explanatory	Numerical
9.	GV_WGTCDTR	Truck weight code	Explanatory	Categorical
10.	OA_AGE	Age of Occupant	Explanatory	Numerical
11.	OA_BAGDEPLY	Air Bag System Deployed	Explanatory	Categorical
12.	OA_HEIGHT	Height of Occupant	Explanatory	Numerical
13.	**OA_MAIS**	**Maximum known Occupant AIS**	**Response**	**Categorical**
14.	OA_MANUSE	Manual belt system use	Explanatory	Numerical
15.	OA_SEX	Occupant's Sex	Explanatory	Categorical
16.	OA_WEIGHT	Occupant's Weight	Explanatory	Numerical
17.	VE_GAD1	Deformation Location	Explanatory	Categorical
18.	VE_PDOF_TR	Clock Direction for Principal Direction of Force	Explanatory	Numerical
19.	GV_FOOTPRINT	Vehicle Footprint	Explanatory	Numerical
20.	VE_ORIGAVTW	Average Track Width	Explanatory	Numerical
21.	VE_WHEELBAS	Vehicle Wheel Base	Explanatory	Numerical

3 Data Selection and Preprocessing

We followed the following process to analyze the dataset:

Figure 1: Process to Explore the data



3.1 Tools Used

- **R** – Used for Data exploration, imputation using kNN, Naïve Bayes modeling
- **Genie** – Naïve Bayes and Tree Augmented Naïve Bayes Network Modeling and Analysis

3.2 Dealing with Missing Values

Filling up missing Values in Numerical variables with kNN imputation in R: To proceed further, all the missing values were replaced with imputed values using k-Nearest Neighbour Imputation based on a variation of the Gower Distance for numerical, categorical, ordered and semi-continuous variables. It generates multiple imputations for incomplete multivariate data. A small piece of code has been written in R to implement imputation on numerical data.

Table shown below summarizes the **key statistical indicators** of the attributes that have been imputed. The **Original dataset** and **Imputed dataset** have the almost the **same Mean, Median** and **standard deviation** after imputation:

Table 2: Imputed Dataset

		Original Dataset				Imputed Dataset			
S. No	Variables	Rows	Mean	Standard Deviation	Median	Rows	Mean	Standard Deviation	Median
1	GV_CURBWGT	20204	1617.26	393.57	1530	20247	1618	391.6	1530
2	GV_DVLAT	14049	0.04	13.02	0	20247	0.05	13	0

3	GV_DVLONG	14049	-14.76	17.66	-15	20247	-14.97	18.66	-16
4	GV_ENERGY	14049	505.24	645.74	306	20247	495.2	657.6	291
5	GV_LANES	20244	3.28	1.36	3	20247	3.28	1.23	3
6	GV_MODELYR	20247	2003.62	2.77	2003	20247	2004	2.89	2003
7	GV_OTVEHWGT	18147	1630.16	411.35	1550	20247	1626	411.4	1550
8	GV_SPLIMIT	20016	40.73	11.24	40	20247	40.73	11.14	40
9	OA_AGE	20190	40.17	17.37	37	20247	40.16	18.1	37
10	OA_HEIGHT	17508	170.84	10.75	170	20247	171	9.89	170
11	OA_MAIS	19203	0.91	1.04	1	20247	0.89	1	1
12	OA_MANUSE	19774	0.88	0.32	1	19774	0.8847	0.21	1
13	OA_WEIGHT	17599	78.72	19.64	77	20247	78.42	17	77
14	VE_ORIGAVTW	20014	154.75	7.66	154	20247	154.8	8	154
15	VE_WHEELBAS	20238	281	28.72	272	20247	282	28.63	281
16	VE_PDOF_TR	18298	152.62	67.51	135	20247	152.5	67	135

3.2 Correlation Check Of Numerical Values

The second step is to analyze the **multi-collinearity effects** in between the numerical variables and **eliminate the highly-correlated** variables from the analysis of **Naive Bayes**. Data analysis tool in Excel has been used to compute the correlation matrix using **Pearson's coefficient** and for better interpretation, the results have been compiled in a correlation chart. Please find below the correlation matrix among all numeric variables:

