Data Analysis of

Canada Car Accidents: 1999 - 2014

Term Project Report

SCS 3250-015, Group 3

**Boris Korotkov**

**Chaewon Kim**

**Patricia Luo**

**Ryan He**

April 2018

Table of Contents

[Project Overview 2](#_Toc511055699)

[Data Preparation 2](#_Toc511055700)

[What was the data source? 2](#_Toc511055701)

[How good was the data quality? 2](#_Toc511055702)

[How did we procure the data? 3](#_Toc511055703)

[What tools/code did we use to prepare for our analysis? 3](#_Toc511055704)

[What challenges did we face? 3](#_Toc511055705)

[Analysis 5](#_Toc511055706)

[What trends, correlations and/or patterns we saw in the data? 5](#_Toc511055707)

[Individual analysis and the result we discovered: 5](#_Toc511055708)

[Conclusions 25](#_Toc511055709)

[What we learned from the data set? 25](#_Toc511055710)

[What …. 27](#_Toc511055711)

[Appendices 28](#_Toc511055712)

[A-1 sample dataset 28](#_Toc511055713)

[A-2 Sample Codes 28](#_Toc511055714)

# Project Overview

We analyzed the dataset containing collision data for car accidents in Canada from 1999 to 2014. Given various information pertaining to collision, vehicle and person, we conducted analyses to understand the trends and patterns of collisions and discover conditions that are more prone to accidents.

We created an open repository on [GitHub](https://github.com/rynho/3250-G3) containing background information, codes (via Jupyter Notebook) and reports to allow members to easily collaborate and track changes.

# Data Preparation

## What was the data source?

We selected the [Canadian Car Accidents 1999-2014](https://www.kaggle.com/tbsteal/canadian-car-accidents-19942014/data) dataset (provided by Transport Canada) from Kaggle: an open platform for predictive modeling and analytics competition using real life datasets provided by companies and users.

## How good was the data quality?

The overall quality of the dataset was decent except there were several missing values, unknown/inapplicable values and different data types. For example, besides the 17 vehicle types defined in ***vehicle type*** (numbers in ‘str’), there were four nonnumeric values:

* ‘NN’ not applicable, e.g., “dummy” vehicle record created for the pedestrian;
* ‘QQ’ choice other than defined vehicle types;
* ‘UU’ unknown values; and
* ‘XX’ jurisdiction does not provide the data.

Since every column contained some nonnumeric values, Pandas treated them as strings, which required manipulation of data when trying to utilize numeric-only features (e.g. scatterplots).

Moreover, the values in the **c\_wday (Day of week)** column represented specific days of the week (e.g. 1 = Monday, 7 = Sunday) instead of representing days of month, which created some difficulty when trying to use time series index for our analysis (detail in below sections).

## How did we procure our data?

The full dataset, in **csv** format (including header), included around 6 million records of car accidents in Canada over the period of 15 years from 1999 to 2014 and the pdf file was the data dictionary. Each row of the dataset represented registered records for each person involved in those specific collisions, and there were 22 columns presenting information pertaining to collision, vehicle and person level data elements of each collisions.

We downloaded the dataset (.csv) from the URL mentioned above, used system command to perform some sense check, then used *pandas.read\_csv* method to load the full dataset into a *DataFrame*. See Appendix **[A-2.R1]**.

## What tools/code did we use to prepare for our analysis?

We utilized Python (Pandas, NumPy) for data cleaning and preparing. Some of our data cleaning include the following but are not limited to:

* Converted data type from ‘str’ to ‘int’ for certain numeric columns;
* Converted non-numeric values to NaN and used drop.na() as they do not present significant insights; and
* Composed a ***date*** column combining ***year*** and ***month*** for possible usage in the analysis. See **[A-2.R2]** for the sample code.

## What challenges did we face?

Some of the challenges we faced while preparing the data for analysis include the following but are not limited to:

* The dataset did not include ‘day’ portion of (YY/MM/DD) but a column ***day of week (c\_wday)*** representing specific days of the week (e.g. 1 = Monday, 7 = Sunday). We decided to combine ***year*** and ***month*** column to create a time series and kept the ***day of week*** for certain analyses.
* We experimented two different ways to engineer the **date** column and noted their advantages and disadvantages. We discovered that *pandas.DatetimeIndex* method is faster, but the result included ‘day’ part (YY/MM/DD e.g. 1999/01/01) when it actually represents *1999 January, Monday*. On the contrary, while the *pandas.PeriodIndex* with ‘M’ frequency accurately shows only the year and month, the method is about 10 times slower given the huge dataset. We decided to use one or the other in our individual analyses to compare the difference.
* All values in the dataset were ‘str’ type. We decided to change the data type when we created a duplicate of the dataset/sliced data to proceed with further analysis accordingly.
* Considering that some unknown values may present significant insights into our analysis, we decided to keep them in the general dataset but treat them differently in individual analyses. However, the unknown values in date/time columns are dropped as they not only represent tiny portion of the data and are insignificant, but also complicate the analysis.

