

**UK
ROAD
ACCIDENT
STATISTICS
DATA
WAREHOUSING**



1.Introduction

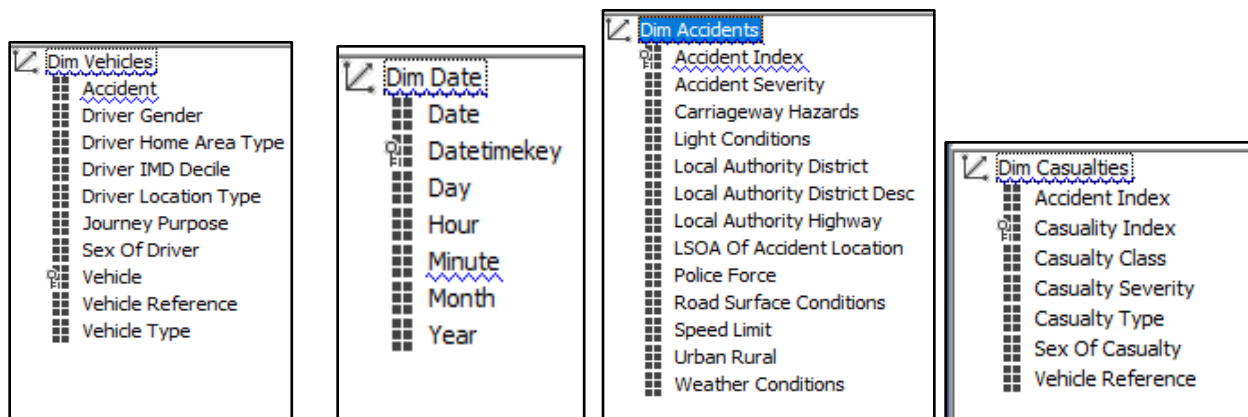
Car crashes are scary, and they happen a lot much more than they should. While some accidents are simple rear-end collisions, a lot of accidents that occur are a much bigger deal.

The Department for Transport has some really interesting data which helps public people to know the most important causes of fatal car accidents. Given the importance of human lives, the purpose of this project is to explore and gain a better understanding of some of the factors that affect the number of vehicle crashes by querying the designed UKaccidentscube.

2.Cube Building and Deployment steps

2.1. Dimension Structure

We have created multidimensional structure using UK accidents database. Dimensions are based on various things related to accidents like vehicles involved in the accident, date when accidents occurred, accidents details and casualty's information. Also, we have created some hierarchies which would help in data processing and properties of dimensions are modified carefully. We have created many MDX queries to provide deeper insights about the different factors involved in accident.

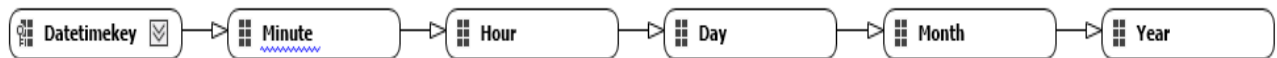
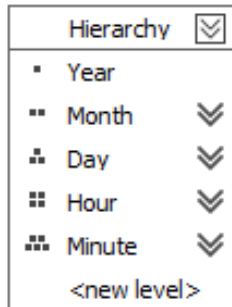


2.2. Hierarchy Structure

We have created Date, Accidents and Vehicles hierarchies.

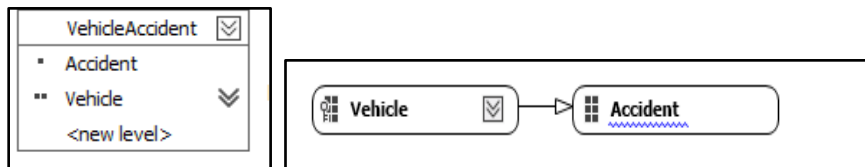
Date hierarchy:

We have created date hierarchy which consists of attributes Year, Month, day, Hour and Minute.



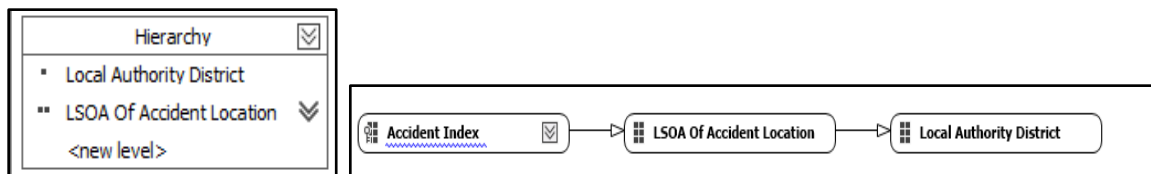
Vehicle hierarchy:

We have created vehicle hierarchy which consists of attributes vehicle and accident.

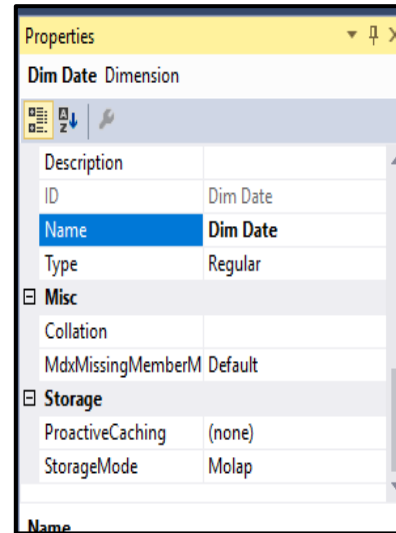
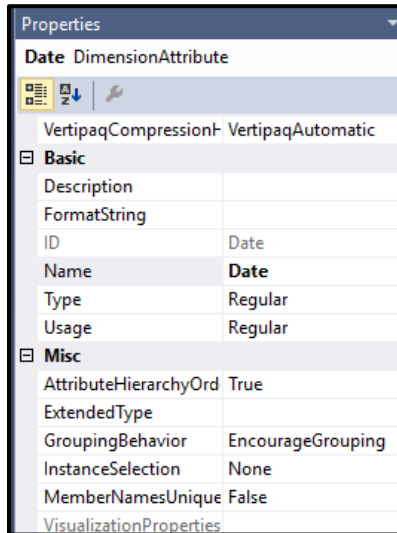


Accidents hierarchy:

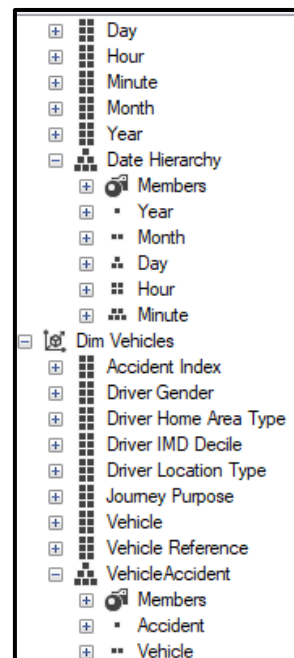
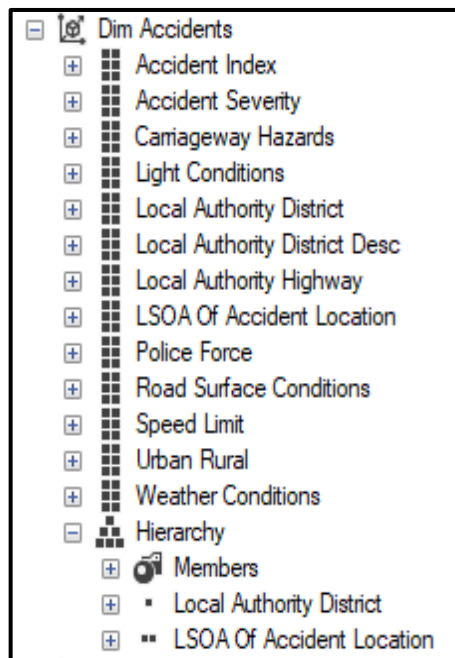
We have created accidents hierarchy which consists of local authority district and LSOA of accident location.



We have also changed properties of dimension according to hierarchy requirement.

