Fatalities in car accidents, a case of study in Canada

José Chacón, Alejandra Cruces, Mario Tapia

2023-07-16

Contents

1 Introduction

This report presents the results of the Statistical Learning course project that aimed to analyze and predict car accidents in Canada based on a data set from 1994 to 2004. The data set contained information about the date, time, weather, road conditions, vehicle type, driver age, and injury severity of each accident among others. The project involved several steps, such as data cleaning and filtering, exploratory data analysis, creation of new variables, undersampling of the majority class, modeling with logistic regression and k-nearest neighbors, and answering relevant data science questions. The main objective of the project was to identify the most effective way to allocate resources for traffic surveillance in Canada, by identifying the factors that influence the likelihood and severity of car accidents. Additionally, we compare the performance of two different machine learning algorithms for classification.

2 Dataset

Our research uses a dataset of Canadian Car Accidents from 1994 to 2014, which was constructed by Transport Canada. Transport Canada is a federal institution from Canada that promotes safe, secure, efficient and environmentally responsible transportation. The dataset is download from Kaggle (link). The data set consists of 22 categorical variables and 5,860,405 observations. Each observation corresponds to a person who was involved in a car crash. Therefore, one car crash can have multiple observations depending on the number of people involved. The variables have different numbers of categories, ranging from 2 to more than 30 (see Annex). The variables can be grouped into three main categories: Collision level, Vehicle level and Person level.

Our research aims to examine the factors that influence the occurrence of fatalities in car crashes. For this purpose, we configure the data set by car crash level. We also acknowledge that the data set is highly imbalanced, as most of the car crashes do not result in fatalities. We address this issue by using an undersampling technique that we will explain later in this report.

3 Exploratory Data Analysis

In this report, we will perform an Exploratory Dat Analysis on the data set of car accidents from Canada. We will clean and filter the data, explore the variables, distributions, trends and relationships in the data, as well as identify any potential issues or limitations.

3.0.1 Cleaning and Filtering of Data

We start by loading the data stored in a .csv file into the cars variable, keeping the columns names as headers.

```
#Load Data
cars <- read.csv("NCDB_1999_to_2014.csv", header=TRUE)
```

Table 1: Head of the Dataset (columns 1 to 7)

C_YEAR	C_MNTH	C_WDAY	C_HOUR	C_SEV	C_VEHS	C_CONF
1999	01	1	20	2	02	34
1999	01	1	20	2	02	34
1999	01	1	20	2	02	34
1999	01	1	08	2	01	01
1999	01	1	08	2	01	01
1999	01	1	17	2	03	QQ

Table 2: Head of the Dataset (columns 8 to 14)

C_RCFG	C_WTHR	C_RSUR	C_RALN	C_TRAF	V_ID	V_TYPE
UU	1	5	3	03	01	06
UU	1	5	3	03	02	01
UU	1	5	3	03	02	01
UU	5	3	6	18	01	01
UU	5	3	6	18	99	NN
QQ	1	2	1	01	01	01

Table 3: Head of the Dataset (columns 15 to 22)

V_YEAR	P_ID	P_SEX	P_AGE	P_PSN	P_ISEV	P_SAFE	P_USER
1990	01	M	41	11	1	UU	1
1987	01	M	19	11	1	UU	1
1987	02	F	20	13	2	02	2
1986	01	M	46	11	1	UU	1
NNNN	01	M	05	99	2	UU	3
1984	01	M	28	11	1	UU	1

The tables 1, 2 and 3 show that most of the variables are character type, except for Collision severity (C_SEV), which is categorical, and Year (C_YEAR), which is numerical. Hence, we will treat all the variables as categorical in our analysis, as explained in the subsequent sections. A detailed description of each variable is given in the Annex. We will also examine each level of each variable in depth, since there are many levels. The Exploratory Data Analysis section will present the categories of the variables more clearly. We should note that some values such as UU, QQ and XX indicate missing information.

3.0.2 Removing missing values

Before exploring the dataset, we checked if there were any null values across its columns.

```
na_count <-sapply(cars, function(y) sum(is.na(y)))
print(na_count)
```

```
## C_YEAR C_MNTH C_WDAY C_HOUR  C_SEV C_VEHS C_CONF C_RCFG C_WTHR C_RSUR C_RALN
##      0      0      0      0      0      0      0      0      0      0      0
## C_TRAF   V_ID V_TYPE V_YEAR   P_ID  P_SEX  P_AGE  P_PSN P_ISEV P_SAFE P_USER
##      0      0      0      0      0      0      0      0      0      0      0
```

The last chunk of code returned zero null values in the dataset. As there could also be null values encoded as blank spaces, we identified 3 rows whose values were empty under column C_VEHS (Number of vehicles involved in collision).

```
#Evaluating blank space
na_count <-sapply(cars, function(y) sum(y == ""))
print(na_count)
```

```
## C_YEAR C_MNTH C_WDAY C_HOUR  C_SEV C_VEHS C_CONF C_RCFG C_WTHR C_RSUR C_RALN
##      0      0      0      0      0      3      0      0      0      0      0
## C_TRAF   V_ID V_TYPE V_YEAR   P_ID  P_SEX  P_AGE  P_PSN P_ISEV P_SAFE P_USER
##      0      0      0      0      0      0      0      0      0      0      0
```

It could be possible that there is corrupted data in these 3 rows, so we analyzed them before taking any action.

```
cars[cars$C_VEHS == "",]
```

```
##         C_YEAR C_MNTH C_WDAY C_HOUR C_SEV C_VEHS C_CONF C_RCFG C_WTHR C_RSUR
## 5400116   2013     07      3     16     2            QQ     01      1      1
## 5400117   2013     07      3     16     2            QQ     01      1      1
## 5500948   2013     10      6     15     2            QQ     UU      1      Q
##         C_RALN C_TRAF V_ID V_TYPE V_YEAR P_ID P_SEX P_AGE P_PSN P_ISEV P_SAFE
## 5400116      3     UU   99     NN   NNNN   01     M    19    99      2     NN
## 5400117      3     UU   99     NN   NNNN   02     M    18    99      1     NN
## 5500948      1     UU   99     NN   NNNN   01     M    54    99      2     NN
##         P_USER
## 5400116      3
## 5400117      3
## 5500948      3
```

As we mentioned before, there are special categorical variables assigned to accidents where certain information is unknown or the data was not provided by the jurisdiction. These 3 rows included many of these special values, so clearly the accident data was collected although having missing important information.

So it's safe to not consider these 3 rows.

```
cars = cars %>% filter(C_VEHS != "")
```

About these special categorical values, while it makes sense that the data were tagged with them for historical purposes, for our research they are missing data so they don't add any value.

These values are the following:

```
* U/UU/UUUU: Unknown,
* NN/NNNN: Data element is not applicable,
* QQ: Choice is other than the preceeding values,
* XXXX: Jurisdiction does not provide this data element.
```

So we updated the dataframe in order to not consider rows that include these values, with the exception of C_WTHR and C_CONF where the value QQ can be of interest for the study.

```
cars = cars %>% filter(C_YEAR != 'U' & C_YEAR != 'UU' &
                       C_MNTH != 'U' & C_MNTH != 'UU' &
                         C_WDAY != 'U' & C_WDAY != 'UU' &
                         C_HOUR != 'U' & C_HOUR != 'UU' &
                         C_VEHS != 'U' & C_VEHS != 'UU' &
                         C_CONF != 'U' & C_CONF != 'UU' &
                         C_RCFG != 'U' & C_RCFG != 'UU' &
                         C_RCFG != 'QQ' &
                         C_MNTH != 'U' & C_MNTH != 'UU' &
                         C_WTHR != 'U' & C_WTHR != 'UU' &
                         C_RSUR != 'U' & C_RSUR != 'UU' &
                         C_RALN != 'U' & C_RALN != 'UU' &
                         C_RALN != 'Q' &
                         C_TRAF != 'U' & C_TRAF != 'UU' &
                         C_TRAF != 'QQ' &
                         P_PSN != 'NN' & P_PSN != 'QQ' &
                         V_ID != 'U' & V_ID != 'UU' &
```

```
                      V_TYPE != 'U' & V_TYPE != 'UU' &
                      V_YEAR != 'U' & V_YEAR != 'UU' &
                      V_YEAR != 'NNNN' & V_YEAR != 'UUUU' &
                      V_YEAR != 'XXXX' &
                      P_ID != 'U' & P_ID != 'UU' &
                      P_SEX != 'U' & P_SEX != 'UU' &
                      P_AGE != 'U' & P_AGE != 'UU' & P_AGE != 'NN' &
                      P_PSN != 'U' & P_PSN != 'UU' &
                      P_ISEV != 'U' & P_ISEV != 'UU' &
                      P_ISEV != 'N' &
                      P_SAFE != 'U' & P_SAFE != 'UU' &
                      P_SAFE != 'NN' & P_SAFE != 'QQ' &
                      P_USER != 'U' & P_USER != 'UU')
```

After that, we checked the unique values for every column in the dataframe in order to confirm there were no more of these special categorical values.

```
lapply(cars, unique,decreasing=FALSE)
```

```
## $C_YEAR
##  [1] 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013
## [16] 2014
##
## $C_MNTH
##  [1] "01" "02" "03" "04" "05" "06" "07" "08" "09" "10" "11" "12"
##
## $C_WDAY
## [1] "1" "2" "3" "4" "5" "6" "7"
##
## $C_HOUR
##  [1] "15" "09" "20" "05" "08" "14" "07" "13" "11" "22" "10" "18" "16" "12" "17"
## [16] "06" "03" "21" "01" "23" "00" "04" "19" "02"
##
## $C_SEV
## [1] 2 1
##
## $C_VEHS
##  [1] "01" "02" "03" "04" "06" "09" "05" "07" "13" "08" "12" "14" "10" "26" "16"
## [16] "71" "11" "21" "27" "15" "46" "25" "31" "18" "56" "23" "36" "17" "20" "19"
## [31] "29" "28" "38" "32" "22" "35" "33" "72" "40" "44" "58" "30" "77" "24" "34"
## [46] "39" "51" "57" "43" "37"
##
## $C_CONF
##  [1] "QQ" "34" "03" "01" "33" "21" "04" "24" "35" "31" "02" "23" "32" "06" "36"
## [16] "41" "05" "22" "25"
##
## $C_RCFG
##  [1] "01" "02" "03" "05" "04" "06" "08" "07" "09" "10"
##
## $C_WTHR
## [1] "1" "3" "4" "2" "6" "5" "7" "Q"
##
## $C_RSUR
```

```
##  [1] "1" "2" "5" "3" "Q" "7" "4" "6" "8" "9"
##
## $C_RALN
## [1] "1" "3" "4" "2" "5" "6"
##
## $C_TRAF
##  [1] "06" "01" "05" "18" "03" "04" "07" "08" "10" "16" "17" "02" "13" "11" "15"
## [16] "09" "12"
##
## $V_ID
##  [1] "01" "02" "03" "04" "05" "06" "07" "08" "09" "13" "10" "11" "12" "14" "15"
## [16] "16" "19" "20" "21" "22" "23" "25" "26" "18" "28" "31" "33" "34" "35" "36"
## [31] "37" "38" "39" "40" "41" "42" "43" "44" "45" "46" "47" "50" "53" "56" "57"
## [46] "60" "61" "62" "65" "68" "71" "17" "24" "27" "29" "30" "32" "48" "49" "51"
## [61] "52" "54" "55" "58" "59" "63" "64" "66" "67" "69" "70" "72" "73" "74" "75"
## [76] "76" "77"
##
## $V_TYPE
##  [1] "01" "06" "08" "11" "07" "09" "17" "14" "05" "18" "10" "23" "21"
##
## $V_YEAR
##   [1] "1995" "1992" "1988" "1989" "1986" "1990" "1994" "1998" "1984" "1993"
##  [11] "1987" "1996" "1991" "1999" "1997" "1985" "1981" "1983" "1977" "1979"
##  [21] "1982" "1978" "1974" "1980" "1973" "1970" "1975" "1971" "1976" "1968"
##  [31] "1969" "1972" "1966" "1967" "1945" "2000" "1965" "1950" "1959" "1955"
##  [41] "1958" "1964" "1909" "1923" "1960" "1963" "1914" "1908" "1953" "1906"
##  [51] "1925" "1949" "1938" "1907" "1961" "1917" "1962" "1944" "1956" "1930"
##  [61] "1931" "1939" "1951" "1946" "1952" "1947" "1957" "1943" "1954" "1901"
##  [71] "1948" "1937" "1935" "1926" "1941" "1932" "1912" "1920" "1903" "1933"
##  [81] "1919" "2001" "1913" "1940" "2002" "1916" "1929" "1928" "1942" "1918"
##  [91] "2003" "1924" "1922" "1915" "1934" "2004" "2005" "1904" "1927" "2006"
## [101] "2007" "2008" "1911" "2009" "2010" "2011" "2012" "2013" "1921" "2014"
## [111] "2015"
##
## $P_ID
##  [1] "01" "02" "03" "04" "05" "06" "07" "08" "09" "10" "11" "12" "13" "14" "15"
## [16] "16" "17" "18" "19" "20" "21" "22" "23" "24" "25" "26" "27" "28" "29" "30"
## [31] "31" "32" "33" "34" "35" "36" "37" "38" "39" "40" "41" "42" "43" "44" "45"
## [46] "46" "49" "47" "48" "50" "51" "52" "53" "54" "55" "56" "57" "58" "59" "60"
## [61] "61" "62" "63" "64" "65" "66" "67" "68" "69" "70" "71" "72" "73" "74" "75"
## [76] "76" "77" "78" "79" "80" "81" "82" "83" "84" "85" "86" "87" "88" "89" "90"
## [91] "91" "92" "93"
##
## $P_SEX
## [1] "M" "F"
##
## $P_AGE
##  [1] "17" "33" "70" "38" "34" "30" "18" "68" "28" "37" "50" "20" "53" "71" "26"
## [16] "51" "41" "63" "25" "23" "55" "36" "16" "35" "60" "13" "44" "61" "65" "75"
## [31] "79" "32" "62" "85" "21" "49" "19" "29" "92" "57" "86" "52" "31" "74" "59"
## [46] "48" "43" "15" "12" "76" "39" "73" "69" "04" "27" "47" "22" "54" "66" "45"
## [61] "42" "80" "46" "78" "40" "24" "82" "11" "07" "72" "56" "64" "14" "84" "58"
## [76] "01" "77" "67" "09" "02" "10" "06" "08" "03" "05" "83" "88" "81" "87" "89"
## [91] "95" "90" "91" "94" "99" "93" "98" "96" "97"
```

```
##
## $P_PSN
##  [1] "11" "13" "23" "21" "12" "22" "32" "96" "33" "31" "98" "97"
##
## $P_ISEV
## [1] "1" "2" "3"
##
## $P_SAFE
## [1] "02" "01" "13" "12" "09" "10"
##
## $P_USER
## [1] "1" "2" "4" "5"
```