# Analysis

## What trends, correlations and/or patterns did we see?

In the following section, we addressed the most interesting findings. However, refer to the attached Jupyter Notebooks for all details, and the analyses not mentioned here. We analyzed the nature of the collision records by considering different factors/conditions, as well as the relationships/correlations between those factors/conditions to identify trends and patterns. Before doing each analysis, we sliced the dataset pertaining to only the columns needed in order to maintain original data integrity, make debugging easier, and avoid reloading the huge dataset which takes longer.

## Individual analysis and the result we discovered:

1. **Collision trend, correlation and seasonality**

To analyze the collision trend over the time, we created a ‘date’ column using the pd.*PeriodIndex* function (collision year and month as arguments) and set it as the index of our dataset (see sample code in **[A-2.B1]**).

* 1. Overall collision diagram

In order to examine the trend for both fatal and non-fatal collisions, we created two more corresponding columns by using the collision severity (**c\_sev = 1 fatal , c\_sev =2 non-fatal)** column. (see sample code in **[A-2.B2]**). Then we grouped the *‘fatal’* and ‘*non-fatal’* columns by date and summed the number of collisions (see sample code in **[A-2.B3]**).



When we plotted the result we noticed that collisions have a seasonal pattern, but it was difficult to identify the exact pattern since the time period was a 15-year scale. Thus, we created another plot with a smaller range (most recent 5 years). From the graph above, we identified a seasonality for non-fatal collisions as 12 months; the number of collision peaks is in mid-year and decreases in the first quarter (Q1).and off-peak is in the first quarter (Q1).

While we were able to see a seasonality pattern for non-fatal collisions, the fatal collision line resembled that of a straight line. We assumed that this is due to the very small number of the fatal collisions in comparison to non-fatal ones, so we created a plot for fatal collisions only for the most recent 5 years of data.



The graph above confirmed our assumption that fatal collisions have the seasonal pattern as non-fatal collisions e.g. Increases and peaks mid-year and starts to decrease and reaches off-peak in Q1.

* 1. Fatal vs. Non-fatal collisions: Autocorrelation & Correlation

To further confirm our seasonality assumption of fatal and non-fatal collisions, we examined the autocorrelation of fatal and non-fatal collisions for the most recent 5 years (see sample code in **[A-2.B4]**).



By looking at the autocorrelation diagram (left), we noticed that fatal and non-fatal collisions have a strong autocorrelation with a 12-month lag thus solidifying our assumption. Furthermore, we used the df.corr() to find out how strongly they were correlated. The correlation diagram (right) with the regression line shows that they are strongly correlated; in fact, the correlation coefficient was determined as 0.87. (see sample code in **[A-2.B5]**)

1. Number of vehicles involved in collision **(c\_vehs)** by Week Day (c\_wday) and Hour (c\_hour)

To gain better understanding, we counted the records for certain values in *c\_vehs*, and confirmed that every vehicle involved in that specific accident had its own record (e.g., there are 58 records in the dataset have 57 vehicles involved, the outstanding one could be a pedestrian.

We wanted to observe the pattern of number of vehicles involved in collision over **(c\_vehs)** over *Week Day* ***(c\_wday)***and *Hour* ***(c\_hour)****.* We grouped the **c\_vehs** by **c\_wday** and the results are shown below (left). In a span of a week, number of collisions increase towards Friday (peak) and starts decreasing towards Sunday. (see sample code in **[A-2.B6]**).



To observe the trend in regard to hours, we grouped the **c\_vehs** records by **c\_hour**. From the graph above (right), we saw that there are two local maxima: one at 8:00-8:59AM and the second at 16:00-16:59pm. These peaks reflect morning and afternoon rush hours (see sample code in **[A-2.B7]**).



The graph on the left shows the trend of number of vehicles involved in collision over time. The maximum number of vehicles involved in a single accident was 77 but since it was hard to observe a pattern, so we selected the top 5 frequent number of vehicles involved in collision to do our analysis. We can see that the overall number of vehicles involved in collision decreased over time and that 2 vehicles were most likely to be involved in a collision followed by 1 and 3 vehicles. The graph on the right shows the trend of number of vehicles involved in collision at different hours. The two peaks are prominent at Hour 8 and Hour 16 which resembles the exact same pattern for the total number of vehicles involved in collision at different hours.