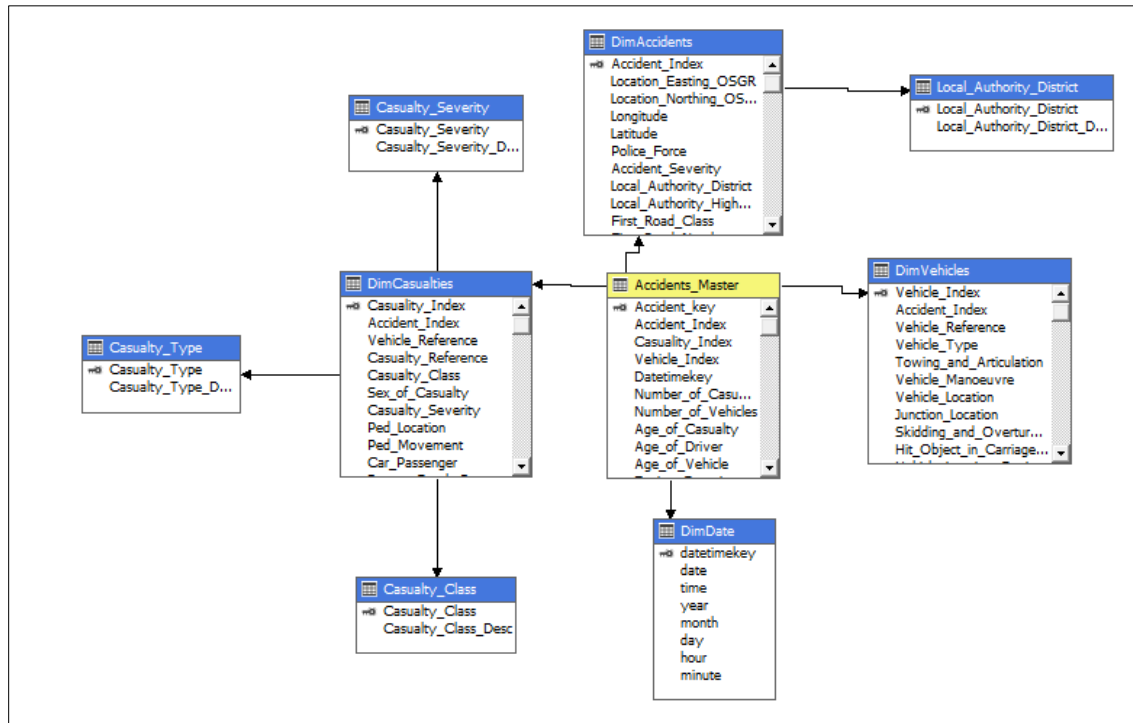


Following screenshot shows the SSMS hierarchy structure which we got after cube deployment.



2.3. Cube Structure

This is a cube structure which consists of Accident Master Fact table and casualties, accidents, vehicles, date dimensions.



2.4. Calculated Measures

We have created named calculations based on the numerical measures Number_of_Casualties, Number_of_Vehicles, Age_of_casualty, Age_of_driver, Age_of_vehicle_, and Engine_Capacity_CC in the Accidents Master facts table.

Average age of vehicle measure:

We have calculated **average age of vehicle** using age of vehicle and number of vehicles.

Name: [Average Age of Vehicle]

Parent Properties

Parent hierarchy: Measures

Parent member: [Change]

Expression

[Measures].[Age Of Vehicle]/[Measures].[Number Of Vehicles]

Additional Properties

Format string: [v]

Visible: True

Non-empty behavior: [v]

Associated measure group: Accidents Master

Display folder: [v]

Color Expressions

Font Expressions

Average age of casualty measure:

We have calculated **average age of casualty** using age of casualty and number of casualties.

Name: [Average Age Of Casualty]

Parent Properties

Parent hierarchy: Measures

Parent member:

Expression

[Measures].[Age Of Casualty]/[Measures].[Number Of Casualties]

Additional Properties

Format string: "#,##0.00;-#,##0.00"

Visible: True

Non-empty behavior:

Associated measure group: Accidents Master

Display folder:

Color Expressions

Font Expressions

Casualty per vehicle measure:

We have calculated **casualty per vehicle** using number of casualties and number of vehicles.

Name: [Casualty Per Vehicle]

Parent Properties

Parent hierarchy: Measures

Parent member:

Expression

[Measures].[Number Of Casualties]/[Measures].[Number Of Vehicles]

Additional Properties

Format string: "#,##0.00;-#,##0.00"

Visible: True

Non-empty behavior:

Associated measure group: Accidents Master

Display folder:

Color Expressions

Font Expressions

CountHighway:

We have calculated **CountHighway** using distinct count of Local Authority Highway

Name: [CountHighway]

Parent Properties

Parent hierarchy: Measures

Parent member: [Change]

Expression

DistinctCount([Dim Accidents].[Local Authority Highway].members)

Additional Properties

Format string: "#,##0.00;-#,##0.00"

Visible: True

Non-empty behavior:

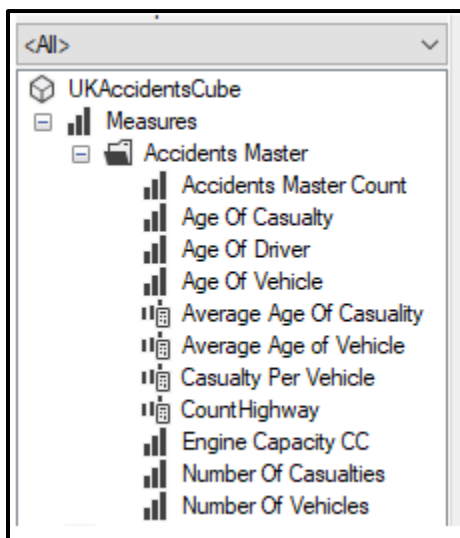
Associated measure group: Accidents Master

Display folder:

Color Expressions

Font Expressions

Following screenshot shows the SSMS calculated measure structure which we got after cube deployment.



2.5. Partitions

For partitions we split the data source into three groups based on date:

1. 1st Partition - 2005-2008
2. 2nd Partition - 2009-2012
3. 3rd Partition - 2013-2015

In this setup, we will be able to keep historical data which is unchanging in three separate partitions that consists of years from (2005-2015)

Partition Name	Source	Estimated Rows	Storage Mode	Aggregation Design
1 Accidents Master 200...	SELECT [dbo].[Accidents_Master].[Accident_...	1950724	MOLAP	AggregationDesign30P...
2 Accidents Master 200...	SELECT [dbo].[Accidents_Master].[Accident_...	1601684	MOLAP	AggregationDesign30P...
3 Accidents Master 201...	SELECT [dbo].[Accidents_Master].[Accident_...	1098451	MOLAP	AggregationDesign30P...

2.6. Aggregations

With aggregations, we try different percentages for performance improvement. We found that at 70%, the cube does not create as many aggregations and as far as resources and processing times are concerned, we could afford to keep this percentage.

Aggregations	Estimated Partition Size	Partitions
2	1950724	Accidents Master 2005-2008
2	1601684	Accidents Master 2009-2012
2	1098451	Accidents Master 2013-2015

3.Exploratory Search

3.1. What Is the Distribution of Injury Severity by Different Factors?

Identifying the potential factors contributing to the injury severity of individuals involved in car accidents is of special importance in the safety and insurance industries. Having that said, in the queries below, we have particularly focused on car accidents to identify the distribution of injury severity in car occupants based on the different factors.

3.1.1. What is the breakdown of all accidents by the injury severity?

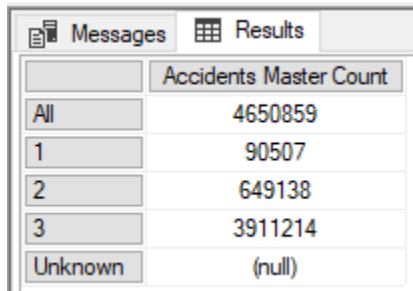
SELECT

[Measures]. [Accidents Master Count] ON COLUMNS,

[Dim Accidents]. [Accident Severity]. [Accident Severity].allmembers

ON ROWS

FROM [UKAccidentsCube]



The screenshot shows a SQL Server Enterprise Manager window with a 'Messages' tab and a 'Results' tab. The 'Results' tab displays a table with two columns: 'Accident Severity' and 'Accidents Master Count'. The data is as follows:

Accident Severity	Accidents Master Count
All	4650859
1	90507
2	649138
3	3911214
Unknown	(null)

According to the query output, the majority of accident in this dataset led to **slight injuries (3)** followed by **serious injuries (2)**. The smallest group of injuries are the **Fatal (1)** ones.