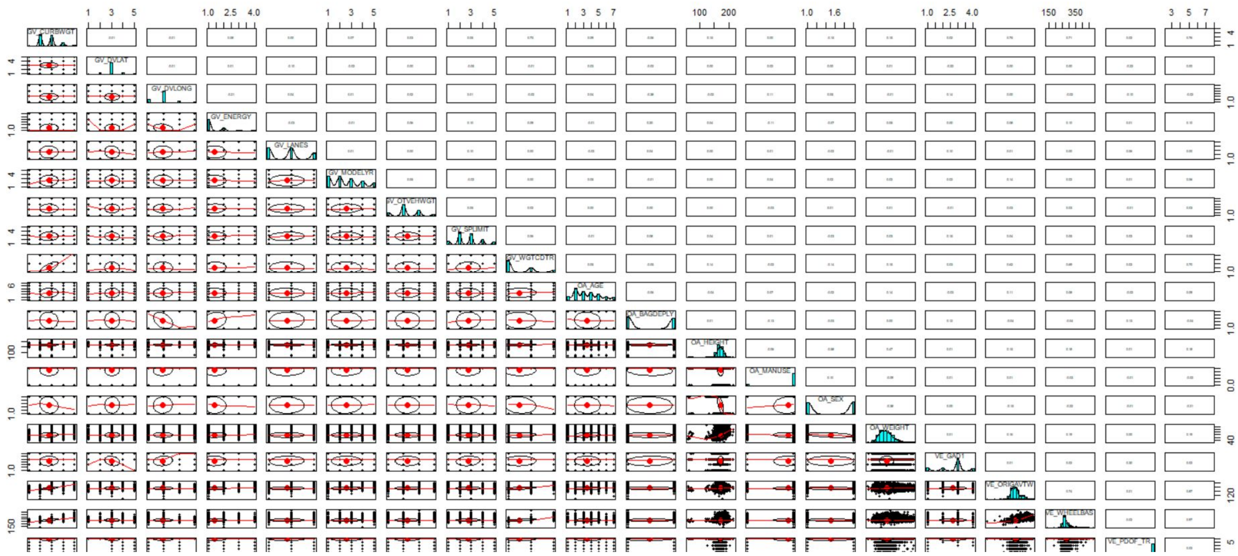
Table 3: Correlation Matrix

	GV_CURBWGT	GV_DVLAT	GV_DVLONG	GV_ENERGY	GV_LANES	GV_MODELYR
GV_CURBWGT	1.00000	0.00409	0.00900	0.08131	0.00803	0.06757
GV_DVLAT	0.00409	1.00000	-0.00546	-0.06049	-0.10239	-0.01338
GV_DVLONG	0.00900	-0.00546	1.00000	-0.29869	0.01331	0.01759
GV_ENERGY	0.08131	-0.06049	-0.29869	1.00000	-0.02264	-0.00658
GV_LANES	0.00803	-0.10239	0.01331	-0.02264	1.00000	0.01223
GV_MODELYR	0.06757	-0.01338	0.01759	-0.00658	0.01223	1.00000
GV_OTVEHWGT	0.02627	-0.00625	0.00483	0.07214	0.00074	0.05581
GV_SPLIMIT	0.05182	-0.05526	-0.02968	0.09981	0.09140	-0.01599
OA_AGE	0.08409	0.02679	0.04399	-0.00560	-0.02206	0.05710
OA_HEIGHT	0.15703	0.00544	-0.02349	0.04305	0.00048	0.00020
OA_MAIS	-0.06111	0.05023	-0.22317	0.36677	-0.03903	-0.06091
OA_MANUSE	-0.01087	0.02307	0.13011	-0.09956	0.01255	0.04177
OA_WEIGHT	0.15777	0.02171	-0.01403	0.05077	-0.00903	0.02259
VE_ORIGAVTW	**0.79453**	0.01123	0.01491	0.08015	0.00685	0.14398
VE_WHEELBAS	**0.76721**	0.00939	-0.01268	0.09766	0.00124	0.01535
VE_PDOF_TR	-0.03066	-0.43379	0.53289	0.01352	0.07727	0.03074
GV_FOOTPRINT	**0.81796**	0.01090	-0.00484	0.09804	0.00170	0.06008

	GV_OTVEHWGT	GV_SPLIMIT	OA_AGE	OA_HEIGHT	OA_MAIS	OA_MANUSE
GV_CURBWGT	0.02627	0.05182	0.08409	0.15703	-0.06111	-0.01087
GV_DVLAT	-0.00625	-0.05526	0.02679	0.00544	0.05023	0.02307
GV_DVLONG	0.00483	-0.02968	0.04399	-0.02349	-0.22317	0.13011
GV_ENERGY	0.07214	0.09981	-0.00560	0.04305	0.36677	-0.09956
GV_LANES	0.00074	0.09140	-0.02206	0.00048	-0.03903	0.01255
GV_MODELYR	0.05581	-0.01599	0.05710	0.00020	-0.06091	0.04177
GV_OTVEHWGT	1.00000	0.04748	0.00465	-0.00106	0.08593	-0.02962
GV_SPLIMIT	0.04748	1.00000	-0.00969	0.03886	0.09343	0.00954
OA_AGE	0.00465	-0.00969	1.00000	-0.04131	0.09509	0.07422
OA_HEIGHT	-0.00106	0.03886	-0.04131	1.00000	-0.03024	-0.06326
OA_MAIS	0.08593	0.09343	0.09509	-0.03024	1.00000	-0.18441
OA_MANUSE	-0.02962	0.00954	0.07422	-0.06326	-0.18441	1.00000
OA_WEIGHT	0.00037	0.03518	0.13197	0.47439	0.05822	-0.08692
VE_ORIGAVTW	0.03137	0.03973	0.11018	0.14532	-0.06106	0.00680
VE_WHEELBAS	0.02830	0.05776	0.07753	0.18233	-0.04687	-0.03264
VE_PDOF_TR	0.03690	0.14054	0.00270	-0.00421	-0.07389	0.04353
GV_FOOTPRINT	0.03011	0.05565	0.09005	0.18049	-0.05424	-0.02295