After removing missing values, the dataset ended up having 3497249 rows, being reduced by approximately 2 million rows.

3.0.3 Reducing the dataset to accident level

We decided to focus on the conditions at the time of the accident, rather than the characteristics of each individual involved. This choice was motivated by the greater number and relevance of the variables related to the former perspective. These variables include factors such as weather, week day, and month. To perform this analysis, we assigned a unique ID to each accident based on these features, and then we removed the duplicate entries by keeping only one accident per ID. After this, we discarded the variables that were not related to the environmental factors of the accidents.

```
cars["ID"] <- paste(cars$C_YEAR, cars$C_MNTH, cars$C_WDAY, cars$C_HOUR, cars$C_SEV,
                    cars$C_VEHS, cars$C_CONF, cars$C_RCFG, cars$C_WTHR, cars$C_RSUR,
                    cars$C_RALN, cars$C_TRAF, sep = "")

#We will keep a copy of the original dataset
#in case that then we want to create additional features
cars_copy <- cars

#Keep one accident by ID, eliminate duplicate accidents
cars <- cars[!duplicated(cars[ , "ID"]), ]
```

3.1 Exploring Data Distributions

To understand more our dataset and variables we carried out an Exploratory Data Analysis divided in four parts:

1. Periodicity of accidents

An evaluation of how the number of accidents has increased/decreased in the period 1999 - 2014 and then evaluate how variables like day of the week, hour or seasonal factors could influence it.

2. Variables distribution

A general overview of how frequent some dataset variables values are compared to the others values this variable could take.

3. Differences between fatal and non-fatal accidents in variable distribution

An evaluation of which variable values could be most likely to appear depending of the severity of the accident according to its frequency in our dataset.

4. Person level analysis

An exploration of some factors that could be related to a fatality at a person level like the use of seatbelt or the age of a person.

Periodicity of accidents

- By year (Figure 1.a): A decrease over the years is seen, this effect could be attributed to the result of public policies in favour of the improvement of road designs, road surfaces, traffic signs and vial education. Also, it could be attributed to the improvement of technology and security standard in cars.
- By day of the week (Figure 1.b): Car accidents peaks on Friday, while decreasing over the weekend with a minimum on Sunday.
- By hour (Figure 1.c): The peak time of cars accidents is during the afternoon between 15-17 pm.
- Grouping by day of the week and hour two things stand out (Figure 1.d):
  1. The frequency of accidents occurring past midnight increases on Saturday and Sunday (left side of each plot)
  2. During the weekdays (Monday - Friday) accidents in the morning are most likely to happen.
- Grouping by month and day (Figures 1.e, 2 and 3): The months with highest frequency of accidents are November, December and January which could mean there's a factor of seasonality involved, given that winter starts in December and prolongues until February.
- Periodicity of fatal accidents by day of the week and hour (Figure 4): The plot clearly shows that fatal accidents reach a peak on friday and specially during the weekend, not only for those that happens in the early morning but also for those in the afternoon.

3.1.1 Variables distribution

Collision level variables

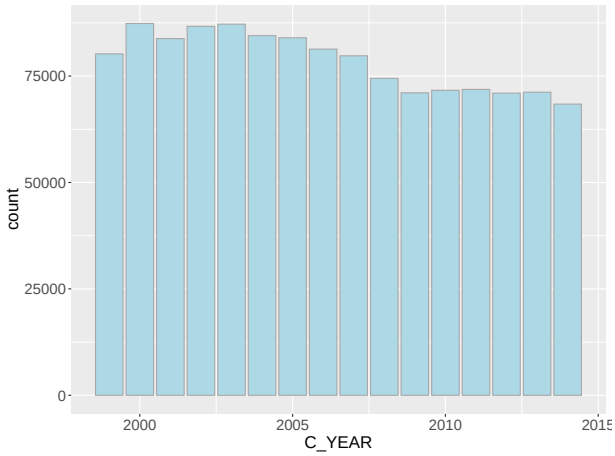
- Severity of accident (Figure 5.a): As expected, in the dataset there are way more non-fatal accidents than fatal ones.
- Number of vehicles involved in collision (Figure 5.b): Because of the number of values that this variable could take, we consider the top 10 frequent values. The typical accident scenario is the one where two vehicles are involved.
- Collision configuration: Because there are 3 categories involving collisions, first we do a general overview of the frequencies (Figure 6.a) and then we focus in each category.

  The types of collisions with highest frequencies are:

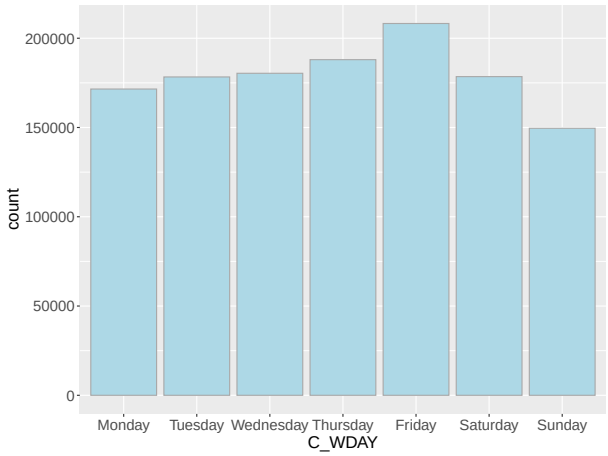
  06: Other single vehicle collision configuration (Figure 6.b)

  21: Rear-end collision (a vehicle crashes into the one in front of it) (Figure 6.c)

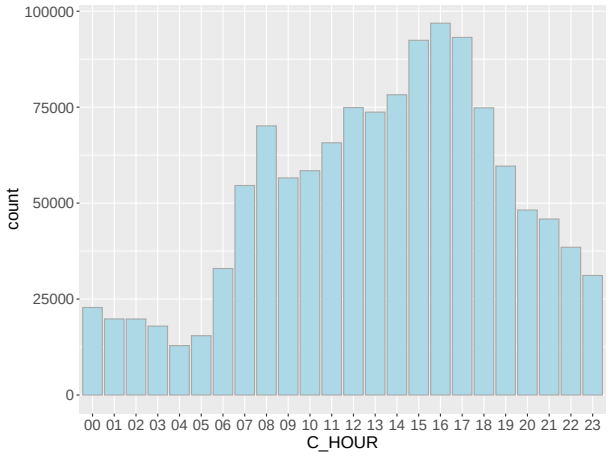
  35: Right angle collision (vehicles traveling on perpendicular streets when one driver fails to yield the right of way to the other) (Figure 6.d)



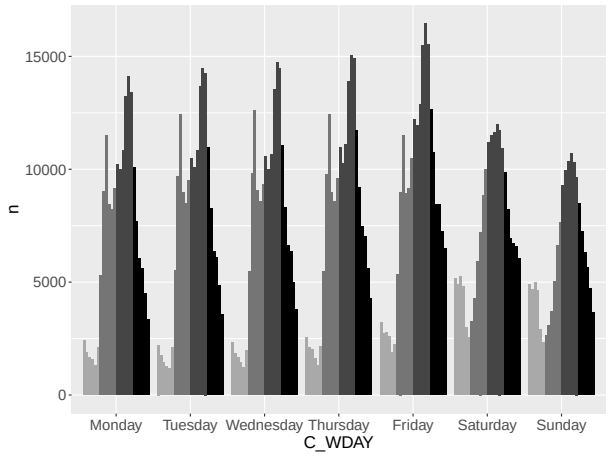
(a) Year distribution



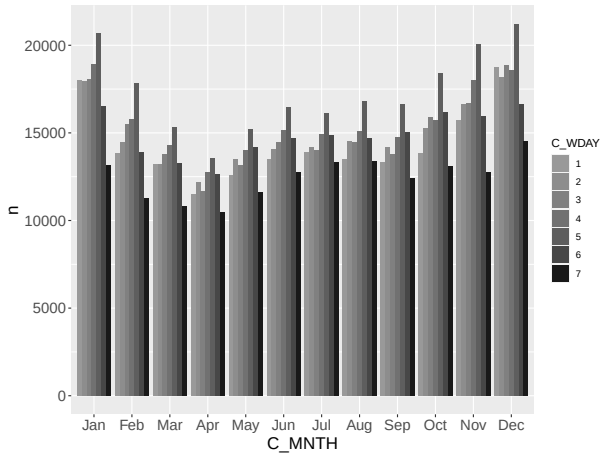
(b) Day of the week distribution



(c) Hour distribution

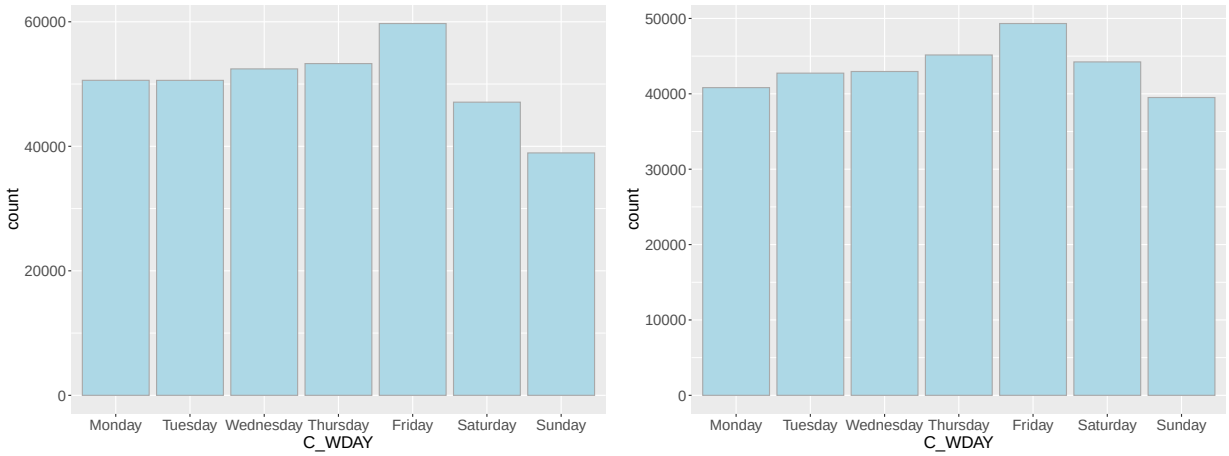


(d) Frequency of accidents by day of the week and hour



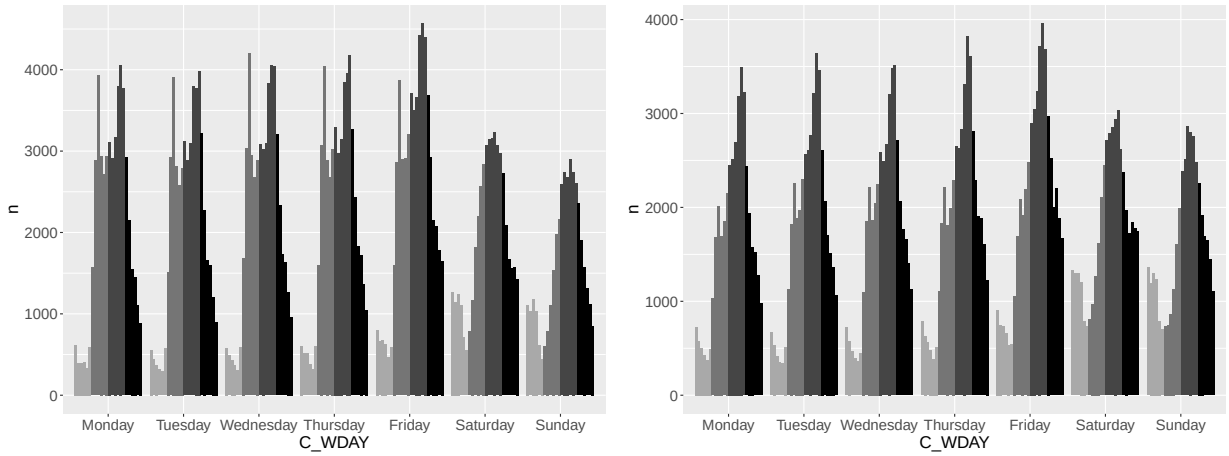
(e) Frequency of accidents by month and day of the week