1. **Road configuration, weather, road surface and traffic control**

The further analysis is similar in nature and will demonstrate the impact of four different collision factors to collision and fatality rate.

* 1. Road configuration **(c\_rcfg)**

In addition to generating a bar plot showing the relation between collision severity and road configuration, we also plotted a second axis using the *twinx()* method of a subplot to show the fatal percentage for collisions happened in different road configuration (the blue curve below with the y axis on right).



While mid-block and intersection collisions represent the biggest portion of collisions, the passing or climbing lane collisions have the highest fatality rate (~12%). A possible reason could be the different speed between vehicles during collision contact and the vulnerable position of the vehicle changing the lane.

* 1. Weather **(c\_wthr)** & Road Surface **(c\_rsur)**



From the graph above we can see that surprisingly, neither snow nor rain contributed to more collisions and fatality rate. In fact, the majority of collisions actually took place in clear and sunny days. The most dangerous weather for drivers was weather with limited visibility like as fog, smog and mist and weather with strong wind. The fatality rate in such weather condition was about 4%.



The analysis done for road surface above brought to our attention that dry and normal road surface does not guarantee a safe trip. Dry road leads in the statistics while fatality rate is not significant (~1.5%). The most life-threatening collisions happen on sandy, dirty and muddy roads with fatality rate 3.5% and 4.5% respectively.

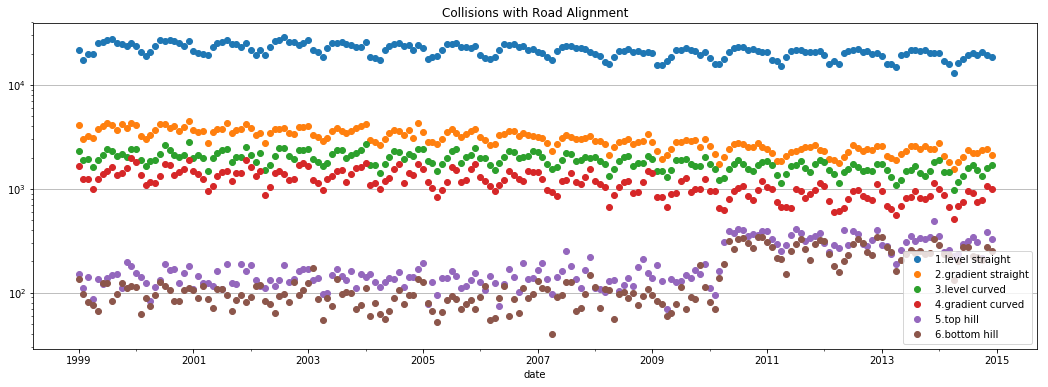
* 1. Traffic Control **(c\_traf)**

Traffic controls are designed to inform, guide and control traffic to enhance the safety of motorists and pedestrians. As we can see from the graph below, most number of collisions happened when traffic signals were fully operated, and no traffic control was present. The highest fatality rate occurred when collisions happened in reduced speed zone and railway crossing with signs only, around 7% and 8% respectively. While high fatality rate near railway signs can be explained by collision with trains, the high collision rate in reduced speed zone was not easy to comprehend. One possible explanation could be collision involving minors: since the reduced speed zone signs are usually present near schools and day care services.



1. **Road Alignment (c\_raln)**
   1. Road Alignment & Time relation

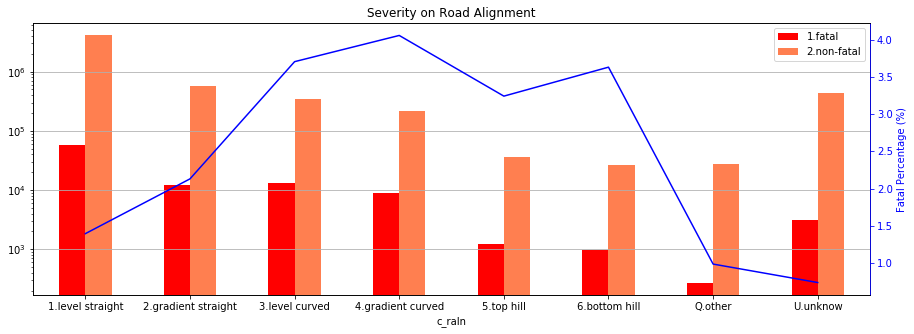
We sliced and grouped the data by *time* and *road alignment*, used the *size* method to aggregate the sum for each possible (*date, c\_raln*) pair into a two-level indexed *pandas.Series* and used the *unstack* method to transform into a new *Dataframe* (sample code in **[A-2.R4]**).



