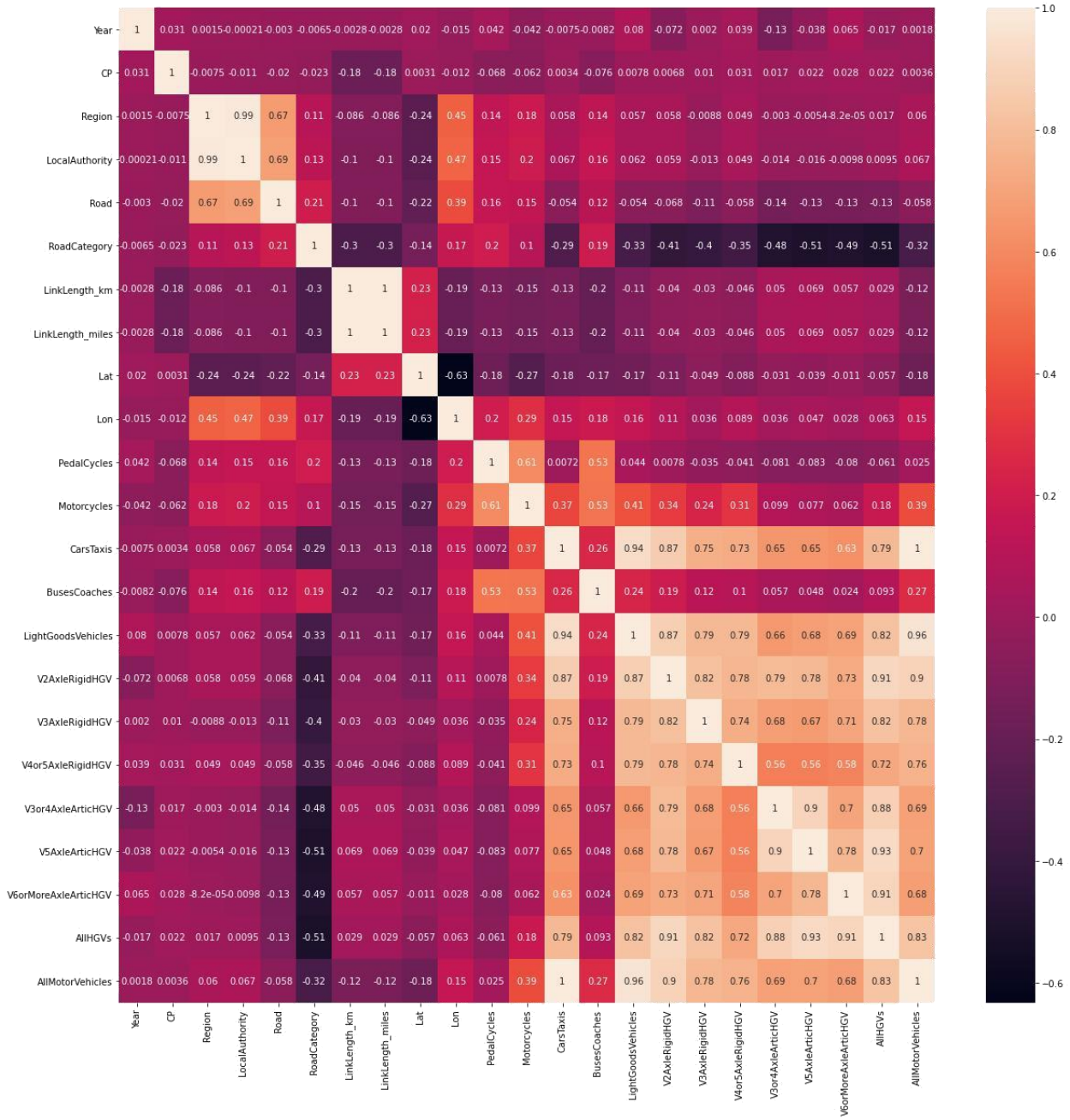
UK Traffic Accidents: A story through data

The project involved preliminary analyses on the UK Traffic Accidents Dataset which has data from years 2000-2016 and contains data points on location of accidents, road linkages, the kind of vehicle involved, and more. The analyses had two phases to it: Exploratory Data Analysis (EDA), and then a mix of Frequent Pattern Mining as well as Cluster Analysis.

Phase 1: Exploratory Data Analysis (EDA)

For the first phase, prior to visualising any data points, we had to preprocess the data. We started off by checking for any duplicate rows (there were none) to ensure correct data entry. Next, we searched for any *NaN* values and found out that only two attributes contained missing values. Since there were 37,573 entries without data on 'Estimation_method' and 'Estimation_method_detailed' attributes, we decided to drop both the attributes as we won't be finding them interesting for our analysis later on. The other option was to drop all instances with missing values, and that was not viable as it would rob dataset of 37,353 instances. Furthermore, after assessing the unique values of each attribute (to check for misspelling or data entry errors), the attribute 'RoadCategory' which we might have used later had some values with letters in lower case while most were in upper case (example, 'Pu' instead of 'PU'). To correct this, we ensured all the letters were upper case by using a string method *str.upper()* on the attribute. This concluded the preprocessing, which was followed by visualisations for EDA.

