FIT5147 Data Visualization Assignment

FIT 5147 - Data Exploration and Visualization

Jayesh Parab (27148572)

Executive Summary

The report emphasizes the importance of data visualization and exploration in real life situations. It shows how collected data can be explored and analyzed critically using a selected dataset of Crashes Last Five Years from Victorian government site. The data set contains information of all the reported accident that took place in Victoria from 2012 to 2017 with respect to cause, demographics, location, time frame and region. The report also focusses on the importance of learning R language, how to use RStudio software, loading data in RStudio and developing interactive visualization using shiny package of R.

The aim of the report is to analyze the causes of accidents and suggest preventive measures in order to reduce the number of accidents through qualitative analysis of the selected dataset.

Link: https://www.data.vic.gov.au/data/dataset/crashes-last-five-years

Contents

1 Introduction 1
1.1 Purpose 1
1.2 Dataset Details 1
1.3 Problem 1
2 Data Wrangling 2
2.1 Selecting the dataset 2
2.2 Data Cleaning 2
2.2.1 Missing Data 2
2.2.2 Format of data 3
2.3 Data Checking 4
2.3.1 Latitude and longitude checking 4
2.3.2 Data types Checking 5
3 Data Exploration 5
4 Five Sheet Methodology 7
4.1 Sheet 1 7
4.2 Sheet 2 7
4.3 Sheet 3 8
4.4 Sheet 4 8
4.5 Sheet 5 8
5 Interactive Visualization 9
5.1 Tab 1: Top City 9
5.1.1 Graph 1: Accident per City 10
5.1.2 Graph 2: Major accident suburbs 11
5.1.3 Graph 3: Accidents incidents 12
5.2 Tab 2: Time Analysis 12
5.2.1 Graph 1: Heat map 12
5.2.2 Graph 2: Accident by Week Day 13
5.2.3 Graph 3: Accident by Hour 13
5.3 Tab 3: Demographics 14
5.3.1 Graph 1: Demographics by Gender 14

5.3.2 Graph 2: Demographics by Injury 14
5.3.3 Graph 3: Demographics by People 15
5.3.4 Graph 4: Demographics by Injury Type 15
5.3.5 Graph 5: Demographics by Vehicle 16
5.4 Tab 4: Important Parameter 16
5.4.1 Graph 1: Accident by Speed 16
5.4.2 Graph 2: Accident by Road Type 17
5.4.3 Graph 2: Accident by Road Geometry 17
5.5 Tab 5: More Features 18
5.5.1 Sub Menu 1: Accident Parameter 18
5.5.2 Sub Menu 2: Accident Count by City 18
5.5.3 Sub Menu 3: Frequent accident locations 19
6 Importance of each panel 20
7 Solution 21
8 Conclusion 21

1 Introduction

The report provides analysis on selected dataset 'Crashes', with the help of a number of visualizations. Accidents can never be prevented completely, but their count can be reduced. Therefore, the aim is find the predominant causes of accidents and then provide a solution, which can be implemented to reduce the number of accidents taking place. The reports also discusses ways to check whether the dataset selected in clean or no as well as steps taken to wrangle data in a format which can be used conveniently.

1.1 Purpose

The main purpose of this exercise is to critically analyze the crashes dataset in order to find the major accident causes as well as suggest steps to reduce the accident count. This analysis will thus provide valuable insights which can be used to develop solutions in order to solve the business question of reducing the accident count. Furthermore, this analysis will thus help Victorian government to protect and safeguard public and private property as well as reduce loss of life by reducing the accident count.

1.2 Dataset Details

Following are the details of the dataset selected:

Characteristics	Value
Date Published	02/07/2015
Date Updated	27/07/2016
Agency	VicRoads
Link	https://www.data.vic.gov.au/data/dataset/crashes-last-five-years
Important Parameters	Time frame Details: Accident time, Accident Date, Day of Week. Location details: Latitude, Longitude, Suburb Name, Region Name, Road Type, Road Geometry Demographic Details: Type of Vehicles Involved, Number of People involved including Type and Gender, Number of People Injured including Injury type Other Details: Light Conditions, Police Attention Required, Hit and Run Case, Vehicle ran Off Road

1.3 Problem

The important observations discovered after exploring the dataset were as follow:

- Many accident took place due to bad light conditions. Bad light conditions included no street light installed and street light present but not working.
- Accident occurring due to intake of alcohol were numerous.
- Accident involving hit & run case and police involvement were substantial.

- Accidents in which vehicle goes off road were also high.

Generated problems:

- Can accidents occurring due to insufficient light reduced?
- Is there a solution to stop people from driving after in taking alcohol?
- How to reduce the count of hit and run accident cases?
- How to help police to reach the accident spot quickly in accidents which requires police involvement?
- How to prevent vehicles from going off roads?

2 Data Wrangling

2.1 Selecting the dataset

The dataset selected should be available in proper format or it should be easy to convert to proper format like csv or json files. Always prefer to take dataset from reliable sources like government sites. On selecting the dataset, it is better to remove the information one is not going to use in analysis. It increases computational speed as well.

The columns selected from crashes dataset are as follow:

```
##################################################################################################
# Step 1: Loading Data into R
##################################################################################################
crashes_complete <- read.csv('Crashes_Last_Five_Years.csv')
#View(crashes_complete)

##################################################################################################
# Step 2: Subsetting the required columns and removing the rest for efficiency
##################################################################################################
crashes <- crashes_complete[c("OBJECTID","ACCIDENT_NO","ACCIDENT_DATE","ACCIDENT_TIME","ACCIDENT_TYPE",
                      "DAY_OF_WEEK","HIT_RUN_FLAG","LIGHT_CONDITION","POLICE_ATTEND","ROAD_GEOMETRY",
                      "SPEED_ZONE","RUN_OFFROAD","LONGITUDE","LATITUDE","LGA_NAME","REGION_NAME",
                      "TOTAL_PERSONS","INJ_OR_FATAL","FATALITY","SERIOUSINJURY","OTHERINJURY",
                      "NONINJURED","MALES","FEMALES","BICYCLIST","PASSENGER","DRIVER","PEDESTRIAN",
                      "PILLION","MOTORIST","UNKNOWN","ALCOHOL_RELATED","UNLICENCSED","NO_OF_VEHICLES",
                      "HEAVYVEHICLE","PASSENGERVEHICLE","MOTORCYCLE","PUBLICVEHICLE","DEG_URBAN_NAME",
                      "RMA")]
#View(crashes)
```

2.2 Data Cleaning

It is very hard to find the clean dataset. Most of the times values will be missing or data set will be incomplete.

2.2.1 Missing Data

Some values were missing in the DAY_OF_WEEK column so I used Accident date to generate the values.

```
#################################################################################################
# Step 3: Adding missing values to DAY_OF_WEEK
#################################################################################################

summary_table <- table(crashes$DAY_OF_WEEK)
View(summary_table)
crashes$DAY_OF_WEEK <- weekdays(as.Date(crashes$ACCIDENT_DATE))
summary_table <- table(crashes$DAY_OF_WEEK)
View(summary_table)
```

2.2.2 Format of data

Sometimes the data is not missing but it is not entered in correct format. Therefore one has to reformat it.