From the graph above, we saw that most collisions happened on *Straight and level* road alignment. A possible reasoning could be that drivers were not cautious enough and drove at a high speed given the easy road.

* 1. Road Alignment **(c\_raln)** & Collision Severity **(c\_sev)**

The graph below shows the relationship between collision severity and road alignment and presents the fatality rate at each road alignment.



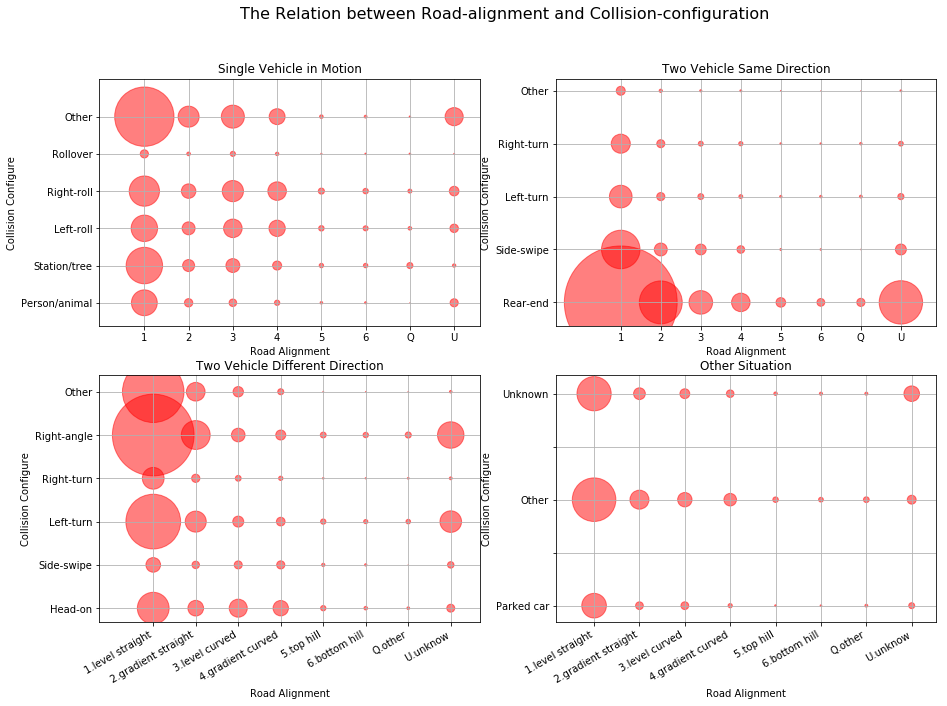
Generally, there were less fatal accidents compared to non-fatal accidents (less than 4% in any given cases) and to no surprise, the Straight-level road alignment had a lower fatality rate, even though the total number of collisions was much higher than other cases. On the other hand, the curved alignment led to more severe accidents; this could be caused by limited sight.

* 1. Road Alignment & Collision Configuration



*Collision configuration* **(*c\_conf*)** column indicates one or two vehicles involved in collision, their motion, direction and relative position during a collision. From the graph above, we can see that the most common collision configuration was read end collision, followed by right angle collision.

Considering the complexity of collision configurations, we grouped them into four categories by number of vehicles and their directions and plotted them into four subplots.

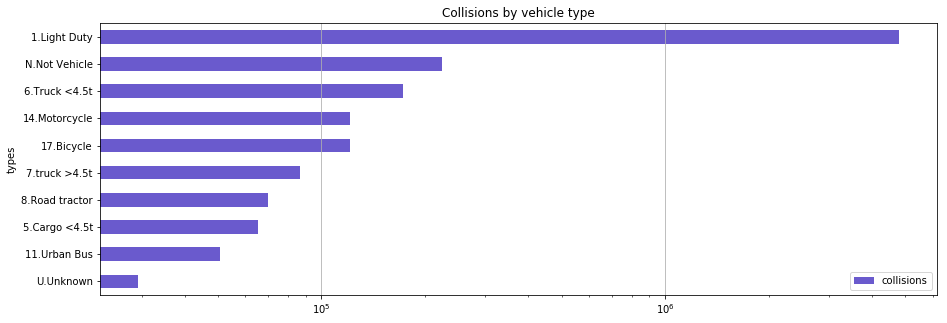


The size of each bubble indicates the number of collisions in each setting and the scale for all subplots were equal, which makes the results comparable. In all cases, there was a clear trend that as road-alignment improved, the collision chances increased due to higher speed and less caution. Disregarding the unknown or "other" conditions, we can see from each case that:

* Most collision for one car accident occurs from hitting a static object, followed closely by right and left roll on to shoulder.
* For two cars in same direction, most accident situations are rear-end collision, which clearly increases along road-alignment.
* For two cars in different direction, right-angle collision is the most frequent, probably in an intersection. While left turn is also risky, right turn shows very low risk: a result of the inherent complexity of left turn.