3.1.2. What is the frequency of Fatal (1) accident with Male and Female drivers? Is there any obvious difference based on the gender?

SELECT

{[Measures]. [Accidents Master Count]}

} ON 0,

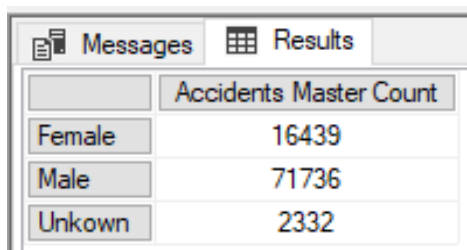
[Dim Vehicles]. [Driver Gender]. [Driver Gender] ON 1

FROM

[UKAccidentsCube]

WHERE

([Dim Accidents]. [Accident Severity]. & [1])



The screenshot shows a SQL Server Enterprise Manager window with a 'Messages' tab and a 'Results' tab. The 'Results' tab displays a table with two columns: 'Driver Gender' and 'Accidents Master Count'. The data is as follows:

Driver Gender	Accidents Master Count
Female	16439
Male	71736
Unknown	2332

As it is obvious from the query result, in the 11 years time span from 2005 to 2015, the number of Fatal accidents with Male drivers are four times the number of Fatal accidents with Female drivers. This output could be a clue to do statistical test significance.

3.1.3. What is the number of Fatal (1) accident with at least one male driver above 60 years old is involved?

SELECT

{[Measures]. [Age of Driver]]On Columns,

```

FILTER ([Dim Accidents]. [Accident Index]. [Accident Index]),
[Measures]. [Age of Driver]>=60 and [Measures]. [Age of Driver]<=90) On Rows
FROM [UKAccidentsCube]
Where [Dim Accidents]. [Accident Severity]. &[1]

```

There are 3934 such accidents.

Note: Since we realize that there are **anomalies** in the Age of Driver attribute, many drivers with more than 100 years old, we decided to put 90 as the upper boundary.

	Age Of Driver
200501TA00087	76
200501TA00106	75
200501TA00270	75
200501TA09001	83
200501TB00106	60
200501TB00180	60
200501TC00030	64
200501TC00221	83
200501TC00285	67
200501TC00303	67
200501TC00350	75

```

Messages Results
Executing the query ...
Obtained object of type: Microsoft.AnalysisServices.AdomdClient.CellSet
Formatting.
Cell set consists of 3934 rows and 2 columns.
Done formatting.
Run complete

```

3.1.4. Which combination of road surface condition and Light Condition led to the most Fatal accident

```

SELECT

```

```

{{[Measures]. [Accidents Master Count]}} ON 0,

```

```

NON EMPTY

```

```

Crossjoin (

```

```

{{[Dim Accidents]. [Road Surface Conditions]. [Road Surface Conditions]}- {[Dim
Accidents]. [Road Surface Conditions]. & [-1]}- {[Dim Accidents]. [Road Surface
Conditions]. [All]. UNKNOWNMEMBER}

```

```

,

```

```

{{[Dim Accidents]. [Light Conditions]. [Light Conditions]}- {[Dim Accidents]. [Light
Conditions]. & [7]}- {[Dim Accidents]. [Light Conditions]. [All]. UNKNOWNMEMBER}

```

```

) On 1

```

FROM [UKAccidentsCube]

Where [Dim Casualties]. [Casualty Severity]. & [1]

Messages		Results
		Accidents Master Count
1	1	20755
1	4	4737
1	5	162
1	6	4466
2	1	6469
2	4	3232
2	5	110
2	6	3696
3	1	81
3	4	27
3	6	45
4	1	381
4	4	138
4	5	2
4	6	250
5	1	36
5	4	22
5	5	4
5	6	29

Using the data dictionary we realized that there are **Missing** and **Unknown** values in the both attribute, Surface and weather condition. To exclude them The **Except function** was used. Then, using the Crossjoin function we got the desired result to look at. Surprisingly, most Fatal accident were happened in the daytime and on the Dry road condition. One interpretation could be, drivers' perception that they could drive on high-speed in that environment which led to more accident.

3.2. Querying UKAccidentCube to answer questions of interest

In this section of document, we applied a variety of MDX queries to see how satisfactorily our cube could answer those. In addition to that, different MDX function was used to test the proper functionality of our cube.

3.2.1. Knowing the LSOA of each accident location, display the allocation of them to each Local Authority District Area to see which Local Authority District Areas has a greater number of LSOA and probably are busier in terms of number of car accidents.

WITH SET [DistrictHighway]

AS

Hierarchize (

{[Dim Accidents]. [Hierarchy]. [Local Authority District]. & [1],

[Dim Accidents]. [Hierarchy]. [Local Authority District]. & [104],

[Dim Accidents]. [Hierarchy]. [Local Authority District]. & [101]

[Dim Accidents]. [Hierarchy]. [LSOA Of Accident Location]. & [1] & [E01000586],

[Dim Accidents]. [Hierarchy]. [LSOA Of Accident Location]. & [1] & [E01000919],
 [Dim Accidents]. [Hierarchy]. [LSOA Of Accident Location]. & [1] & [E01000946],
 [Dim Accidents]. [Hierarchy]. [LSOA Of Accident Location]. & [1] & [E01004758],
 [Dim Accidents]. [Hierarchy]. [LSOA Of Accident Location]. & [102] & [E01005294],
 [Dim Accidents]. [Hierarchy]. [LSOA Of Accident Location]. & [102] & [E01005151],
 [Dim Accidents]. [Hierarchy]. [LSOA Of Accident Location]. & [104] & [E01005340],
 [Dim Accidents]. [Hierarchy]. [LSOA Of Accident Location]. & [104] & [E01005335])
 SELECT {[Measures]. [Accidents Master Count]} ON COLUMNS,
 [DistrictHighway] ON ROWS
 From [UKAccidentsCube]

	Accidents Master Count
1	31518
E01000586	55
E01000919	69
E01000946	63
E01004758	214
101	12706
E01005151	49
E01005294	95
104	15161
E01005335	132
E01005340	105

Since the dataset provides information about Lower layer Super Output Areas (LSOAs), their location and the Local Authority District to which they belong, we were interested to know the accidents LSOAs along their Local Authority District. Using the **Hierarchize function**, we got the anticipated result so that we could answer question such as, which Local Authority District

3.2.2. Display the application of analytical functions on the Cube.

Local Authority District	LSOA Of Accident Location	Local Authority District...	Number Of Vehicles
1	E01002819	Westminster	3
1	E01002820	Westminster	9
1	E01002826	Westminster	3
1	E01002827	Westminster	10
1	E01002829	Westminster	24
1	E01002832	Westminster	4
1	E01002852	Westminster	239
1	E01002854	Westminster	203
1	E01002859	Westminster	45
1	E01002863	Westminster	13
1	E01002886	Westminster	53
1	E01002891	Westminster	222
1	E01004646	Westminster	118
1	E01004647	Westminster	214
1	E01004648	Westminster	323
1	E01004649	Westminster	205
1	E01004650	Westminster	109
1	E01004651	Westminster	176
1	E01004652	Westminster	72

The screen shot from Visual Studio cube browser shows the relationship among attributes that have been used in the analytic functions below.

```

WITH
MEMBER [Measures]. [MaxDistrict_1_Vehicles] AS
MAX ([Dim Accidents]. [Hierarchy]. [Local Authority District]. & [1].children,
[Measures]. [Number of Vehicles])
MEMBER [Measures]. [MinDistrict_1_Vehicles] AS
MIN ([Dim Accidents]. [Hierarchy]. [Local Authority District]. & [1].children,
[Measures]. [Number of Vehicles])
MEMBER [Measures]. [SumDistrict_1_Vehicles] AS
SUM ([Dim Accidents]. [Hierarchy]. [Local Authority District]. & [1].children,
[Measures]. [Number of Vehicles])
MEMBER [Measures]. [AvgDistrict_1_Vehicles] AS
Round (AVG ([Dim Accidents]. [Hierarchy]. [Local Authority District]. & [1].children,
[Measures]. [Number of Vehicles]), 2)

SELECT
{[Measures]. [MaxDistrict_1_Vehicles], [Measures]. [MinDistrict_1_Vehicles],
[Measures]. [SumDistrict_1_Vehicles], [Measures]. [AvgDistrict_1_Vehicles]} ON
COLUMNS,
[Dim Date]. [Year]. [Year] on rows
FROM [UKAccidentsCube]