For Example: In crashes dataset speed of 30km/hr was entered without space and ACCIDENT_TYPE for collision with some other object was entered in small alphabets.

```
#################################################################################################
# Step 24: Changing in crashes$ACCIDENT_TYPE "collision with some other object" to "Collision with some other object"
#################################################################################################

crashes$ACCIDENT_TYPE <- sub("collision with some other object", "Collision with some other object", crashes$ACCIDENT_TYPE)
View(crashes)

#################################################################################################
# Step 25: Changing in crashes$ACCIDENT_TYPE "collision with some other object" to "Collision with some other object"
#################################################################################################

crashes$SPEED_ZONE <- sub("30km/hr", "30 km/hr", crashes$SPEED_ZONE)
View(crashes)
write.csv(crashes, "C:/Users/JATESH/Documents/Data Visualization/crashes.csv", row.names=F)
```

It also included removing brackets from some LGA_NAME and replacing ‘Unk.’ to ‘Unknown’ in LIGHT_CONDITION column.

```
#################################################################################################
# Step 12: Replacing "Unk." in LIGHT_CONDITION with "Unknown"
# Reducing the ambiguity of Unk.
#################################################################################################
crashes$LIGHT_CONDITION <- sub("Unk.", "Unknown", crashes$LIGHT_CONDITION)
summary_table <- table(crashes$LIGHT_CONDITION)
View(summary_table)
crashes_final <- crashes
View(crashes)
#################################################################################################
# Step 13: Formatting the LGA_NAME to remove parenthesis and making it uniform
# Eg:- (Falls Creek) to Falls Creek
#################################################################################################
#library(sqldf)
#names_with_bracket <-sqldf("select * from crashes where LGA_NAME LIKE '%(%'")
#View(names_with_bracket)
#distinct_names_with_bracket <-sqldf("select distinct LGA_NAME from names_with_bracket")
#View(distinct_names_with_bracket)

crashes$LGA_NAME <-  gsub("\\(|\\)", "", crashes$LGA_NAME)
```

2.3 Data Checking

2.3.1 Latitude and longitude checking

In the data is checked for various conditions like whether longitude and latitude are correct. This can be found out by mapping them using leaflet library.

```
#################################################################################################
# Step 15: Checking Latitude and Longitude
#################################################################################################
library(leaflet)
leaflet(data = crashes) %>% addTiles() %>%
  addMarkers(
    ~LONGITUDE,
    ~LATITUDE,
    popup = ~as.character(OBJECTID),
    clusterOptions = markerClusterOptions()
  )
```

There were three columns in crashes dataset total people, males and females. Therefore I verified whether total people = males + females. Not all columns matched so I had to create a new column UNKNOWN_GENDER and add the difference to it.

```
#################################################################################################
# Step 6: Verifying whether TOTAL_PERSONS = MALES + FEMALES + UNKNOWN
#################################################################################################

gender_count_checking<-crashes[!(crashes$TOTAL_PERSONS == crashes$MALES + crashes$FEMALES +
                       crashes$UNKNOWN),]
view(gender_count_checking)
crashes$UNKNOWN <- crashes$TOTAL_PERSONS - (crashes$MALES + crashes$FEMALES)
colnames(crashes)[colnames(crashes)=="UNKNOWN"] <- "UNKNOWN_GENDER"
crashes <- crashes[,c("OBJECTID","ACCIDENT_NO","ACCIDENT_DATE","ACCIDENT_TIME","ACCIDENT_TYPE",
                      "DAY_OF_WEEK","HIT_RUN_FLAG","LIGHT_CONDITION","POLICE_ATTEND","ROAD_GEOMETRY",
                      "SPEED_ZONE","RUN_OFFROAD","LONGITUDE","LATITUDE","LGA_NAME","REGION_NAME",
                      "TOTAL_PERSONS","INJ_OR_FATAL","FATALITY","SERIOUSINJURY","OTHERINJURY",
                      "NONINJURED","MALES","FEMALES","UNKNOWN_GENDER","BICYCLIST","PASSENGER","DRIVER","PEDESTRIAN",
                      "PILLION","MOTORIST","ALCOHOL_RELATED","UNLICENCSED","NO_OF_VEHICLES",
                      "HEAVYVEHICLE","PASSENGERVEHICLE","MOTORCYCLE","PUBLICVEHICLE","DEG_URBAN_NAME",
                      "RMA")]
view(crashes)
gender_count_checking <- crashes[!(crashes$TOTAL_PERSONS== crashes$MALES + crashes$FEMALES +
                       crashes$UNKNOWN_GENDER),]
view(gender_count_checking)
```

The same process was followed for:

- TOTAL_PERSONS = INJ_OR_FATAL + NONINJURED
- INJ_OR_FATAL = FATALITY + SERIOUSINJURY + OTHERINJURY
- TOTAL_PERSONS = BICYCLIST + PASSENGER + DRIVER + PEDESTRIAN + PILLION + MOTORIST
- NO_OF_VEHICLES = HEAVYVEHICLE + PASSENGERVEHICLE + MOTORCYCLE + PUBLICVEHICLE

And whenever match was not found an unknown column was created.

2.3.2 Data types Checking

The data types of columns also need to be verified:

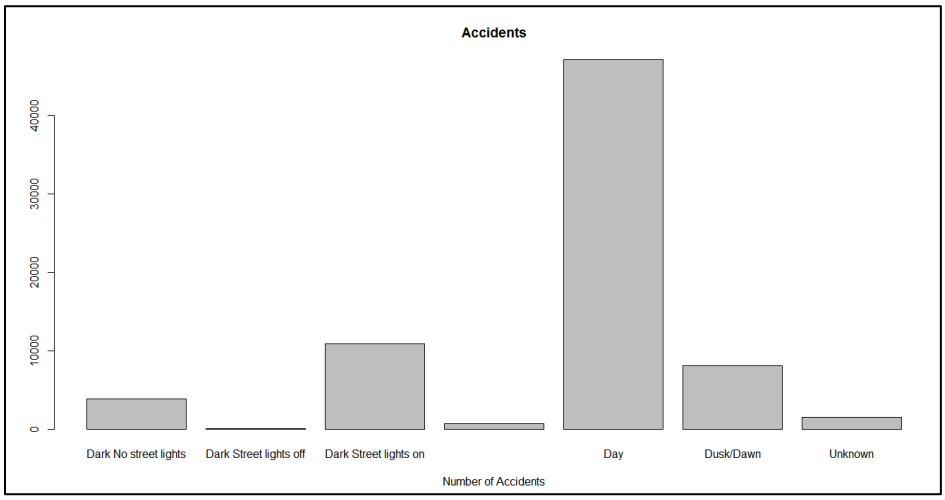
```
#########################################################################################################
# Step 23: Checking whether columns are numeric
#########################################################################################################

is.numeric(crashes$OBJECTID)
is.numeric(crashes$LONGITUDE)
is.numeric(crashes$LATITUDE)
is.numeric(crashes$TOTAL_PERSONS)
is.numeric(crashes$INJ_OR_FATAL)
is.numeric(crashes$FATALITY)
is.numeric(crashes$SERIOUSINJURY)
is.numeric(crashes$OTHERINJURY)
is.numeric(crashes$NONINJURED)
is.numeric(crashes$MALES)
is.numeric(crashes$FEMALES)
is.numeric(crashes$UNKNOWN_GENDER)
is.numeric(crashes$BICYCLIST)
is.numeric(crashes$PASSENGER)
is.numeric(crashes$DRIVER)
is.numeric(crashes$PEDESTRIAN)
is.numeric(crashes$PILLION)
is.numeric(crashes$MOTORIST)
is.numeric(crashes$UNKNOWN_PERSON_TYPE)
is.numeric(crashes$UNLICENCSED)
is.numeric(crashes$NO_OF_VEHICLES)
is.numeric(crashes$HEAVYVEHICLE)
is.numeric(crashes$PASSENGERVEHICLE)
is.numeric(crashes$MOTORCYCLE)
```

3 Data Exploration

A lot of accidents occur due to improper light conditions.

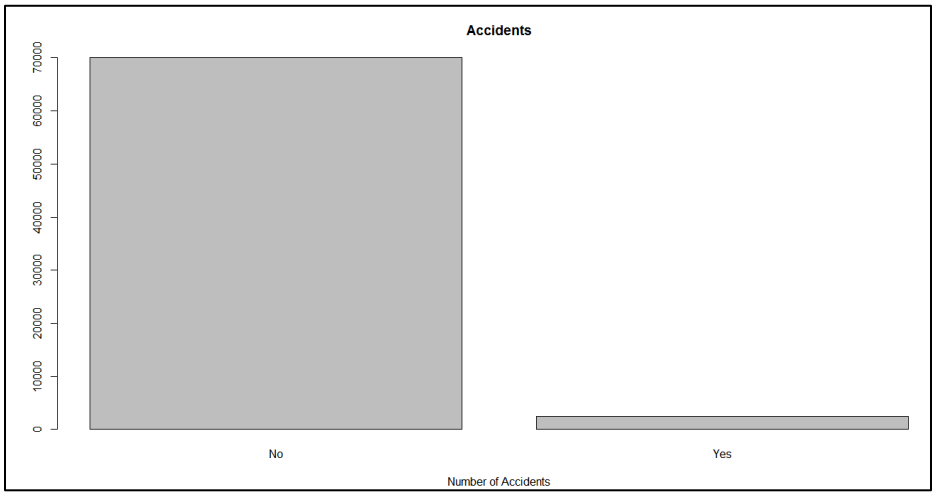
```
#########################################################################
# Step 26: Data exploration
#########################################################################
crashes <- read.csv('crashes.csv')
View(crashes)
counts <- table(crashes$LIGHT_CONDITION )
View(counts)
barplot(counts, main="Accidents",
        xlab="Number of Accidents")

counts_1 <- table(crashes$ALCOHOL_RELATED )
View(counts_1)
barplot(counts_1, main="Accidents",
        xlab="Number of Accidents")
```

	Var1	Freq
1	Dark No street lights	3874
2	Dark Street lights off	149
3	Dark Street lights on	10909
4	Dark Street lights unknown	779
5	Day	47043
6	Dusk/Dawn	8134
7	Unknown	1594



Many accidents also involved alcohol consumption.