1. **Collision configuration**
2. **Vehicle Type**

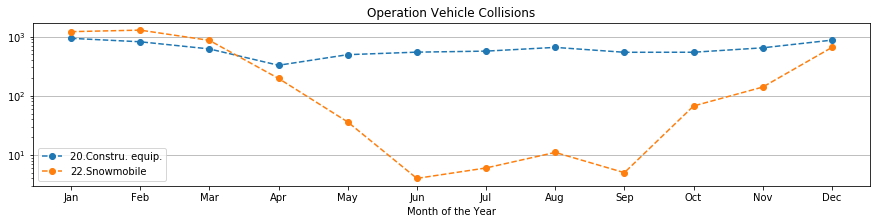
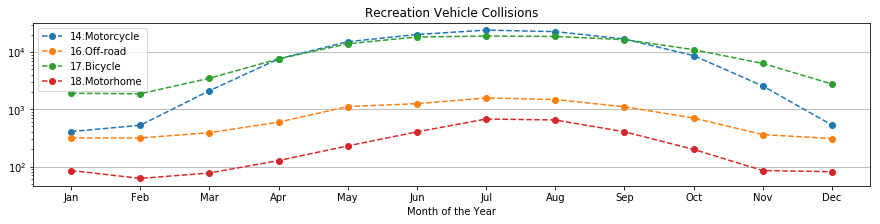
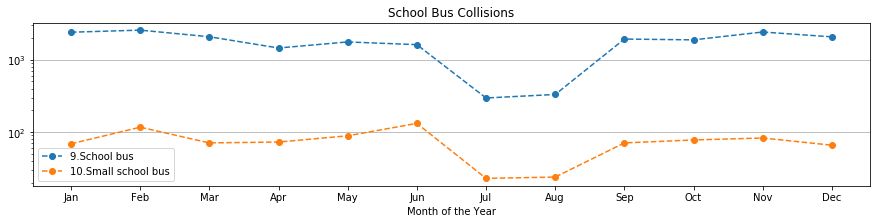
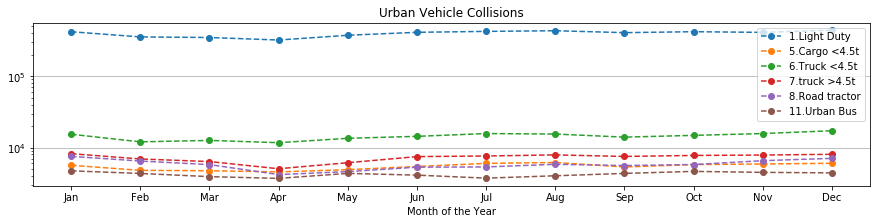
***Vehicle type*** (*v\_type*) indicates the vehicle types involved in accidents. e.g., truck, school bus, bicycle, etc.



From the graph above, we can see that vast majority of collisions were caused by light duty vehicles, followed by light truck, motorcycle and bicycle.

* 1. Vehicle Type & Time Relation

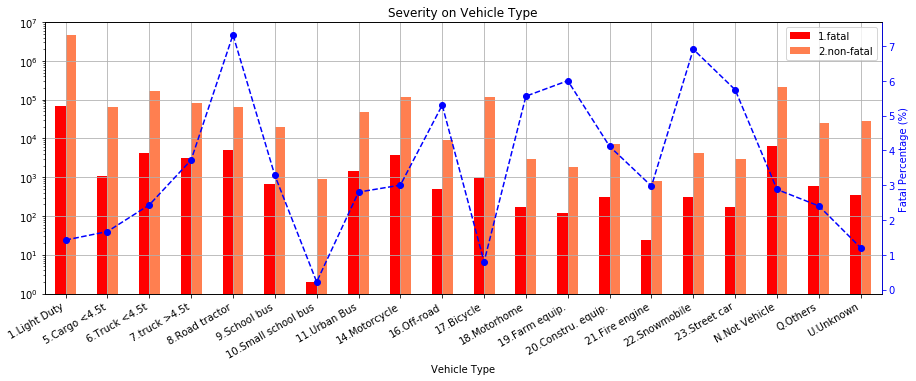
Because there were around 20 different types of vehicles in the dataset, we grouped similar types, and plotted them into a set of graphs, which focused mostly on the seasonal pattern observed for each vehicle types.