The first visualisation we decided to make was a correlation matrix. Correlation signifies how two variables move with each other. However, before constructing the matrix, we identified 4 categorical variables ('Region', 'Road', 'RoadCategory', 'LocalAuthority') which we first encoded to find out correlation. The correlation matrix was plotted as follows:



The diagonal of the matrix is the value 1 since every variable is perfectly correlated with itself. The above heat map shows the correlation between each pair of variables. We will just focus on the moderate and strong relations. A correlation is said to be moderate if the magnitude is between 0.4 - 0.59. It is strong between 0.6-0.79 and it is very strong between 0.8 - 1. An insight we can see is that the types of vehicles are positively related (example, ‘CarTaxis’ and ‘LightGoodsVehicles’). This means that increased accidents of CarsTaxis on a roadlink in a given year meant increased accidents of LightGoodsVehicle. This could give valuable insight for why the accidents increased. Most of the different types of vehicles are strongly correlated with each other.

Some examples of categorized correlations from the above matrix are given below:

Moderate Correlations:

- Region and Lon - Positive
- Local Authority and Lon - Positive
- RoadCategory and V3or4AxleArticHGV - Negative
- RoadCategory and V5AxleArticHGV - Negative
- RoadCategory and V6orMoreAxleArticHGV - Negative
- RoadCategory and AllHGVs - Negative

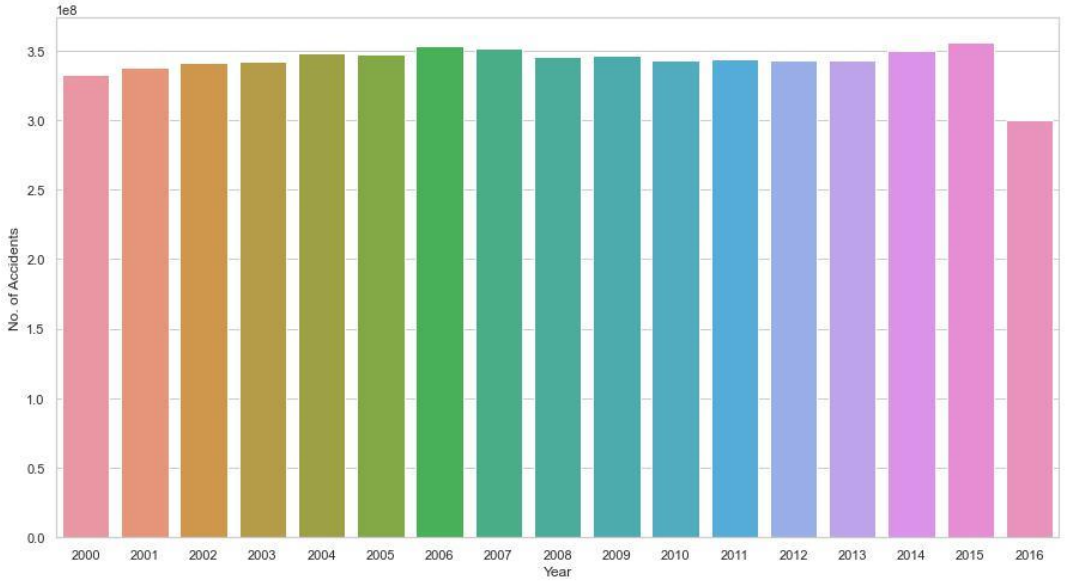
Strong Correlations:

- Region and Road - Positive
- Local Authority and Road - Positive
- Lat and Lon - Negative
- PedalCycles and MotorCycles - Positive
- PedalCycles and BusesCoaches - Positive
- MotorCycles and BusesCoaches - Positive

Very Strong Correlations:

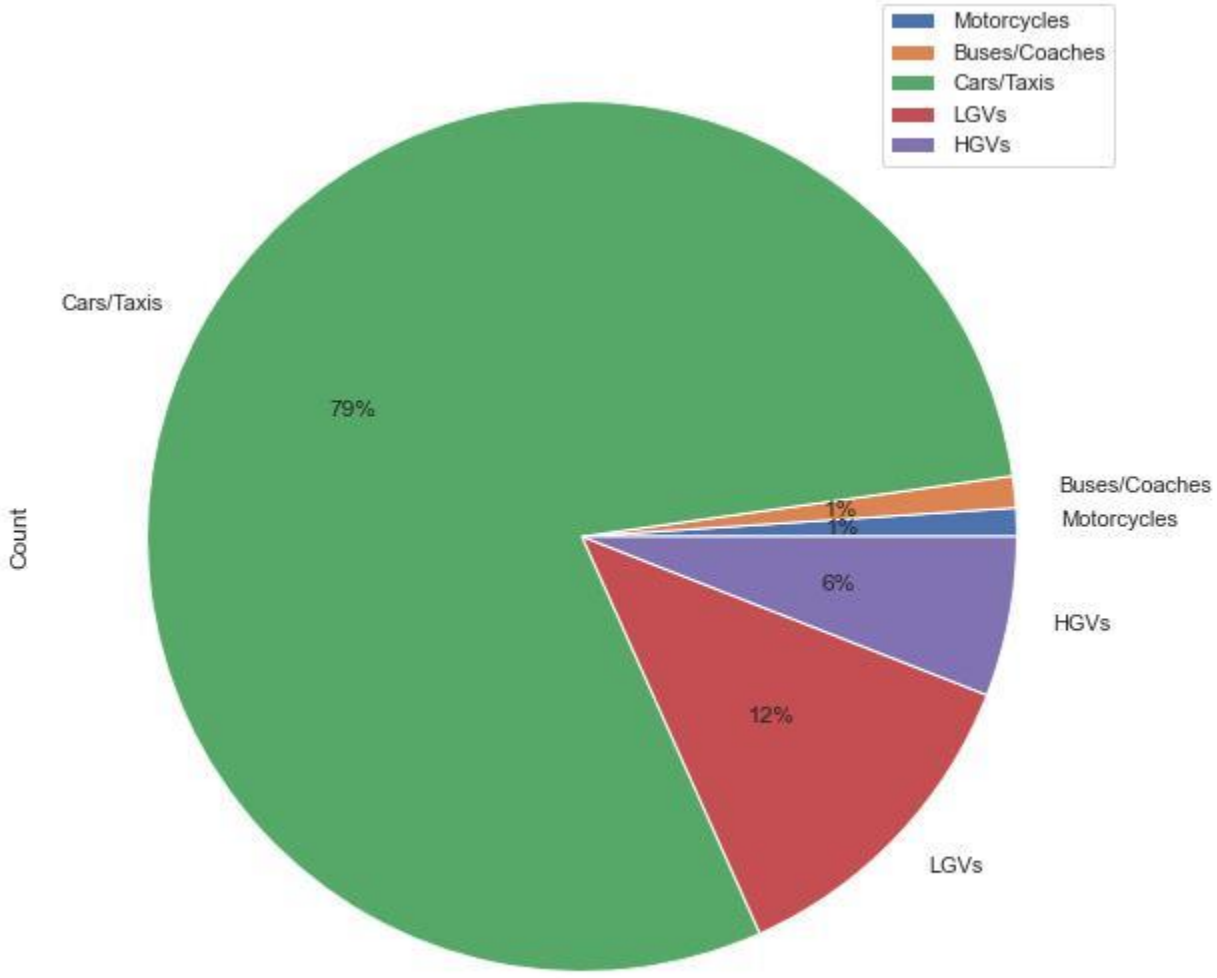
- Region and Local Authority - Positive
- Link_length_km and Link_length_miles - Positive
- CarsTaxis and LightGoodsVehicles - Positive