	Var1	Freq
1	No	70035
2	Yes	2447

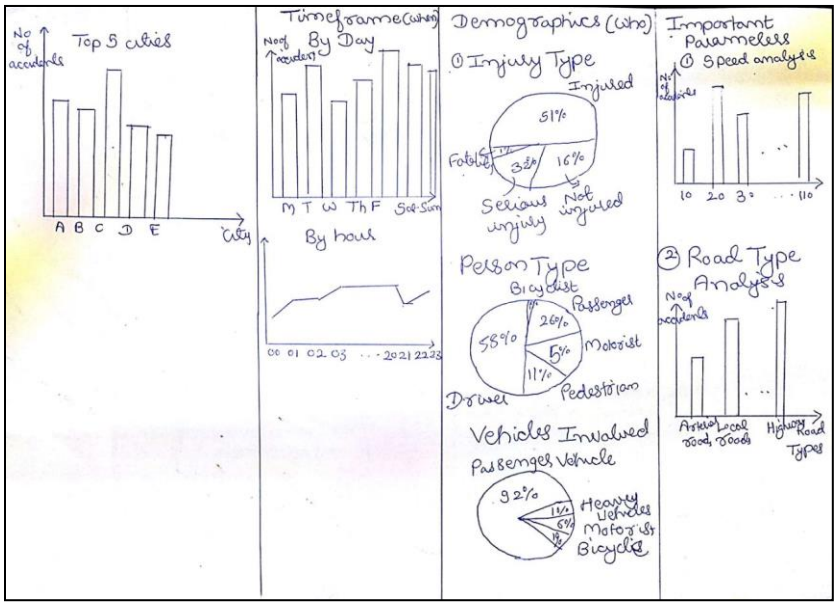


Same analysis was done hit and run, vehicle going off road and police involvement.

4 Five Sheet Methodology

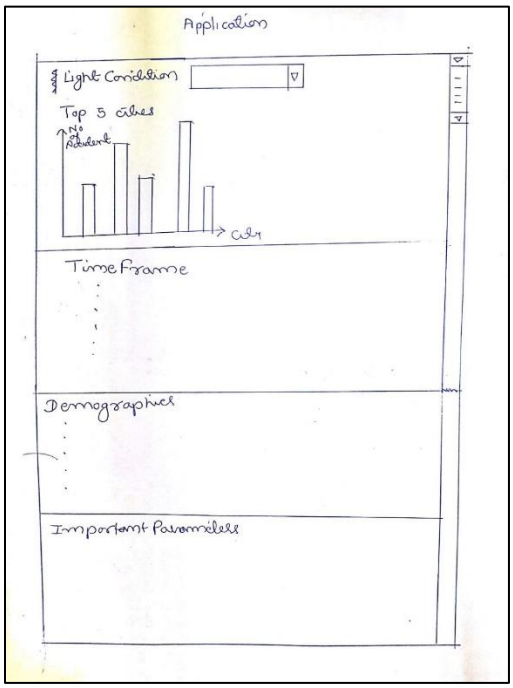
4.1 Sheet 1

In this sheet the dataset of crashes was explored with respect to information content like top cities, time, demographics and important parameters.



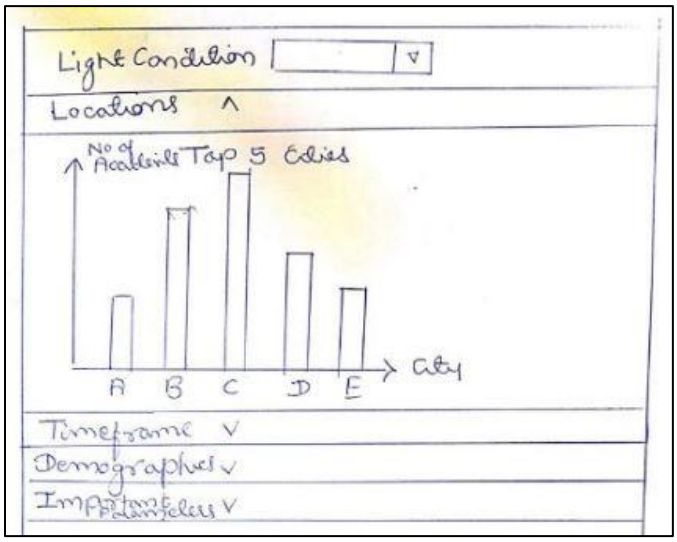
4.2 Sheet 2

An interactive scroll down page application was developed which was interactive but had bad user interface.



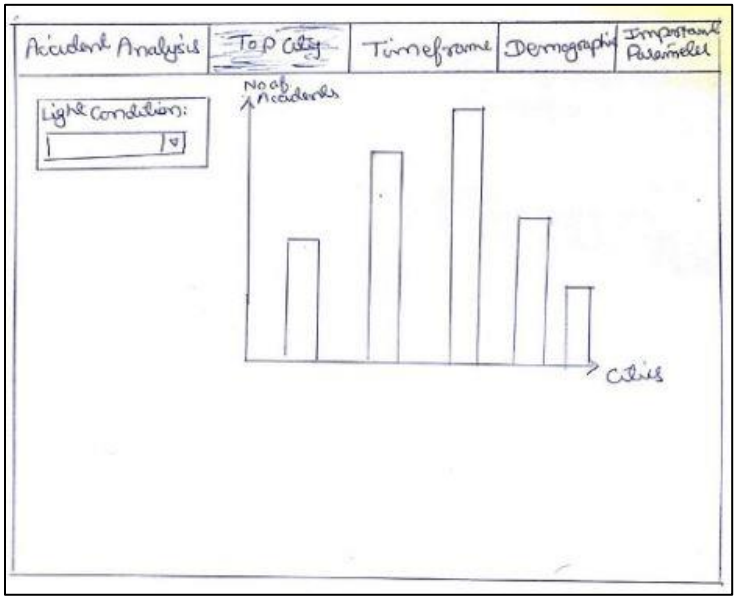
4.3 Sheet 3

An interactive page with drop down buttons was the second design. The advantages were that it was interactive and better user interface as compared to first. However, the drawback was that user interface was not that good.



4.4 Sheet 4

An interactive page with tabs was the final design. The advantages were it was interactive and possessed a good user interface.