By looking at the plots above, different vehicle types showed different observations:

* The number of collisions for urban vehicles (light duty, trucks, road tractor, and bus) is flat across all seasons, indicating that urban vehicles are in duty all throughout the season.
* School bus collisions decreased in July and August due to summer vacation.
* More accidents related to bicycle, motorcycle, off-road vehicle and motorhome occurred from April to October, as they are used for outdoor activities during warmer temperature, whereas more snow-mobile collisions were prevalent from October to April.
  1. Vehicle Type & Severity

Similar to the analysis for ***road alignment*** and ***severity***, we plotted the results for fatal and non-fatal collisions by different vehicle types with their corresponding fatality rates with a blue dotted line.



From the figure above, we see most collisions occurred with light-duty vehicles, followed by trucks, motorcycle, and bicycle. However, accidents with the highest fatality rate were caused by road tractor and Snowmobile (about 7%), followed by street car, farm equipment and motorhome (around 6%). It is also noticeable that small school bus has the least fatal collisions as well as that of bicycle.

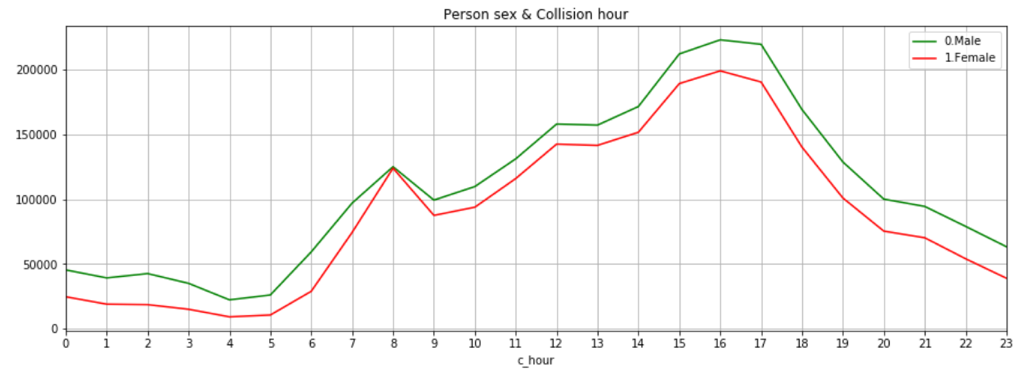
Apart from some external factors and conditions, human factors also play a significant role in contributing to the likelihood of a collision. Therefore, the following analysis will pay more attention to some person-level data elements, focusing on person sex, age, position, and safety device used.

1. **Person sex**

***Person sex*** (*p\_sex*) indicates the gender of the individuals involved in accidents. For convenience, all unknown and missing values were dropped.

* 1. Person Sex & Collision Hour

We grouped the data by p\_sex and by using the count method, we found that 2,608,529 men and 2,116, 502 women were involved in accidents. Thus, we assumed that men are more likely to be involved in an accident than women. The graph below shows the relation of gender involved in collision and collision hour.



The graph above follows the observed pattern of the ‘hour’ plots we earlier on and our assumption was confirmed that at all hours, men are more likely to be involved in a collision than women.

* 1. Person Sex & Medical Treatment Required

*Medical Treatment Required*(***p\_isev***) indicates the degree of casualty. After dropping all unknown and missing values, the data indicated three levels: *No injury*, *Injury* and *Fatality*. We used same method as earlier, we aggregated the result into a two-level index table (***p\_sex*** and ***p\_isev***), and plotted a bar chart.



By looking at the plot above, we can conclude that females are more likely to get injured in an accident, compared to that of male and fatality for both of the genders were relatively lower than that of no injury and injury.

1. **Person age**

*Person age*(***p\_age***) represents age from 0 to 99 years old. 0 indicates less than 1 years old and 99 indicates 99 years or older. We grouped people into 10 age groups e.g. 10 represents 11 to 20 years, 20 represents 21 to 30 and so on.

* 1. Person Age & Collision Hour

Like in “Person Sex & Collision Hour” analyze, we defined x-axis as collision hour in one day. Also, 10 different-color lines represent 10 age-groups.



The graph shows that the age group most prone to accident is 21 to 30 years old. In the morning (8:00 am to 11:59am), three age-groups (21 to 30; 31 to 40; 41 to 50) showed similar number of accidents. We also observed that age groups involving working people are most likely to get into accidents during rush hour (8:00-8:59am and 16:00-17:59pm).

