ECS784U - DATA ANALYTICS - 2017/18

Coursework

Group Members:

Chia-Chen Hsieh 170339562

Po-Kai Chen 170160900

Wei-Shiang Lian 170509682

Yi-Chen Wu 150358581

Submission Date: 13/04/2018

Contents

DATA ANALYTICS COURSEWORK

Abstract

Traffic is one of the important topics related to livelihood. In this report, we are going to use the traffic accidents data given by Kaggle website which is recorded by the UK police forces to do the data analysis. We aim to learn more information through reading the data from different perspectives, besides, we hopefully can generate some brief suggestions to help with improving the traffic issues as from the department of transport report in the UK, there were many road deaths every year.
We are going to present the content in the following order, from the background, how we read data, deal with the missing data, and the analysis content will be described in four main ways, which are descriptive statistic, time series analysis, heat map analysis, and map of traffic accidents analysis. Finally, the conclusion and the recommendations for the traffic topic which we found will be presented as well.

Introduction

It is crucial to have the data analysis skill in the contemporary society, as it can be widely applied in a lot of fields. Not only for the computer science, but also for the market places.

The aim of this report is to analyze the data we got from the website, which is the 1.6 million UK traffic accidents. Since this topic is highly related to the livelihood field, we are going to present the data analysis results in a descriptive statistic aspect, time series analysis perspective, heat map analysis, as well as map of traffic accidents analysis to get deeper understanding of the results data showed to us. Additionally, we are going to see if we can integrate some brief conclusion and recommendations after the data analysis and other related literatures.

Background Information

Livelihood Topic- Traffic Accidents

The livelihood problem has always been an important topic for the government and residents. For instance, the department of transport published regularly the related topic reports on their website for people to review. In their report -- reported road casualties in Great Britain: quarterly provisional estimates year ending June 2017, we can learn that there were 1,710 road deaths in the year ending June 2017, and this is not statistically different from the year ending 2016. In another word, regarding the traffic accidents, there were road deaths and injured every year. (Road accidents and safety statistics guidance - GOV.UK, 2018)

Therefore, it is critical to learn more information from the traffic incident data to get the insight of possible countermeasures that could be undertaken to improve with the conditions from an overall view in the UK.

Data Resource

The data can be accessed on the Kaggle website, the link is shown as below: [online] https://www.kaggle.com/daveianhickey/2000-16-traffic-flow-england-scotland-wales/data [Accessed 11 Apr. 2018].

Read Data- Manage Multiple Documents into One

As the data we used is in three different excel documents, which are the content of traffic incidents recorded in 2005-2007, 2009-2011, and 2012-2014. There is a lack of data in 2008. Since the data is not totally completed, in this report that we are going

to change low memory = False, and recognized 'null' as na by setting na value=['NULL'] with the default setting. In this way, we can merge the three files into a single one to do the data analysis from an overall perspective. The codes will be described as below.

```
data2005_2007 = pd.read_csv('data/accidents_2005_to_2007.csv', na_values=['NULL'], low_memory = False)
data2009_2011 = pd.read_csv('data/accidents_2009_to_2011.csv', na_values=['NULL'], low_memory = False)
data2012_2014 = pd.read_csv('data/accidents_2012_to_2014.csv', na_values=['NULL'], low_memory = False)
dataAADF = pd.read_csv('data/ukTrafficAADF.csv', na_values=['NULL'], low_memory=False)
#combine data from 2005 - 2014
data = pd.concat([data2005_2007, data2009_2011, data2012_2014])
data['Date'] = pd.to_datetime(data['Date'], format='%d/%m/%Y', errors='coerce')
```

https://pandas.pydata.org/pandas-docs/stable/generated/pandas.read_csv.html

Missing Data- Data Cleaning

There are seven main ways to clean the missing data, which are listwise deletion, recover the values, educated guessing, average imputation, common-point imputation, regression substitution, and multiple imputation. Given the way to deal with the data depends on the specific situations, in this part we will briefly introduce each of the way in the following. (Jeff, 2015)

Listwise Deletion

Regarding listwise deletion, it can be used when the data is large enough, as long as your data is large enough, then it is highly likely that we can just drop data without substantial loss of statistical power. In other words, this way is to delete all data from any participant with missing values. However, it is crucial to make sure that the missing values are random and that you are not inadvertently removing a class of participants. (Jeff, 2015)

Recover the Values

Sometimes it might be possible to contact the participants and ask them to fill out the missing values. For instance, during the in-person studies, Jeff (2015) claimed that they have found having an additional check for missing values before the participants helps.

Educated Guessing

Thirdly, Jeff (2015) said that it is generally believed that not the preferred action, but sometimes it can be considered as a way to infer a missing value. For instance, like

those presented in a matrix, if the participant responds with all “4s”, assuming that the missing value is a 4 in terms of the related questions.

Average Imputation

This way is to use the average value of the responses from the other participants to fill in the missing value. For example, if the average of the 30 responses of the questions is a 4.1, use a 4.1 as the imputed value. Although Jeff (2015) claimed that this choice is not always recommended as it will artificially reduce the variability of your data, however, it makes sense in some cases.

Common-Point Imputation

Regarding this way, it is to use the middle point or most commonly chosen value for a rating scale. For instance, Jeff (2015) said that on a five-point scale, substitute a 3 as the middle point, or a 4 as the most common value in many cases. This way is considered as more structured than guessing, however, it is still a risky option. Jeff (2015) recommended that it is better to use it unless there are good reasons and supportive data.

Regression Substitution

We can use multiple-regression analysis to estimate a missing value. This technique is to deal with missing SUS scores, in this case, for example, you would need enough data to create stable regression equations and predict the missing values automatically. (Jeff, 2015)

Multiple Imputation

This is the way which currently is recognized as the most popular approach to take the regression idea further and take advantage of correlation between responses, but also the most sophisticated one. Following this way, software manages plausible values based on the correlations for the missing data and then averages the simulated datasets by incorporating random errors in the predictions. (Jeff, 2015)

Regarding the data that we used for the traffic incidents analysis, since the missing value plays 7.7785e-05 part of the data, we are going to drop NA to delete all data from any participant with missing values, which is the first way to deal with the missing data.

Aims

We aim to do the data analysis toward the data we found from the website, which is recorded by the UK police force to have an advanced insight of It. On the other hand, by doing the analysis, we would like to provide some suggestions toward this topic to help with the improvement of the quality of livelihood.

Exploratory Analysis- Data Preparation

From the beginning, we found that the data type of the time variable is originally a string one. In order to smoothly analyze the data, we change it from a string type into 'datetime' one. Additionally, we add some columns for the later application.

```
#change the format of data['Date'] into datetime
data['Date'] = pd.to_datetime(data['Date'], format='%d/%m/%Y', errors='coerce')
#add 'month' & 'day' variable
data['Month'] = data['Date'].dt.month
data['Day'] = data['Date'].dt.day
#add 'hour' variable
data['Hour'] = pd.to_datetime(data['Time'], format='%H:%M',
errors='coerce').dt.hour.astype('int')
```

As we will analyze the data in the following orders, which are descriptive statistic and then time series analysis. Therefore, given the part below is for time series analysis.

```
data2005_2007 = data[(data["Year"] >= 2005) & (data["Year"] <= 2007)].copy()
data2009_2014 = data[(data["Year"] >= 2009) & (data["Year"] <= 2014)].copy()
Casualties2005_2007byYM = data2005_2007.groupby(['Year',
'Month'])['Number_of_Casualties'].sum().to_frame().reset_index()
Casualties2005_2007byYM['Index'] = pd.date_range('2005-01', periods =
Casualties2005_2007byYM.shape[0], freq = 'M')
Casualties2005_2007 =
data2005_2007.groupby(['Date'])['Number_of_Casualties'].sum().to_frame().reset_index(
)
Casualties2009_2014 =
data2009_2014.groupby(['Date'])['Number_of_Casualties'].sum().to_frame().reset_index(
)
ts_2005_2007 = pd.Series(Casualties2005_2007.Number_of_Casualties.values,
Casualties2005_2007.Date)
ts_2009_2014 = pd.Series(Casualties2009_2014.Number_of_Casualties.values,
Casualties2009_2014.Date)
```

Data Analysis Implication

1.1 Descriptive Statistic

From the very beginning, we are going to check basic information of all the variables that we have in our data.

```
data.describecribe()
```

	Location_Easting_OSGR	Location_Northing_OSGR	Longitude	Latitude	Police_Force	Accident_Severity	Number_of_Vehicles
count	1.504049e+06	1.504049e+06	1.504049e+06	1.504049e+06	1.504150e+06	1.504150e+06	1.504150e+06
mean	4.396214e+05	3.001584e+05	-1.436625e+00	5.258940e+01	3.020536e+01	2.838190e+00	1.831606e+00
std	9.511616e+04	1.610084e+05	1.398078e+00	1.449889e+00	2.551603e+01	4.018423e-01	7.147586e-01
min	6.495000e+04	1.029000e+04	-7.516225e+00	4.991294e+01	1.000000e+00	1.000000e+00	1.000000e+00
25%	3.750600e+05	1.782600e+05	-2.373902e+00	5.149016e+01	6.000000e+00	3.000000e+00	1.000000e+00
50%	4.399600e+05	2.688300e+05	-1.403714e+00	5.230913e+01	3.000000e+01	3.000000e+00	2.000000e+00
75%	5.230600e+05	3.981510e+05	-2.215100e-01	5.347858e+01	4.500000e+01	3.000000e+00	2.000000e+00
max	6.553700e+05	1.208800e+06	1.759398e+00	6.075754e+01	9.800000e+01	3.000000e+00	6.700000e+01

Number_of_Casualties	Day_of_Week	Local_Authority_(District)	1st_Road_Class	1st_Road_Number	Speed_limit	Junction_Detail	2nd_Road_Class
1.504150e+06	1.504150e+06	1.504150e+06	1.504150e+06	1.504150e+06	1.504150e+06	0.0	1.504150e+06
1.350960e+00	4.118607e+00	3.476149e+02	4.087999e+00	1.009919e+03	3.900540e+01	NaN	2.675084e+00
8.253345e-01	1.924405e+00	2.594292e+02	1.428936e+00	1.823518e+03	1.413993e+01	NaN	3.205539e+00
1.000000e+00	1.000000e+00	1.000000e+00	1.000000e+00	-1.000000e+00	1.000000e+01	NaN	-1.000000e+00
1.000000e+00	2.000000e+00	1.100000e+02	3.000000e+00	0.000000e+00	3.000000e+01	NaN	-1.000000e+00
1.000000e+00	4.000000e+00	3.220000e+02	4.000000e+00	1.290000e+02	3.000000e+01	NaN	3.000000e+00
1.000000e+00	6.000000e+00	5.180000e+02	6.000000e+00	7.250000e+02	5.000000e+01	NaN	6.000000e+00
9.300000e+01	7.000000e+00	9.410000e+02	6.000000e+00	9.999000e+03	7.000000e+01	NaN	6.000000e+00

2nd_Road_Number	Urban_or_Rural_Area	Year
1.504150e+06	1.504150e+06	1.504150e+06
3.815684e+02	1.353871e+00	2.009370e+03
1.302555e+03	4.783534e-01	3.013497e+00
-1.000000e+00	1.000000e+00	2.005000e+03
0.000000e+00	1.000000e+00	2.006000e+03
0.000000e+00	1.000000e+00	2.010000e+03
0.000000e+00	2.000000e+00	2.012000e+03
9.999000e+03	3.000000e+00	2.014000e+03

As the graphs showed above, we can learn all the variables we have in our data.

In this step, we are going to visualize the data in various ways by generating the plot which is grouped by year, month, day of week, and the time of each day. The codes will be presented as below.

```
fig = plt.figure(figsize=(15, 10))

ax1 = fig.add_subplot(411)
data.groupby('Year')['Number_of_Casualties'].sum().plot(ax = ax1, kind = 'bar')
ax1.set_title('Total number of Casualties by year')

ax2 = fig.add_subplot(412)
data.groupby('Month')['Number_of_Casualties'].sum().plot(ax = ax2, color = 'red', kind =
'bar')
ax2.set_title('Total number of Casualties by Month')

ax3 = fig.add_subplot(413)
data.groupby('Day_of_Week')['Number_of_Casualties'].sum().plot(ax = ax3, color =
'aqua', kind = 'bar')
ax3.set_title('Total number of Casualties by Day_of_Week')

ax4 = fig.add_subplot(414)
data.groupby('Hour')['Number_of_Casualties'].sum().plot(ax = ax4, color = 'darkslateblue',
kind = 'bar')
ax4.set_title('Total number of Casualties by Hour')

plt.tight_layout()
fig.savefig('Casualties.png')
```

We can find out that the number of accident decreased from 2005 to 2011, and increase a little in 2012, then it dropped down in the next year, but no dramatically change after 2013. The overall chart is shown as below as chart-1. Besides, in terms of the day of the week chart from chart-1, it is clear that Sundays had the lowest number of accidents, and Fridays had the highest ones.