Our next category for EDA was yearly analysis. We first summed up the attribute 'AllMotorVehicles' by making a *groupby* object based on year to get a data frame consisting of the total number of accidents for each year for Motorcycles, CarsTaxis, BusesCoaches, Light Goods Vehicles (LGVs), and all Heavy Goods Vehicles (HGVs). We obtained the following visualisation:

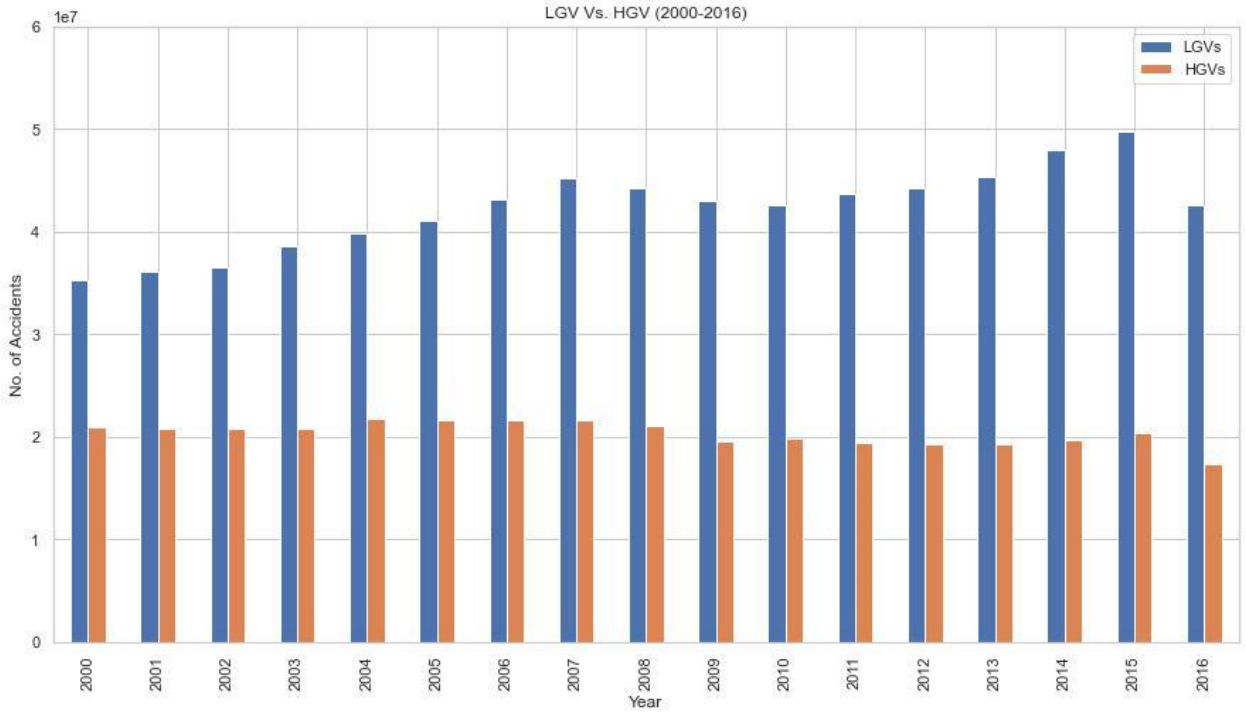


The general trend over the years can be seen as increasing slightly, however there is a need to take into account the population growth (and hence the traffic congestion/growth) in UK which perhaps might be a hypothesis to test whether the number of accidents have stayed stagnant or with slight increase despite a boom in vehicles on the road due to increasing population. One interesting thing to notice is that the last year in the dataset (2016) has a sudden drop in the number of accidents - this can be due to two reasons; a) either the data for the complete year of 2016 was not present, or b) a policy introduced during the time caused a sudden drop in accidents (not establishing causation, however, but is interesting nonetheless for latter analyses).