* 1. Person Age & Medical Treatment Required

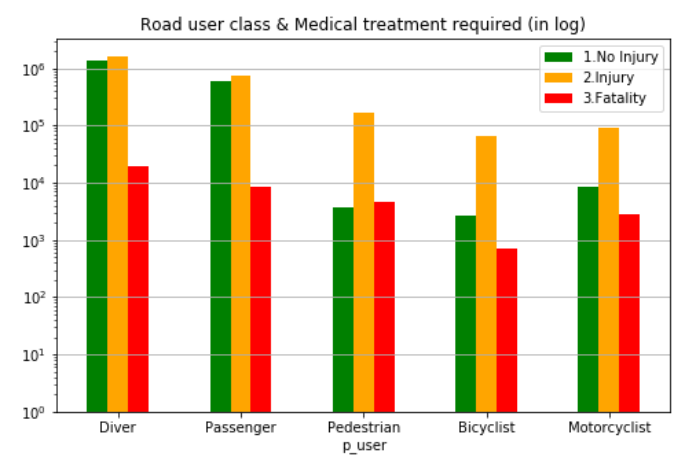


From the graph above, young adults (20-30) show the highest number of injuries while those aged above 90 are have lower overall casualty results compared to that of other age groups. Statistically, from 1999 to 2014 only 185 people above the age of 90 died in accidents whereas 7064 young adults died.

1. **Road user class**

***Road user class*** (*p\_user*) shows who was involved when accidents occurred. Dataset included 5 road users (driver, passenger, pedestrian, bicyclist, motorcyclist and unknown) with codes from 1 to 6. For consistency, the unknown values were dropped.

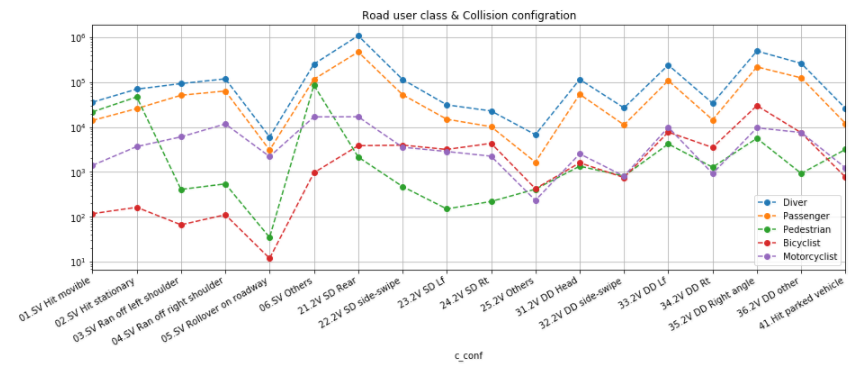
* 1. Road User & Medical Treatment Required



The major group involved in accidents is drivers since they are twice as likely to get hurt or die than passengers. In particular, for each three degrees of casualty, the figure of drivers exceeds the sum of the rest of road users. On the contrary, among all 5 user groups, bicyclists are the safest with the lowest injuries and deaths.

Additionally, as far as drivers and passengers are concerned, no injury and injury themselves account for half of the total cases since fatality only accounts for about 0.06% in both road user groups. A collision has above 85% chance to cause an injury when either pedestrians, bicyclists or motorcyclists are involved in.

* 1. Road User & Collision Configuration



In the end, it is no surprise that driver and passenger have the identical pattern, both of them are more likely to involve in a rear-end collision, and the second common type is right-angle collision. Single vehicle rollover on roadway causes pedestrian or bicyclists to injure or die is infrequent. However, pedestrians are more likely to involve in any other single vehicle collision configuration; for bicyclists, they are more likely to get hurt in a right-angle collision. Among all types of accidents, Motorcyclists get a middling ranking.

# Conclusions

## What ….

Sample sample sample sample sample

* List
* List
* List

Sample sample sample sample sample

# Appendices

## A-1 sample dataset

* Data dictionary:



* Sample data:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **C\_YEAR** | **C\_MNTH** | **C\_WDAY** | **C\_HOUR** | **C\_SEV** | **C\_VEHS** | **C\_CONF** | **C\_RCFG** | **C\_WTHR** | **C\_RSUR** | **C\_RALN** |
| 1999 | 1 | 1 | 20 | 2 | 2 | 34 | UU | 1 | 5 | 3 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **C\_TRAF** | **V\_ID** | **V\_TYPE** | **V\_YEAR** | **P\_ID** | **P\_SEX** | **P\_AGE** | **P\_PSN** | **P\_ISEV** | **P\_SAFE** | **P\_USER** |
| 3 | 1 | 6 | 1990 | 1 | M | 41 | 11 | 1 | UU | 1 |

## A-2 Sample Codes

1. **[A-2.B1]** create new column ‘date’ with data type as ‘period[M]’ and set this column as an index (in “G3\_Project\_Boris.ipynb”).