```

	MaxDistrict_1_Vehicles	MinDistrict_1_Vehicles	SumDistrict_1_Vehicles	AvgDistrict_1_Vehicles
2005	426	1	5476	44.16
2006	430	1	5763	46.85
2007	434	1	5439	45.32
2008	422	1	5246	43.72
2009	456	1	4932	41.1
2010	406	1	5081	41.65
2011	504	1	5153	41.89
2012	546	1	5742	45.57
2013	458	1	5394	42.47
2014	517	1	5804	44.65
2015	475	1	5758	43.29

3.2.3. Display the number of casualties in each Local Authority District and Rank the Local Authority District based on the number of people affected by car crashes in each of them, show the result from highest to lowest.

As we mentioned before each Local Authority District has many LSOAs. The dataset also has the number of cars which were involved in any car accidents in each LSOA for each particular year. Having all those information, we decided to apply some **analytic functions** such as **Min** and **Max** to find the LSOAs of Westminster Local Authority District with maximum and minimum number of cars involved in the car accidents (the min and max were returned), also **Sum function** was used to provide the total number of cars involved in any car accidents in Westminster. Finally **Round function** was used to reduce the decimals. All of those findings

WITH

SET OrderedDistricts AS Order

([Dim Accidents]. [Local Authority District Desc]. [Local Authority District Desc].members, [Measures]. [Number Of Casualties], BDESC)

MEMBER [Measures]. [Casualty Rank] AS Rank

([Dim Accidents]. [Local Authority District Desc].CurrentMember, OrderedDistricts)

SELECT {[Measures]. [Casualty Rank], [Measures]. [Number Of Casualties]} ON 0,

NON EMPTY (Order ([Dim Accidents]. [Local Authority District Desc]. [Local Authority District Desc].MEMBERS, [Measures]. [Casualty Rank], ASC)) ON 1

FROM [UKAccidentsCube]

	Casualty Rank	Number Of Casualties
Swale	1	353254
Birmingham	2	182032
Leeds	3	134932
Liverpool	4	111562
Taunton Deane	5	105659
Bradford	6	105236
Manchester	7	92104
Kirklees	8	79005
Salford	9	78091
Sheffield	10	77495
Cherwell	11	76727
Glasgow City	12	71691
Doncaster	13	67196
Wakefield	14	64053
County Durham	15	63825
Hertsmere	16	59133
East Riding of Yorkshire	17	57545
North Lincolnshire	18	57039
Cornwall	19	54888
Milton Keynes	20	54772

Using the **Order** and **Rank function** we were able to order the Local Authority Districts based on their number of casualties and also allocate rank to each of those Local Authority Districts. Swale with the most affected people has the Rank 1.

3.2.4. What is the total number of casualties in the last 5 days of 2005? How does it differ from 2015?

Select [Measures]. [Number Of Casualties] on columns,
 LastPeriods (5, [Dim Date]. [Date]. & [2005-12-31T00:00:00]) on rows
 From [UKAccidentsCube]

Select [Measures]. [Number Of Casualties] on columns,
 LastPeriods (5, [Dim Date]. [Date]. & [2015-12-31T00:00:00]) on rows
 From [UKAccidentsCube]

	Number Of Casualties		Number Of Casualties
2005-12-27	3208	2015-12-27	1505
2005-12-28	2826	2015-12-28	914
2005-12-29	1480	2015-12-29	1588
2005-12-30	1937	2015-12-30	1044
2005-12-31	1842	2015-12-31	2408

Using the **LastPeriods** function we are able to see the anticipated outputs and give a sense of comparison.

3.2.5. For each year, show the Month with the heights number of Casualties affected in car crashes, in each location

WITH SET [Highest Casualty Rate] AS
 GENERATE ([Dim Date]. [Date Hierarchy]. [Year].MEMBERS,
 TopCount
 (Descendants ([Dim Date]. [Date Hierarchy].CurrentMember,
 [Dim Date]. [Date Hierarchy]. [Month], SELF), 1, [Measures]. [Number Of Casualties]))
 SELECT
 NON EMPTY {[Highest Casualty Rate]*[Measures]. [Number Of Casualties]} on 0,
 NON EMPTY [Dim Vehicles]. [Driver Location Type]. [Driver Location Type] on 1
 FROM [UKAccidentsCube]

	12	7	10	8	11	9	11	10	9	10	2
	Number Of Ca...	Number Of C...	Number Of Casualties	Number Of Casualties	Number Of Casualties	Number Of Casualties	Number Of Casualties	Number Of Casualties	Number Of Casualties	Number Of Casualties	Number Of Casualties
Rural Area	11262	11322	8582	8383	9068	15157	6454	7667	55242	6991	(null)
Small Town	8944	7821	8133	10232	7259	5645	5826	5169	9784	4779	(null)
Unknown	19145	19078	14683	16770	11360	9727	97486	10134	81484	9002	97431
Urban Area	59143	58276	60915	50055	53213	45379	42786	46486	240337	61430	(null)

By Iterating through Sets using **Generate with Descendants Function** we were able to find each month (Jan=1 to Dec=12) with highest number of casualties in each year (2005 to 2015 from left to right). In addition by breaking down the number on different area we had the anticipated cross table to look at for the purpose of comparison.

3.2.6. Show one application of VehicleAccident user-defined hierarchy.

SELECT {}

ON COLUMNS,

DESCENDANTS

([Dim Vehicles]. [VehicleAccident]. [Accident]. & [200501BS70092])

ON ROWS

FROM [UKAccidentsCube]

200501BS70092
189
190
191

Creating this hierarchy, we can identify all the cars in each accident by using **Descendants function**.

3.2.7. Apply formatting to check if the cube provide the result properly.

WITH MEMBER MEASURES.CELLPROPERTY AS [Measures]. [Accidents Master Count]

, FORE_COLOR=RGB (0, 0,255)

, BACK_COLOR=IIF ([Measures]. [Accidents Master Count]>400000, RGB (255, 0, 0), RGB (0, 255, 0))

, FONT_SIZE=10

, FORMAT_STRING='#, #.000'

SELECT MEASURES.CELLPROPERTY ON 0,

[Dim Date]. [Date Hierarchy]. [Year].MEMBERS ON 1

FROM [UKAccidentsCube]

CELL PROPERTIES VALUE, FORMATTED_VALUE, FORE_COLOR, BACK_COLOR, FONT_SIZE

	CELLPROPERTY
2005	525,546.000
2006	500,310.000
2007	480,505.000
2008	444,363.000
2009	427,676.000
2010	401,198.000
2011	395,395.000
2012	377,415.000
2013	358,094.000
2014	377,091.000
2015	363,266.000

Using the **Formatting** we were able to fill the year with Accident Master Count over 400000 in red. Otherwise it is green

3.2.8. For all calendar years, find the 10 Local Authority District with the lowest accident occurrence.

SELECT

Non empty

BOTTOMCOUNT

([Dim Accidents]. [Local Authority District Desc]. [Local Authority District Desc].**Members**, 10, [Measures]. [Accidents Master Count]
) ON COLUMNS,

[Dim Date]. [Year]. [Year].**Members ON ROWS**

FROM [UKAccidentsCube]

	Orkney Islands	Shetland Islands	Western Isles	Berwick-upon-Tweed	Teesdale	London Airport (Heathrow)	Oswestry	South Shropshire	Alnwick
2005	176	207	301	575	706	157	415	809	657
2006	168	154	198	413	460	44	355	699	826
2007	94	106	173	266	550	234	635	570	643
2008	92	39	404	430	296	135	686	569	547
2009	88	357	107	100	74	117	130	86	122
2010	186	340	129	(null)	(null)	117	(null)	(null)	(null)
2011	115	170	62	(null)	(null)	98	(null)	(null)	(null)
2012	79	82	121	(null)	(null)	87	(null)	(null)	(null)
2013	103	253	46	(null)	(null)	120	(null)	(null)	(null)
2014	99	72	98	(null)	(null)	208	(null)	(null)	(null)
2015	27	79	87	(null)	(null)	139	(null)	(null)	(null)

Using **BOTTOMCOUNT** function we created the desired result.