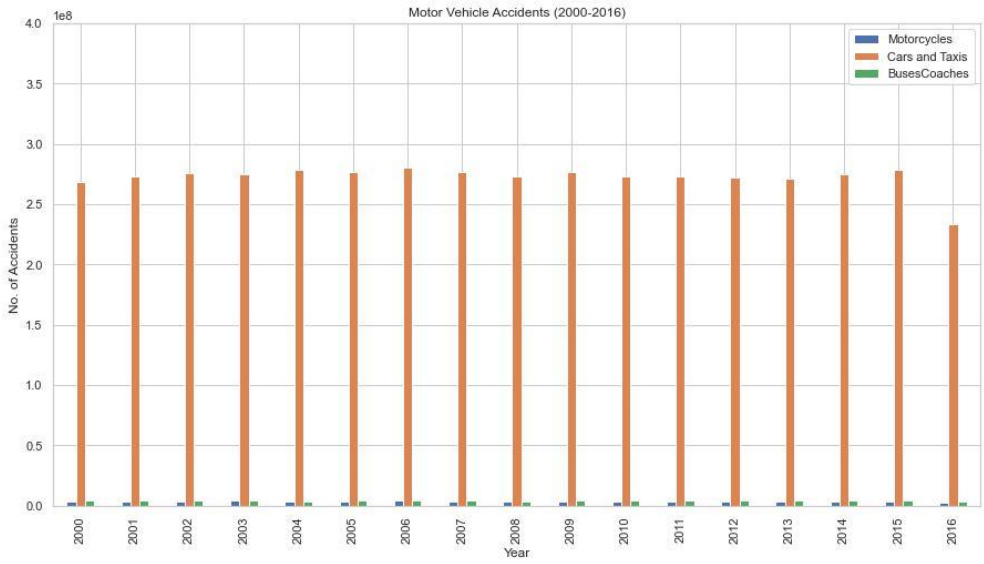
Next up in EDA is the Category-wise analysis based on the type of vehicles. In terms of the total distribution of accidents across categories, the below pie chart shows that **Car/Taxis** take up a sweeping majority of **79%** of total accidents followed by LGVs at **12%** and HGVs at **6%.**



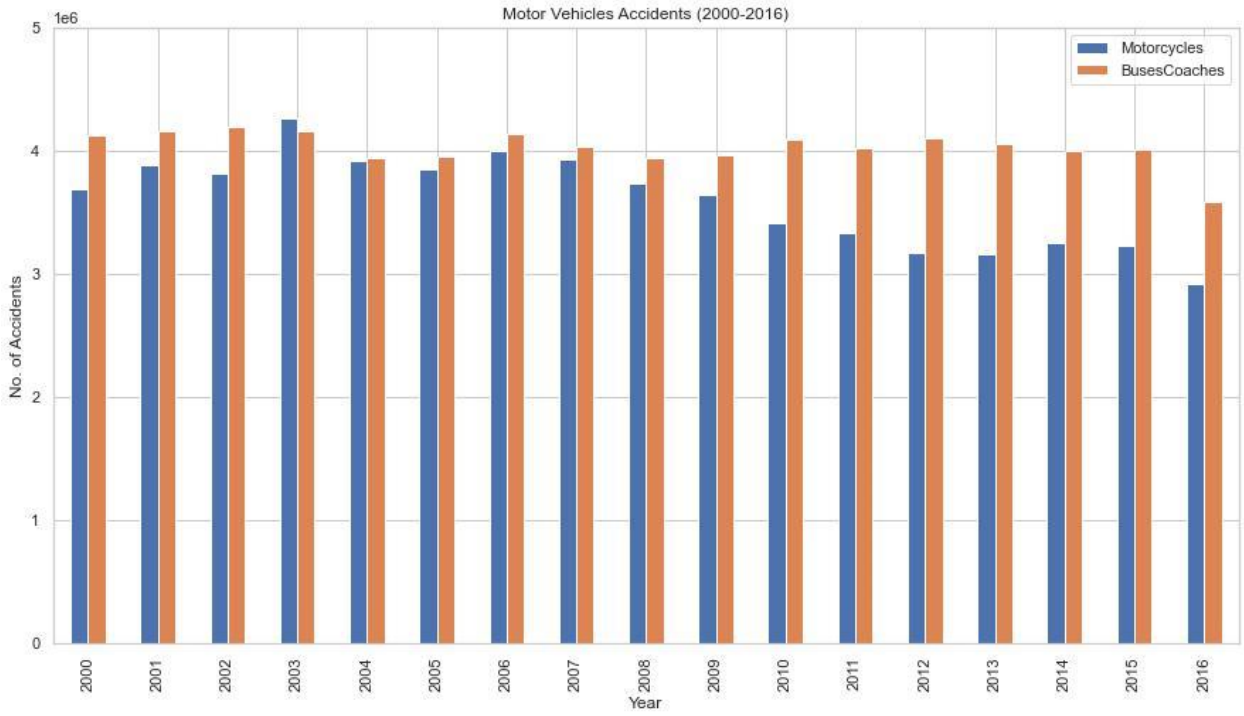
Furthermore, we took a route to compare the accident numbers of different types of vehicles over the years. The first category was LGVs vs HGVs:



It can be seen that Light Goods Vehicles have shown consistently higher numbers compared to Heavy Good Vehicles, and are increasing over the years on average. Heavy Good Vehicles show a stable, consistent number of accidents over the years. Next, we plot the category of 'MotorVehicles' which includes Motorcycles, Cars/Taxis, and Buses/Coaches:

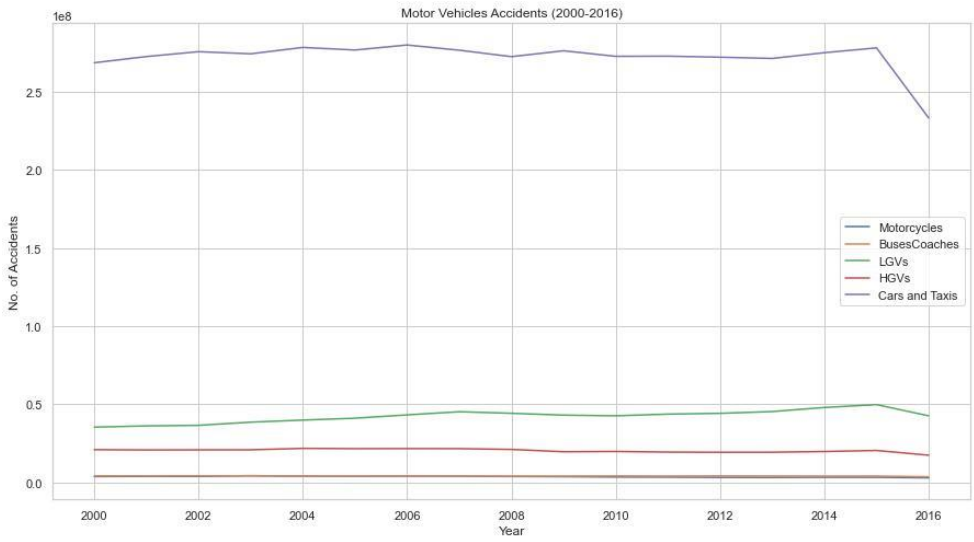


We noticed here that accidents for cars and taxis are consistently very high over the years on average (compared to the other two categories of motorcycles and buses/coaches). Since these 3 are not comparable together, we will compare motorcycles with buses/coaches:

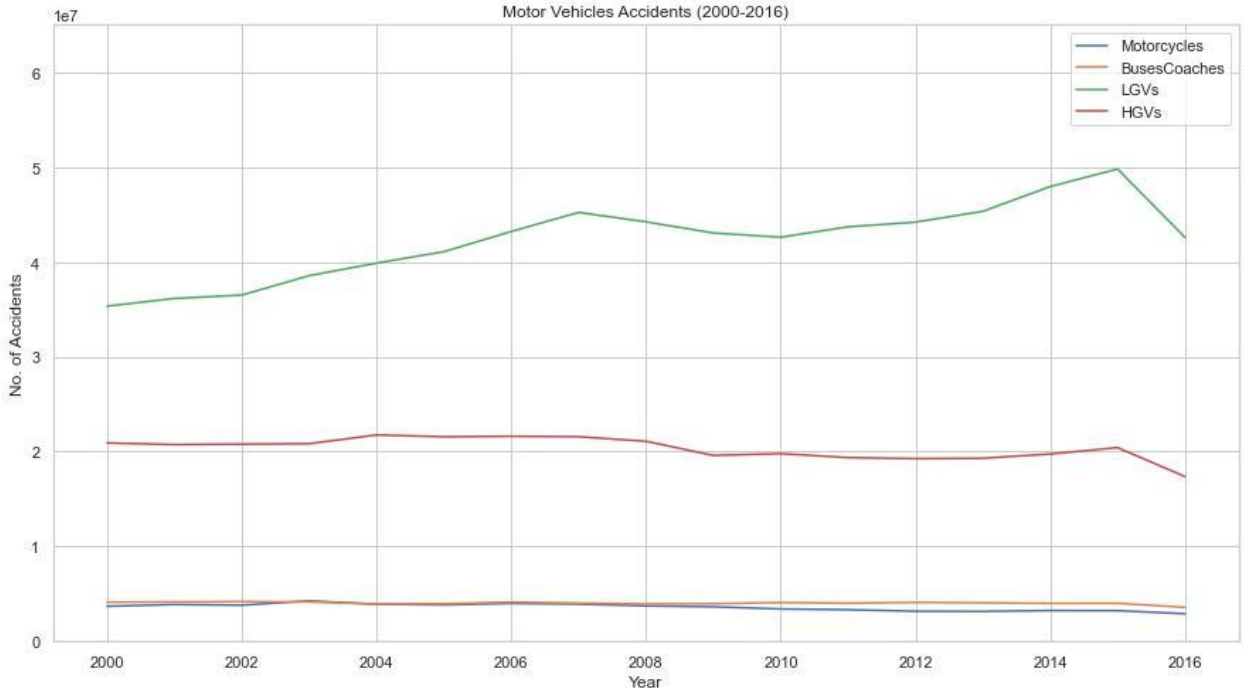


From the above visualisation, we can see that accidents for buses and coaches are fluctuating along 4,000,000 accidents for most of the years, while there is a decreasing trend starting from the year 2007 onwards for Motorcycle accidents. This can be interesting as it may show a reason or cause (perhaps due to a policy of helmets being mandatory or a separate motorcycle lane) which may be inferred from later analyses.