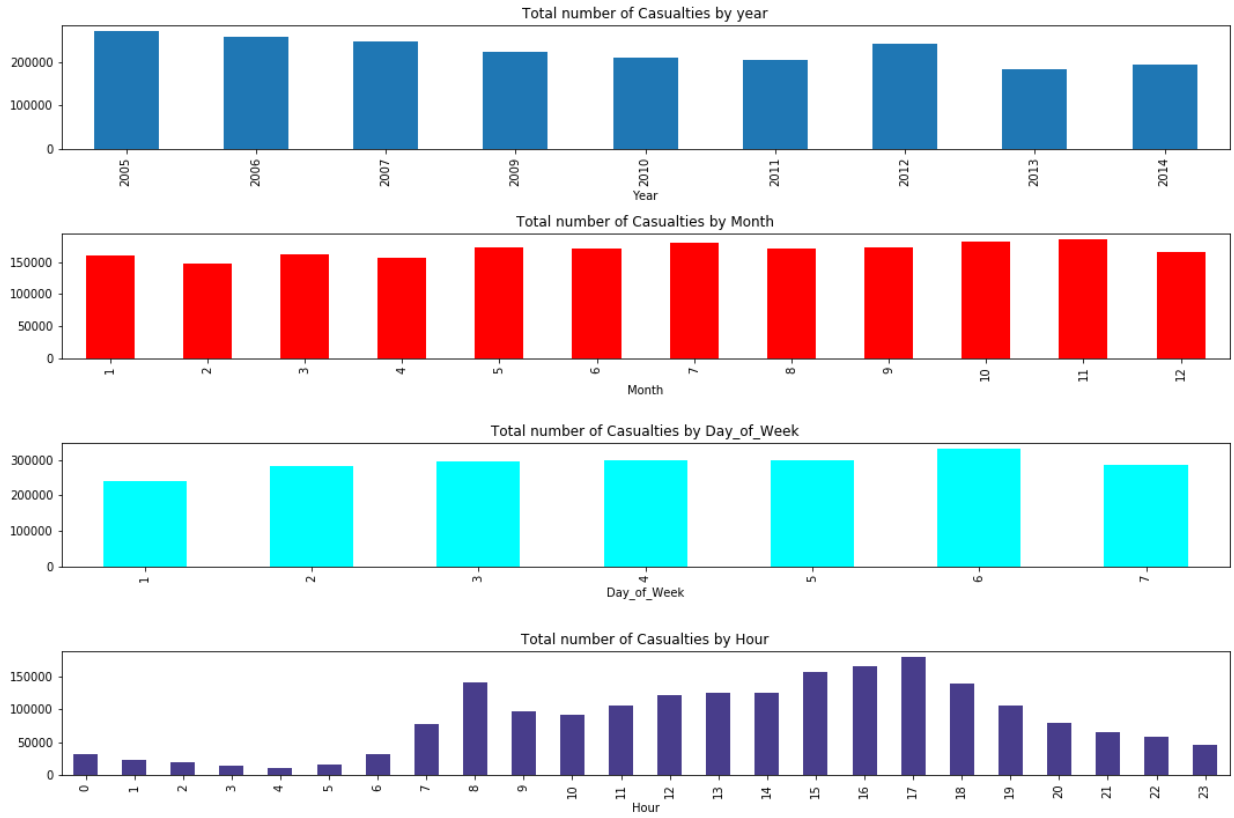
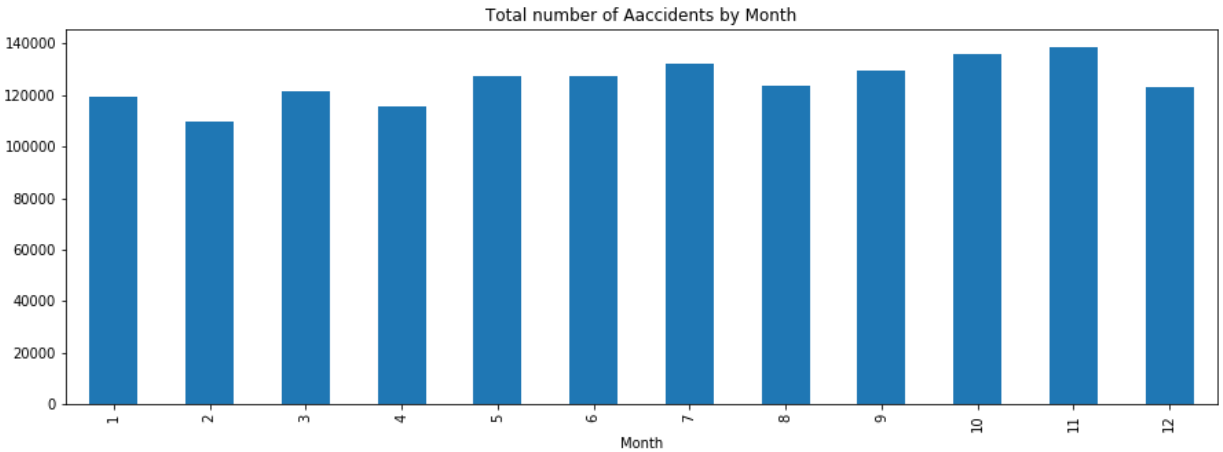
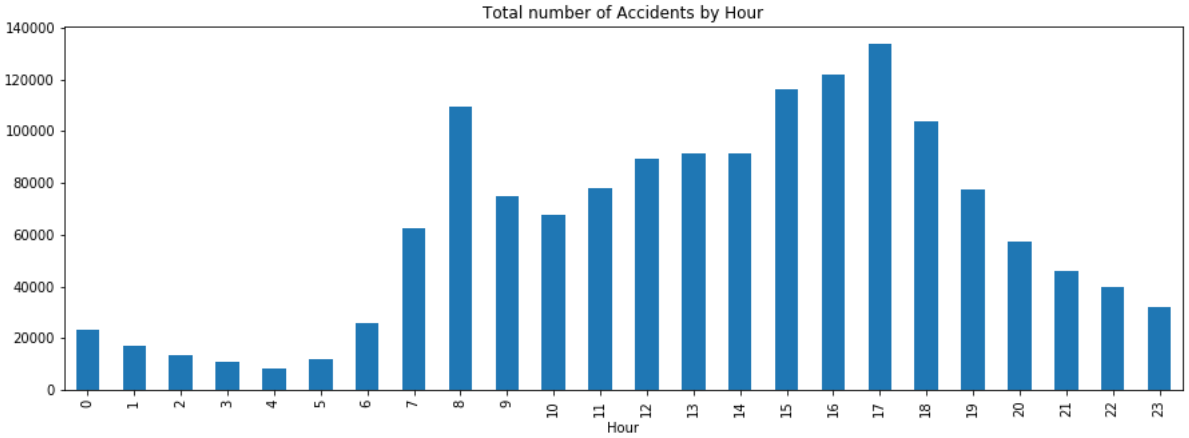


Chart-1. Total number of Causalities grouped in different time period variables

Given the charts as below, regarding the monthly aspect of the chart, it is clear that the number of accidents are in February, on the other hand, November had the highest number of accidents as the peak of the whole year.

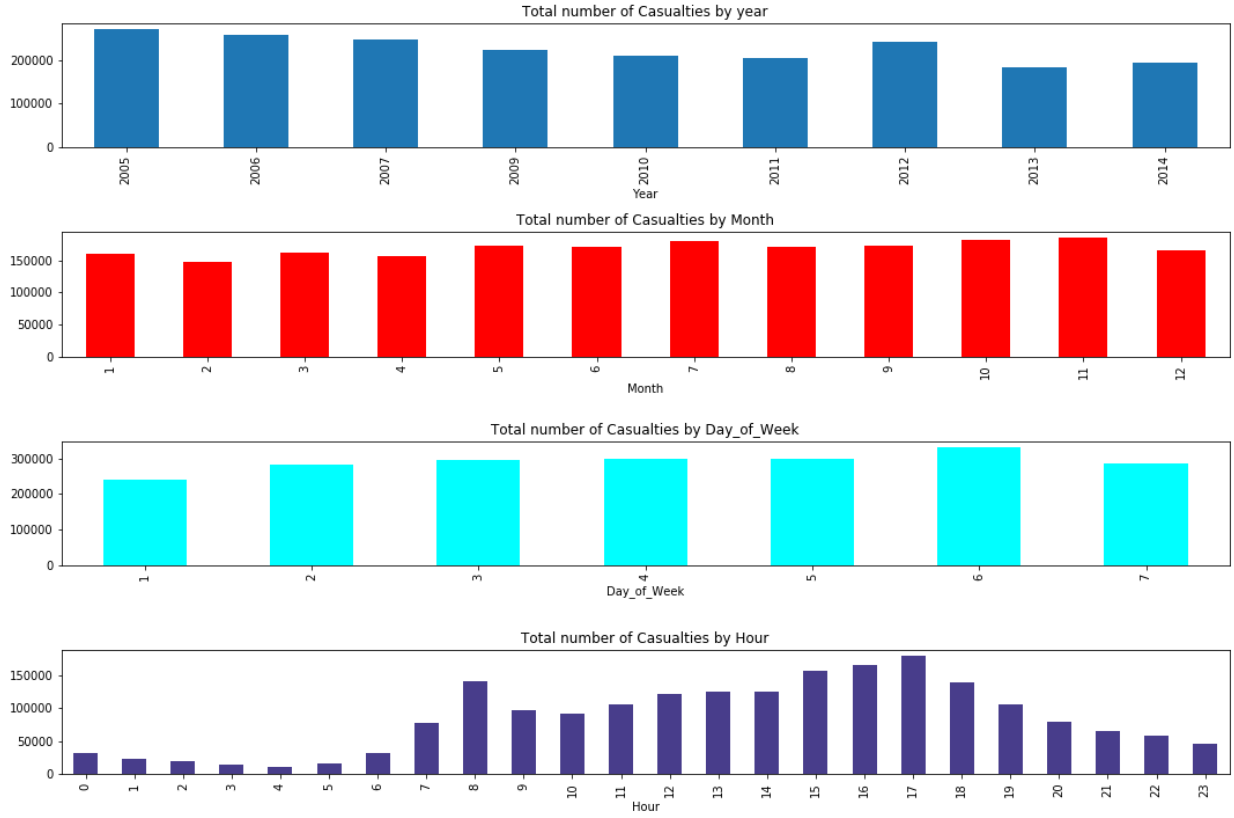


Month	1	2	3	4	5	6
number	119579	109902	121263	115490	127626	127442
Month	7	8	9	10	11	12
number	132042	123501	129576	135992	138545	123075



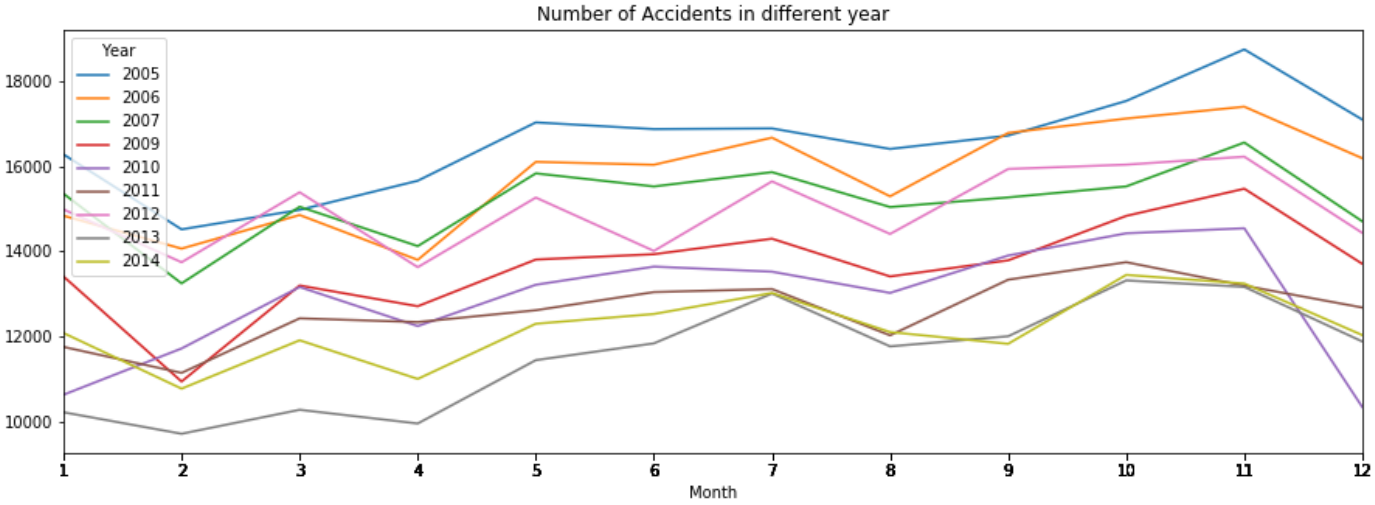
As for the data that is grouped by hour above, 4 a.m. has the lowest amount of accidents, which is 8,347. Then the number increased rapidly to the local peak at 8 a.m. with number 109,622, followed by a drop. After that, the value increased steadily to the maximum at 5 p.m. Not surprisingly, the highest value happened at the commute time while off work.