4. Conclusion

Using the UK Road Accidents dataset as the multidimensional database for this project, all the required steps to create a MDX cube was completed. The deployment was done successfully and a variety of functions were used to test the proper functionality of the cube.

There are quite many good/solid works out there however, this project would be merely another journey at its beginning for us.

1. The Goal of Creating the Mining Models

In this part of the project, we have created a mining structure that allows us to predict whether a potential car occupant (driver or passenger) will get injured while driving. So we spatially are focused on the Accident Severity attribute as the predictable one. Our target attribute has three different states namely 1 for Fatal 2 for Serious and 3 for the Slight type of car accident.

This prediction model building is meant for a policy-making or public education program where the goal is to efficiently create an instruction for the safe driving. Having the different factors related to the car accidents, we could identify the best combination of them (e.g. weather, Light, Speed, Road Surface, etc.) to avoid fatal car accident and bring that knowledge to the public.

2. Creating Mining Structure and the Mining Models

The very first step of this phase was creating a structure named “Accident Severity DMX” upon which all the three mining models were built.

```

CREATE MINING STRUCTURE [Accident Severity DMX]
(
    [Accident Index] LONG KEY,
    [Accident Severity] LONG DISCRETE,
    [Carriageway Hazards] TEXT DISCRETE,
    [Light Conditions] LONG DISCRETE,
    [Local Authority District] LONG DISCRETE,
    [Local Authority Highway] TEXT DISCRETE,
    [LSOA of Accident Location] TEXT DISCRETE,
    [Police Force] TEXT DISCRETE,
    [Road Surface Conditions] TEXT DISCRETE,
    [Speed Limit] LONG DISCRETIZED,
    [Urban Rural] TEXT DISCRETE,
    [Weather Conditions] TEXT DISCRETE
)
WITH HOLDOUT (30 PERCENT or 1000 CASES)

```

Here the mining models were added to the structure by using the ALTER MINING STRUCTURE statement. Using the below queries, we will be able to compare all of the models. Also, since there is no Holdout Seed to ensure that an exact partition is used while we run our models several times, all the models were processed at once to get the fairest and most accurate comparison.

```

ALTER MINING STRUCTURE [Accident Severity DMX]
ADD MINING MODEL [Decision Tree DMX]
(
    [Accident Index],
    [Accident Severity] PREDICT,
    [Carriageway Hazards],
    [Light Conditions],
    [Local Authority District],
    [Local Authority Highway],
    [LSOA of Accident Location],
    [Police Force],
    [Road Surface Conditions],
    [Speed Limit],
    [Urban Rural],
    [Weather Conditions]
) USING Microsoft_Decision_Trees
WITH DRILLTHROUGH

```

```
ALTER MINING STRUCTURE [Accident Severity DMX]
ADD MINING MODEL [Naïve Bayes]
(
  [Accident Index],
  [Accident Severity] PREDICT,
  [Carriageway Hazards],
  [Light Conditions],
  [Local Authority District],
  [Local Authority Highway],
  [LSOA of Accident Location],
  [Police Force],
  [Road Surface Conditions],
  [Speed Limit],
  [Urban Rural],
  [Weather Conditions]
) USING Microsoft_Naive_Bayes
```

```
ALTER MINING STRUCTURE [Accident Severity DMX]
ADD MINING MODEL [Neural_Network]
(
  [Accident Index],
  [Accident Severity] PREDICT,
  [Carriageway Hazards],
  [Light Conditions],
  [Local Authority District],
  [Local Authority Highway],
  [LSOA of Accident Location],
  [Police Force],
  [Road Surface Conditions],
  [Speed Limit],
  [Urban Rural],
  [Weather Conditions]
) USING Microsoft_Neural_Network
```

The code below fulfilled the final task which was training the models using the UK Road Accident data source.

```

INSERT INTO MINING STRUCTURE [Accident Severity DMX]
(
  [Accident Index],
  [Accident Severity],
  [Carriageway Hazards],
  [Light Conditions],
  [Local Authority District],
  [Local Authority Highway],
  [LSOA of Accident Location],
  [Police Force],
  [Road Surface Conditions],
  [Speed Limit],
  [Urban Rural],
  [Weather Conditions]
)
OPENQUERY ([UK Accidents Database],
'SELECT
  Accident_Index,
  Accident_Severity,
  Carriageway_Hazards,
  Light_Conditions,
  Local_Authority_District,
  Local_Authority_Highway,
  LSOA_Of_Accident_Location,
  Police_Force,
  Road_Surface_Conditions,
  Speed_Limit,
  Urban_Rural,
  Weather_Conditions
FROM dbo.DimAccidents')

```

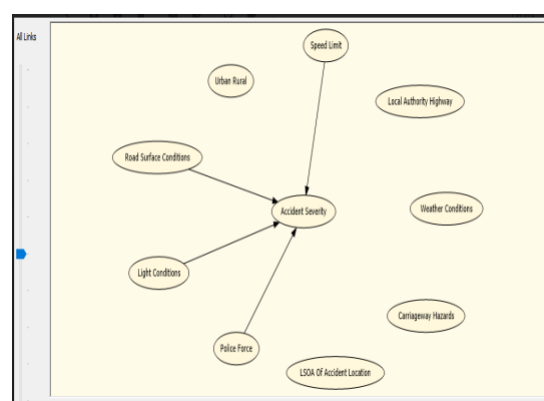
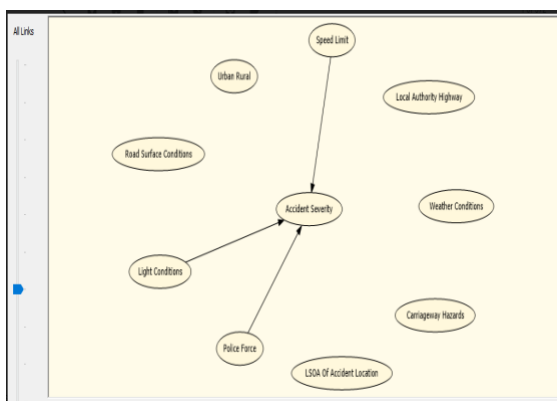
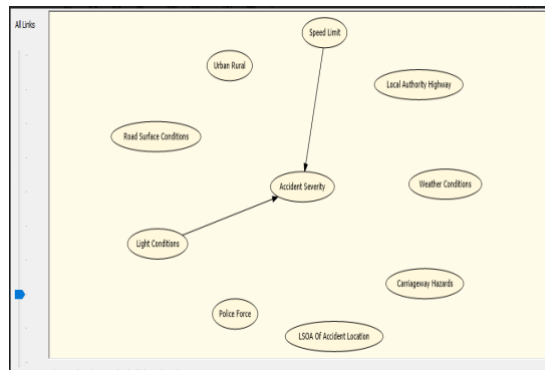
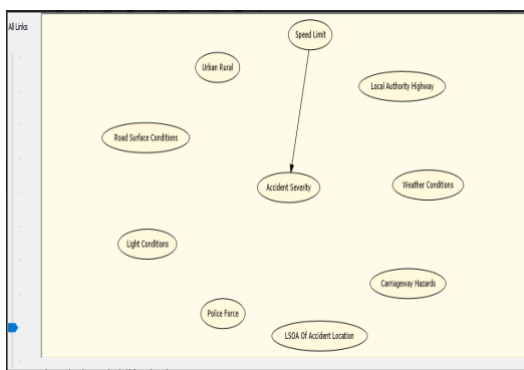
3. Model Interpretation

One of the challenges we were facing in training our models was the imbalance of the dataset in terms of the number of instances of different categories of injury severity. Also, to fix the problem our attempt to merging some levels together didn't work since the level 3 was still the majority of the response column. Since we did not have any direct access to the data to manipulate it and address this issue, we went ahead and trained our prediction models with an unbalanced dataset. We believe this issue was the main reason for the model's tendency to predict most of the cases in the test data as 3 (Fatal Injury). Having that said, we decided to be focused on the prediction accuracy of Fatal Injury (level 3) while comparing the models to find the best one.