Contd..	**OA_WEIGHT**	**VE_ORIGAVTW**	**VE_WHEELBAS**	**VE_PDOF_TR**	**GV_FOOTPRINT**
GV_CURBWGT	0.15777	**0.79453**	**0.76721**	-0.03066	**0.81796**
GV_DVLAT	0.02171	0.01123	0.00939	-0.43379	0.01090
GV_DVLONG	-0.01403	0.01491	-0.01268	0.53289	-0.00484
GV_ENERGY	0.05077	0.08015	0.09766	0.01352	0.09804
GV_LANES	-0.00903	0.00685	0.00124	0.07727	0.00170
GV_MODELYR	0.02259	0.14398	0.01535	0.03074	0.06008
GV_OTVEHWGT	0.00037	0.03137	0.02830	0.03690	0.03011
GV_SPLIMIT	0.03518	0.03973	0.05776	0.14054	0.05565
OA_AGE	0.13197	0.11018	0.07753	0.00270	0.09005
OA_HEIGHT	0.47439	0.14532	0.18233	-0.00421	0.18049
OA_MAIS	0.05822	-0.06106	-0.04687	-0.07389	-0.05424
OA_MANUSE	-0.08692	0.00680	-0.03264	0.04353	-0.02295
OA_WEIGHT	1.00000	0.15558	0.18668	-0.00706	0.18684
VE_ORIGAVTW	0.15558	1.00000	**0.73978**	-0.02582	**0.87237**
VE_WHEELBAS	0.18668	**0.73978**	1.00000	-0.03040	**0.96653**
VE_PDOF_TR	-0.00706	-0.02582	-0.03040	1.00000	-0.03035
GV_FOOTPRINT	0.18684	**0.87237**	**0.96653**	-0.03035	1.00000

Figure 2: Correlation Matrix in R



The results of **correlation matrix** show that **GV_CURBWGT, GV_FOOTPRINT, VE_ORIGAVTW** and **VE_WHEELBAS** have **high correlation** among themselves. Hence, we decided to keep the **GV_CURBWGT** and eliminate the rest of the variables.

4 Feature Engineering

Before proceeding with classification using Bayesian networks, we used Range values to **bin** the continuous variable in R.

Table 4: Variables Binning

Variable	**Datatype**	**No. of Bins**	**Binning Methodology**
GV_WGTCDTR	Categorical	3	Already categorized in dataset
OA_BAGDEPLY	Categorical	2	Already categorized in dataset
OA_SEX	Categorical	2	Already categorized in dataset
VE_GAD1	Categorical	4	Already categorized in dataset
OA_MAIS	Categorical	4	Intuitive, divided into 4 injury levels
GV_MODELYR	Numerical	5	Intuitive, uniform distribution of years
GV_CURBWGT	Numerical	5	Binning based on range using R
GV_DVLAT	Numerical	5	Binning based on range using R
GV_DVLONG	Numerical	4	Binning based on range using R
GV_ENERGY	Numerical	4	Binning based on range using R
GV_LANES	Numerical	3	Intuitive, division of lanes
GV_OTVEHWGT	Numerical	4	Binning based on range using R
GV_SPLIMIT	Numerical	5	Binning based on range using R
OA_AGE	Numerical	7	Binning based on range using R

VE_PDOF_TR	Numerical	6	Binning based on range using R

The details of binning process of different numerical variables is described below:

OA_MAIS (Class Variable)	Divided in to: • **Minor injury:** Class 0 and Class 1 (negligible probability of death) • **Moderate Injury:** Class 2 (less probability of death (1-10%)) • **Severe Injury:** Class 3 and Class 5 (medium probability of death (5-50%)) • **Death:** Class 5 and Class 6 (No chance of Survival)
GV_WGTCDTR	Divided in to: ‘**Passenger Car**’, ‘**Truck (<=6000 lbs.)**’ and ‘**Truck (<=10000 lbs.)**’
OA_BAGDEPLY	Divided into ‘**Deployed**’ and ‘**Not Deployed**’
OA_SEX	Divided into ‘**Male’** and ‘**Female**’
VE_GAD1	Separate groups are created for **‘Left**’, ‘**Right**’, ‘**Rear**’ and ‘**Front**’
GV_MODELYR	Separate groups are created for cars manufactured before 2002, between 2002 and 2008 and post 2008
GV_CURBWGT	Separate groups are created for vehicle below 800 kg, between 800 to 1400 kg, between 1400 to 2000 kg, between 2000 to 2600 kg and more than 2600 kg
GV_DVLAT	Separate groups are created for speed: less than -20 kmph, between -20 to 0 kmph, between 0 to 20 kmph and more than 20 kmph
GV_DVLONG	Separate groups are created for speed less than -20 kmph, between -20 to 0 kmph, between 0 to 20 kmph and more than 20 kmph
GV_ENERGY	Separate groups are created for values less than 600, between 600 to 1200, between 1200 to 1800 and more than 1800
GV_LANES	Separate groups are created for values less than 3, between 3 to 5 and more than 5.
GV_OTVEHWGT	Separate groups are created for values less than 1000,

	between 1000 to 1500, between 1500 to 2000 and more than 2000
GV_SPLIMIT	Separate groups are created for values less than 30, between 30 to 45, between 45 to 60 and more than 60
OA_AGE	Separate groups are created for values less than 25, between 25 to 35, between 35 to 50, between 50 to 60 and more than 60
VE_PDOF_TR	Separate groups are created for values less than 25, between 25 to 35, between 35 to 45, between 45 to 55 and more than 55