Next, we generated the bar chart based on the number of casualties, and the trend showed which is quite similar to that of the number of accident.

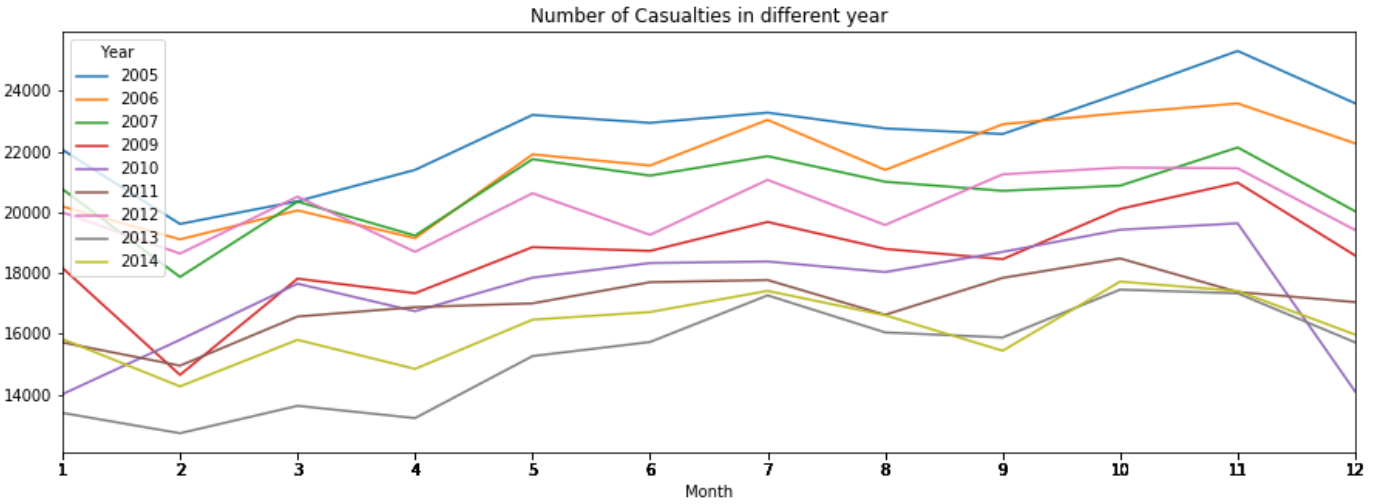


Apart from grouping the data by the previous methods, we can also group it by year and month simultaneously to make the trend in each year more clearly.

```
group_by_year_month = data.groupby(['Year',
'Month'])['Number_of_Casualties'].sum().to_frame().reset_index()
# plt.figure(figsize=(15, 15))
fig, ax = plt.subplots(figsize=(15, 5))
group_by_year_month.pivot(columns='Year', values= 'Number_of_Casualties', index =
'Month').plot(ax = ax)
# group_by_year_month.plot()
ax.set_title('Number of Casualties in different year')
ax.set_xticks(group_by_year_month['Month'].values)
fig.savefig('Number of Casualties in different year.png')
```

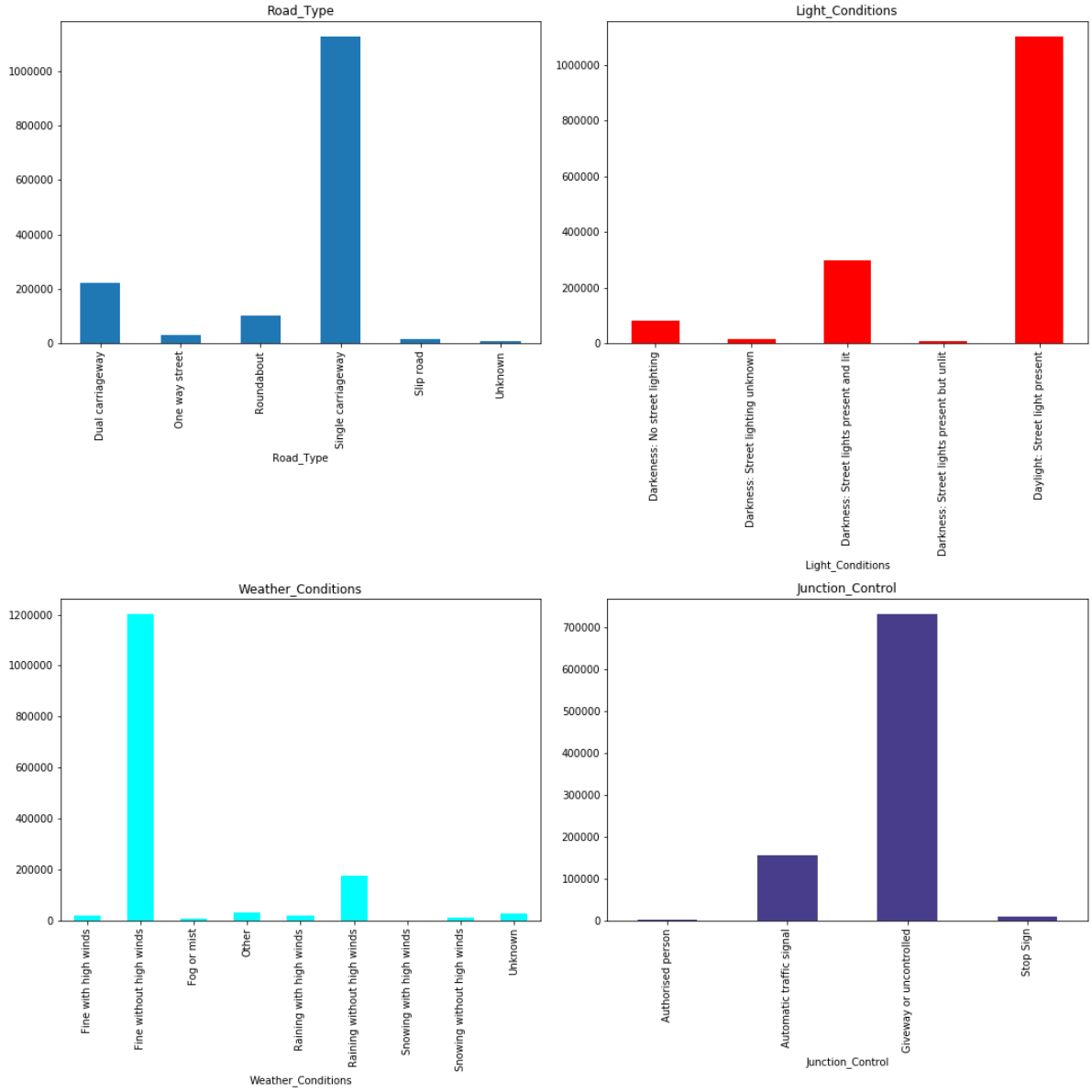


As the line chart above showed, February is the month with the least number of accidents, and November is the peak. Furthermore, the year with the highest number of accidents is in 2005, and there is no significant different tendency each year.



The similar trends can also be observed in the number of casualties.

We also are curious about how would the results show the difference between the type of road, light condition, weather condition, junction control as the variable.

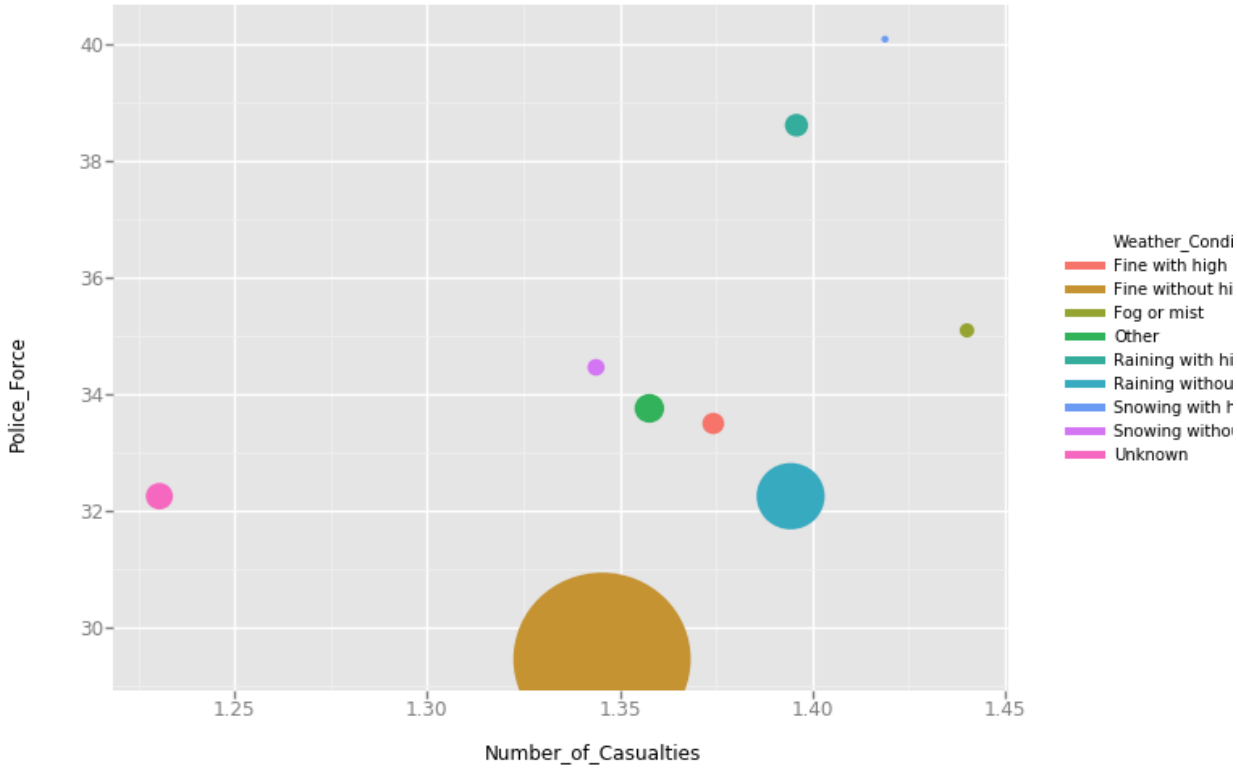


In this part, we can see that most of the accidents happened under single carriageway when we placed the road type as the variable. The highest number of accidents happened under a circumstance of daylight with street light present. Regarding the weather condition, fine weather without high winds has the highest value.