3.1. Decision Tree Model

The dependent variable of the decision tree is Accident Severity. Which has three classes, 1(Fatal), 2(Serious) and 3 (Slight)? The root of this tree contains all 1779653 observations in training dataset and it could expand through **14 levels** to show the final partitioning of all the records (i.e. most-end leaves). The most important attribute to determine how to separate different levels of Accident severity is the Speed Limit attribute followed by a Light condition, Urban Rural, Road Surface Condition, and Police Force as the second most important ones which depend on the Speed Limit measure.

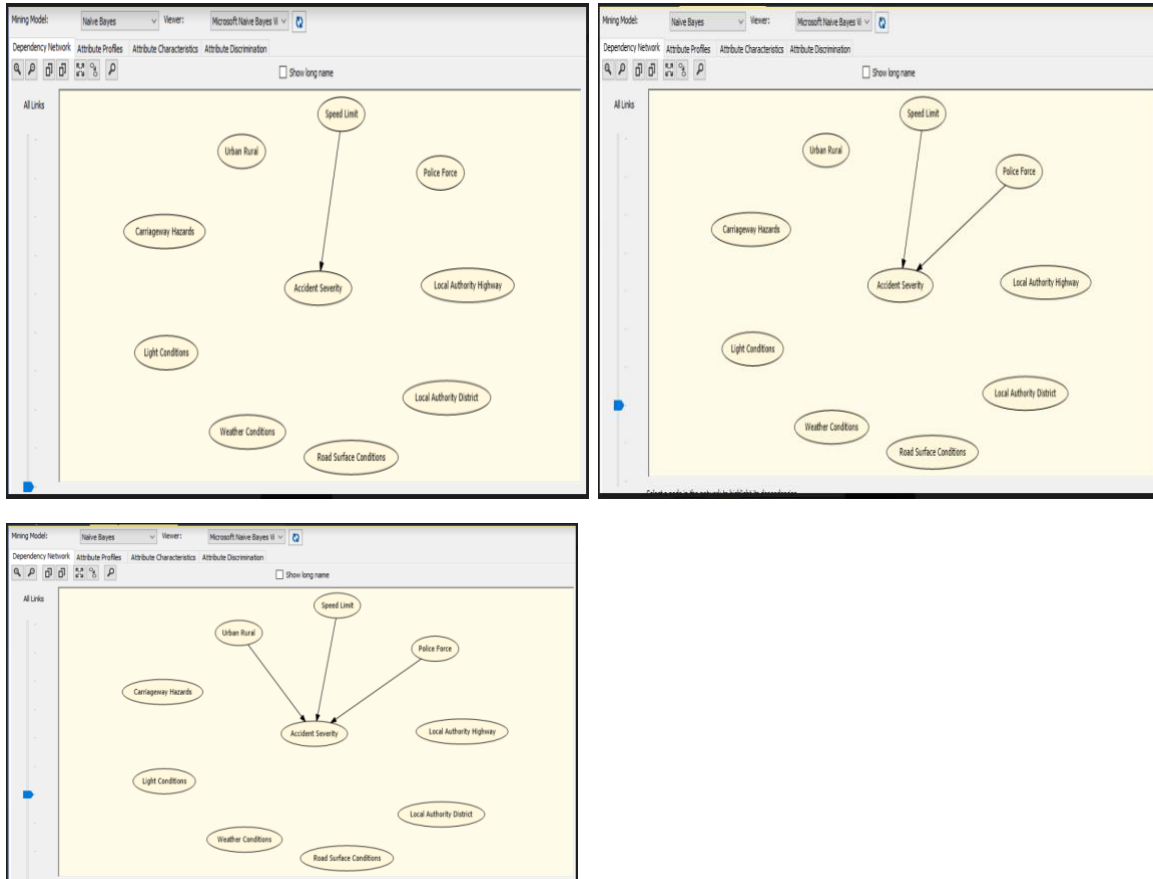
The Speed Limit over 58 seems to cause to the most fatal accident hence if we wanted to create a standard for divers this piece of information could be used as a guide.



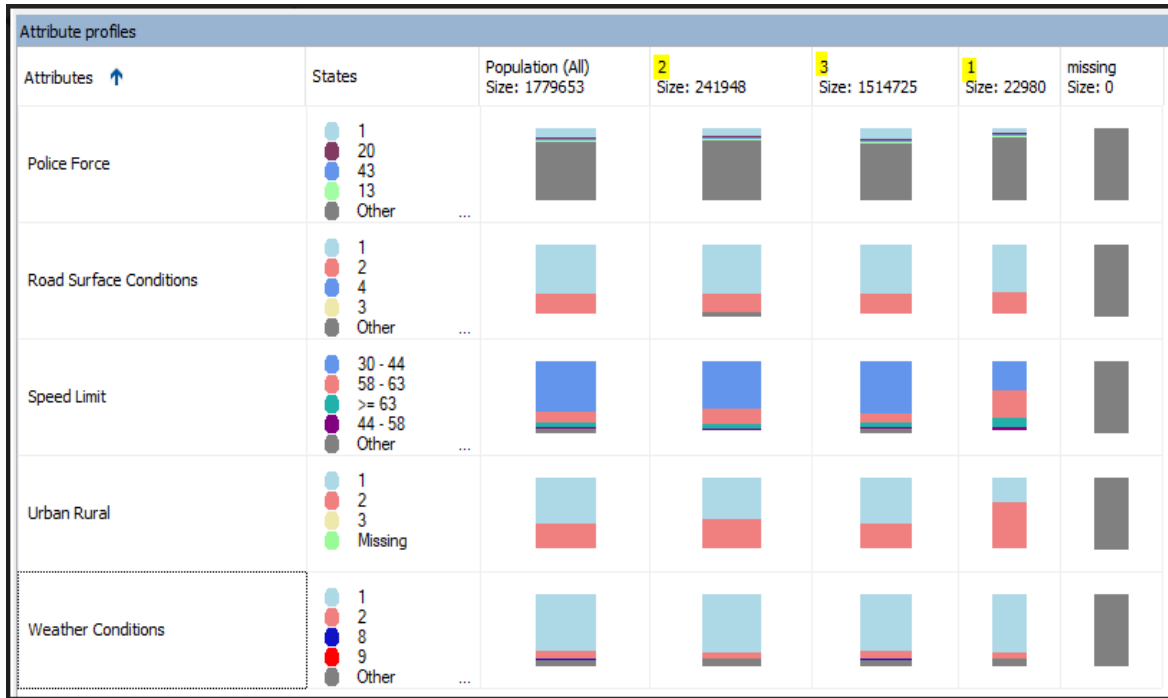
3.2. Naïve Bayes Model

According to the Dependency Network of this mining model, the most important attribute in identifying Accident severity **Speed Limit** followed by **Police Force** and **Urban Rural** variable. The result is in consistency with our Decision Tree model however, the **Road**

Surface Condition variable is the least important one to predict our target variable in this model.