Figure 1: Periodicity of accidents



(a) Winter (b) Summer

Figure 2: Day of the week distribution by season



(a) Winter (b) Summer

Figure 3: Frequency of accidents by day of the week and hour

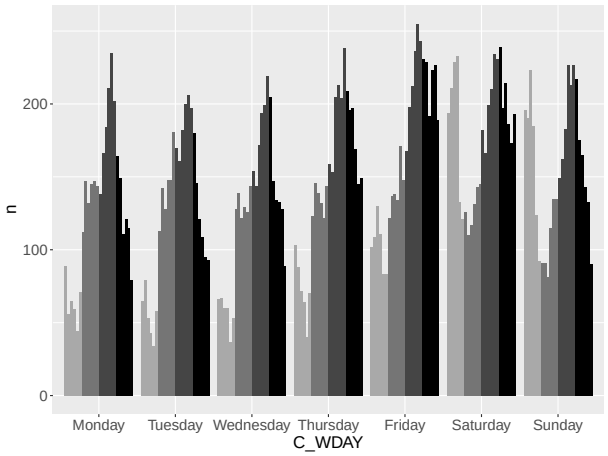
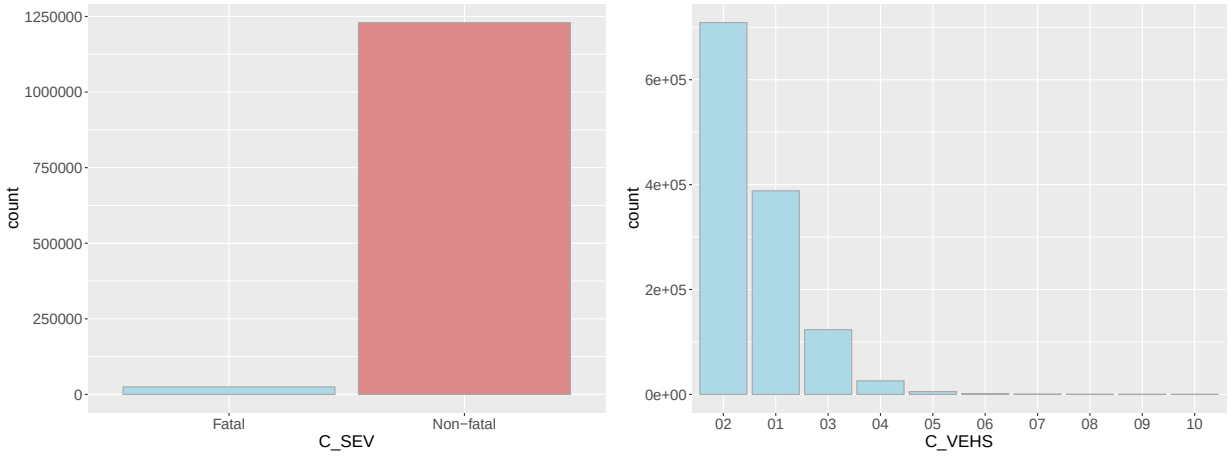


Figure 4: Frequency of fatal accidents by day of the week and hour

- Roadway configuration (Figure 7.a): The top-2 roadway configuration values are understandable given that we've previously found that the most frequent collisions are the ones involving cars crashing into another car in front of them and ones involving vehicles traveling in perpendicular roads .
- Weather condition (Figure 7.b): Besides clear and sunny condition being the most frequent, we should consider in the analysis when it's cloudy, raining or snowing.



(a) Severity of accidents distribution

(b) Number of vehicles involved in collision distribution

Figure 5: Collision level variables Part 1

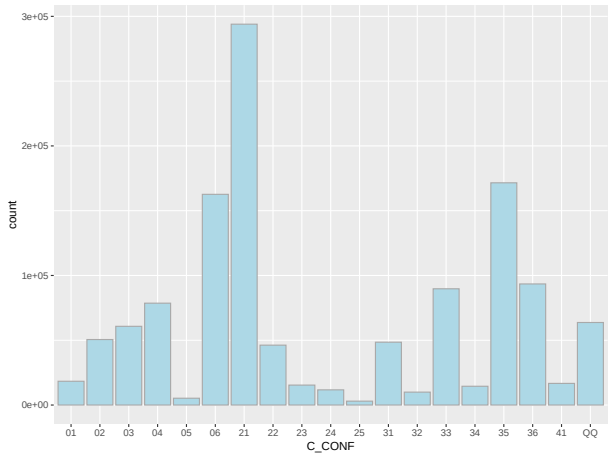
- Road surface (Figure 8.a): There's a seasonal factor involved with road surface with snow, wet and icy characteristics being part of the most frequent values.
- Road alignment (Figure 8.b): Accidents with straight road alignment are more usual but roads with gradient and curved characteristics are important to consider in posterior analysis.
- Traffic control (Figure 8.c): The common case in the dataset is that accidents occur where no traffic control is present.

Vehicle level variables

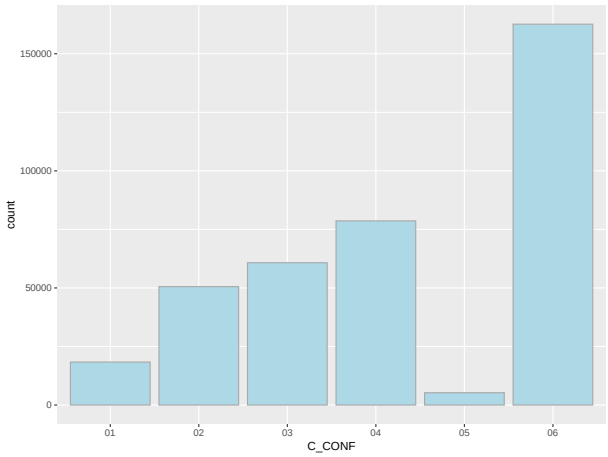
- Vehicle type (Figure 9.a): Light duty vehicles are by a large margin the most frequent vehicle types involved in accidents.
- Vehicle model year (Figure 9.b): This variable on its own doesn't give us much clues about the characteristics of an accident.

3.1.2 Person level variables

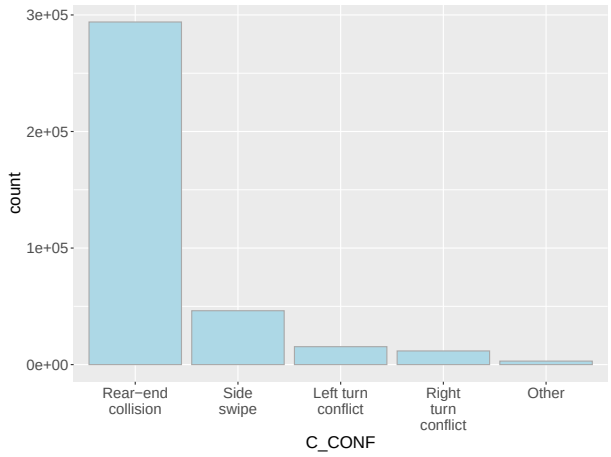
- Person involved in the collision by gender (Figure 10.a): In the dataset, there are more males involved in an accident than females.
- Person involved in the collision by age (Figure 10.b): In the dataset, the age range of 20-30 is the one more frequent to be part of a car accident.



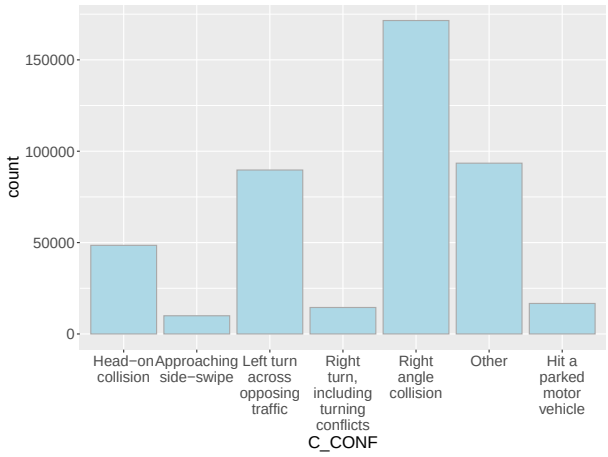
(a) Collision configuration distribution