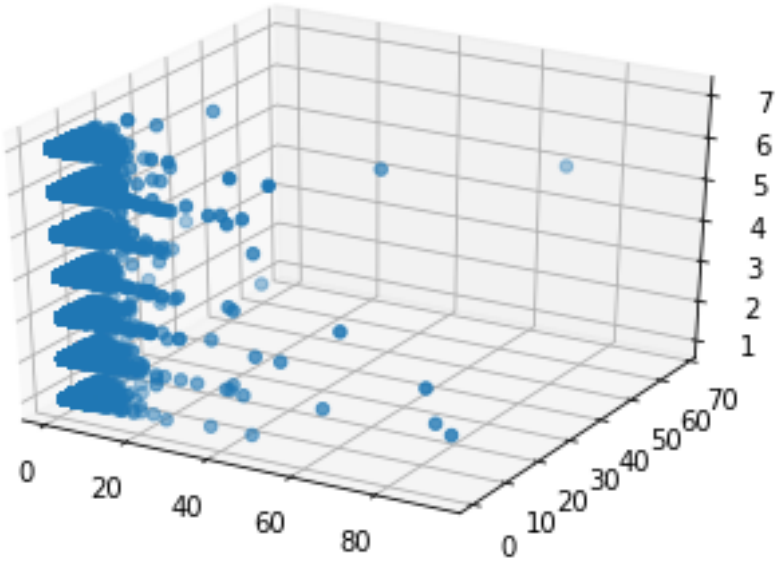
Also, in terms of the junction control chart, we can find out that the item of giveway or uncontrolled junction is the condition of most accidents happened. Although we can learn the meaning from each chart directly, but readers can get further information through comparing with other conditions. For example, firstly we found that fine weather without high winds is the most dangerous one, we can also analyse

the data on specific roads, so that we can realise that some roads might be more dangerous under some specific weather condition.

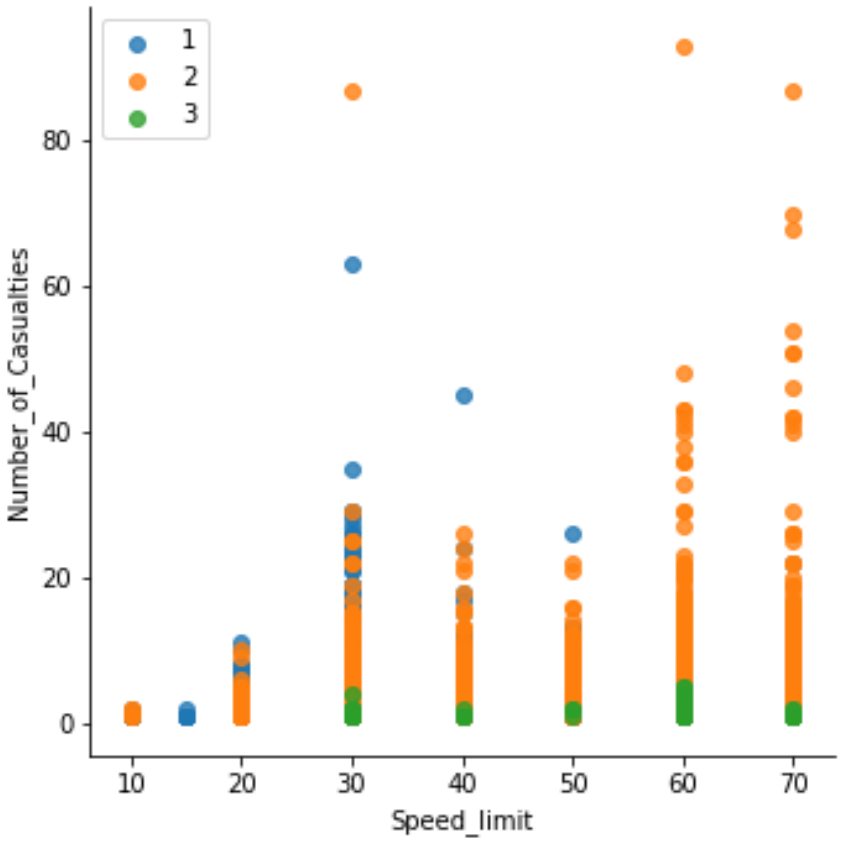
Next, we are going to review the relationship between three variables. We analyzed the data by the weather condition and used ggplot to generate the image, which showed the average of the number of police_force and the involved casualties under different weather conditions. The bigger circle means the higher percentage.



Apart from the ggplot, we can also use 3D plot to visualize the data. In jupyter notebook, we can rotate the 3D plot to view the relationship between 3 variables.



In the end of this part, we are going to observe the relationship between speed limit, number of casualties, and whether it is rural area. We use seaborn package to visualize it.



1.2 Time Series Analysis

A time series is a series of data points indexed (or listed or graphed) in time order. Most commonly, a time series is a sequence taken at successive equally spaced points in time. Thus, it is a sequence of discrete-time data. Examples of time series are heights of ocean tides, counts of sunspots, and the daily closing value of the Dow Jones Industrial Average.

Time series analysis comprises methods for analysing time series data in order to extract meaningful statistics and other characteristics of the data. Time series forecasting is the use of a model to predict future values based on previously observed values. While regression analysis is often employed in such a way as to test theories that the current values of one or more independent time series affect the current value of another time series, this type of analysis of time series is not called "time series analysis", which focuses on comparing values of a single time series or multiple dependent time series at different points in time. Interrupted time series analysis is the analysis of interventions on a single time series.

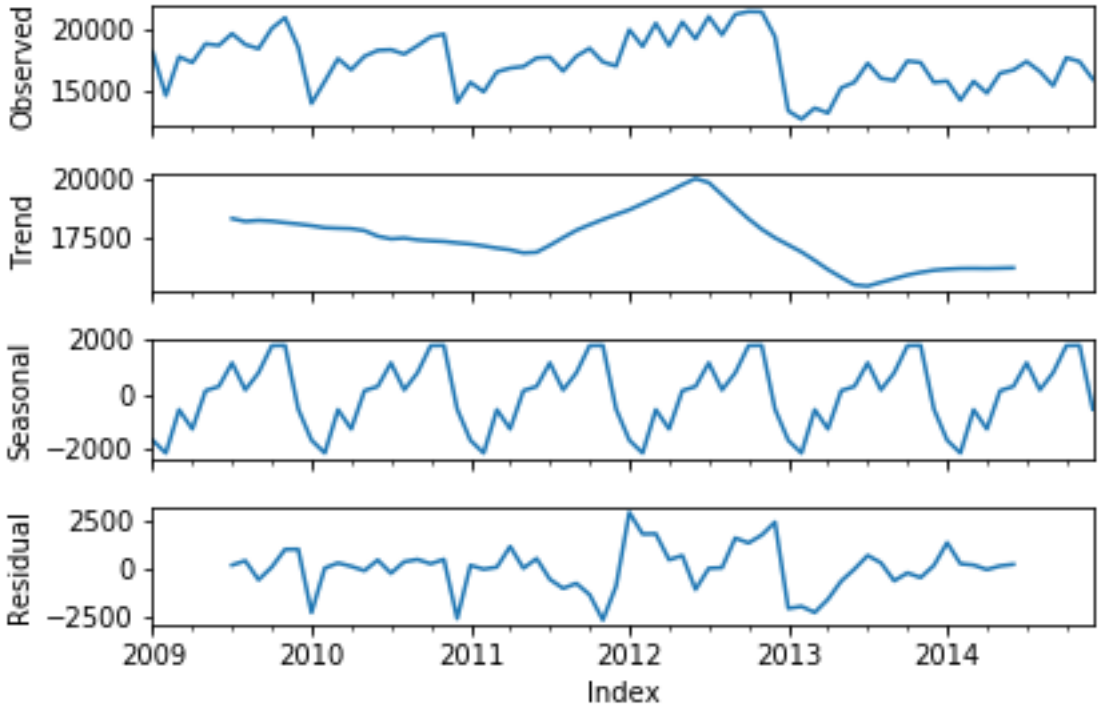
Since we do not have the data in 2008, so we choose to use the data from 2009 – 2014.

Then add an index that show the month.

```
Casualties2009_2014byYM['Index'] = pd.date_range('2009-01', periods =
Casualties2009_2014byYM.shape[0], freq = 'M')
```

After that, use the api, statsmodels.api to do the time series analysis.

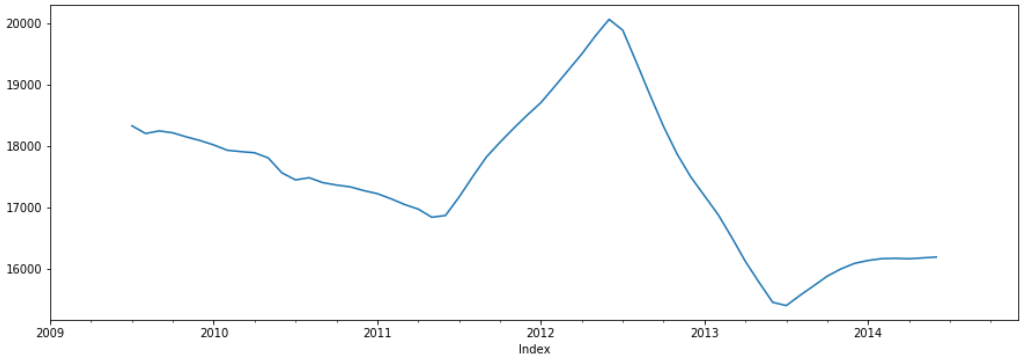
```
Casualties2009_2014byYMTS =
pd.Series(Casualties2009_2014byYM.Number_of_Casualties.values,
Casualties2009_2014byYM.Index)
Casualties2009_2014by_YM_TS_Analysis =
sm.tsa.seasonal_decompose(Casualties2009_2014byYMTS,  freq = 12)
```



The image above shows the Number_of_Casualties, and the trend of it. At the seasonal part, we can learn that the Number_of_Casualties tend to be high in summer.

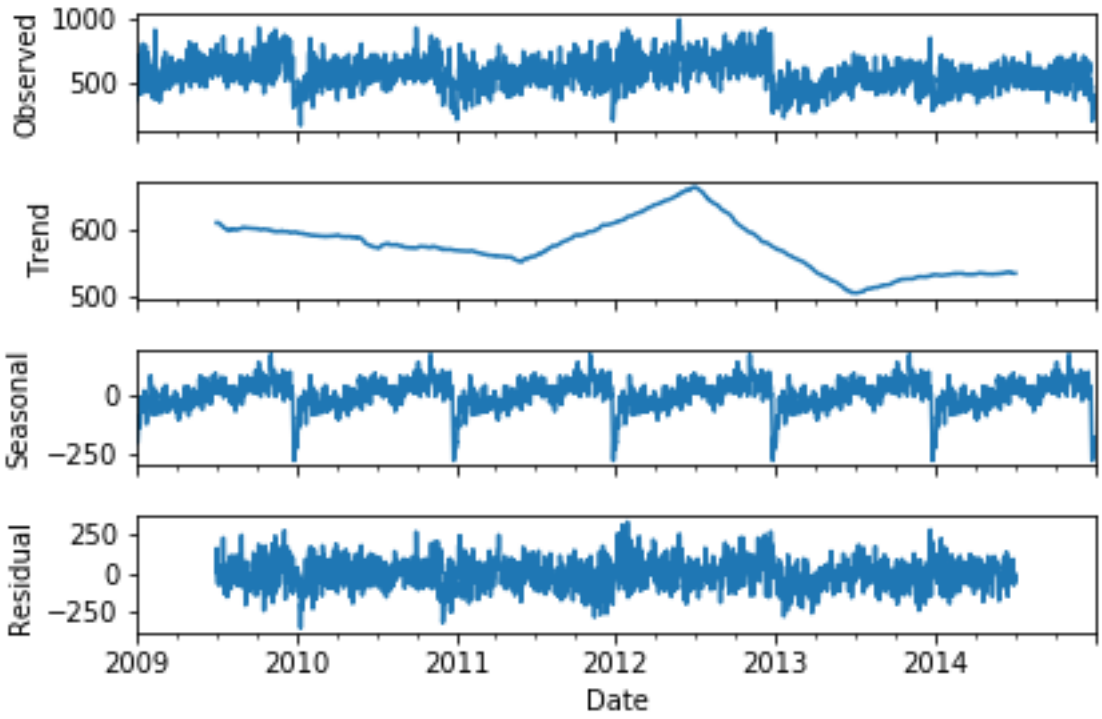
We can call the trend as well, just typed in the codes as below:

```
fig = plt.figure(figsize=(15, 5))
trend2009_2014 = Casualties2009_2014by_YM_TS_Analysis.trend.plot()
```

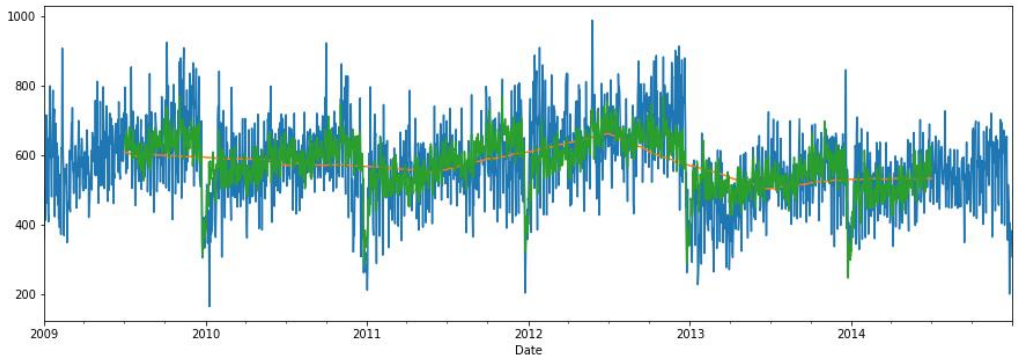


After that, we analysed the time_series by day instead of by month. We changed the 'freq' part in 'sm.tsa.seasonal_decompose' to 365 to do the time_series that one round for 365 day.

```
Casualties2009_2014 =
data2009_2014.groupby(['Date'])['Number_of_Casualties'].sum().to_frame().reset_index(
)
ts_2009_2014 = pd.Series(Casualties2009_2014.Number_of_Casualties.values,
Casualties2009_2014.Date)
stl_2009_2014 = sm.tsa.seasonal_decompose(ts_2009_2014, freq=365)
```



As the graph showed above, we got the similar trend as that in the previous one. As a result, we overlap the observed data, trend, and seasonal effects.