Just to show the accident count trends over time across all categories we also made line plots:

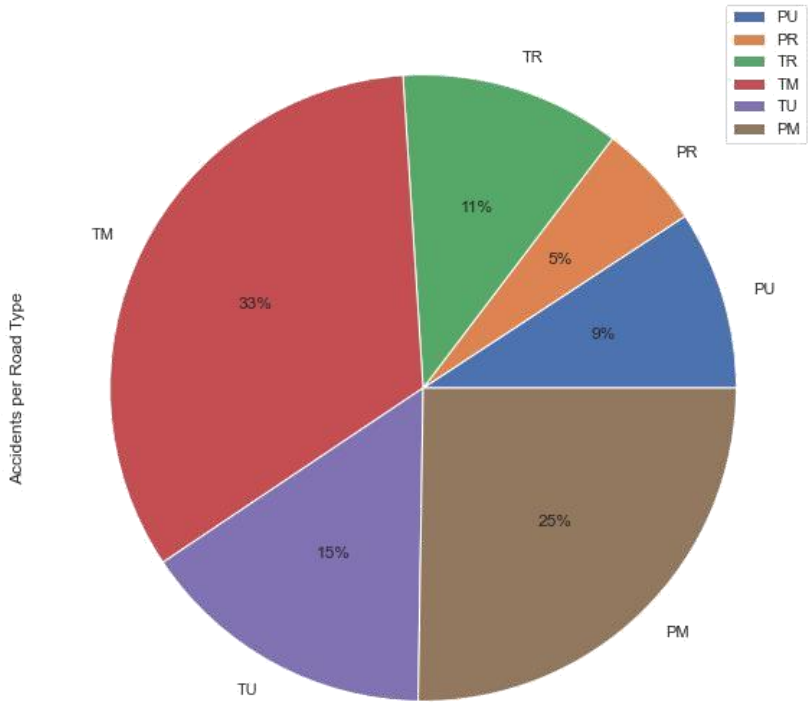


This plot just shows the need to address accidents in cars and taxis which have been consistently very high over the years on average (compared to all other categories). However, we do see a sharp dip from 2015-2016 in both Cars/Taxis and LGVs which again might be due to the reasons described above for a dip in accidents overall. Since the rest categories are not comparable together with cars and taxis, we will eliminate cars and taxis from the graph and zoom in on the variations in other categories:



We can now see the trend of each of these categories. LGV accidents have been increasing on average and are the highest among these categories consistently in terms of accidents over the years. Buses/Coaches and motorcycles follow a very similar stable trend with similar numbers which are lower than the HGV accidents. HGV accidents have slightly decreased over the years.

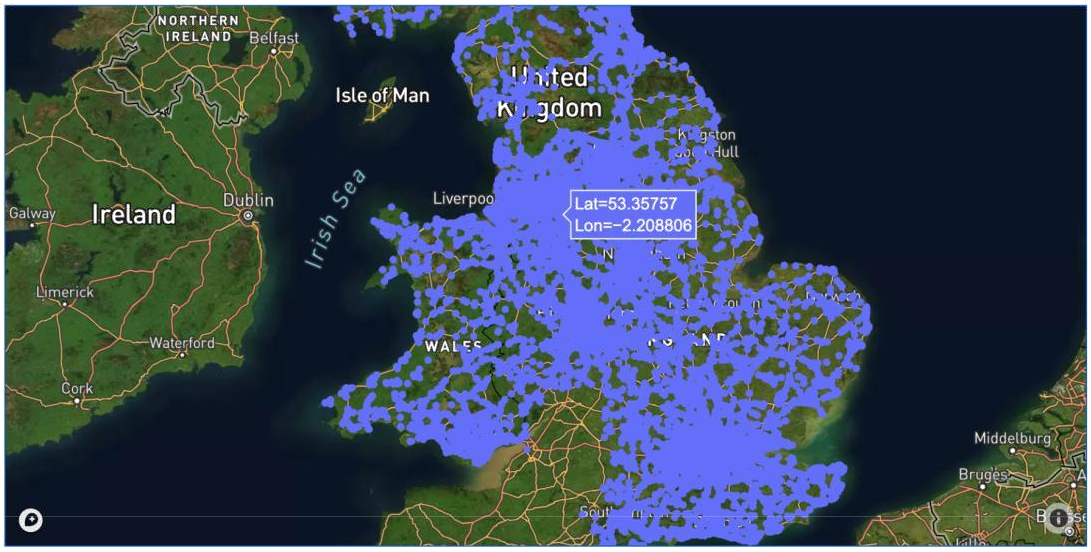
Our next analysis as part of EDA is focused on regional/location-based analyses. First, we made a pie chart based on the ratios of accidents per their road type specifically:



The pie chart above shows the distribution of the ratio of accidents per road type. The top three ratios are of the following road types: TM, PM, and TU.

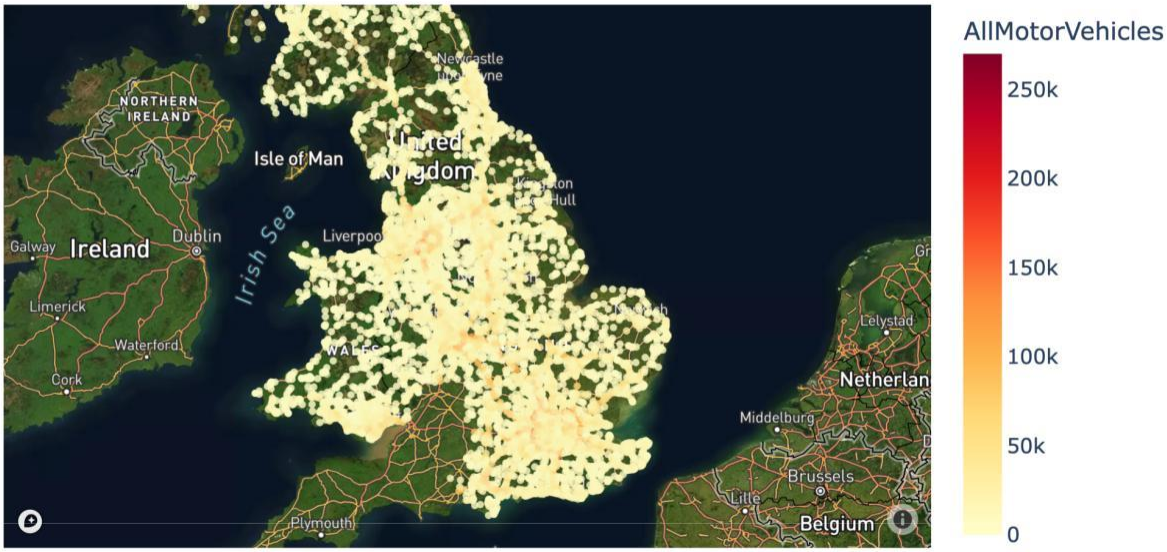
Towards the final part of our EDA we visualised different attributes by superimposing the datapoints interactively onto the map of UK using *plotly* by incorporating the longitudinal and latitudinal values of each accident. For this, we would like to cite the article which we took inspiration from: https://medium.com/technology-hits/working-with-maps-in-python-with-mapbox-and-plotly-6f454522ccdd.