(b) Collision involving single vehicle in motion

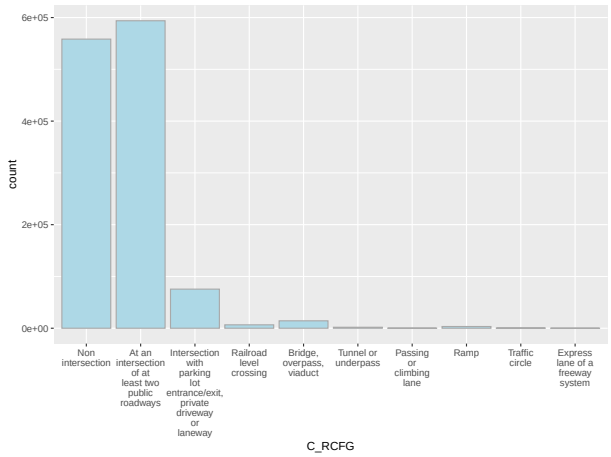


(c) Collision involving two vehicles in motion - same direction of travel

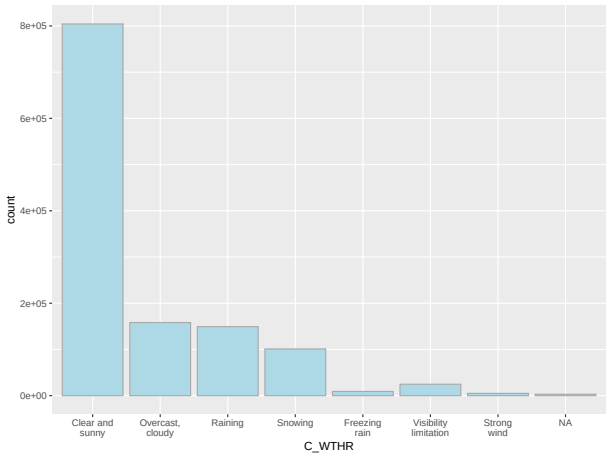


(d) Collision involving two vehicles in motion - different direction of travel

Figure 6: Collision configuration by category

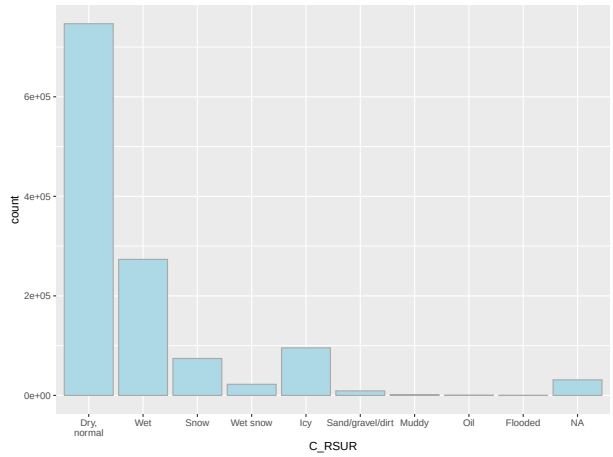


(a) Roadway configuration distribution

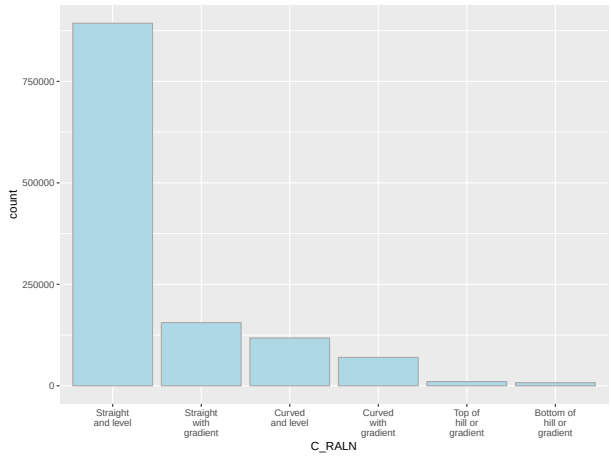


(b) Weather condition distribution

Figure 7: Collision level variables Part 2



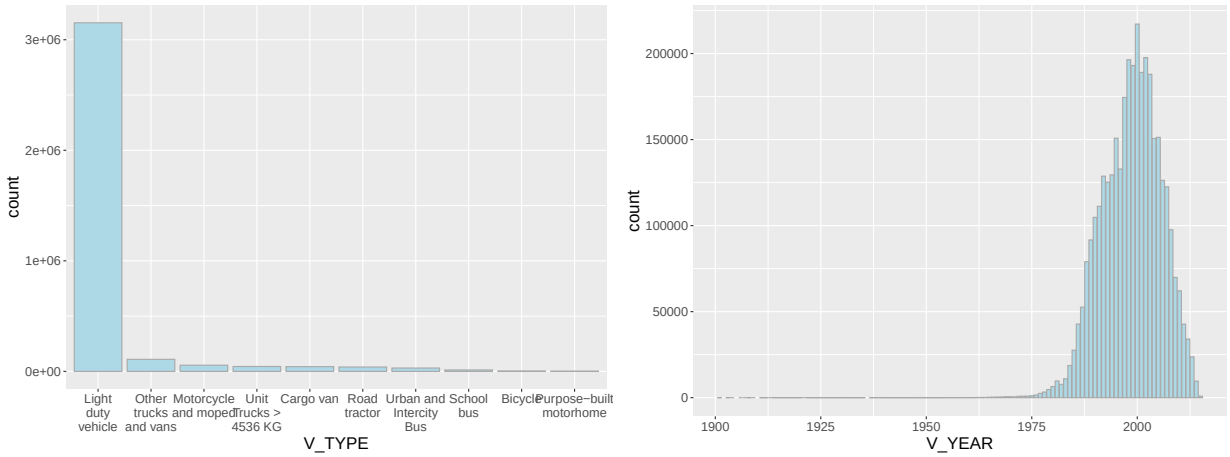
(a) Road Surface distribution



(b) Road Alignment distribution

(c) Traffic Control distribution

Figure 8: Collision level variables Part 3

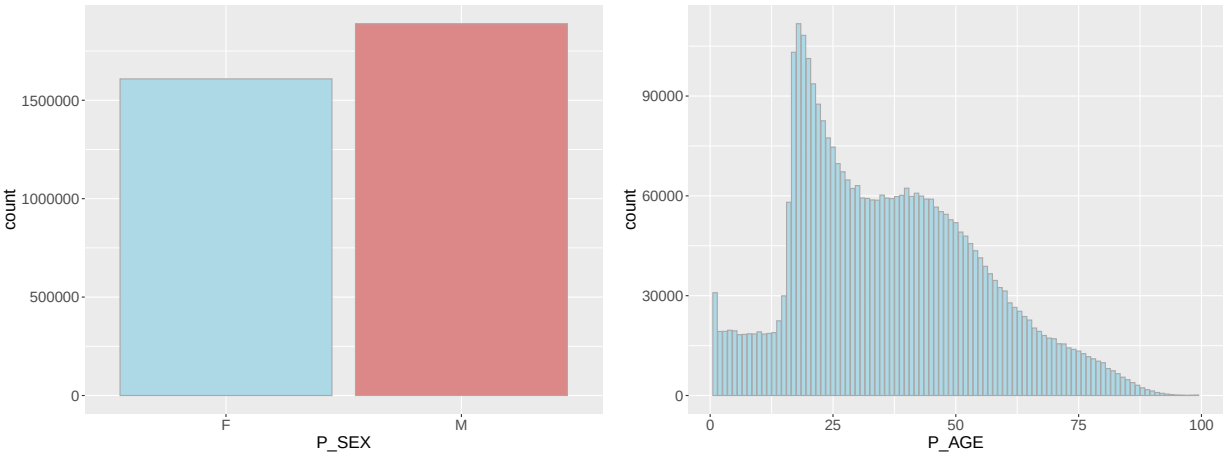


(a) Vehicle type distribution

(b) Vehicle model year distribution

Figure 9: Vehicle level variables

- Person involved in the collision position in the vehicle (Figure 11): As expected, most of the positions distributions are from drivers.
- Medical treatment for a person involved in an accident (Figure 12.a): People involved in a car accident are more likely to receive medical treatment for injuries.
- Safety device used by a person involved in an accident (Figure 12.b): By a large margin, safety device was used.
- Road user class (Figure 13) : There are more car accidents were there's a motor vehicle driver/passenger affected than the ones where pedestrians are.



(a) Person sex distribution

(b) Person age distribution

Figure 10: Person level variables

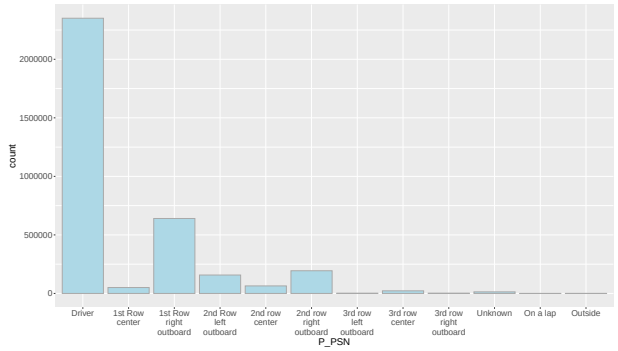
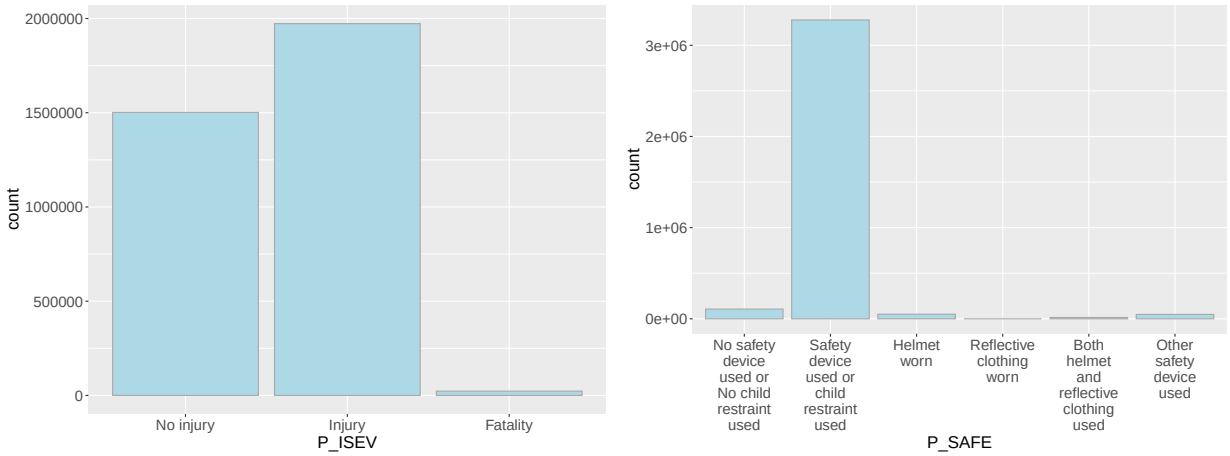


Figure 11: Person position distribution



(a) Medical treatment required distribution

(b) Safety device used distribution

Figure 12: Person level variables Part 2

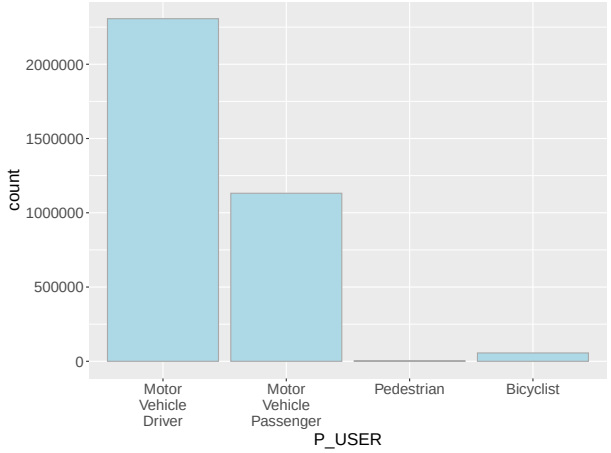
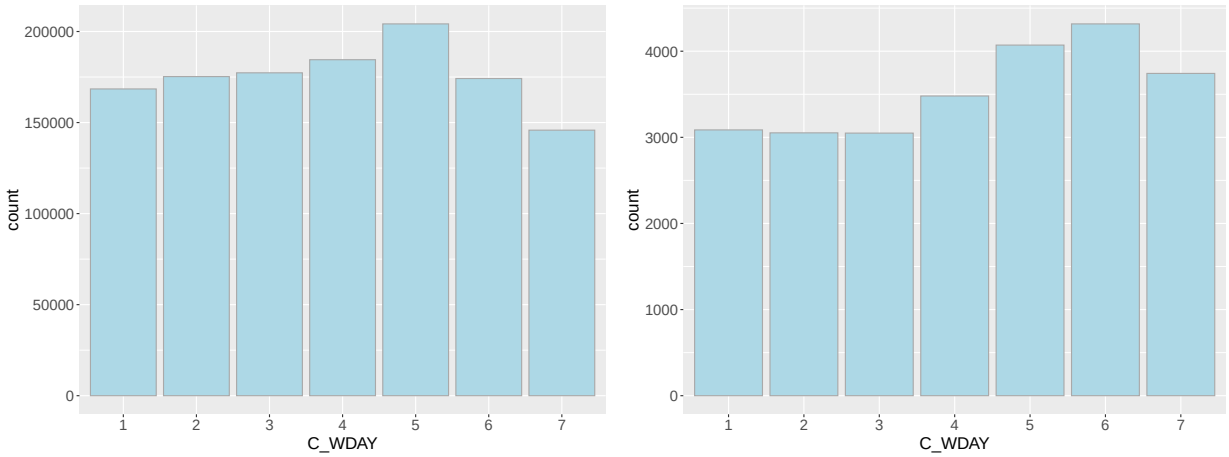


Figure 13: Road user class distribution

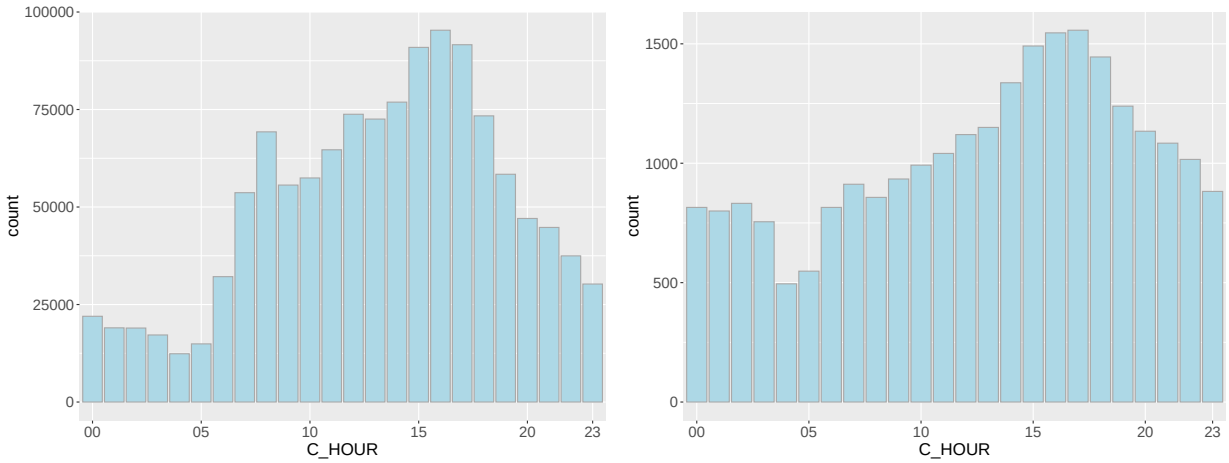
3.1.3 Differences between fatal and non-fatal accidents in variables distribution

- Day of the week (Figure 14): While there's a decrease in the number of non-fatal accidents on Saturday and Sunday, for fatal accidents, the quantity of those increases from Thursday to Saturday.
- Hour of the day (Figure 15): Besides the fact that fatal accidents occur more frequently during early morning, it's good to point out that there's an increase of accidents taking place during 8-9am, probably because of people commuting to their workplace.