For the second section of the time series analysis, we are going to forecast the number of accidents based on the past data.

Firstly, we do the stationarity test,

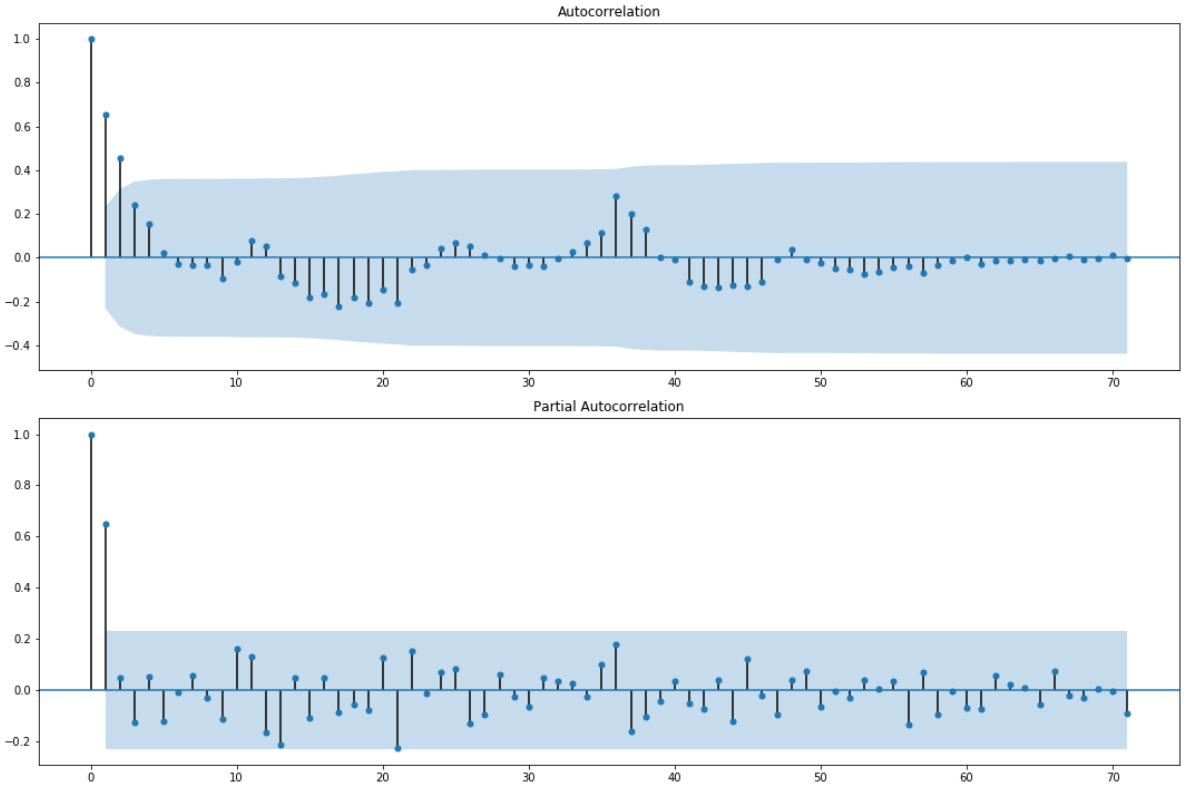
```
sm.tsa.stattools.adfuller(Casualties2009_2014byYMTS)
```

And get the result below,

```
(-3.7701501451985,
 0.0032279161250606996,
 0,
 71,
 {'1%': -3.5260046468256072,
  '10%': -2.5889948363419957,
  '5%': -2.9032002348069774},
 1033.3212894422879)
```

Since the p-value is `0.0032279161250606996` (less than 0.05), we would not do any differencing.

Then, we plot the acf, and pacf plot.



In this step, we are going to use the time series function to choose the parameter for ARMA model.

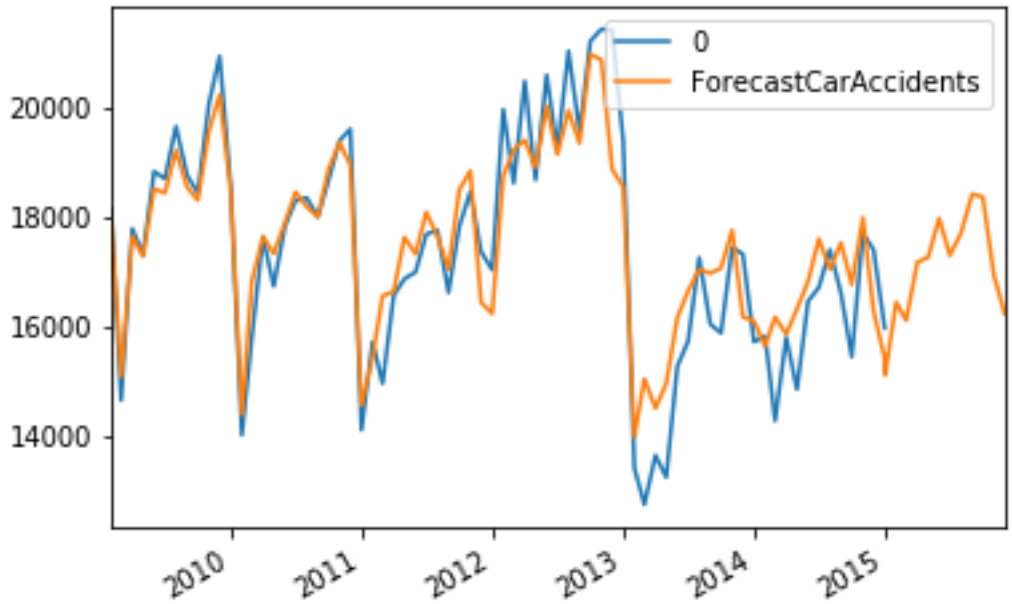
```
res = sm.tsa.arma_order_select_ic(Casualties2009_2014byYMTS.values, ic=['aic', 'bic'], trend='nc')
res.aic_min_order
```

As the result is (2, 1), therefore, we will choose to use ARMA (2, 1) model.

After that, we build the ARMA (2, 1) model to forecast the number of accident in 2015.

```
from statsmodels.tsa.statespace.sarimax import SARIMAX
(p,d,q) = (2,0,1)
mod = SARIMAX(Casualties2009_2014byYMTS.values, order = (p,d,q), seasonal_order = (p,d,q, 12))
res = mod.fit()
from dateutil.relativedelta import *
date_list = pd.date_range(start = max(Casualties2009_2014byYMTS.index), periods = 12, freq = 'MS')
future = pd.DataFrame(index = pd.to_datetime(date_list), columns = ['ForecastCarAccidents'])
pred_df = pd.concat([Casualties2009_2014byYMTS.to_frame(), future])
pred_df['ForecastCarAccidents'] = res.predict(start = 1, end = len(pred_df.index))
```

Then plot the forecast out as below.



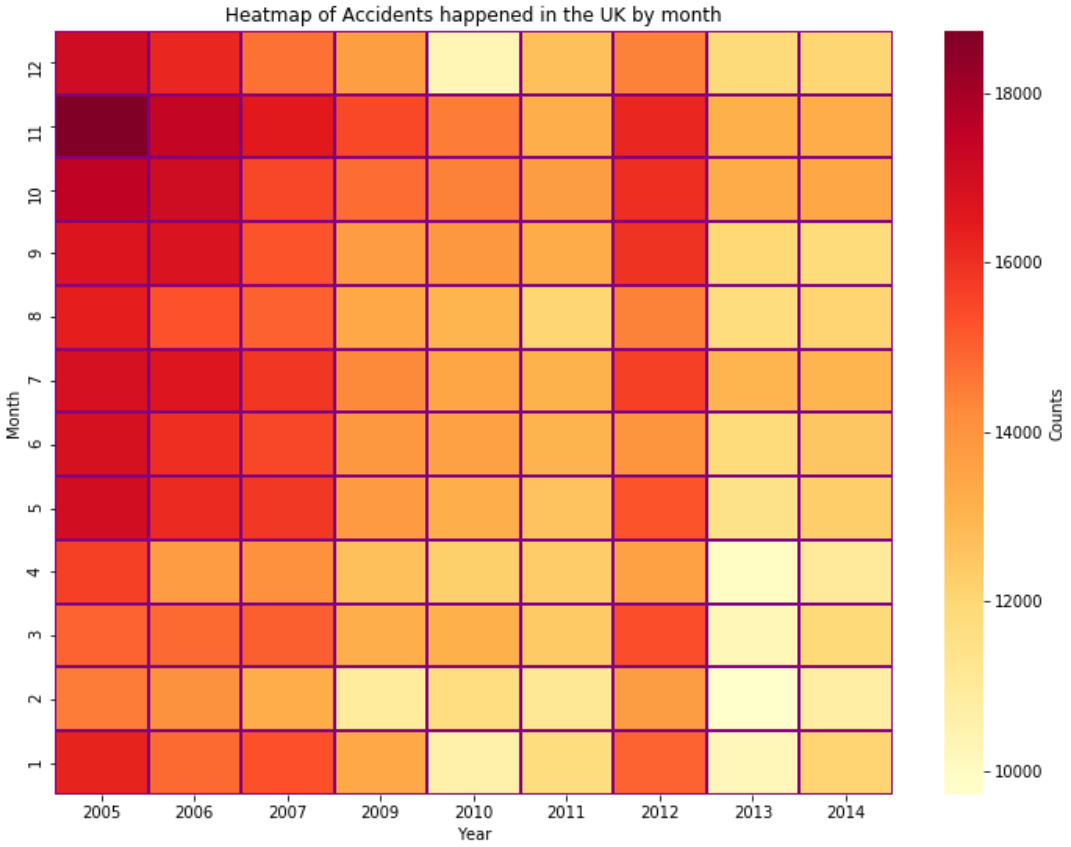
The blue line is the original data, and the orange line is the value we forecast.