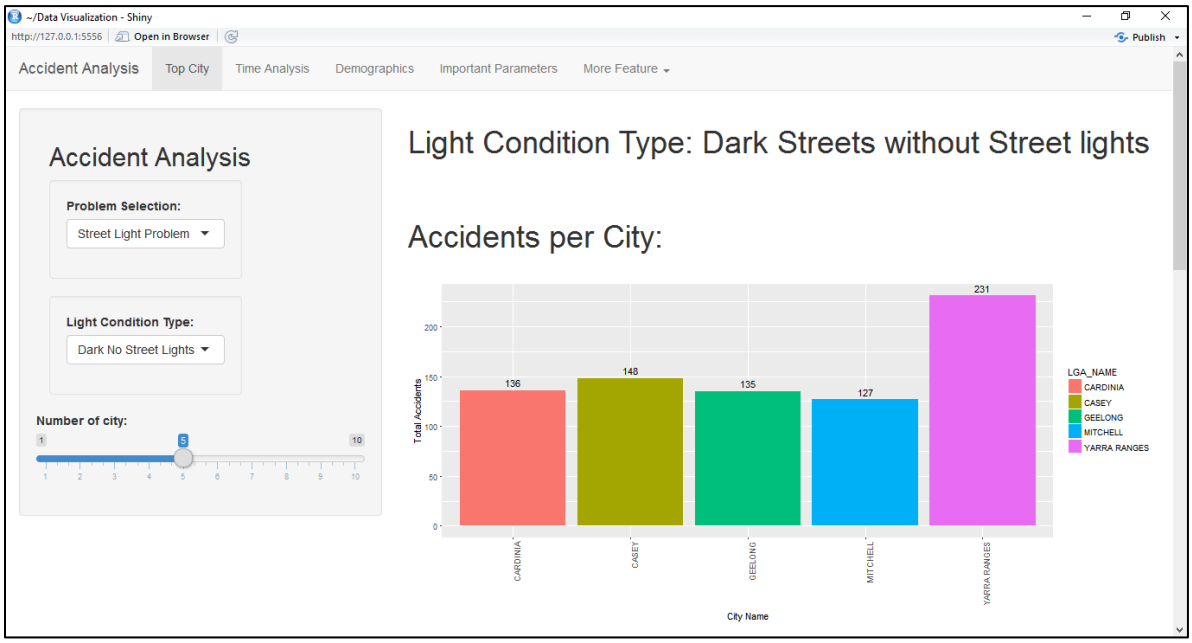
4.5 Sheet 5

The fourth sheet design was selected as final design with the only change that select input directly renders a new UI component.

5 Interactive Visualization

The interactive visualization consists of a shiny application using RStudio. The application is designed to provide top cities where accidents are highest based on different light conditions, alcohol involvement, police involvement, hit and run cases and vehicle running off road. The applications provides the users to select anywhere from 1 to 10 top cities in which they plan to take preventive measures. The idea behind selecting the top cities is that there are always limitations with respect to finance on government or private firms. For example, just because an accident took place at a location due to street light not present, does not provide reason to install street light at that place as the accident could have occurred due human error. The correct approach would be to install street lights in suburb where many accidents takes place due to street lights not present. The second reason for following the top city approach is to implement a solution where would affect the most. For instance, implementing street lights in suburb where accident cost is highest, if successful it will bring down accidents counts by a large number.

The application home screen is as follows:



5.1 Tab 1: Top City

The home screen ‘Accident Analysis’ has a select button which allows you to select the problem type. It has two options street light problem and other problem. It is an example of dynamic UI rendering. Three graphs are shown on this tab.

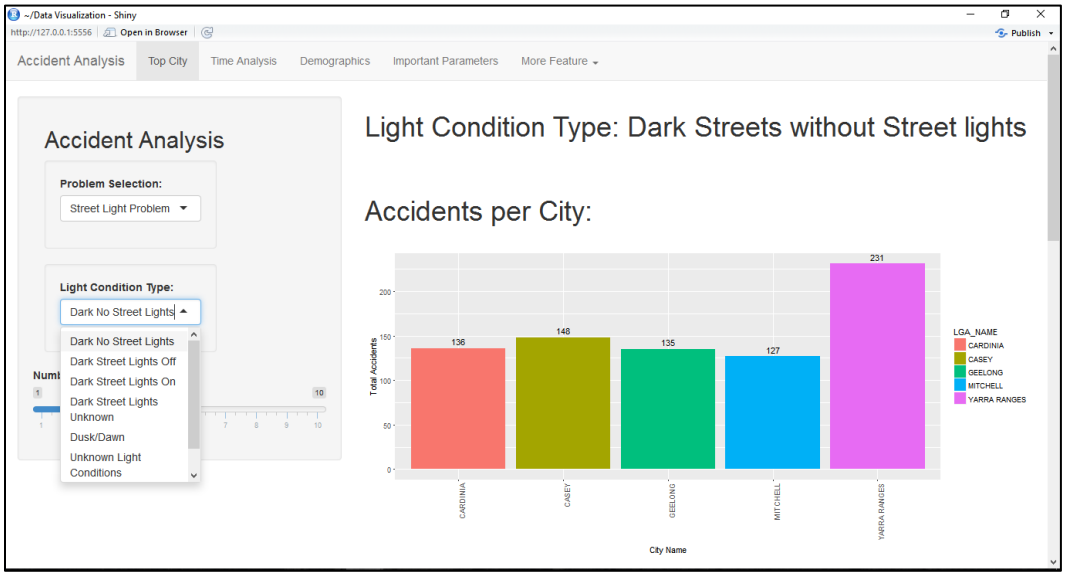
The following table gives the details:

Option selected in first Select Input	Dynamic Select Input Available	Options in Dynamic Select Input
Street Light Problem	Light Condition Type	• Dark No Street Lights • Dark Street Lights Off • Dark Street Lights On

		• Dark Street Lights Unknown • Dusk/Dawn • Unknown Light Conditions • Day
Other Problem	Accident Cause	• Alcohol Involved • Hit And Run Involved • Police Involved • Vehicle Run Off Road