(a) Non-fatal accidents (b) Fatal accidents

Figure 14: Weekday

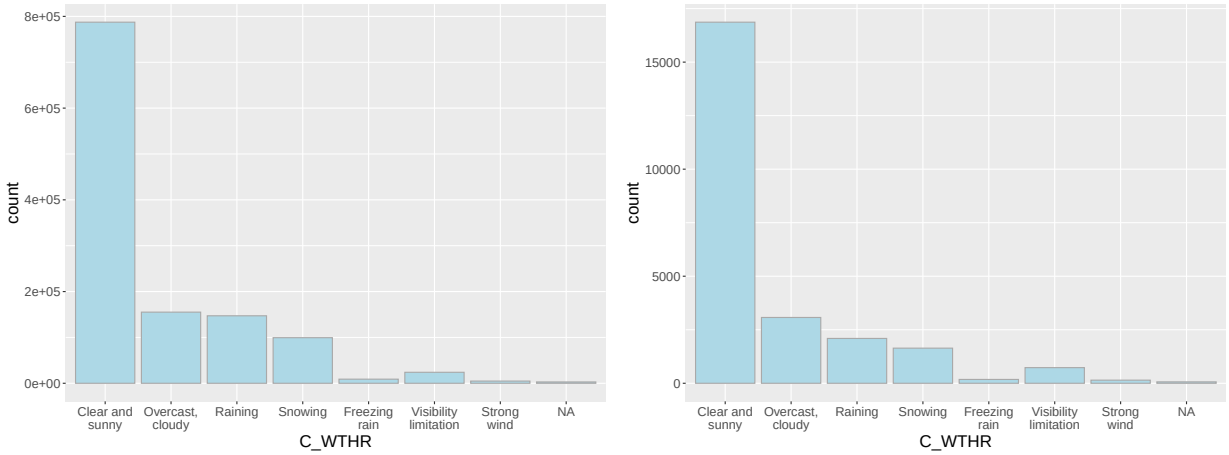


(a) Non-fatal accidents (b) Fatal accidents

Figure 15: Hour of the day

- Weather condition (Figure 16): There's not much of a difference compared to the general variable distribution.
- Number of vehicles (Figure 17): In fatal accidents the frequency of accidents involving one vehicle and those involving 2 are mostly the same.

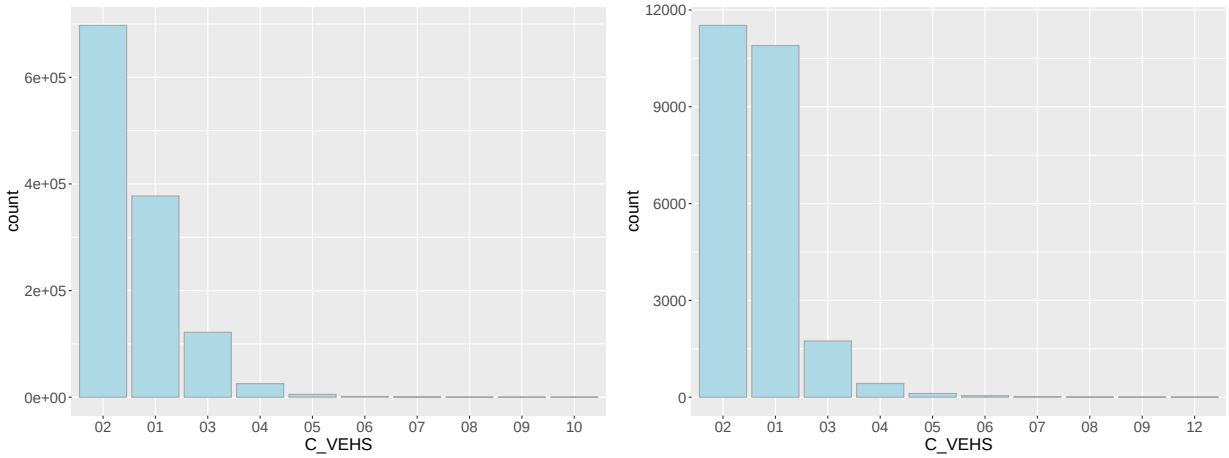
- Roadway configuration (Figure 18): In fatal accidents, accidents occurring in non-intersection of road are more frequent than those occurring in intersection of public roadways. While in non-fatal accidents, those two configuration share similar frequencies.



(a) Non-fatal accidents

(b) Fatal accidents

Figure 16: Weather Condition



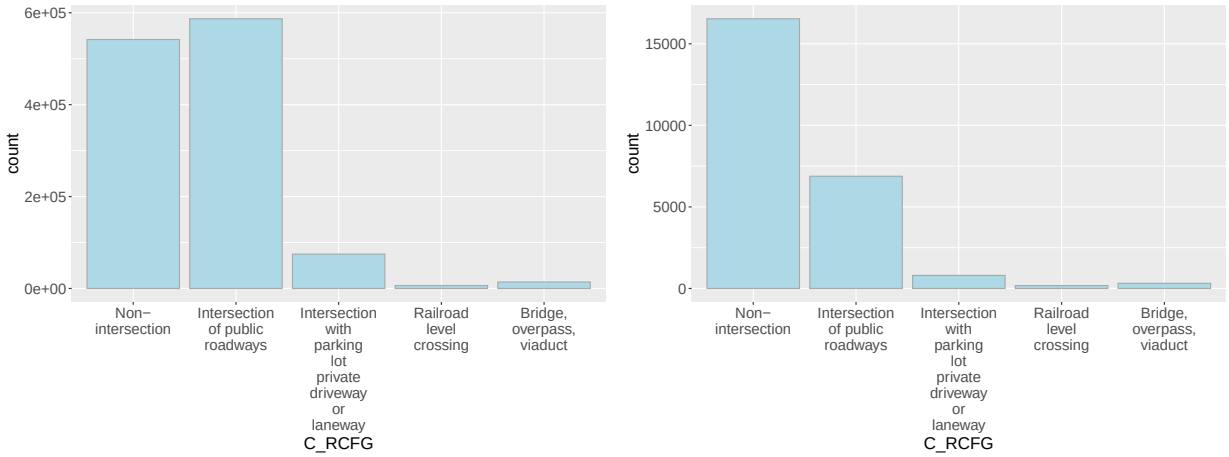
(a) Non-fatal accidents

(b) Fatal accidents

Figure 17: Number of vehicles

- Road Surface (Figure 19): There's not much of a difference compared to the general variable distribution.

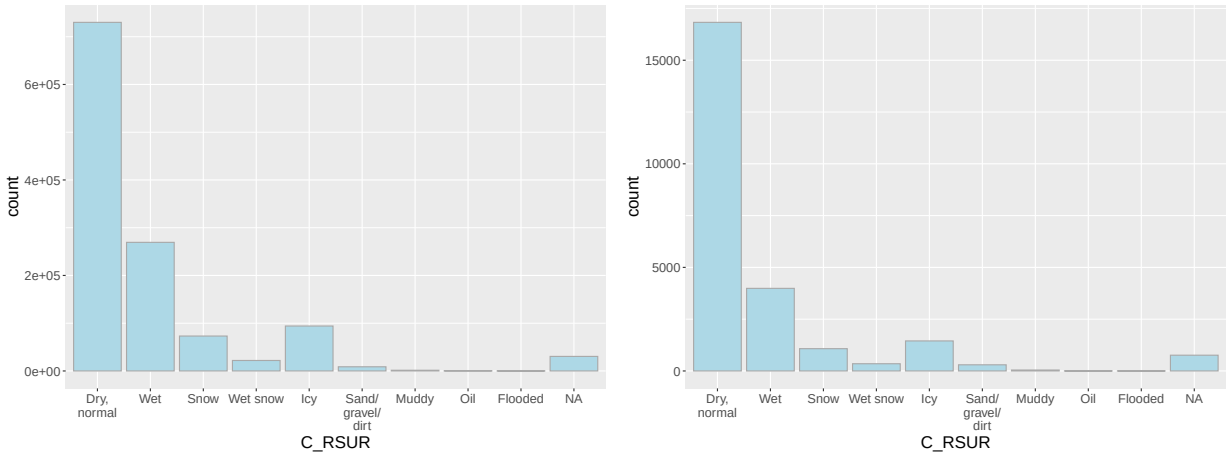
- Road Alignment (Figure 20): Fatalities occurs mainly in curved and level, and curved with gradient roads compared to non-fatalities accidents.
- Vehicle type (Figure 21): There's not much of a difference compared to the general variable distribution.
- Traffic control (Figure 22): Fatal accidents are more likely to occur where no control is present. While in non-fatal accidents, there's a good portion of accidents happening where traffic signs are located.