January	Feburary	March	April	May	
15104.154785	16445.622334	16110.557161	17181.299407	17268.475558	17990.533073
17307.701858	17716.232557	18431.642414	18382.706059	16937.745873	16234.594749

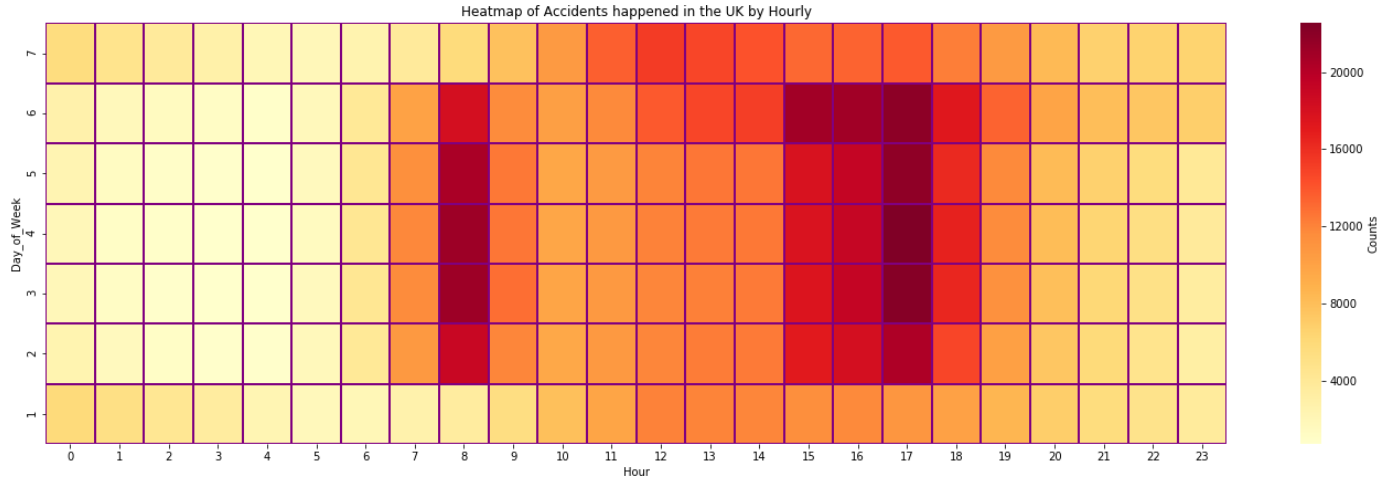
1.3 Heat Map

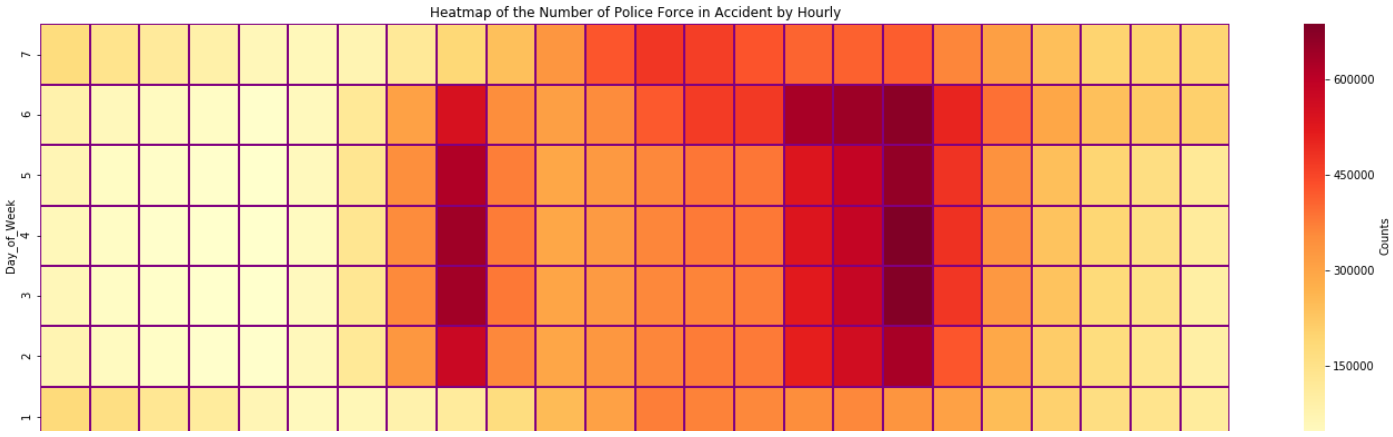
A heat map is a two-dimensional representation of information with the colours and can help the user to visualize simple or complex information. Heat maps are good for showing variance across multiple variables, revealing any patterns, displaying whether any variables are similar to each other, and for detecting if any correlations exist in-between them. For example, heat maps can be used to show the changes in data over time if at least one of the rows or columns are set to time intervals.

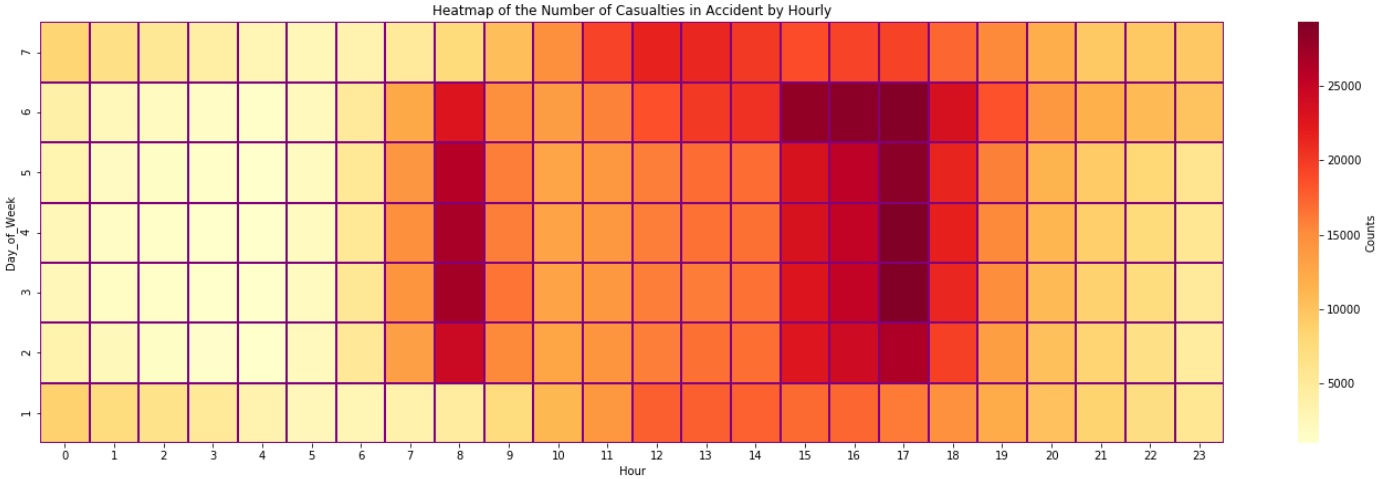
In our case, a heat map could be used to compare the accidents which happened across the year in different months, to see when is the peak period of car accidents happened in the UK. The rows can list down the year to compare, the columns contain each month and the cells include the total number of accident values. We could easily see the changes from the heat map. Therefore, this makes a heat map a useful tool for data analysis. Heat maps and big data are allies in providing meaningful insights, as heat maps are capable of displaying complex data in a simple visual format for the end user.



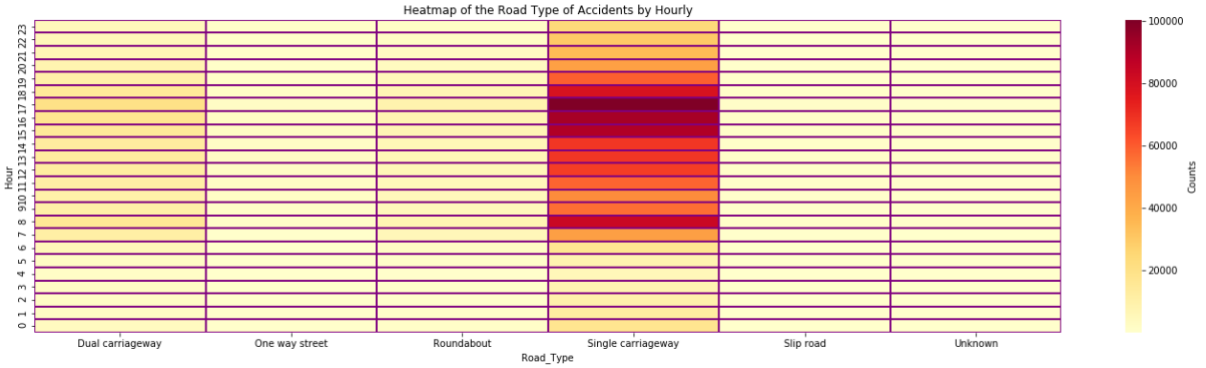
To see it in detail, we changed time intervals from monthly based to daily one, then we can interestingly find out that there is a high-positive correlation between the total number of accidents, the number of police force, and the number of casualties during the following heat map. To be more precise, the car accidents in the UK highly happened during the weekdays, from Monday to Friday, with 7 to 9 o'clock in the morning and 15 to 19 o'clock as peak hours in the afternoon. Following this point of view, we could estimate that it may need more police supports during those time periods to improve the conditions.

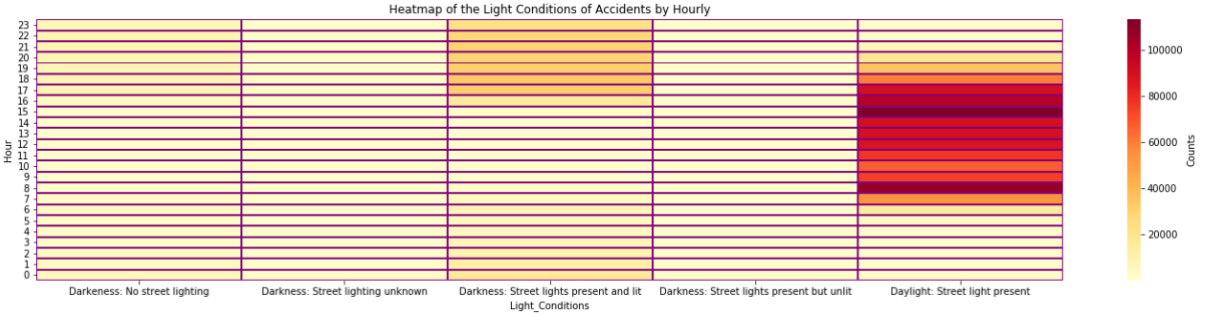


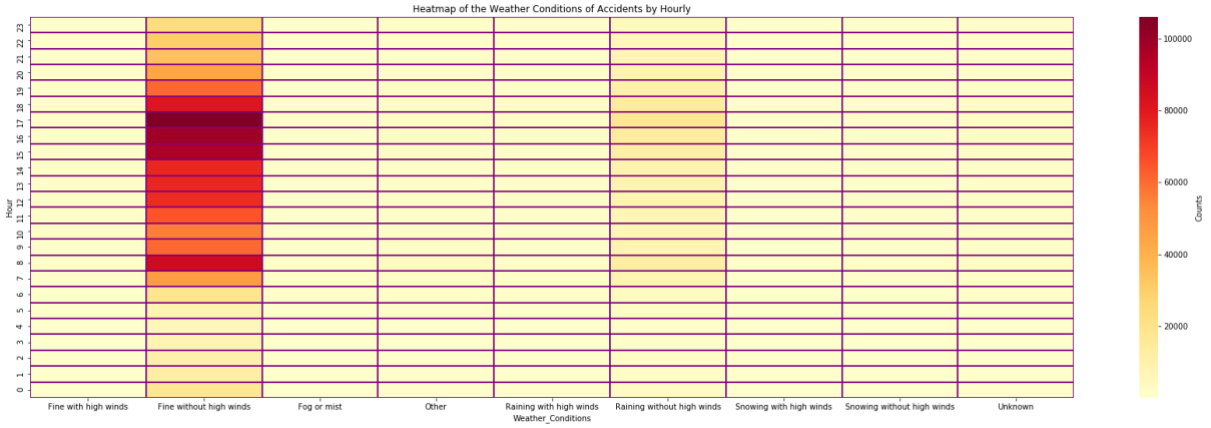


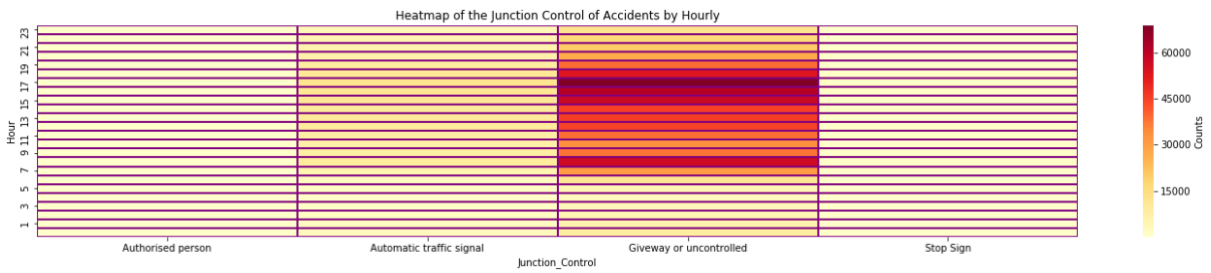


Furthermore, considering the causes of car accidents, we arranged four main conditions in the heat map to compare different situations for the daily view. As the following figures showed, the high percentage of the accidents happened from 7 to 20 o'clock in the single carriageway, in give way or uncontrolled conjunctions, with daylight and street light present, and fine weather without high winds. With the acknowledgement, it could be helpful for reviewing road design and the research for the safer driving campaigns for the UK government.