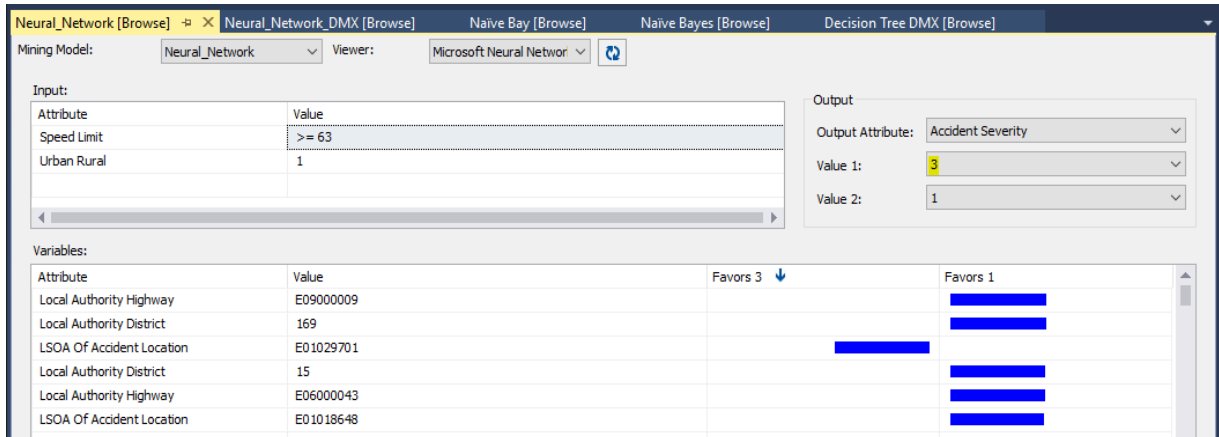


Using the Attribute Profile tool, we realize that in the three categories (Fatal3, Serious2, and slight1), the majority of people have the same distribution regarding the many variables such as weather condition or Surface Condition. However, as the color shown when you look horizontally, the distribution is different between those three categories regarding their Speed Limit, Rural Urban and somewhat the Police Force. And, that is why the algorithm chose those three variables as the basis for the prediction.



3.3. Neural Network Model

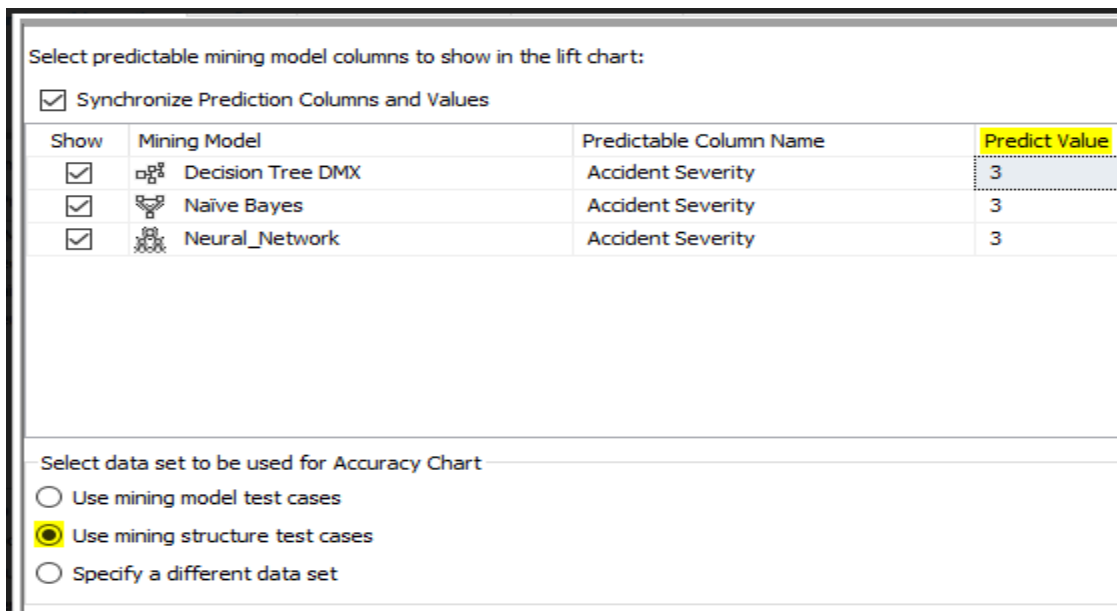
Using a Neural Network mining model, we are able to identify some variables and their values as the input and see how the other variables play roles on the three levels of Accident Severity. Having the general idea from the two previous models and using common sense, we chose Speed Limit >65 and 1 to be specific only on Urban part .in this scenario, it turns out that the LSOA (Lower layer Super Output Areas) play a key role on having or not having Fatal injury such that, in some area those Speed Limit led to fatal injury whereas in some LSOA it is safe to drive with that high speed. The Neural Network Browser Tool provides the ability to get a good insight into the model and effect of input values on the final prediction. The example above was one of our interesting findings.

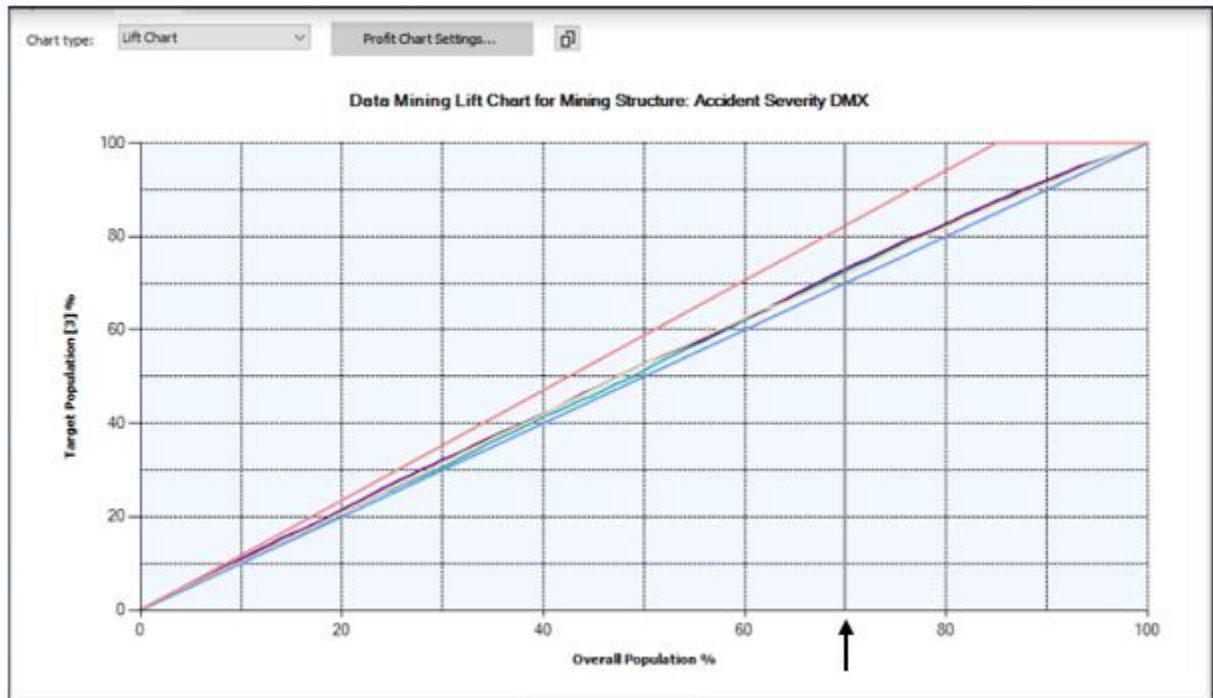


4. Comparison Using Different Criteria

4.1. Lift Chart

As we mentioned before, there are three levels of Accident Severity in our dataset, however, from the business viewpoint, we know that Fatal Accidents are the most important and tragic one among the three. Hence, identifying the best model which could accurately predict the Fatal Accidents is of the special interest. In addition to that, these kinds of accidents create a burden for insurance companies so from the economic viewpoint it worth to be focused on them.





Mining Legend

Population percentage: 70.00%

Series, Model	Score	Target population	Predict probability
Decision Tree DMX	0.89	72.59%	83.73%
Naïve Bayes	0.90	73.06%	80.41%
Neural_Network	0.90	71.76%	82.36%
Random Guess M...		70.00%	
Ideal Model for: D...		82.35%	

As we could see in the Mining Legend table, the Naïve Bayes and Neural Network model with the **Lift Score** of **0.90** outperforms the Decision Tree DMX model. However considering the Target population prediction, the Naïve Bayes perform even better than the neural network.

It is also interesting to consider the **Predict Probability** column since it reports the thresholds for each of the three models. As we could see, Decision Tree is the most conservative one in identifying a new case as a Fatal Accident. Naïve Bayes model has the threshold of 80.41% as the base for making a decision while Neural_Network has selected 82.36%.

The full data set has 1779653 observations with 1514725 Fatal Accident or 85 % of the dataset.