(a) Non-fatal accidents

(b) Fatal accidents

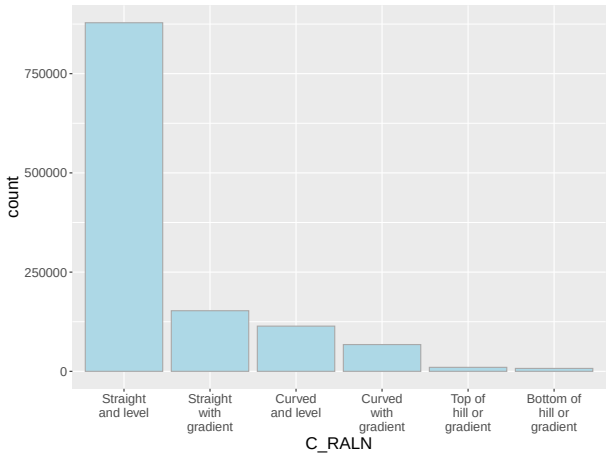
Figure 18: Roadway Configuration



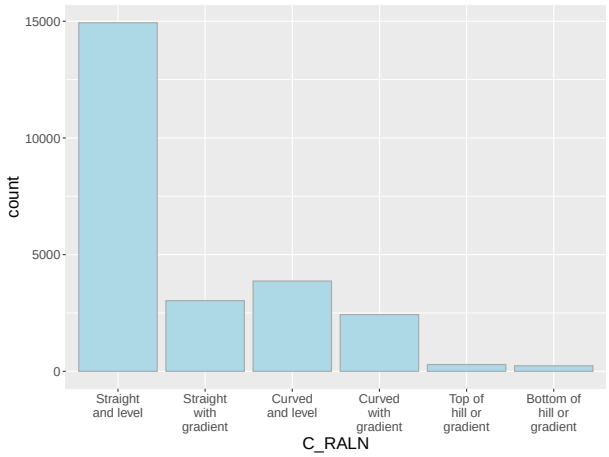
(a) Non-fatal accidents

(b) Fatal accidents

Figure 19: Road Surface

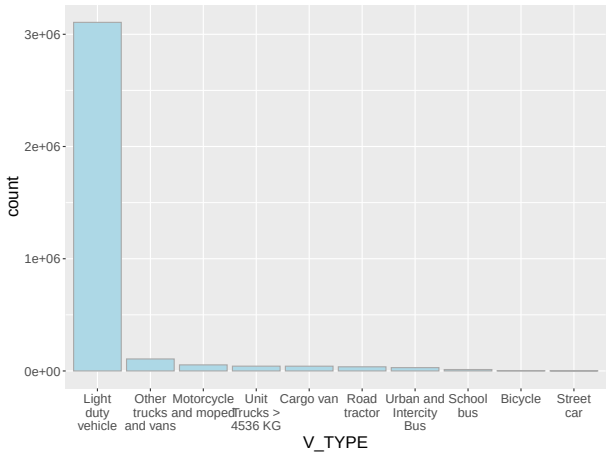


(a) Non-fatal accidents

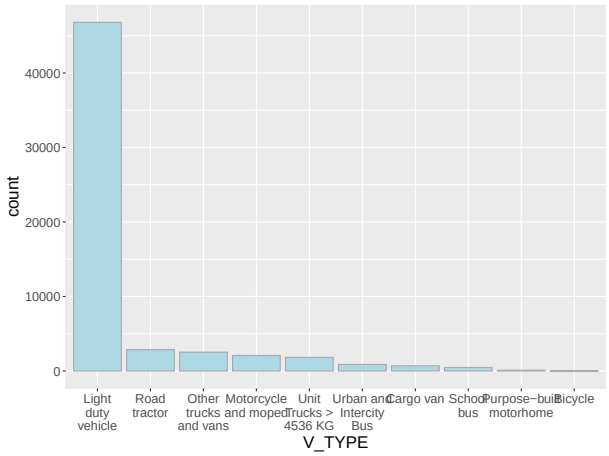


(b) Fatal accidents

Figure 20: Road Alignment

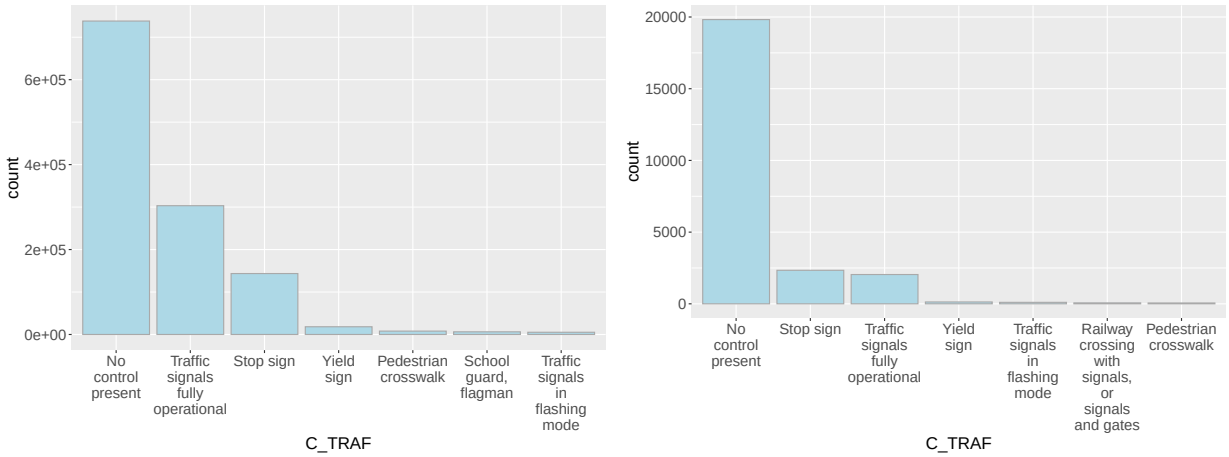


(a) Non-fatal accidents



(b) Fatal accidents

Figure 21: Vehicle Type

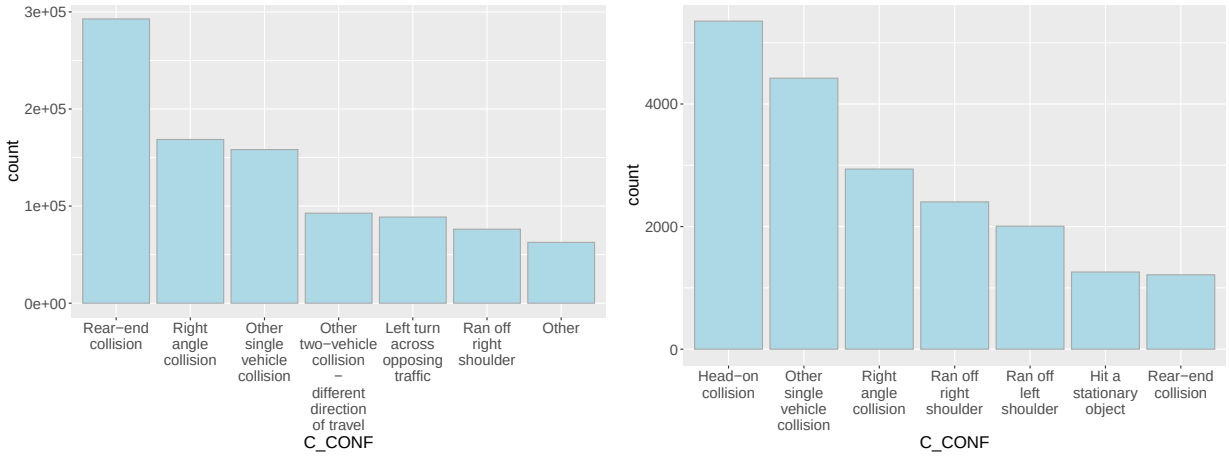


(a) Non-fatal accidents

(b) Fatal accidents

Figure 22: Traffic Control

- Collision Configuration (Figure 23): Here we notice a great difference between non-fatal and fatal accidents. While in non-fatal accidents the frequencies for the values that this variable could take are similar to the ones found in the general variable distribution, this is not the case for fatal accidents. Head-on collisions are the most frequent case for fatal accidents, meaning that could be a key factor for determining the severity of the accident.
- Road user class (Figure 24): Severe accidents between a vehicle and a bicyclists are more likely to end up in a fatality for the second.



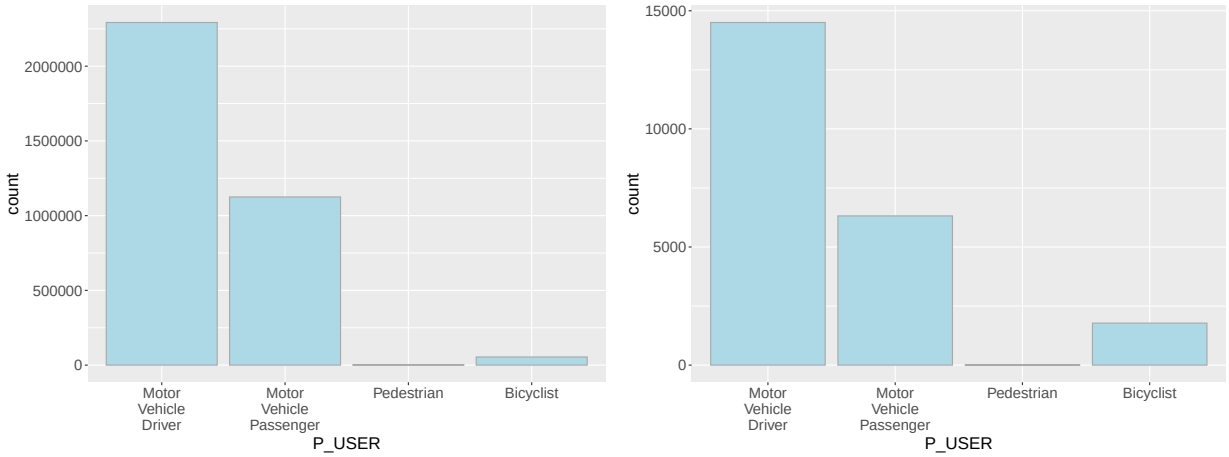
(a) Non-fatal accidents

(b) Fatal accidents

Figure 23: Collision Configuration

3.1.4 Person Level Analysis

- Relationship between medical treatment required and the use of safety device in a fatal accident (Figure 25): When there's a severe accident and a person involved in it is not using a safety device, the most



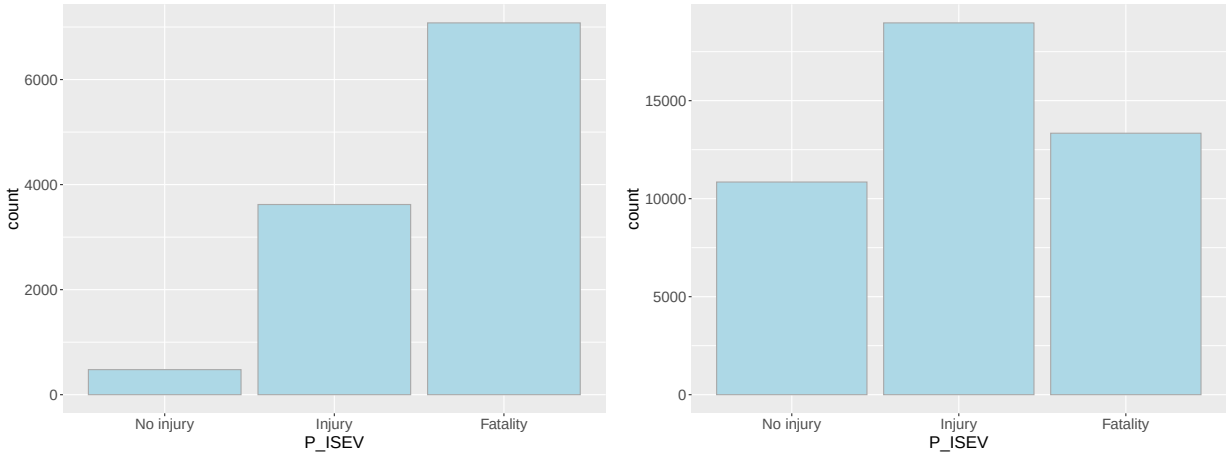
(a) Non-fatal accidents

(b) Fatal accidents

Figure 24: Road User Class

frequent case is that it will lead to a loss of life, while for the ones using a safety device, the most frequent case is that they will end up being injured.

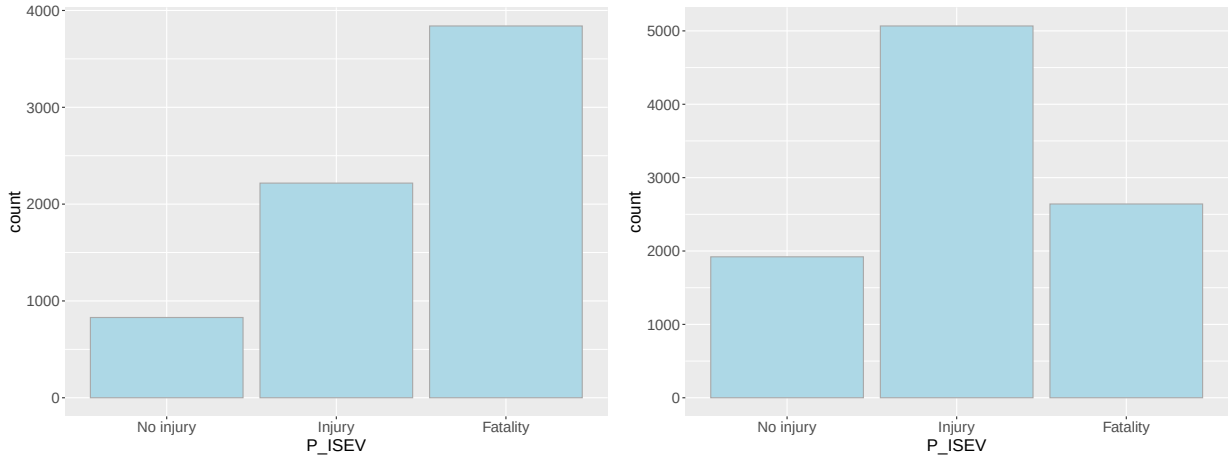
- Relationship between medical treatment required and an elderly involved in a fatal accident (Figure 26.a): When there's a severe accident and an elderly involved in it, the most frequent case is the loss of life for these risk population.
- Relationship between medical treatment required and an underage involved in a fatal accident (Figure 26.b): In the case of the underage, if they are involved in a fatal accident the frequent case is that they will receive medical treatment for an injury.



(a) Without safety device used

(b) With safety device used

Figure 25: Medical treatment for person in a fatal accident by the used of safety device



(a) An elderly involved in a fatal accident

(b) An underage person involved in a fatal accident

Figure 26: Medical treatment required when there is:

3.2 Reduced Data Frame

As we explained, we will focus on the analysis at accident level, so we will remove those variables that only make sense at person level and vehicle level.

```
#Eliminate variables that does not have sense at accident level
cars <- cars[, !names(cars) %in% c("V_ID", "V_TYPE", "V_YEAR", "P_ID", "P_SEX",
                                   "P_AGE", "P_PSN", "P_ISEV", "P_SAFE", "P_USER")]
```

3.3 Data Science questions

In this report, we assume the role of data scientists working for Transport Canada, the federal institution responsible for transportation policies and programs. Our main objective is to find the most effective way to allocate resources for traffic surveillance in Canada. To achieve this, we will address the following questions:

Regarding when/where to allocate resources:

- Which time and day of the week is more probable that an accident with fatalities occurs?
- What kind of weather condition is most prevalent during car accidents with fatalities?
- Is there a correlation between the road configuration and the severity of the accident?
- Is there a correlation between the road alignment and the severity of the accident?

Regarding preventive measures campaigns:

- How does the safety device affect the likelihood of casualties in traffic accidents?
- When there is elderly and underage people involved in a car accident, is it more probable to have fatalities reported?