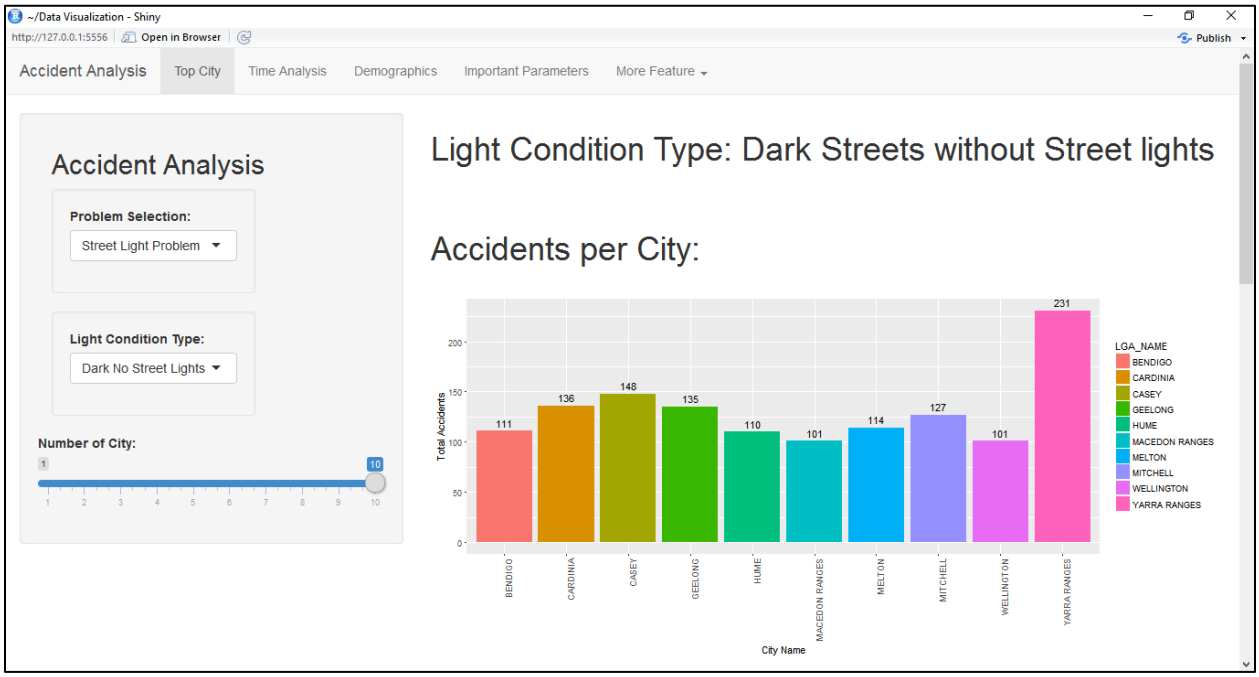
5.1.1 Graph 1: Accident per City

Select Input: Street Light Problem Selected



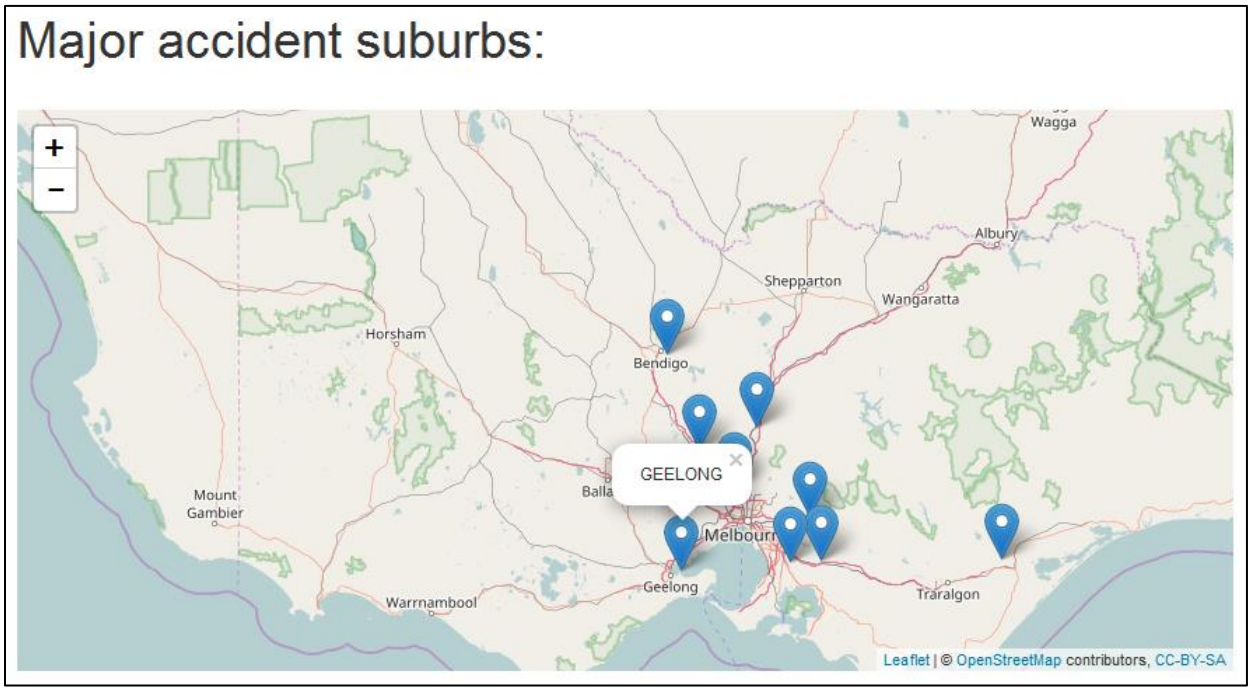
Select Input: Other Problem Selected

The other important thing in the above graph is number of cities can be selected from 1 to 10. Following is the screen shot if 10 is selected in the slider of Number of City:



5.1.2 Graph 2: Major accident suburbs

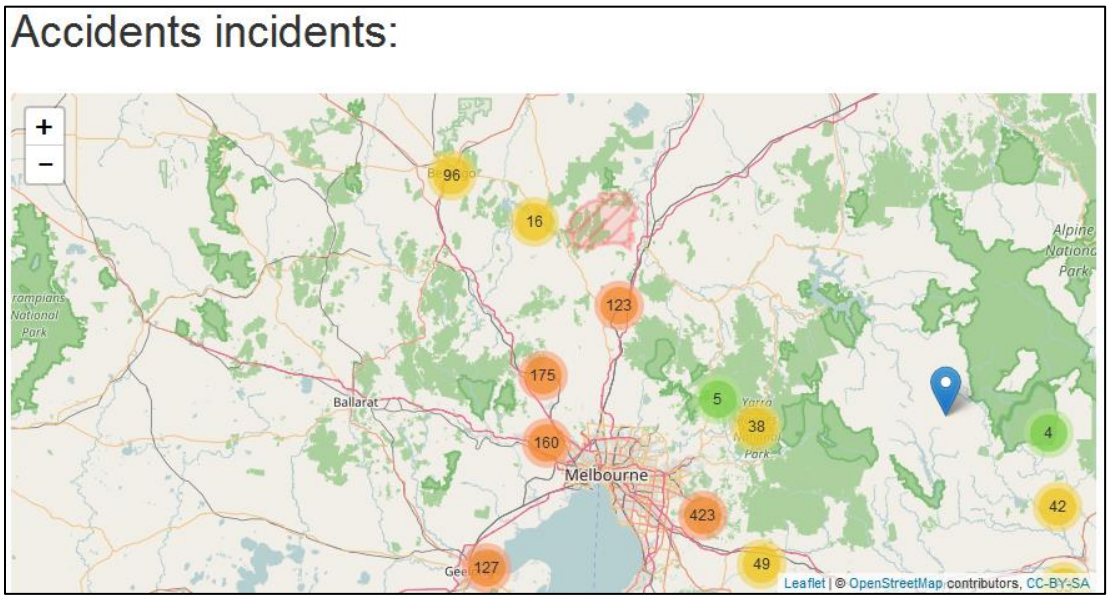
On the same tab this is the second graph shown giving the exact locations of top accidents occurring suburbs:



The important thing about above graph is number of locations shown changes dynamically with respect to 'Number of City:' selected in the slider and on clicking on markers the name of suburb is shown.

5.1.3 Graph 3: Accidents incidents

In this graph the exact accidents locations of the selected number of cities are shown:

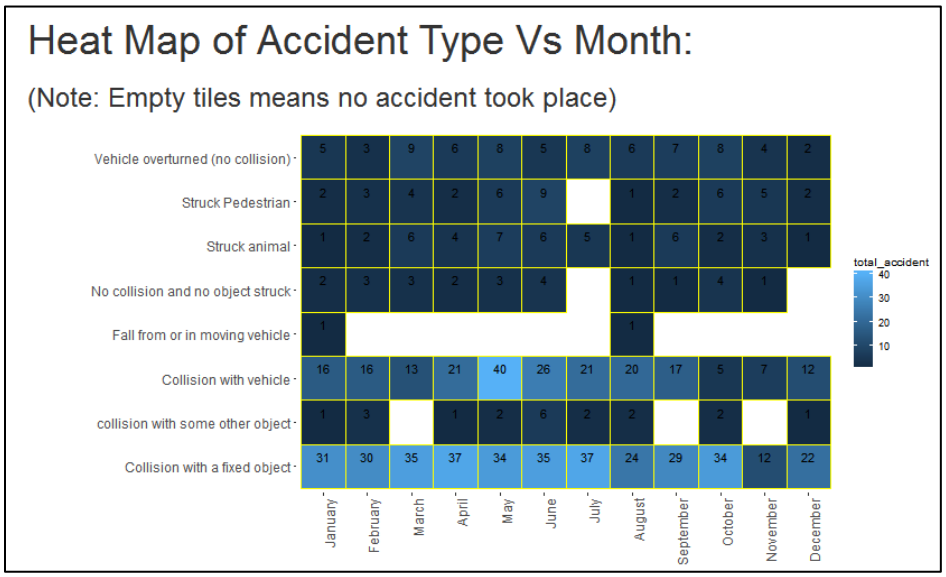


5.2 Tab 2: Time Analysis

Tab 2 has a year slider to select the years. It shows 3 maps heat map (months VS accident type), accident by weekday and accident by hour for selected number of city.