5 MODEL BUILDING

While the business problem is to build and compare the Bayesian networks to predict likely injury level of the vehicle occupants, we used both **"R"** and **Genie** to build Naïve Bayes model, and **only Genie for Tree Augmented Naïve Bayes(TAN)**. Output in **"R"** were not in line with Genie output. Hence, we decided to compare both the models based on Genie output only. "R" algorithm serves as an extended learning to build Naïve Bayes programmatically.

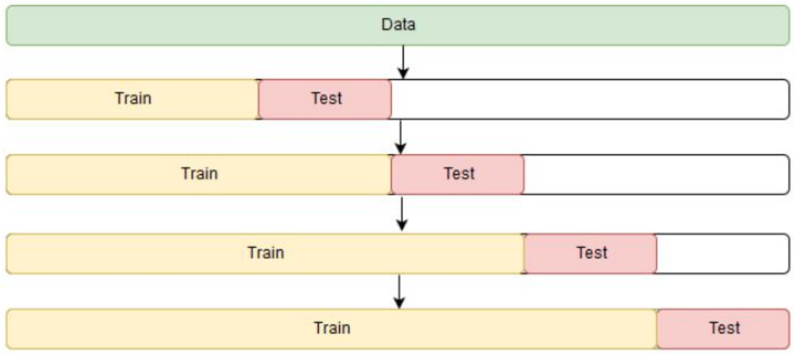
5.1 NAIVE BAYES MODEL (IMPLEMENTED USING R)

The Naive Bayes model is a special form of Bayesian network. This model is mainly used for classification problems. The important feature of Naive Bayes model is that, it has very strong independence assumptions. The final dataset **after imputation** and **binning** is used for modelling.

5.1.1 Training and Validation Set

In a k-fold cross validation, the original data set is divided into k equal parts. Out of those k-parts, one part of the dataset is used for validation and the remaining k-1 parts are used for training the classifier. This process is repeated k-times and each of the parts is used as validation data, exactly once.
The advantage of this method is that it ensures each instance in the dataset is used both, as a training and validation sample and every instance is used exactly once as a validation sample. Here, 4-fold cross validation has been used as shown in figure below:

Figure 3: 5-Fold Cross Validation



5.1.2 Accuracy

Overall accuracy being **82.6%** while the accuracy of individual classes has been shown in the confusion matrix below.

5.1.3 Confusion Matrix

The model can predict the minor injuries and moderate injury with good accuracy and has low accuracy for predicting deaths.

Table 5: Confusion Matrix

Prediction	1	2	3	4
Minor (1)	4829	362	219	37
Moderate (2)	122	28	43	3
Severe (3)	138	58	93	25
Death (4)	45	18	37	17

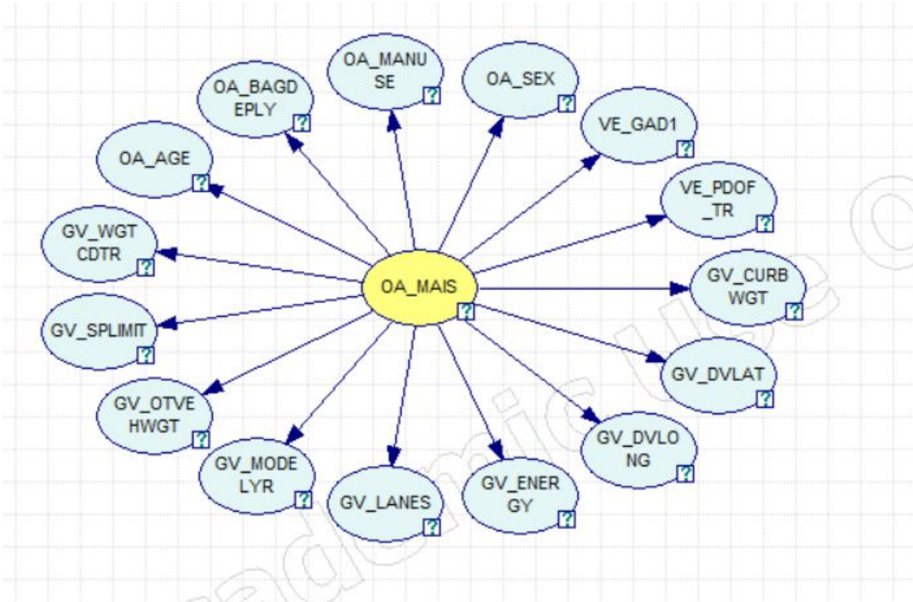
5.2 Naive Bayes Model (Implemented Using Genie)

The same dataset as prepared for R has been used in this model to compare the network within the same tool periphery as TAN to make sure R apis are not influencing the results due to unforeseen default values.