According to the Mining Legend Table:

If we target 70% of the population (1245757 Accidents) with random guess model, we will correctly identify 1060307 Fatal Accident correctly. ($0.70 \times 1514725 = 1060307$)

If we target 70% of the population (1245757 Accidents) with Decision Tree model, we will correctly identify 1099538 Fatal Accident correctly. ($0.7259 \times 1514725 = 1099539$)

If we target 70% of the population (1245757 Accidents) with Neural Network model, we will correctly identify 1085603 Fatal Accident correctly. ($0.7167 \times 1514725 = 1085603$)

If we target 70% of the population (1245757 Accidents) with Naïve Bayes model, we will correctly identify 1106658 Fatal Accident correctly. ($0.7306 \times 1514725 = 1106658$)

As the Mining Legend Table shows, using any of the prediction models is better than relying on the random guess. In fact, if we would like to target 70% of the whole population for an education campaign, by using the Naïve Bayes model and having people driving habits, we could predict whether they would have a Fatal Accident or not. In that case, our claim is 73.6% accurate and it is more accurate than a random guess and the two other models.

4.2. Classification Matrix

Counts for Decision Tree DMX on Accident Severity				
	Predicted	2 (Actual)	3 (Actual)	1 (Actual)
2	2	0	0	0
3	3	132	850	18
1	1	0	0	0

Counts for Naïve Bayes on Accident Severity				
	Predicted	2 (Actual)	3 (Actual)	1 (Actual)
2	2	3	5	0
3	3	129	841	17
1	1	0	4	1

Counts for Neural_Network on Accident Severity				
	Predicted	2 (Actual)	3 (Actual)	1 (Actual)
2	2	0	1	0
3	3	132	849	18
1	1	0	0	0

- Correct Classification Caught by Decision Tree DMX model
 $= (850) / (850 + 18 + 132) = 0.85$ or **85%**
- Correct Classification Caught by Naïve Bayes model
 $= (3 + 841 + 1) / (3 + 5 + 129 + 841 + 17 + 4 + 1) = 0.845$ or **84.5%**
- Correct Classification Caught by Neural Network model
 $= (849) / (1 + 132 + 849 + 18) = 0.849$ or **85%**

By calculating the accuracy measure on the test portion of the dataset, it turns out that **all three models** perform almost the same in terms of **accuracy**. There is a slight difference between Naïve Bayes and the two others, however, it is not noticeable.

4.3. Cross-Validation Approach

The number of false negative cases in each of the prediction models refers to the cases that their accident severity has been wrongly identified (For example a truly fatal accident (3) has been identified as the slight severity accident (1) by mistake). This kind of error, sometimes, could incur more cost to the people than false positive error.

Looking at the average number of false negative cases in each of the mining models below, we realized that Decision Tree model provides us with **Zero** False Negative cases and the Naïve Bayed model has only 5 however, the Neural Network with 189 False Negative occurrences has the worst performance.

Continuing with the two remaining models, Naïve Bayes and Decision Tree, and comparing them regarding the other criteria in the Cross-Validation outputs, we realized that they have almost the same ability in terms of True Positive cases (i.e. identifying fatal accidents correctly).

Fold Count:	3	Max Cases:	1000	Get Results	
Target Attribute:	Accident Severity	Target State:	3	Target Threshold:	0.1

Decision Tree DMX				
Partition Index	Partition Size	Test	Measure	Value
1	333	Classification	True Positive	284
2	333	Classification	True Positive	284
3	334	Classification	True Positive	285
			Average	284.334
			Standard Deviation	0.4716
1	333	Classification	False Positive	49
2	333	Classification	False Positive	49
3	334	Classification	False Positive	49
			Average	49
			Standard Deviation	0.000e+000

1	333	Classification	True Negative	0.000e+000
2	333	Classification	True Negative	0.000e+000
3	334	Classification	True Negative	0.000e+000
			Average	0.000e+000
			Standard Deviation	0.000e+000
1	333	Classification	False Negative	0.000e+000
2	333	Classification	False Negative	0.000e+000
3	334	Classification	False Negative	0.000e+000
			Average	0.000e+000
			Standard Deviation	0.000e+000

1	333	Likelihood	Log Score	-0.4619
2	333	Likelihood	Log Score	-0.4674
3	334	Likelihood	Log Score	-0.4664
			Average	-0.4652
			Standard Deviation	0.0024
1	333	Likelihood	Lift	-0.0026
2	333	Likelihood	Lift	-0.0081
3	334	Likelihood	Lift	-0.008
			Average	-0.0062
			Standard Deviation	0.0026
1	333	Likelihood	Root Mean Square Error	0.1612
2	333	Likelihood	Root Mean Square Error	0.159
3	334	Likelihood	Root Mean Square Error	0.1608
			Average	0.1603
			Standard Deviation	0.0009

Naive Bayes				
Partition Index	Partition Size	Test	Measure	Value
1	333	Classification	True Positive	278
2	333	Classification	True Positive	279
3	334	Classification	True Positive	280
			Average	279.001
			Standard Deviation	0.8167
1	333	Classification	False Positive	48
2	333	Classification	False Positive	44
3	334	Classification	False Positive	48
			Average	46.668
			Standard Deviation	1.8851
1	333	Classification	True Negative	1
2	333	Classification	True Negative	5
3	334	Classification	True Negative	1
			Average	2.332
			Standard Deviation	1.8851
1	333	Classification	False Negative	6
2	333	Classification	False Negative	5
3	334	Classification	False Negative	5
			Average	5.333
			Standard Deviation	0.4713
1	333	Likelihood	Log Score	-0.8128
2	333	Likelihood	Log Score	-0.8089
3	334	Likelihood	Log Score	-0.9003
			Average	-0.8408
			Standard Deviation	0.0422
1	333	Likelihood	Lift	-0.3535
2	333	Likelihood	Lift	-0.3496
3	334	Likelihood	Lift	-0.4419
			Average	-0.3817
			Standard Deviation	0.0427
1	333	Likelihood	Root Mean Square Error	0.2721
2	333	Likelihood	Root Mean Square Error	0.2976
3	334	Likelihood	Root Mean Square Error	0.2875
			Average	0.2857
			Standard Deviation	0.0105

Neural_Network				
Partition Index	Partition Size	Test	Measure	Value
1	333	Classification	True Positive	0.000e+000
2	333	Classification	True Positive	0.000e+000
3	334	Classification	True Positive	285
			Average	95.19
			Standard Deviation	134.4173
1	333	Classification	False Positive	0.000e+000
2	333	Classification	False Positive	0.000e+000
3	334	Classification	False Positive	49
			Average	16.366
			Standard Deviation	23.1103
1	333	Classification	True Negative	49
2	333	Classification	True Negative	49
3	334	Classification	True Negative	0.000e+000
			Average	32.634
			Standard Deviation	23.1103
1	333	Classification	False Negative	284
2	333	Classification	False Negative	284
3	334	Classification	False Negative	0.000e+000
			Average	189.144
			Standard Deviation	133.9457
1	333	Likelihood	Log Score	0.000e+000
2	333	Likelihood	Log Score	0.000e+000
3	334	Likelihood	Log Score	-1.1007
			Average	-0.3676
			Standard Deviation	0.5192
1	333	Likelihood	Lift	0.000e+000
2	333	Likelihood	Lift	0.000e+000
3	334	Likelihood	Lift	-0.6423
			Average	-0.2145
			Standard Deviation	0.3029
1	333	Likelihood	Root Mean Square Error	NaN
2	333	Likelihood	Root Mean Square Error	NaN
3	334	Likelihood	Root Mean Square Error	0.6674
			Average	NaN
			Standard Deviation	NaN

5. Which model is the winner? (Findings based on the best model)

The different measures in the outputs of the model building phase enabled us to look at our three models from different viewpoints (The table below). Now it is time to choose the best one based on our specific need which is, predicting the consequence of a person driving habit in terms of Accident Severity. Having that said, we prefer a model with less number of false negative cases and the highest level of accuracy so that, we have less error to identifying a level 3 severity as the level 2 or 1 by mistake.

Model	Lift Score	Improvement		True Positive	Accuracy
		From Random Guess	False Negative		
Naïve Bayes	0.90	3.06%	5	284	84.5%
Decision Tree	0.89	2.5%	0	279	85%
Neural Network	0.90	1.7%	189	95	85%