1.4 Map of Traffic Accidents

Traffic Accidents Density Analytics

From the very beginning, in order to visualize the data on the map, we used the geographic locations from the traffic accidents and put them into the graph. Given the image below (Figure1), it shows that the spots of traffic accidents cover pretty much the entire Britain. In fact, we have accidents reported at nearly every area in the UK, which is quite astonishing. In the part of data processing, we use the latitude and longitude of the location of each accident as the basis for drawing density maps. Due to the long-term and a large amount of data, we tackled this information to draw a complete picture of all the roads in the UK.

In addition, it can be noted that most of the highlights are concentrated in the British metropolises such as London, Birmingham or Manchester. Accordingly, accident rates have a positive correlation with the population, in another word, more people brought out higher traffic flow.

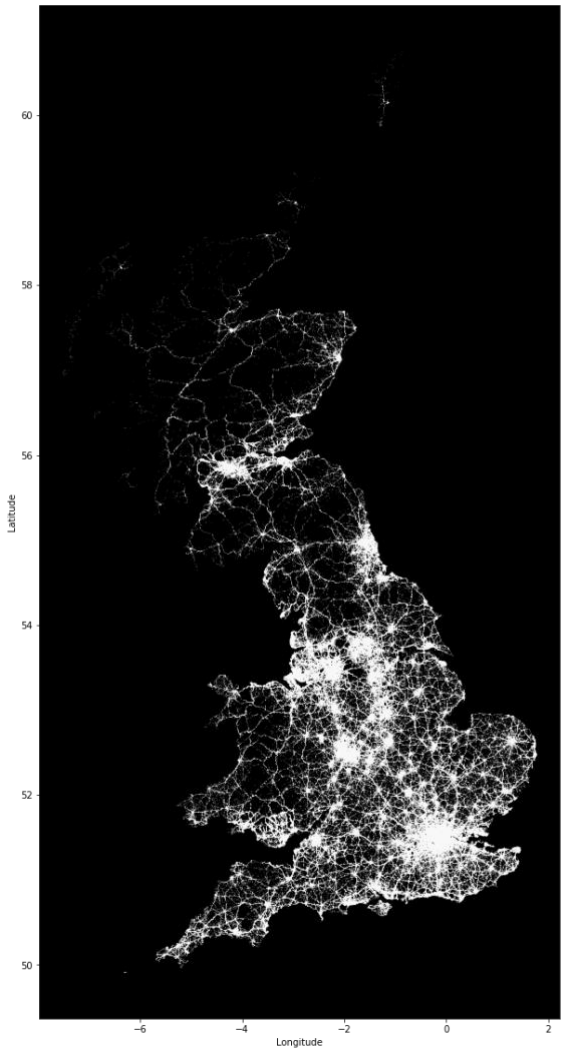


Figure1. Traffic Accident Density Map

Accidents Severity Density Analytics

In this part, we are going to discuss in depth the different column of the traffic accident data. Firstly, we studied the data according to the severity of traffic accidents. The severity is divided into three levels: 1, 2 and 3, as the number getting higher, which means the worse condition of the accident. Secondly, from the quantity of data, we showed the number of accidents as a histogram (Figure2), and we found that the level 3 accounts for the most proportion.

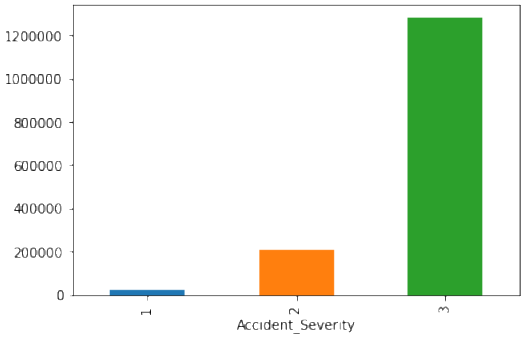


Figure2. The Histogram of Traffic Accident Severity

Therefore, when we draw the results on a density map, it can be recognized that most of the points are clustered on a graph with severity level 3. From three maps as below (Figure3), it is clear to learn that there are many points centred on the location of the metropolis, corresponding to the above-mentioned relationship between traffic flow and the number of accidents.

Figure3. The Density Map of Accident Severity

Speed Limits Density Analytics

In this section, we used the speed limit of traffic accidents provided by the data as an indicator of differentiation, and draw it into a histogram and a density map for analysis. For the next step, we can see that on the roads of 30 speed limit and their accident rate is the highest one, followed by the speed limit of 60, and the units used here are miles per hour.

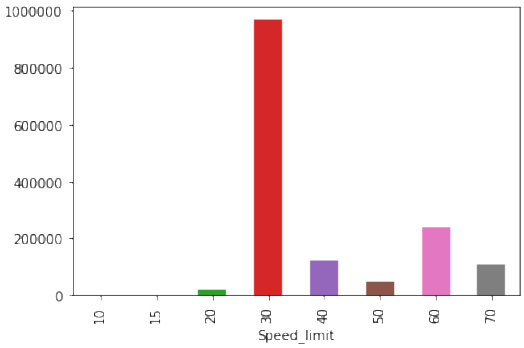


Figure4. The Histogram of Speed Limits

Then we look at the density map drawn by the speed limit (Figure5). Here we can see that in the case of the speed limit of 10 and15, there is almost no points can be found on the density map. Up to the speed limit of 30 is the highest peak. In addition, because the location of the speed limit 70 is mostly on the highway, the density map shows the geographical distribution of the major highways.

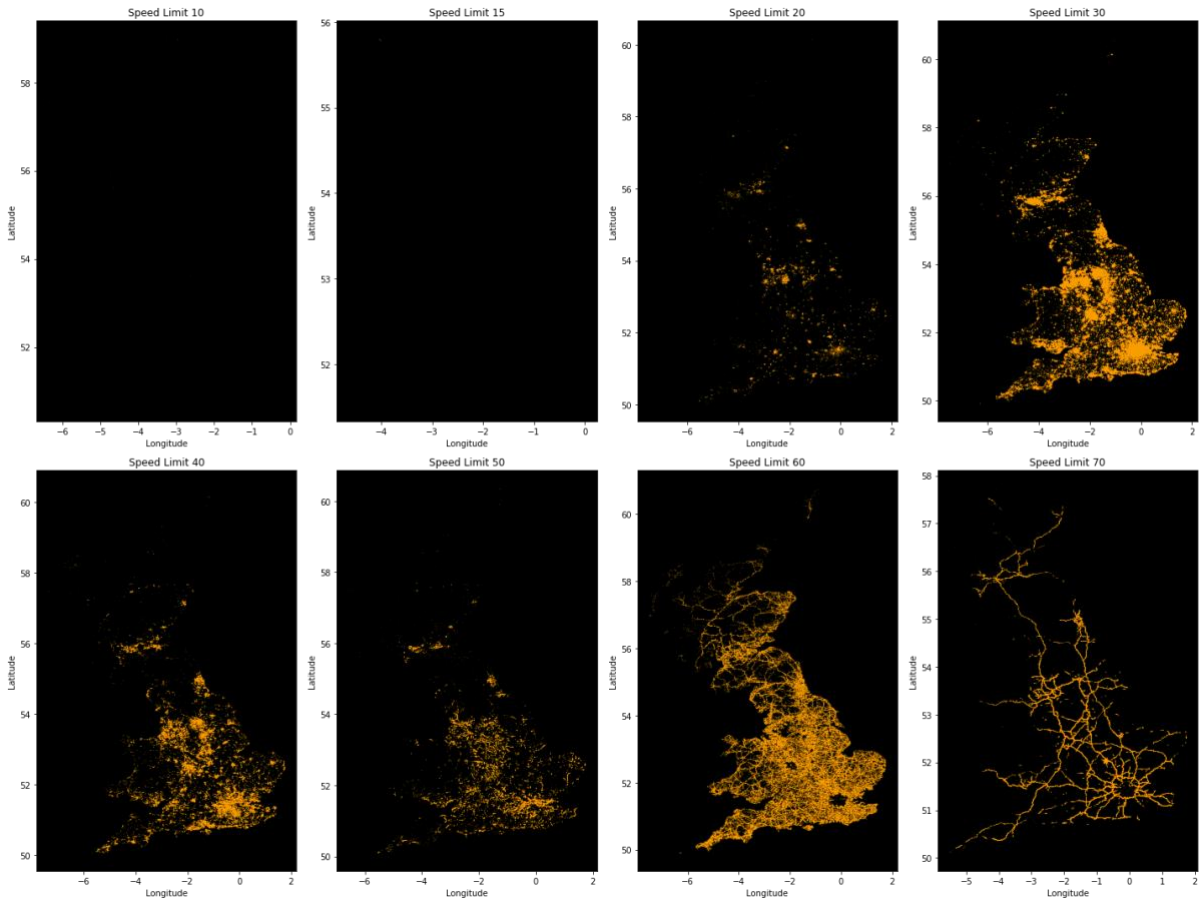


Figure5. The Density Map of Speed Limits

Road Types Density Analytics

In the final part, we consider the types of roads that occur in traffic accidents as different factors. According to the number of roads, we can find that the single carriageways account for the majority, followed by the dual carriageways.

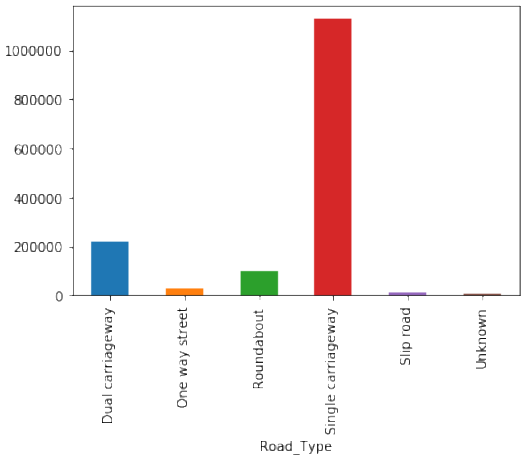


Figure6. The Histogram of Road Types

Furthermore, from the Figure7., it is worth to notice that most of the roads are single carriageways scattered throughout the UK and the dual carriageways are located on the highway, and the rest roads are sporadically dispersed.