|  |
| --- |
| df['date'] = pd.PeriodIndex(df['c\_year'].map(str) + '-' + df['c\_mnth'], freq='M')  df=df.set\_index('date') |

1. **[A-2.B2]** create ‘fatal’ and ‘non\_fatal’ columns based on c\_sev column value (in “G3\_Project\_Boris.ipynb”).

|  |
| --- |
| df['fatal']=np.where(df['c\_sev']==1,1,0)  df['non\_fatal']=np.where(df['c\_sev']==2,1,0) |

1. **[A-2.B3]** group ‘fatal’ and ‘non\_fatal’ columns by date (in “G3\_Project\_Boris.ipynb”).

|  |
| --- |
| df\_1=df.groupby('date')['fatal','non\_fatal'].sum() |

1. **[A-2.B4]** Autocorrelation plot (see line 15 in “G3\_Project\_Boris.ipynb”).

|  |
| --- |
| plt.figure(figsize=(15,5))  for c in df\_1.columns:  autocorrelation\_plot(df\_1[c][-60:],label=c) |

1. **[A-2.B5]** correlation plot andcorrelation coefficientdetermination (in “G3\_Project\_Boris.ipynb”).

|  |
| --- |
| sns.regplot('fatal', 'non\_fatal', data=df\_1)  corr = df\_1.corr() |

1. **[A-2.B6]** group collisions by week day and create a bar plot (in “G3\_Project\_Boris.ipynb”).

|  |
| --- |
| by\_weekday = df.groupby('c\_wday')['c\_sev'].count()  by\_weekday.index = ['Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', 'Sun']  plot2 = by\_weekday.plot(kind='bar',title='Collisions by week day') |

1. **[A-2.B7]** group collisions by day hour day and create a bar plot (in “G3\_Project\_Boris.ipynb”).

|  |
| --- |
| plt.figure(figsize=(15,5))  by\_hour = df.groupby('c\_hour')['c\_sev'].count()  plot3 = by\_hour.plot(kind='bar',title='Collisions by week hour',color='G') |

1. **[A-2.R1]** load the data into a *pandas.DataFrame* (in “G3\_Project\_Ryan.ipynb”).

|  |
| --- |
| accident = pd.read\_csv('NCDB\_1999\_to\_2014.csv', engine = 'python')  *#replace header to lower case*  accident.columns = accident.columns.map(lambda x: x[:].lower()) |

1. **[A-2.R2]** clean and prepare the data for analysis (in “G3\_Project\_Ryan.ipynb”).

|  |
| --- |
| *#Check the non-numeric values in date-time columns*  accident[accident['c\_mnth'].str.contains('[^0-9]')|  accident['c\_wday'].str.contains('[^0-9]')|  accident['c\_hour'].str.contains('[^0-9]')]  *#Make a copy "df" for further analysis, avoid mess up the original data "accident".*  *#Remove all special values (unknown to us) in date-time columns, prepare for using*  *# time series as index.*  df = accident[:]  df[['c\_mnth','c\_wday','c\_hour']]=df[['c\_mnth','c\_wday','c\_hour']].\  replace('[^0-9]+',np.nan,regex=True)replace('[^0-9]+',np.nan,regex=True)  df.dropna(axis=0,subset=['c\_mnth','c\_wday','c\_hour'],inplace=True) |

1. **[A-2.R3]** make a copy of the dataset for easy debugging and faster processing (in “G3\_Project\_Ryan.ipynb”).

|  |
| --- |
| *#Slice the columns I would analyse from original dataset.*  dfa = df.loc[:,['date','c\_year','c\_mnth','c\_wday','c\_hour',  'c\_sev','c\_vehs','c\_conf','c\_wthr','c\_raln',  'v\_type','v\_year']]  *# Make a slice only focusing on c\_raln: road alignment.*  df1 = dfa.loc[:,['date','c\_raln']] |

1. **[A-2.R4]** aggregate by a pair, and reform into a *DataFrame* (in “G3\_Project\_Ryan.ipynb”).

|  |
| --- |
| *#Aggregate the events by month into a summarized dataframe.*  df2 = df1.groupby(['date','c\_raln']).size().unstack() |

1. **[A-2.R5]** calculate accident portion for each collision, and bin the numbers of vehicle involved in each collision (in “G3\_Project\_Ryan.ipynb”).

|  |
| --- |
| *#Calculate accident portion of each event, considering 58 records that have 57 vehicle*  *# involved, that should be just one accident, every relevant record should be only about 1/57.*  df1['acc']= 1/df1.c\_vehs  *#Categorize number of involved vehicles into bins.*  bins = [0,5,10,20,40,60]  df1['cats'] = pd.cut(df1.c\_vehs,bins) |