5.2.1 Network developed in Genie

The network is as show below:

Figure 4: Network in Genie



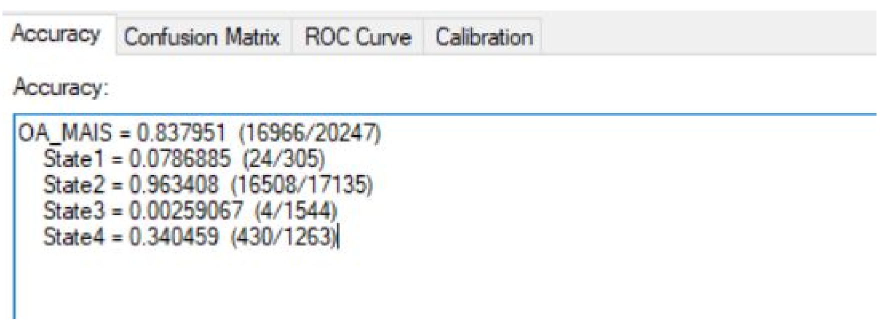
5.2.2 Training and Validation Set

5-fold Cross validation approach was used for training and validating the model.

5.2.3 Results

5-fold validation process was used in Geniei -> validate network option and shown **accuracy of 83.79%** overall. The model is found to be predicting the moderate and death prediction well but not minor (class 1) and Severe (class 3).

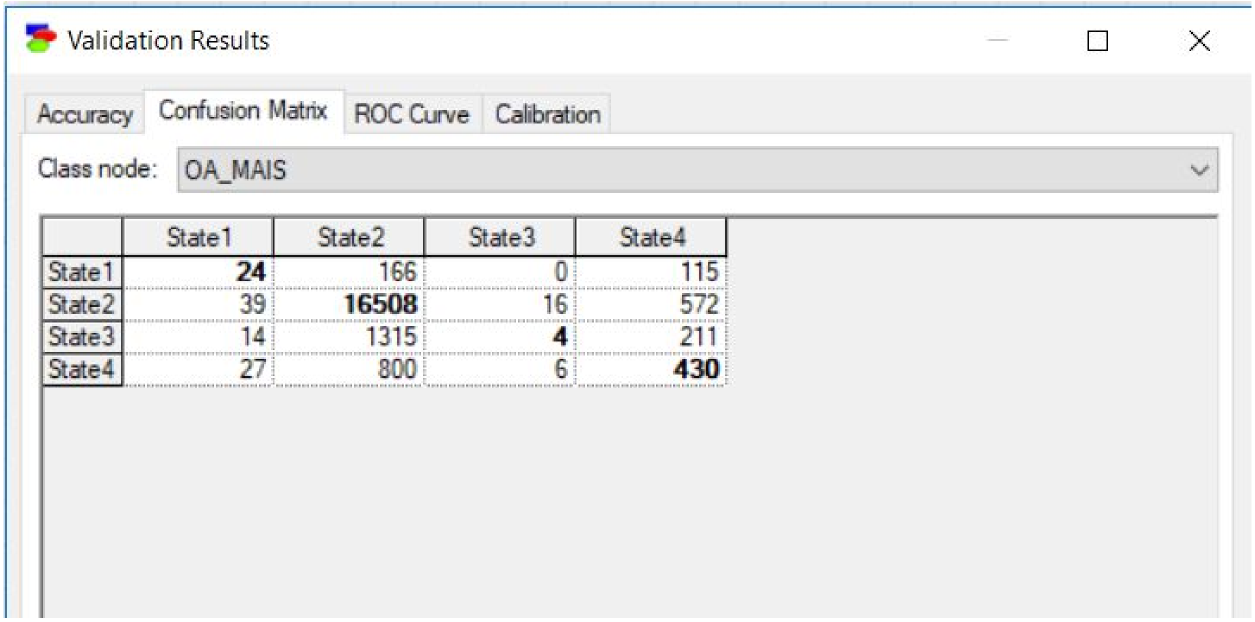
Figure 5: Accuracy shown in Genie



5.2.4 Confusion Matrix

As we can see the State 3 (class 3) has very less representation. This shows signs of data balancing issue and hence our model seems to be overfitted. To overcome this issue oversampling will be recommended to increase the representation of classes with less values. Overall State2 which is **Moderate injury** is predicted with high accuracy. This class has 1-10% of death chance.