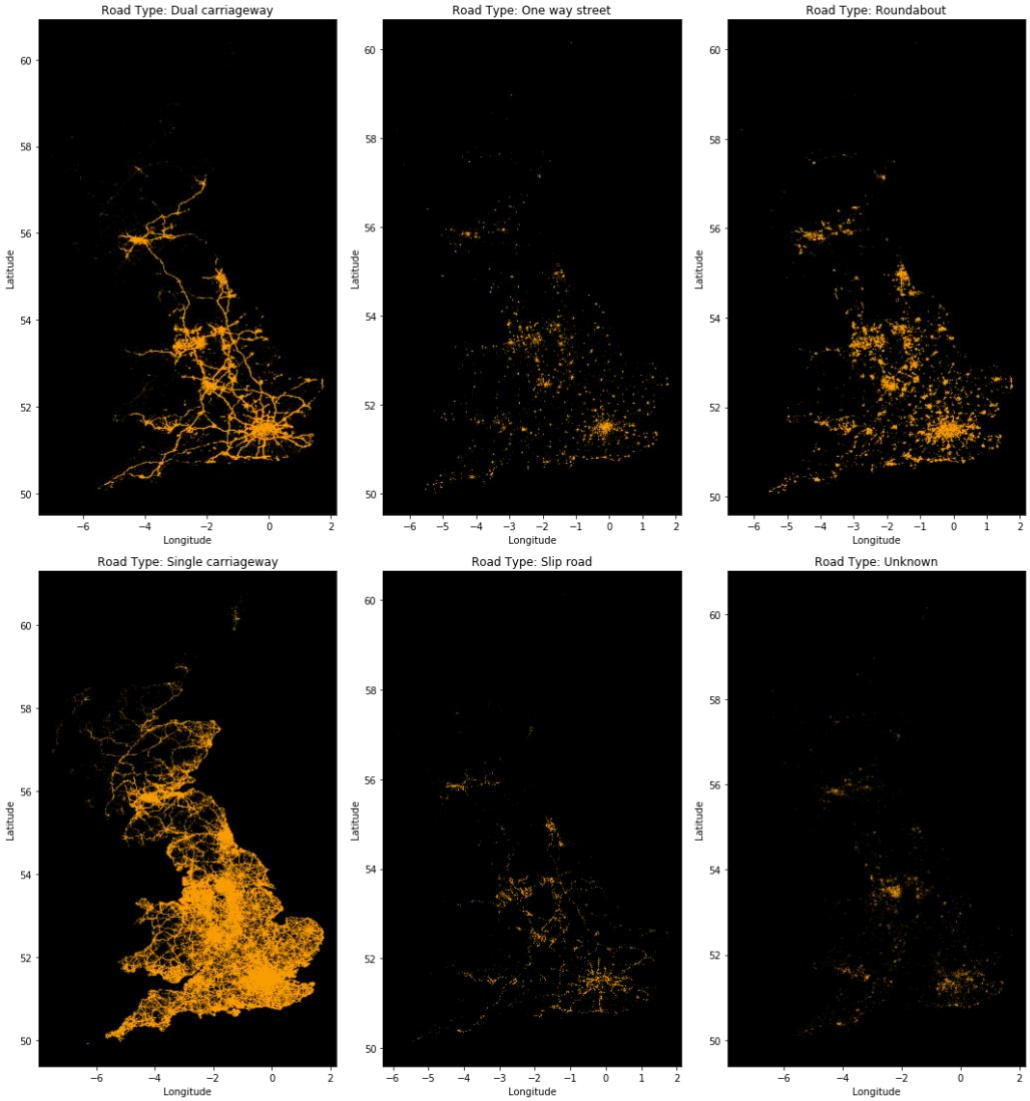


Figure7. The Density Map of Road Types

Traffic Accidents and Flow Density Analytics

Additionally, using the latitude and longitude of the original data to generate the density maps, the location of traffic flow can also be placed on the map, and the intensity of traffic accidents in the area can be displayed on a heat map, and traffic flow on a heat maps can be compared to the in the intensity of traffic accidents to observe the relevance between them.

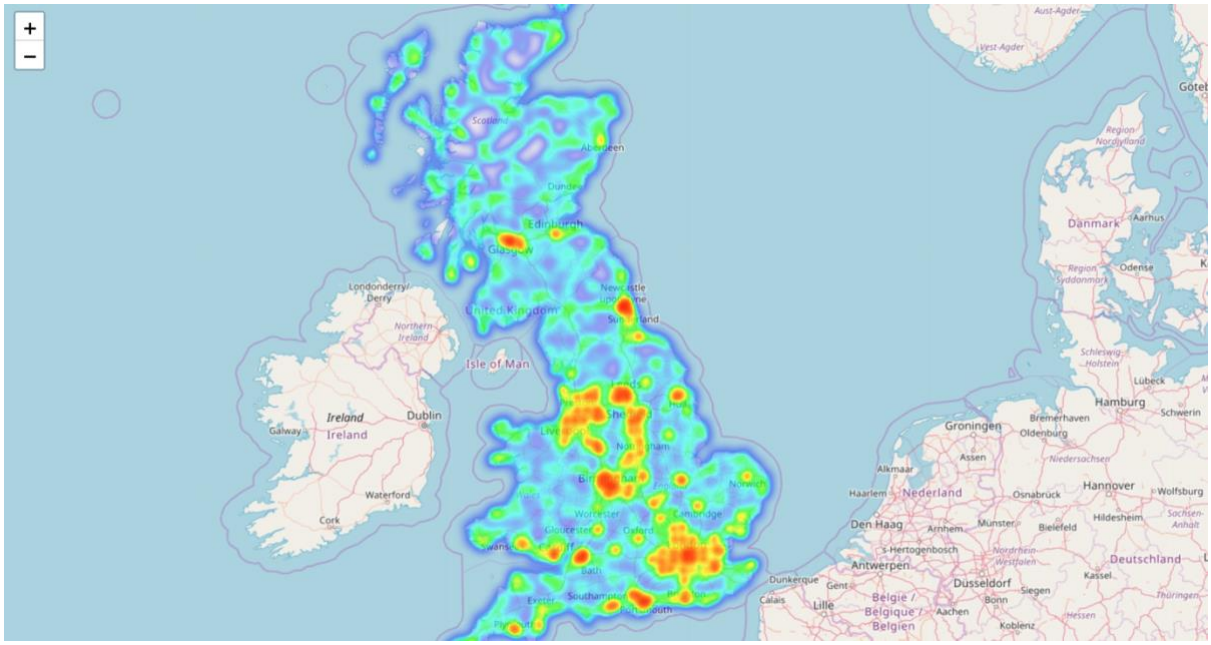


Figure8. The Heat Map of Traffic Accidents

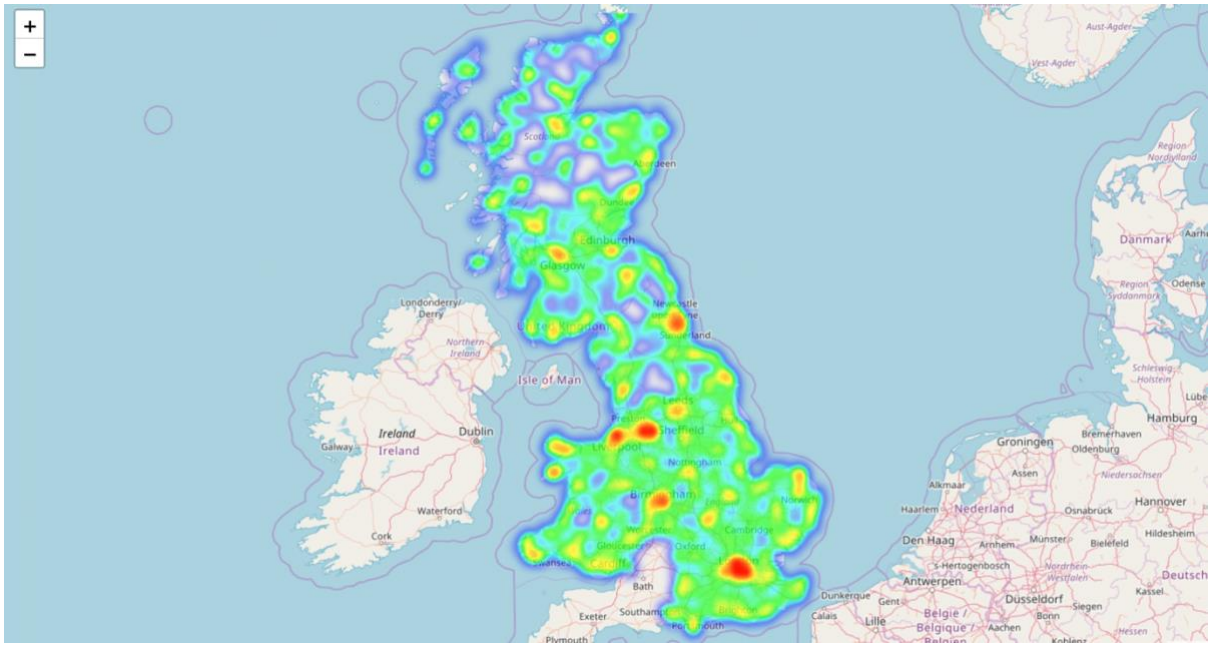


Figure9. The Heat Map of Traffic Flow

During the processing of traffic accidents, geographic locations play a very important role. Through the numbers and figures, it seems reasonable to analyze the composition of data, but through the combination of data and maps can show more intuitive analysis and relevance. For example, comparing the heat map of traffic accidents and flow (Figure8, Figure9), it is significant that the higher traffic flow it is, the higher traffic accidents it becomes by inspecting the red area, which means the high intensity of traffic accidents or flow, and overlapping the above parts to prove the relationship of them.

Conclusion and Recommendations

Conclusion

In the descriptive statistic part, we have learned that total causalities decreased since 2005, although we had a lack of data in 2008, but we can tell from the chart that the number increased in 2012, then down for the next two years.

From the month perspective, November is recognized as the month for happening the highest number of causalities. In the week of the day aspect, it happened the most on Fridays, and during the 24 hours angle, the commute time at around 8 am and 5 pm had the highest values as well.

With the comparison of the different variables charts, fine weather without high winds, day light with street light presented, signal carriageway, and uncontrolled junctions had the highest causalities number as well.

Regarding the time series analysis, we can find that there is a seasonal trend. In addition, we also analyzed the data in different ways, for instance, the heat map, various traffic maps, to cross comparing the results and understand the data in a deeper way.

Recommendations

Therefore, from the overall perspective, it is critical to emphasize the commute safety importance with caution, to pay extra attention and arrange the police force according to the other variables, for example, during the day time from the government's angle. In addition,

Additionally, according to the department of transport's report in the UK, they claimed it has long been known that a considerable proportion of non-fatal causalities are not know to the police, which is different from the hospital survey and records. (Road accidents and safety statistics guidance - GOV.UK, 2018)

As a result, it is fundamental to collect the data completely in order to get close to the accurate traffic conditions, so that not only the government can manage the resources to improve it in a precise way, but also the people who live in the UK can be extra care of their own safety. Even though the topic is related to multiple parameters, for instance, Clarke et al. (2010) said that one of the factor of traffic accident is driver's age and experience. However, we can still start making the efforts on the finding from the data we got so far.

In the content of this report, we have provided an overall perspective of the data we analyzed, for the future suggestion, if researchers would like to discuss further details related to the topic, they can find more other factors which might influence the main topic the most to get more advanced results.

Bibliography

Aljanahi, A.A.M., Rhodes, A.H. and Metcalfe, A.V., 1999. Speed, speed limits and road traffic accidents under free flow conditions. *Accident Analysis & Prevention*, *31*(1-2), pp.161-168.

Andersson, A. K., and Chapman, L. (2011). The impact of climate change on winter road maintenance and traffic accidents in West Midlands, UK. Accident Analysis & Prevention, 43(1), 284-289.

Clarke, D. D., Ward, P., Bartle, C., and Truman, W. (2010). Killer crashes: fatal road traffic accidents in the UK. Accident Analysis & Prevention, 42(2), 764-770.

Clarke, D. D., Ward, P., Bartle, C., & Truman, W. (2010). Older drivers' road traffic crashes in the UK. Accident Analysis & Prevention, 42(4), 1018-1024.

Dave F. H., (2017). *Kaggle*. [ONLINE] Available at: https://www.kaggle.com/daveianhickey/2000-16-traffic-flow-england-scotland-wales/data. [Accessed 12 April 2018].

Golob, T.F. and Recker, W.W., 2003. Relationships among urban freeway accidents, traffic flow, weather, and lighting conditions. *Journal of transportation engineering*, *129*(4), pp.342-353.

Jones, P. M. (1991). UK public attitudes to urban traffic problems and possible countermeasures: a poll of polls. Environment and Planning C: Government and Policy, 9(3), 245-256.

Jeff S. (2015). MeasuringU: 7 Ways to Handle Missing Data. [online] Available at: https://measuringu.com/handle-missing-data/ [Accessed 10 Apr. 2018].

Nagatani, T., 1993. Effect of traffic accident on jamming transition in traffic-flow model. *Journal of Physics A: Mathematical and General*, *26*(19), p.L1015.

Pai, C. W., and Saleh, W. (2007). An analysis of motorcyclist injury severity under various traffic control measures at three-legged junctions in the UK. Safety science, 45(8), 832-847.

Plainis, S., Murray, I. J., and Pallikaris, I. G. (2006). Road traffic casualties: understanding the night-time death toll. Injury Prevention, 12(2), 125-138.

Road accidents and safety statistics guidance - GOV.UK (2018). Reported road casualties Great Britain, provisional estimates: July to September 2017 report. [ONLINE] Available at: https://www.gov.uk/government/statistics/reported-road-casualties-great-britain-provisional-estimates-july-to-september-2017. [Accessed 11 Apr. 2018].

Appendices

Data Sources:

- From Kaggle (Member Requirement): https://www.kaggle.com/daveianhickey/2000-16-traffic-flow-england-scotland-wales/data
- From Google Drive (Download Directly): https://drive.google.com/open?id=1zcMSURU4C5cdf_lhl-_9dDxVcyr8sJZc