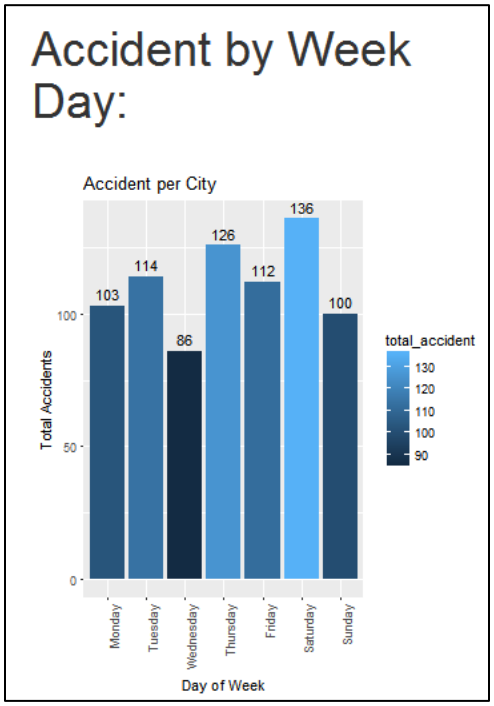
5.2.1 Graph 1: Heat map

It shows the accident type verses month's figures:



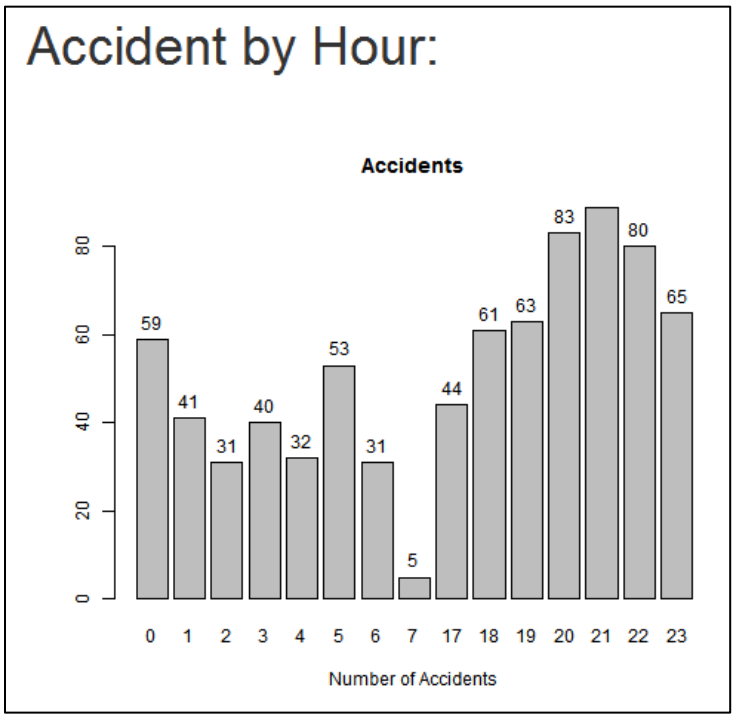
5.2.2 Graph 2: Accident by Week Day

It shows the accident count by Week Day:



5.2.3 Graph 3: Accident by Hour

It gives the accident count by Hours:

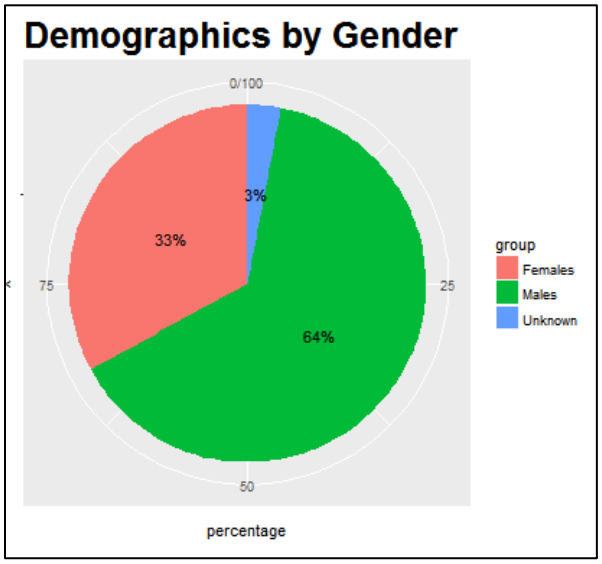


5.3 Tab 3: Demographics

Tab 3 has a year slider to select the years. It also has 5 check boxes to select the type of demographics one wants to see. It shows 5 demographics which include gender, injury, people type, injury type and vehicle type.

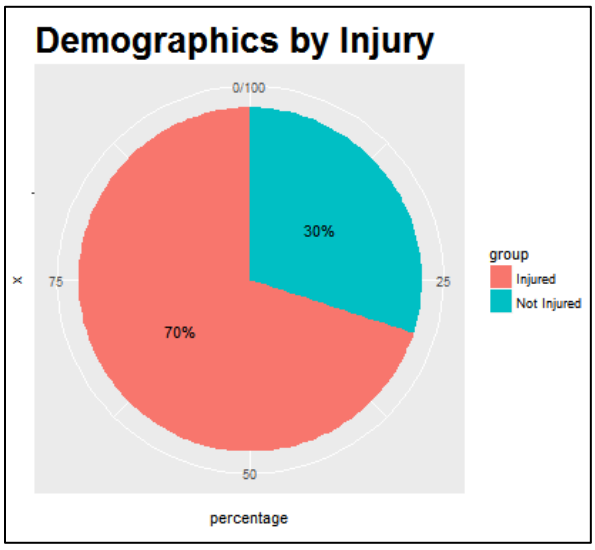
5.3.1 Graph 1: Demographics by Gender

It gives the demographics of Males, Females and Unknown Involved in accident:



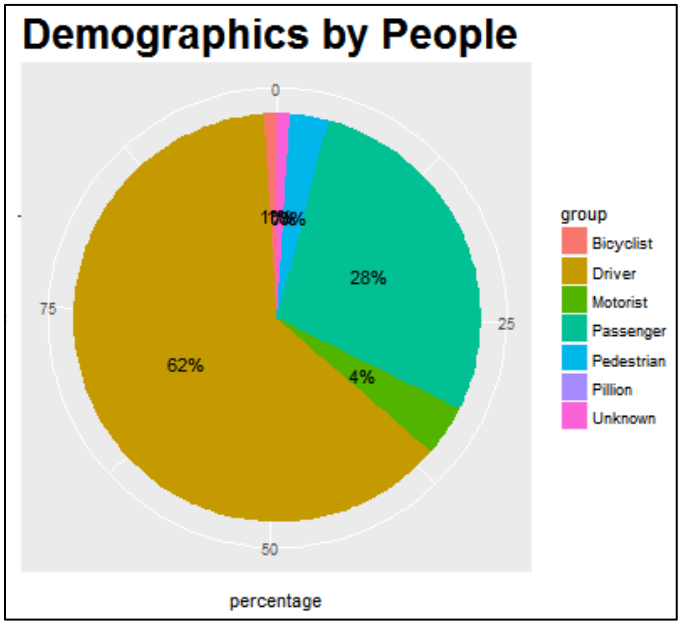
5.3.2 Graph 2: Demographics by Injury

It gives the demographics of Injured and Not Injured involved in accident:



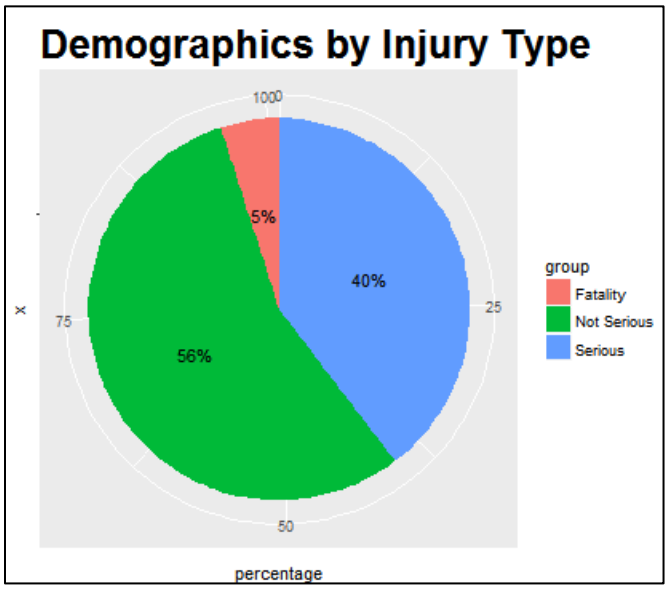
5.3.3 Graph 3: Demographics by People

It gives the demographics of Bicyclist, Drivers, Motorist, Passenger, Pedestrian, Pillion and Unknown involved in accident:



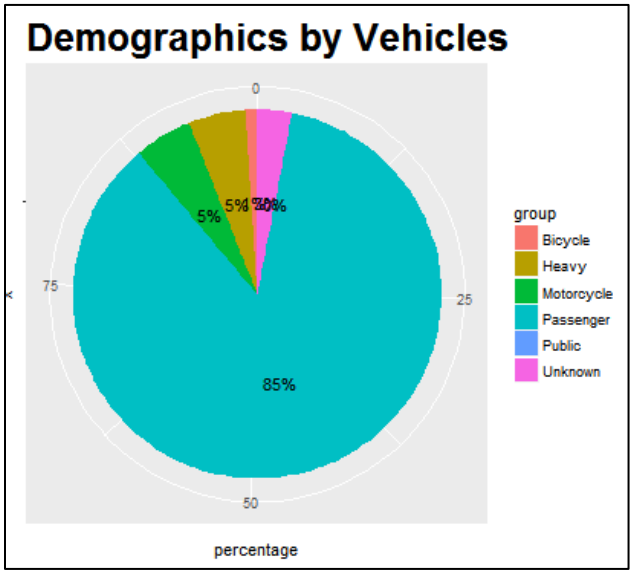
5.3.4 Graph 4: Demographics by Injury Type