Figure 6: Confusion Matrix

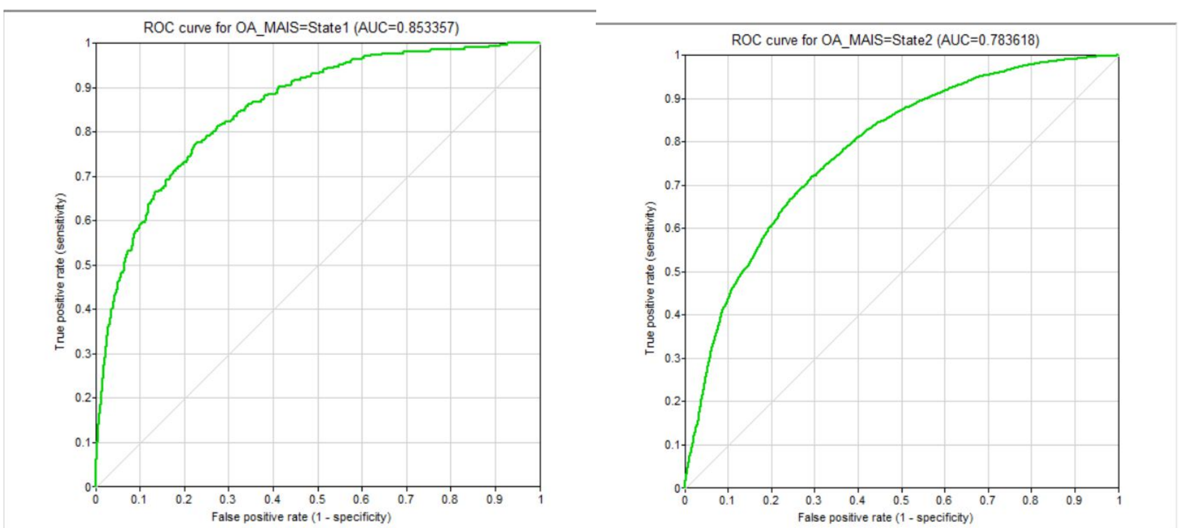


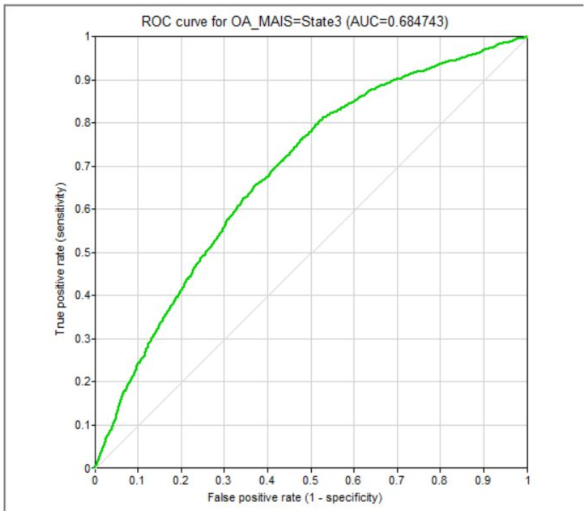
	State1	State2	State3	State4
State1	**24**	166	0	115
State2	39	**16508**	16	572
State3	14	1315	**4**	211
State4	27	800	6	**430**

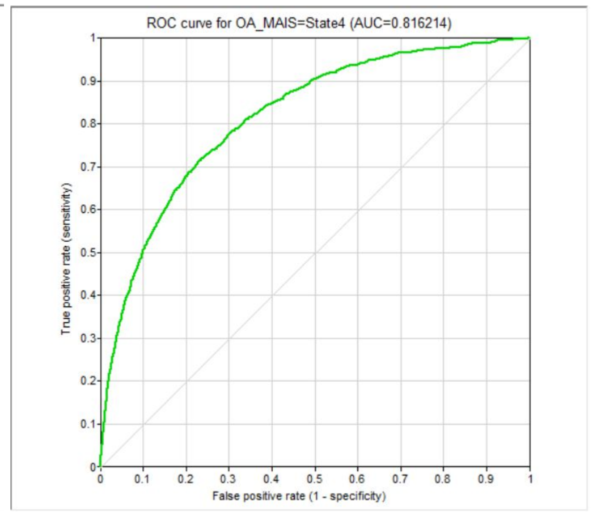
5.2.5 ROC Curves

Based on the ROC curve, we can see that area of curve for State1, State2, and State4 is good shows the high accuracy of tests for these states.

Figure 7: ROC Curves







5.3 TAN Model (Implemented Using Genie)

To compare the performance of Naïve Bayes we built Tree Augmented Naïve Bayes network using Genie. Data used was same as processed above using R.

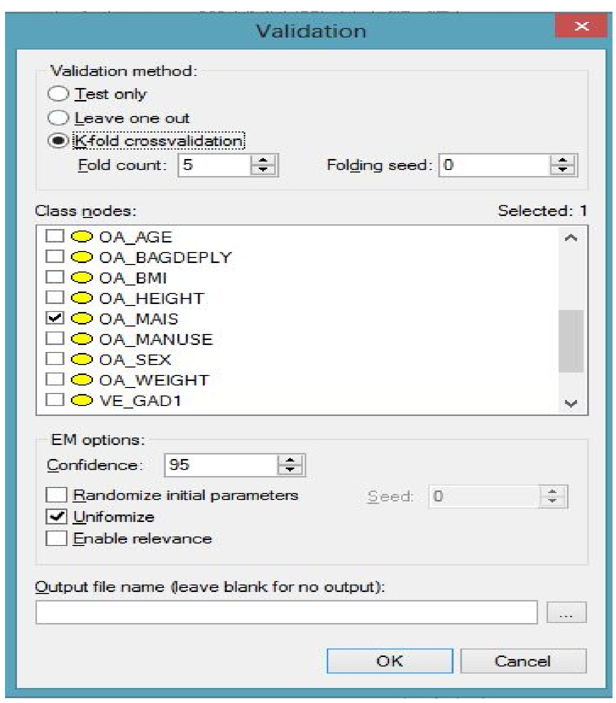
5.3.1 Training and Validation Set

Same 5-fold validation process used.