UK Accident Locations



The heatmap over the UK map shows us that there are approximately 50,000 accidents around all road links. Upon further analysis, we can see that the frequency of accidents increases at the outskirts of major cities.

All UK Accident Locations for Motor Vehicles



The heatmap over the UK map shows us that there are approximately 50,000 accidents around all road links. Upon further analysis, we can see that the frequency of accidents increases at the outskirts of major cities.

Frequent UK Accident Locations for Motor Vehicles

To see frequent accident road links, we limited our dataframe to only >100,000 'AllMotorVehicles' accidents. Judging from the heatmap above we can clearly see that road links near the following cities are most susceptible to motor vehicle accidents: London, Cardiff, Southampton, Nottingham, Manchester, Birmingham, Sheffield, and Glasgow.

Frequent UK Accident Locations for Pedal Cycles

The heatmap of the pedal cycles is similar to the one where we were trying to see locations with the most frequent accidents. These accidents are near the major cities of the United Kingdom.

Part 2: FPM and Clustering Analysis

In this part of the project we had to find out the frequency of accidents that happen in a region and in an year. We used the apriori algorithm that we had learnt in class and we used the efficient_Apriori library since we found it to be the most efficient during our assignment.

The apriori library takes a list of tuples so we needed to change our dataset accordingly. First we made tuples of the region and year and multiplied the tuples by the number of accidents that correspond to that row, so that we have as many objects of the tuple as in our dataset to the frequent pattern mining.

Since the number of accidents were significant we could not have as many objects as number of accidents since our machine would run out of memory. As an alternative we normalised them by dividing them by 100. So throughout the next few parts the number of accidents are in hundreds.

We run apriori with a low min_support since increasing it will result in low values being returned.

Total number of tuples = 116775398 (in hundreds)

Min_support = 0.001

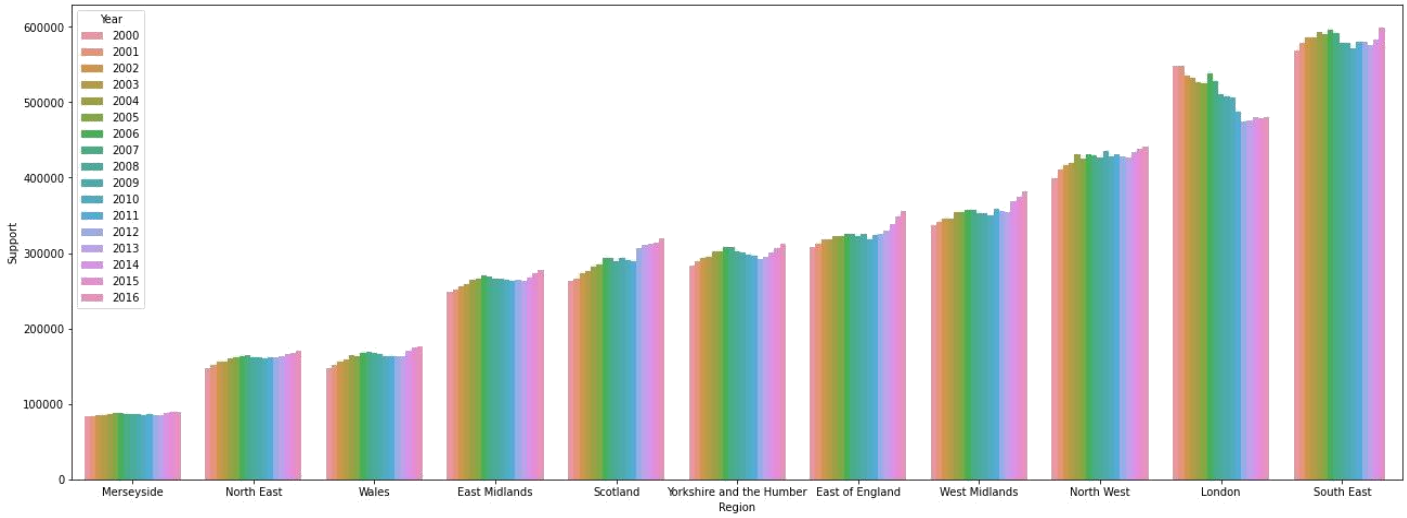
Threshold = 0.001*116775398 = 116776 (in hundreds)

So our threshold is 116776, any pattern that appears more than this threshold will be carried forward. We have used a low min_support so that we have sufficient number of tuples to see the trends.