It gives the demographics of Fatality, Not Serious and Serious injuries involved in accident:



5.3.5 Graph 5: Demographics by Vehicle

It gives the demographics of Bicycle, Heavy vehicles, Motorcycle, Passenger vehicle, Public vehicle and Unknown involved in accident:

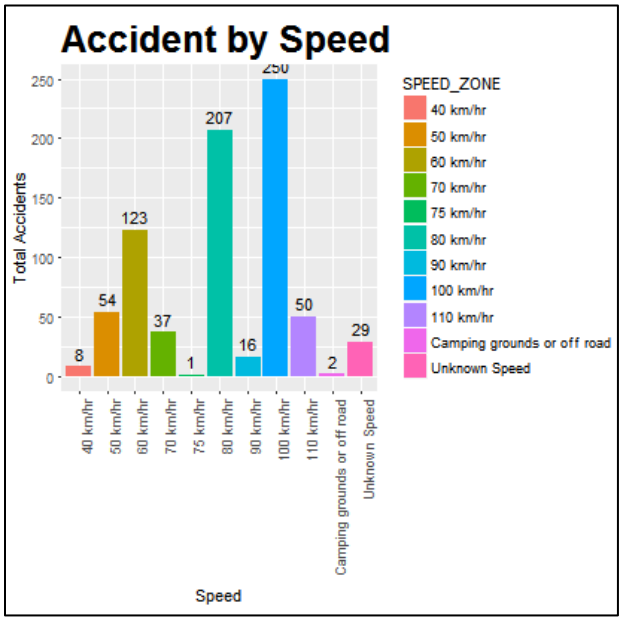


5.4 Tab 4: Important Parameter

Tab 4 has a year slider to select the years. It also has 3 check boxes to select the type of parameter one wants to see. It shows 3 graphs which includes accidents by speeds, accidents by road type and accident by road geometry.

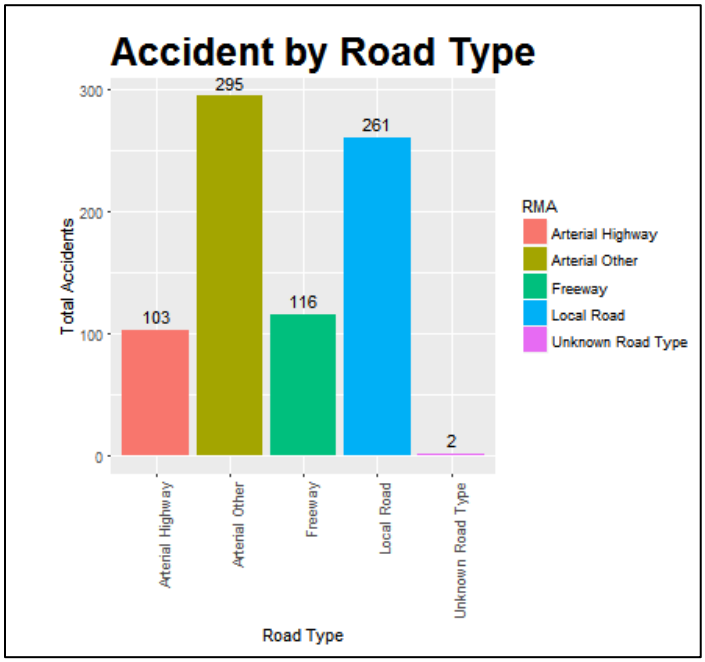
5.4.1 Graph 1: Accident by Speed

It shows the speed at which the accidents have occurred:



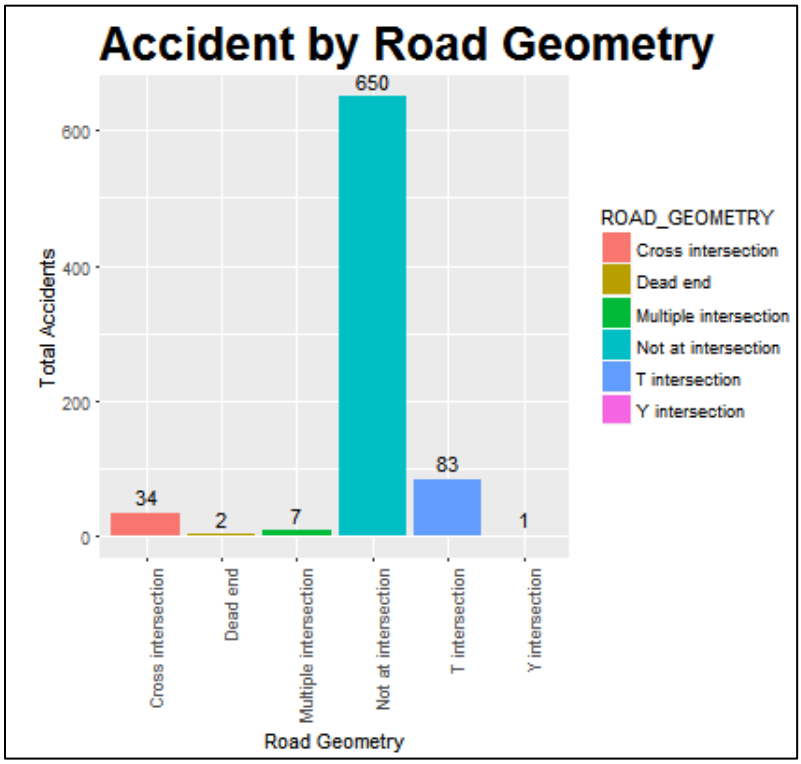
5.4.2 Graph 2: Accident by Road Type

It shows the road type at which the accidents have occurred:



5.4.3 Graph 2: Accident by Road Geometry

It shows the road geometry at which the accidents have occurred:

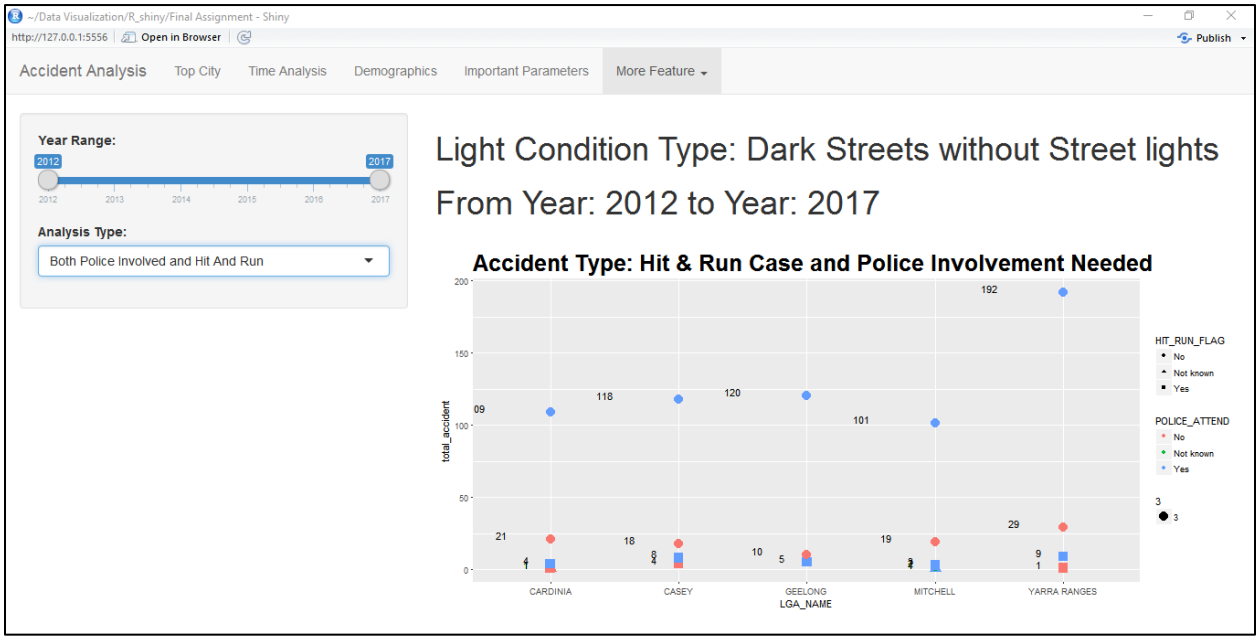


5.5 Tab 5: More Features

It has 3 sub menu accident parameters, accident count by city and frequent accident location.