5.3.2 Results

As shown below, we have used **95% confidence interval** for the validation.

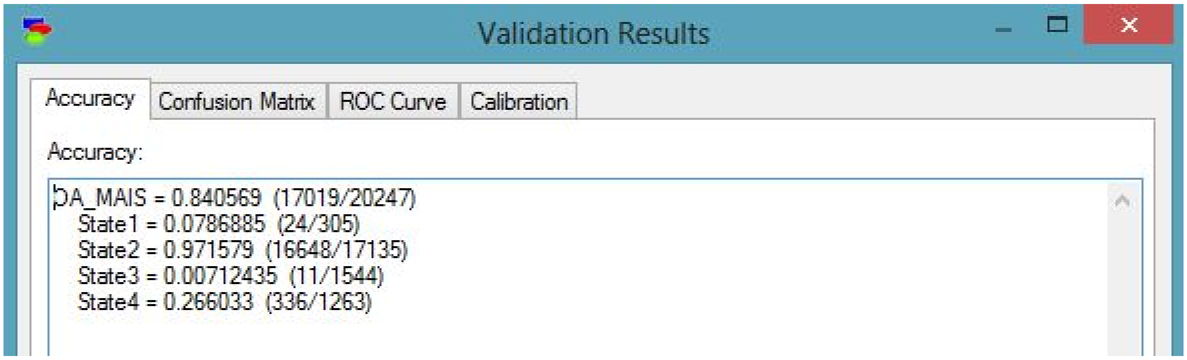
Figure: 8 – Validation Results GENIE Tool



5.3.3 Accuracy

We could achieve overall accuracy of **84%**. Individual class prediction accuracy is as shown below.

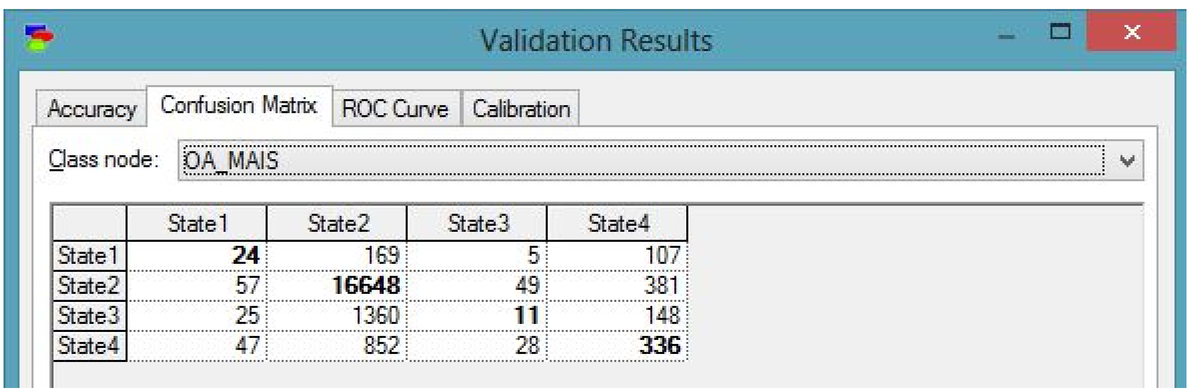
Figure 9: Validation Results



5.3.4 Confusion Matrix

Below figure shows confusion matrix:

Figure 10: confusion Matrix TAN

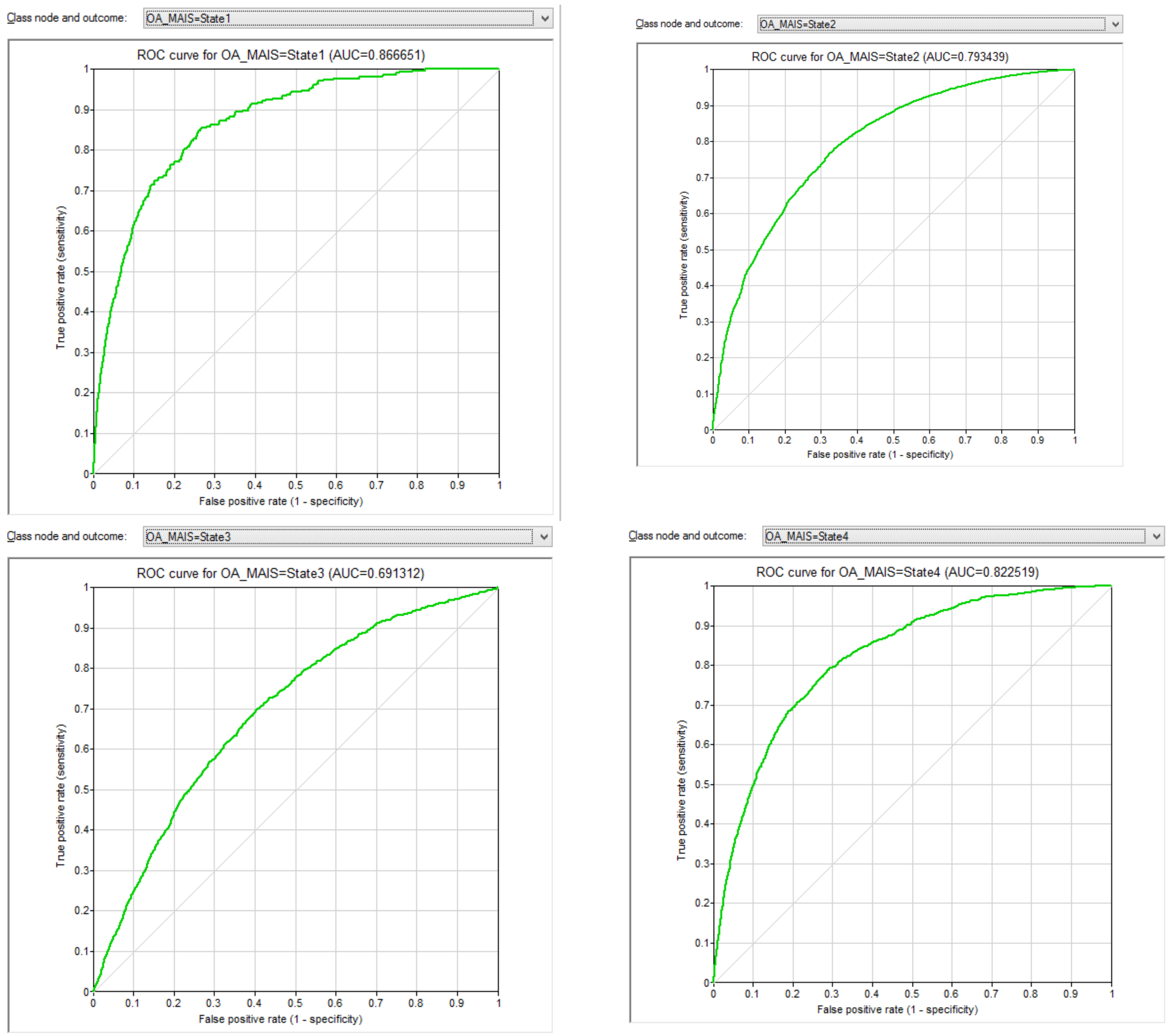


	State1	State2	State3	State4
State1	**24**	169	5	107
State2	57	**16648**	49	381
State3	25	1360	**11**	148
State4	47	852	28	**336**

5.3.5 ROC Curve

Below figure shows the ROC curve for the TAN model for each class. Graph shows the trade-off between the rate at which something is predicted correctly with the rate of False positive rate

Figure 11: ROC Curve

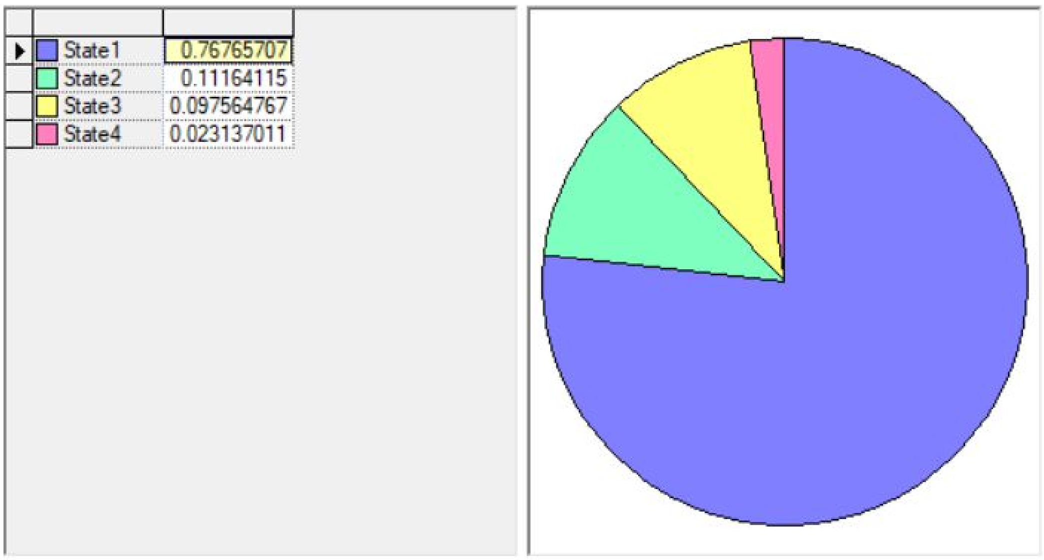


6 Further Network Analysis

6.1 Bag Deployed Vs Not Deployed

If the safety bag was deployed during accident, it is found that **76% chances** is for the injury be in State 1 which is Minor injury. This reinforce the idea that every vehicle should be equipped with safety bag.

Figure 12: Update Belief based on Evidence – Bag Deployed



7 References

- Lecture Notes
- Online Learning resources