Out[17]:

	Year	Region	Support
0	2001	Merseyside	83109
1	2000	Merseyside	84136
2	2003	Merseyside	84893
3	2010	Merseyside	85434
4	2002	Merseyside	85461
...	...	...	...
181	2005	South East	589614
182	2007	South East	592244
183	2004	South East	592892
184	2006	South East	595615
185	2015	South East	599525

186 rows × 3 columns



In the above visualization, we have plotted the Support against Region with hue being the Year. Each bar represents the number of accidents of AllMotorVehicles for a particular year. Upon analysis, we can observe that some regions have similar trends according to change in number of accidents per year. For example, East of England and West Midland have similar trends. These trends allow us to gauge whether the accidents have increased or decreased across years. For example, we can see that in London the number of accidents have decrease across years. Judging from this insight, it would be helpful to see which trafficking policies are being implemented in London and if the similar trend could be translated in other regions through these policies.

For clustering, we used the dataframe to convert it into a kind of a pivot table that shows total number of accidents for each region in each specific year:

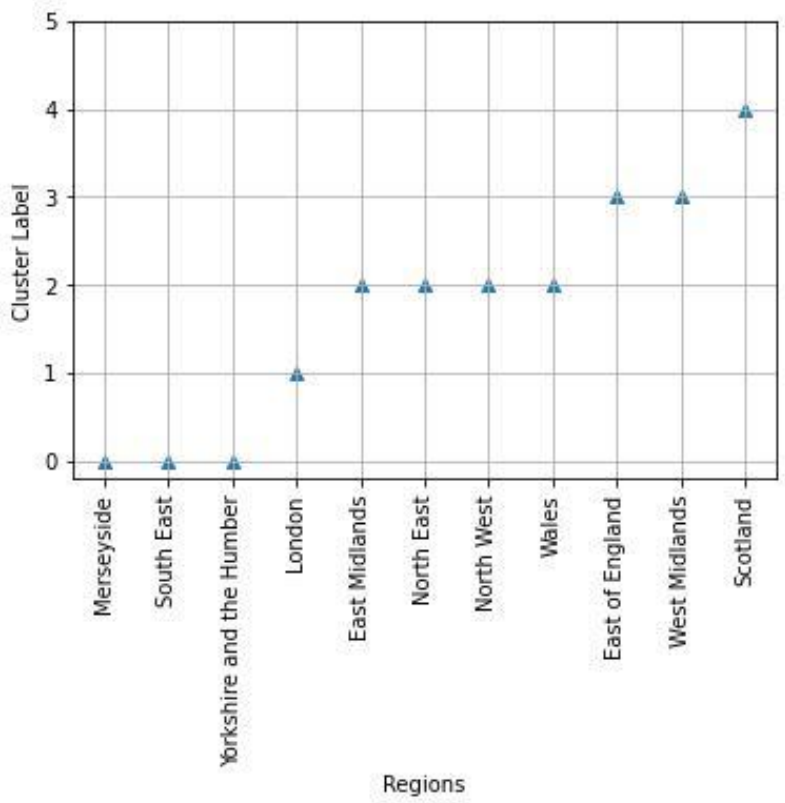
Out[21]:

Year	Region	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2
0	East Midlands	24792283.0	25069902.0	25534456.0	25760086.0	26385719.0	26537520.0	26990474.0	26820695.0	26522368.0	26566058.0	26391819.0	2625221
1	East of England	30692949.0	31232915.0	31756707.0	31788690.0	32245057.0	32242749.0	32539292.0	32425855.0	32144539.0	32399886.0	31790463.0	3229738
2	London	54746708.0	54674758.0	53438044.0	53150243.0	52556110.0	52372693.0	53776517.0	52756704.0	51013416.0	50693165.0	50576363.0	4870458
3	Merseyside	8390257.0	8289196.0	8522751.0	8465178.0	8632153.0	8706532.0	8764608.0	8695362.0	8680273.0	8666500.0	8522277.0	868424
4	North East	14727152.0	15174018.0	15516809.0	15581904.0	16055280.0	16116189.0	16283918.0	16434623.0	16170754.0	16166889.0	16035971.0	1614363
5	North West	39855289.0	40954315.0	41505298.0	41859447.0	43088759.0	42420248.0	42992956.0	42888227.0	42576124.0	43497891.0	42683021.0	4303744
6	Scotland	26203816.0	26568811.0	27210441.0	27546250.0	28126394.0	28332964.0	29284152.0	29220423.0	28843093.0	29273010.0	28908774.0	2888110
7	South East	56742760.0	57747148.0	58447657.0	58459358.0	59172182.0	58845730.0	59446119.0	59107793.0	57726949.0	57693577.0	57024920.0	5787166
8	Wales	14741757.0	15087913.0	15527258.0	15850934.0	16374415.0	16297161.0	16722181.0	16846410.0	16646168.0	16513889.0	16213278.0	1627451
9	West Midlands	33654764.0	34044501.0	34464427.0	34480193.0	35314641.0	35421895.0	35607656.0	35693265.0	35200600.0	35273379.0	34928036.0	3574238
10	Yorkshire and the Humber	28325443.0	28820345.0	29252032.0	29386062.0	30140790.0	30102642.0	30714046.0	30684549.0	30162584.0	29968273.0	29704926.0	2954962

We then splitted the data into regions and number of accidents, followed by using the *Kmeans* clustering library from SKlearn and kept the hyperparameter of number of clusters to 5. The algorithm assigned labels to the regions as follows:

	labels	regions
0	2	East Midlands
1	3	East of England
2	1	London
3	0	Merseyside
4	2	North East
5	2	North West
6	4	Scotland
7	0	South East
8	2	Wales
9	3	West Midlands
10	0	Yorkshire and the Humber

This can also be visually represented better as the following:



The insights we generated from both FPM and clustering analysis are corroborating each other. This can be seen from the fact that London is in a cluster of its own for example in cluster analysis, and in the visualisation from it shows that London is the only one with the downwards trend. Similarly, Scotland has a lot of variations and there aren’t smooth increases and decreases compared to other regions.