Table 4: Dependent variable frequency

C_SEV_FACTOR	Frequency
0	1229743
1	24797

3.4 Dependent variable

We decided to use the binary variable C_SEV as the outcome variable, which represents the presence or absence of a fatality in the accident. For this purpose, we transformed the original variable to match 0 (collision producing non-fatal injury), and 1 (collision producing at least one fatality). In Table 4, the distribution of the dependent variable shows a heavy unbalanced data, with 1,229,743 examples for the majority class and 24,797 for the minority class.

```
cars$C_SEV_FACTOR <- case_when(cars$C_SEV == "1"  ~ "1",
                               cars$C_SEV == "2"   ~ "0")
cars$C_SEV_FACTOR <- factor(cars$C_SEV_FACTOR)
```

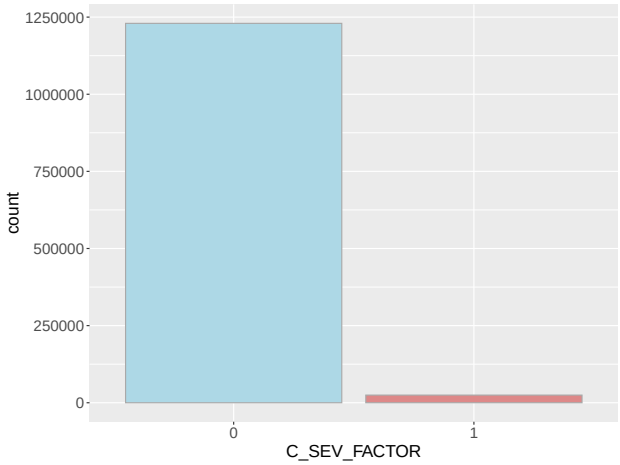


Figure 27: Dependent Variable

3.5 Feature Engineering

3.5.1 Dimensionality reduction

The previous graphs show that some of the categorical variables have many levels with low frequency. To make them easier to interpret and reduce the dimensionality, we grouped some of these levels together. We applied this modification to the following variables:

- Time of the day (C_HOUR): Specific hours are not needed. We rather focus on time frames, ending up with four groups (early morning, morning, afternoon, evening).
- Numbers of cars involved in the accident (C_VEHS): Because of the frequency of the levels, it was modified to 1, 2, and 3 or more.
- Collision configuration (C_CONF): Because of the frequency of the levels, the existent divisions were classified in five groups.

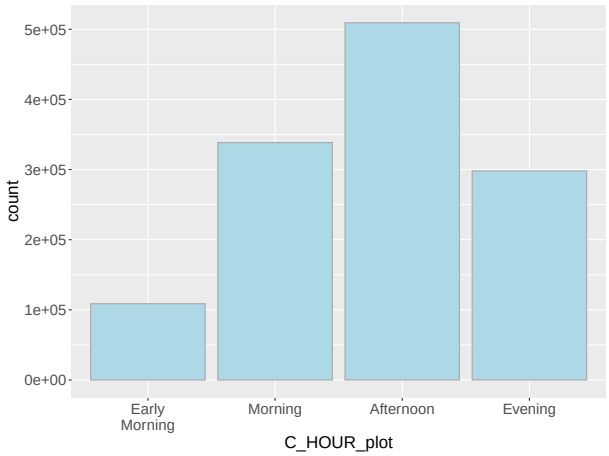
- Roadway Configuration (C_RCFG): Because of the frequency of the levels, the existent divisions were classified in three groups.
- Traffic control (C_TRAF): Because of the frequency of the levels, the existent divisions were classified in six groups.

```
# Level reduction
cars$C_HOUR_AGR <- case_when(
  C_HOUR_INT >= 0 &  C_HOUR_INT <= 5  ~ "Early Morning",
  C_HOUR_INT >= 6 &  C_HOUR_INT <= 11  ~ "Morning",
  C_HOUR_INT >= 12 &  C_HOUR_INT <= 17  ~ "Afternoon",
  C_HOUR_INT >= 18 &  C_HOUR_INT <= 23  ~ "Evening")

cars$C_VEHS_AGR <- case_when(
  cars$C_VEHS == "01"  ~ "1",
  cars$C_VEHS == "02"  ~ "2",
  cars$C_VEHS != "01" & cars$C_VEHS != "02" ~ "+3")

cars$C_CONF_AGR <- case_when(
  cars$C_CONF == "01" | cars$C_CONF == "02" | cars$C_CONF == "03" |
    cars$C_CONF == "04" | cars$C_CONF == "05" | cars$C_CONF == "06"  ~ "1",
  cars$C_CONF == "21" | cars$C_CONF == "22" | cars$C_CONF == "23" |
    cars$C_CONF == "24" | cars$C_CONF == "25"  ~ "2",
  cars$C_CONF == "31" | cars$C_CONF == "32" | cars$C_CONF == "33" |
    cars$C_CONF == "34" | cars$C_CONF == "35" | cars$C_CONF == "36"  ~ "3",
  cars$C_CONF == "41"  ~ "4",
  cars$C_CONF == "QQ"  ~ "5")

cars$C_RCFG_AGR <- case_when(
  cars$C_RCFG == "01"  ~ "1",
  cars$C_RCFG == "02"  ~ "2",
  cars$C_RCFG != "01" & cars$C_RCFG != "02"  ~ "3")

cars$C_TRAF_AGR <- case_when(
  cars$C_TRAF == "01" ~ "1",
  cars$C_TRAF == "03" ~ "3",
  cars$C_TRAF == "04" ~ "4",
  cars$C_TRAF == "06" ~ "6",
  cars$C_TRAF == "18" ~ "18",
  cars$C_TRAF != "01" & cars$C_TRAF != "03" & cars$C_TRAF != "4"
  & cars$C_TRAF != "06" & cars$C_TRAF != "18" ~ "Other")
```

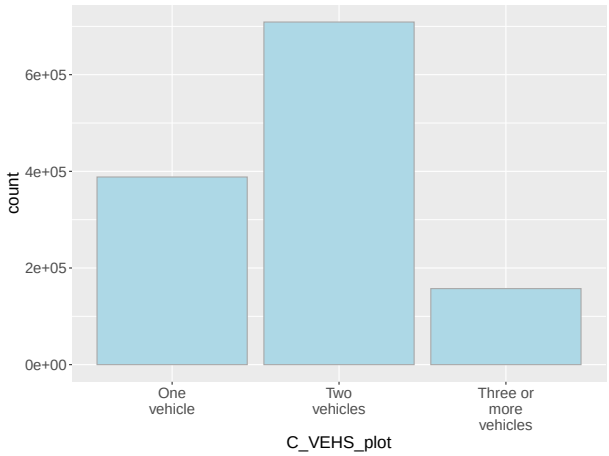
3.6 Transformation of variables to factor

Up to now we have a mix of variable types, but all of them should be categorical. Hence, we transform all the variables into factor. All variables show no specific order.

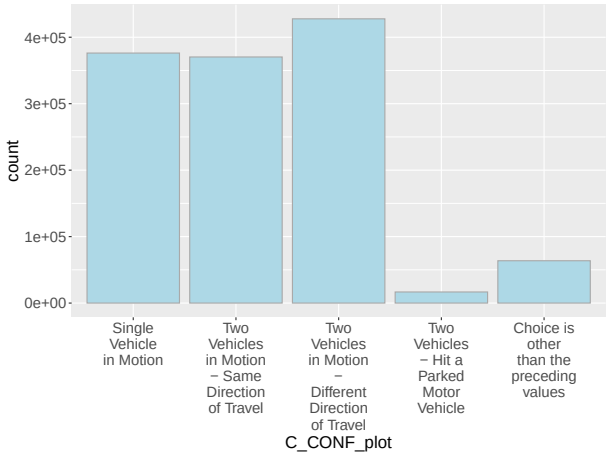
```
#Transform variables to factor
cars$C_MNTH <- factor(cars$C_MNTH,
                      labels = c("jan", "feb", "mar", "apr", "may", "jun", "jul",
                                 "aug", "sep", "oct", "nov", "dic"))
cars$C_WDAY <- factor(cars$C_WDAY,
```



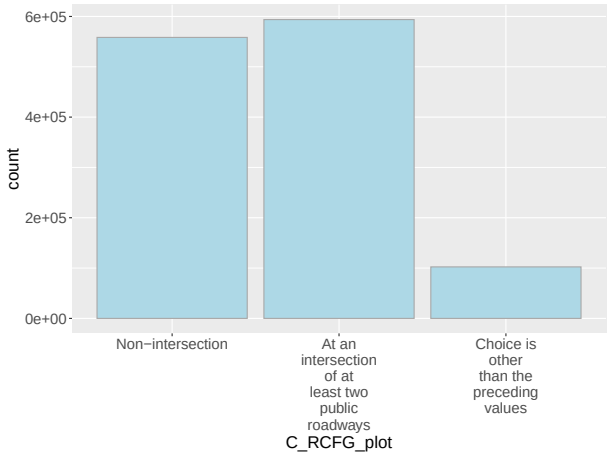
(a) C HOUR AGR distribution



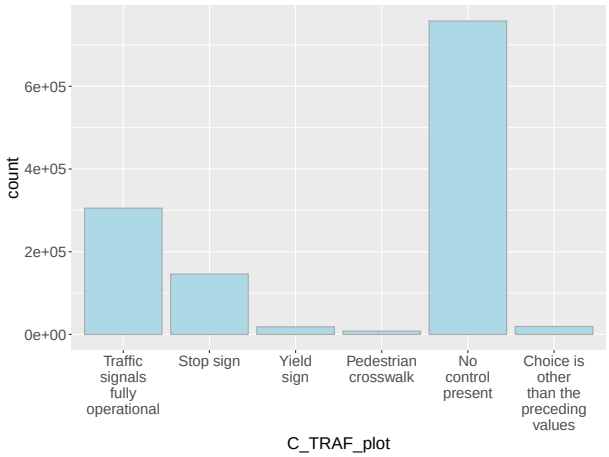
(b) C VEHS AGR distribution



(c) C CONF AGR distribution



(d) C RCFG AGR distribution



(e) C TRAF AGR distribution

Figure 28: Reduced Variables

```
                      labels = c("mon", "tue", "wed", "thu","fri", "sat", "sun"))
cars$C_HOUR_AGR <- factor(cars$C_HOUR_AGR)
cars$C_VEHS_AGR <- factor(cars$C_VEHS_AGR,
                          levels = c("1", "2", "+3"))
cars$C_CONF_AGR  <- factor(cars$C_CONF_AGR,
                           labels = c("single", "two same direction",
                                      "two different direction",
                                      "hit a parked motor vehicle", "other"))
cars$C_RCFG_AGR <- factor(cars$C_RCFG_AGR,
                          labels = c("non-intersection", "at an intersection", "other"))
cars$C_WTHR  <- factor(cars$C_WTHR,
                       labels = c("clear", "overcast", "raining", "snowing",
                                  "freezing rain", "visibility limitation",
                                  "string wind", "other"))
cars$C_RSUR  <- factor(cars$C_RSUR,
                       labels = c("dry", "wet", "snow", "slush", "icy", "sand/dirt",
                                  "muddy", "oil", "flooded", "other"))
cars$C_RALN  <- factor(cars$C_RALN)
cars$C_TRAF_AGR  <- factor(cars$C_TRAF_AGR,
                           labels = c("traffic signals fully operational",
                                      "stop sign", "yield sign", "pedestrian crosswalk",
                                      "no control present", "other"))
```

3.7 Variable creation

We created three new variables from the original ones to preserve the information at person level and to measure its effect on the output. The first variable, P_CHILD, represents the impact of having people under age of majority involved in the accident. The second variable, P_ELD, represents the impact of having people aged 65 or older involved in the accident. The third variable, C_SAFE, represents the impact of having people who did not use safety devices at the time of the accident. After this, we set the values of these variables to 1 when there exists at least one person in the accident who satisfies this condition.

```
cars_copy$P_AGE <- as.numeric(cars_copy$P_AGE)

cars_copy$P_CHILD <- case_when(cars_copy$P_AGE <= 18  ~ "1",
                           cars_copy$P_AGE > 18   ~ "0")

cars_copy$P_ELD <- case_when(cars_copy$P_AGE < 65  ~ "0",
                           cars_copy$P_AGE >= 65   ~ "1")

cars_copy$C_SAFE <- case_when(cars_copy$P_SAFE == "01" | cars_copy$P_SAFE == "13"  ~ "1",
                           cars_copy$P_SAFE != "01" & cars_copy$P_SAFE != "13"  ~ "0")
```

```
#Collapse dummies one by accident
cars_2 <- cars_copy[,c('ID','P_CHILD','P_ELD', 'C_SAFE')] %>%
  group_by(ID) %>%
  summarise_all(max)

#Merge the new variables with our dataset
cars <- merge(cars,cars_2, by = "ID", all.x = TRUE, all.y = FALSE)

#Factor these new variables
```

Table 5: Head of the Final Dataframe

C_SEV_FACTOR	C_MNTH	C_WDAY	C_HOUR_AGR	C_VEHS_AGR	C_CONF_AGR
0	jan	mon	Early Morning	1	single
0	jan	mon	Early Morning	1	single
0	jan	mon	Early Morning	1	single
0	jan	mon	Early Morning	2	two same direction
0	jan	mon	Early Morning	2	two same direction
0	jan	mon	Early Morning	2	two different direction

Table 6: Head of the Final Dataframe

C_TRAF_AGR	C_RSUR	C_WTHR	C_RALN	C_RCFG_AGR	P_CHILD	P_ELD	C_SAFE
stop sign	slush	clear	1	non-intersection	1	0	0
stop sign	icy	clear	1	non-intersection	0	0	1
stop sign	dry	overcast	1	non-intersection	0	0	0
traffic signals fully operational	icy	clear	1	at an intersection	1	0	0
traffic signals fully operational	icy	snowing	1	at an intersection	0	0	0
traffic signals fully operational	snow	snowing	1	at an intersection	0	0	0

```
cars$P_ELD  <- factor(cars$P_ELD)
cars$P_CHILD  <- factor(cars$P_CHILD)
cars$C_SAFE  <- factor(cars$C_SAFE)
```

3.8 Final Dataframe

```
accidents <- cars[,c('C_SEV_FACTOR','C_MNTH','C_WDAY', 'C_HOUR_AGR','C_VEHS_AGR',
                     'C_CONF_AGR', 'C_TRAF_AGR', 'C_RSUR','C_WTHR','C_RALN',
                     'C_RCFG_AGR','P_CHILD','P_ELD', 'C_SAFE')]
nrows.base.df2 <- nrow(accidents)
```

Finally, the data set is composed by 1254540 examples and 14 columns (13 independent variables).