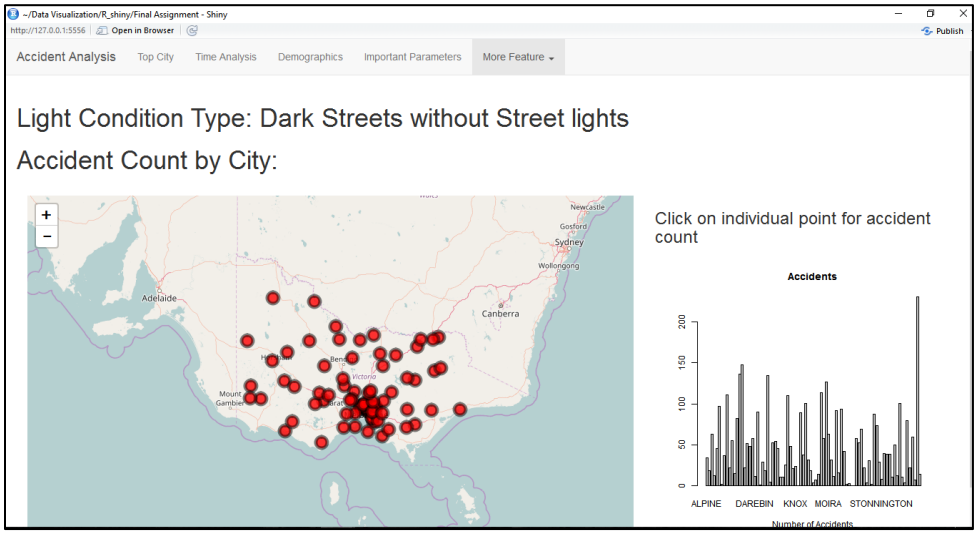
5.5.1 Sub Menu 1: Accident Parameter

It gives the composition of accidents occurred with respect to alcohol consumption, police involvement and hit & run cases. It has a select input to select the type of composition one has to see. It has a year slider to select the years.

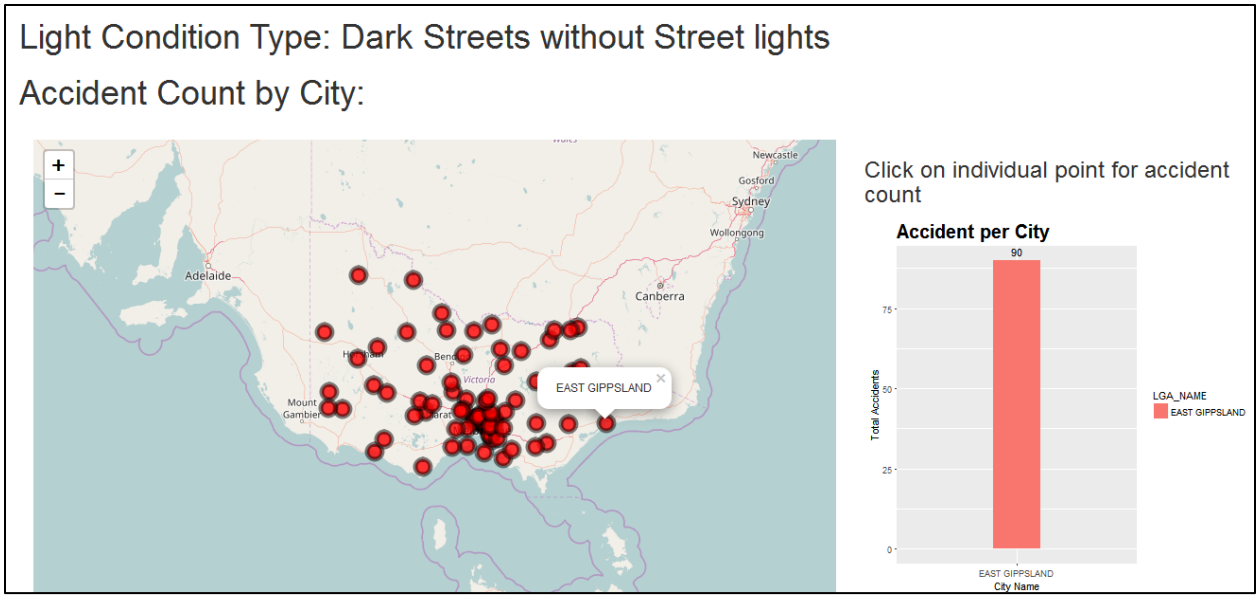


5.5.2 Sub Menu 2: Accident Count by City

It shows a leaflet map of all cities involved in a particular type of accident and on clicking the marker once can see total number of accidents in that city.



On clicking any red point the graph on the right changes.



5.5.3 Sub Menu 3: Frequent accident locations

It shows accident locations where accidents have occurred more than once. On clicking any marker the suburb name along with longitude and latitude is popped out.

6 Importance of each panel

Serial No	Panel Name	Interactive Component	Importance of tab
1	Top City	2 Select Input (one static and other dynamic) and No of city slider	In this tab one can select the problem area he is interested in and then find top cities by various types. One can also see using the cluster map the exact locations where accidents in the top city took place.
2	Time Analysis	Year Slider	For the selected problem and number of cities this tab provide the time analysis with respect to month of accidents, day of occurrences and hour of accident.
3	Demographics	Year Slider and 5 checkboxes each for each type of demographic	For the selected problem and number of cities this tab provide information on demographics of people and vehicles that were involved in the accident.
4	Important Parameter	Year Slider and 3 checkboxes each for each type of Parameter	For the selected problem and number of cities this tab provide information on parameters involved like speed at which accident occurred, road type of accident and road geometry of accident.
5	More Feature: Accident Parameter	Year Slider with Select Input	For the selected problem and number of cities this tab provide information on accident characteristics like whether alcohol involvement was there, were police involved and was there a hit and run case.
6	More Feature: Accident Count by City	NA	For a selected problem this graph plots every suburb involved in that category and on clicking it a count of accident is provided. This is helpful if user is interested in a particular suburb.
7	More Feature: Frequent Accident locations	NA	For a selected problem this graph plots all the locations where accidents have occurred more than once. This is important to analyze because it is hardly a coincidence when accident occurs at the same exact place more than once.

7 Solution

Serial No	Type of Accident	Solutions
1	Street Lights absent	• Planning and implementing new Street lights.
2	Street Light present but not working	• Switching on Street light, if working. • Repairing Street lights not working. • Using light detecting sensors so that street light automatically switches on when light reduces. • Scheduling maintenance of street light regularly.
3	Alcohol Involvement	• Alcohol checking by police: By studying when and where normally drink and drive and punish the offenders to reduce the accident counts. • Increasing public transport on weekends when people normally go out for partying.
4	Hit and Run Cases	• Installing traffic cameras. • Installing police barricades and changing barricades position often.
5	Police Involvement	• Adding new police stations in accident prone areas. • Increase patrolling.
6	Vehicle running off road	• Installing fences.
7	Accidents occurring at junctions	• Can be prevented by installing traffic signals.

8 Conclusion

The shiny application provided can be used as a tool to analyze the accident data in order to find valuable insights. It provides information on how to reduce the accident count as well as provides solution for the same. At the same time it makes us aware that it is very important to prevent accidents as accidents may results in loss of life and damage to property.

In future an application of same sort can be built on real time data providing more insights. Furthermore, it can be combined with weather information to provide updates like storm or rainfall.

Lastly, the assignment has made me aware of the importance of data analysis and how it can be used in real life situations. It has also taught me the importance of learning R language and how to use RStudio effectively.