4 Model

Our objective is to predict whether or not a car accident is likely to end with fatalities or not based on different characteristic of the accident. So, as the output variable is car crash with fatality or not, this problem is a binary classification problem. We decided to apply two popular methods for classification problems: Logistic regression and KNN. Logistic regression is a parametric method that assumes a linear relationship between the input variables and the output variable. It uses a logistic function to model the probability of an outcome, such as whether a car accident is fatal or not. KNN is a non-parametric method that does not make any assumptions about the data distribution. It uses the distance between the input variables and the nearest neighbors to assign a class label, such as fatal or non-fatal.

Both methods have advantages and disadvantages for classifying car accidents. Logistic regression can provide confidence levels for its predictions and can handle multiple input variables easily. However, it may not capture non-linear patterns in the data and may be affected by outliers and collinearity. KNN can handle non-linear solutions and does not require any training. However, it may be slower than logistic regression and may be sensitive to the choice of K and the distance metric.

4.1 Undersampling

The dataset we are working with has a severe class imbalance problem, meaning that the fatal car crash cases are much less frequent than the non-fatal ones. This can affect the model's performance and accuracy, as it tends to predict the majority class more often and neglect the minority class. To address this issue, we apply undersampling, which reduces the number of samples from the non-fatal class to make the dataset more balanced. Another common technique is oversampling, but we choose undersampling because our dataset is large and the difference between the classes is significant. Instead, oversampling would create a lot of synthetic data from the fatal class, which may not be representative of the real cases.

One way to reduce the imbalance between the classes of fatal and non-fatal accidents is to use random under-sampling on the majority class. This can be done with the "ovun" function from the ROSE package in R, which randomly selects a subset of observations from the non-fatal accidents class.

```
set.seed(123)
#We set a seed to replicate results
cars_u <- ovun.sample(C_SEV_FACTOR ~ .,
                      data = accidents, seed = 123, method = "under")$data
```

4.2 Logistic Regression

The dependent variable for this research is C_SEV_FACTOR and stands for accident severity. It takes the value 1 if the car accidents has at least one fatality and 0 if the car crash has non fatality. The logistic regression equation for our problem is:

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_k x_k$$

where p is the probability of the event, in this case the probability of a fatal accident, β_0 is the intercept, $\beta_1, \beta_2, ..., \beta_k$ are the coefficients, and $x_1, x_2, ..., x_k$ are the predictor variables. For our model, all the independent variables are categorical, therefore they were transformed to factor in the previous section.

Logistic regression determines the coefficients that make the observed outcome (non-fatal or fatal accident) most likely using the maximum-likelihood technique.

We use the glm function in R to fit the logistic regression as follows:

```
# Fit the model
model_full <- glm(C_SEV_FACTOR~., data = cars_u, family = binomial)
```

Before going deeper in the analysis of the logistic regression results, we have to check the model assumptions and its performance.

4.2.1 Model Assumptions

4.2.1.1 Multicollinearity (Independency): Look for correlations among categorical variables

To look for dependencies between features, we used Cramer's V test, which is based on the Chi Square statistic. It varies from 0 (corresponding to no association between the variables) to 1 (complete association) and can reach 1 only when each variable is completely determined by the other. The heat map in Figure 29 depicts the strength of association between the different explanatory variables.

```
cramer.matrix <- function(x) {
  names <- colnames(x);
  ndim <- length(names)
  stats <- matrix(nrow=ndim, ncol=ndim, dimnames = list(names, names))
  for (i in 1:ndim) {
    for (j in i:ndim) {
      stats[i,j] = cramerV(x[,i],x[,j])
      if (i != j) {
        stats[j,i] = NA
      }
    }
  }
  return (stats)
}
```

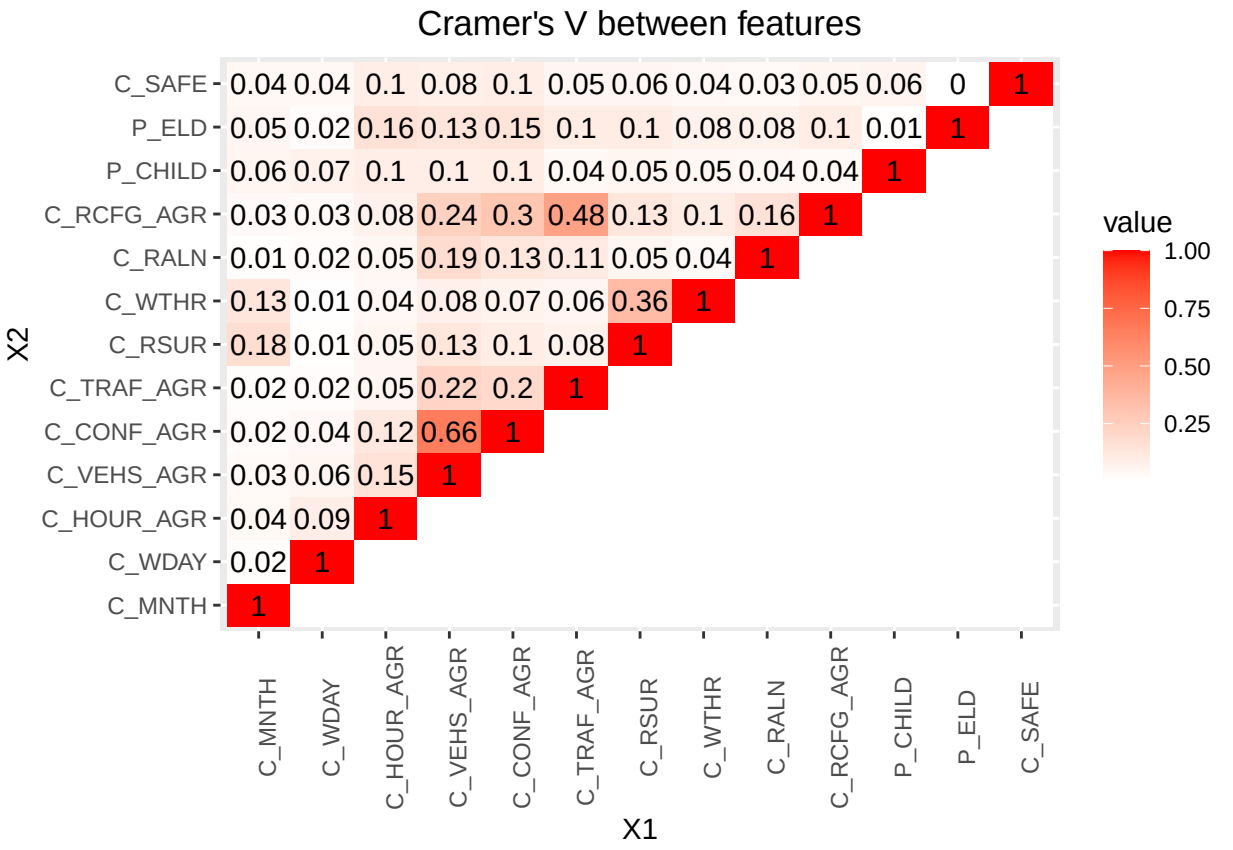


Figure 29: Cramers V between features

Based on these results, it was possible to preliminarily identify pair of variables that are dependent from each other:

- Collision configuration (C_CONF_AGR) and Number of vehicles involved in collision (C_VEHS_AGR): The number of vehicles determines the configuration of the collision.
- Weather conditions (C_WTHR) and Road surface (C_RSUR): The type of weather explains in most cases the characteristics of the surface.

- Traffic control (C_TRAF) and Roadway configuration (C_RCFG_AGR): The type of roadway could explain the presence or absence of traffic signs.

To confirm these findings, we plotted the variables against each other.

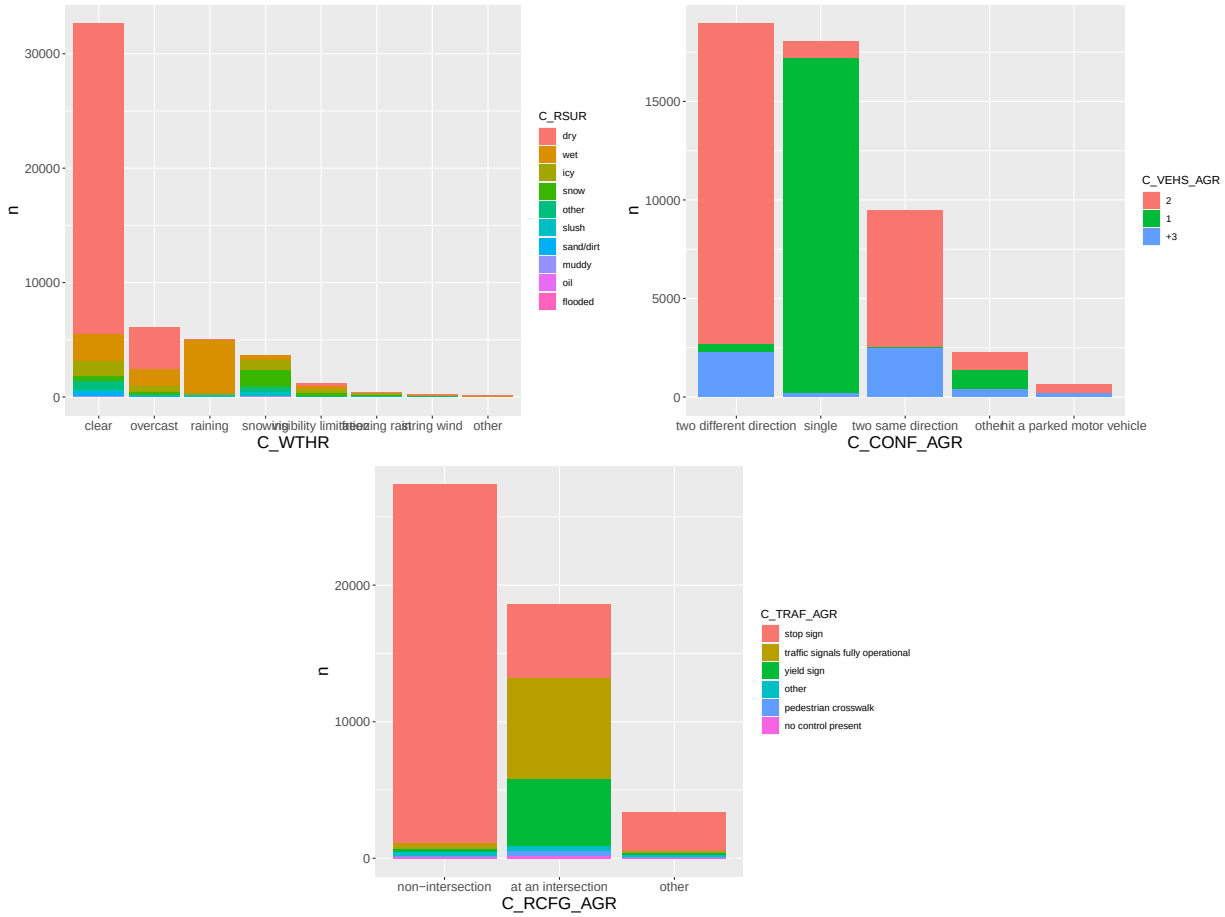


Figure 30: Correlated Variables

It is important to mention that we used the Chi-square statistic to test the independence of the variables, but we preferred the Cramer's V as a measure of association, since it indicates how strong the relationship is, and is also independent from the sample size. Additionally, we considered using Yule's Q, but this metric compares each pair of levels, which was not suitable for our data set with many levels.

Regarding the normality of residuals, logistic regression models are not based on the normal distribution. Therefore, the normality of residuals is not an assumption for logistic regression models. Same applies for KNN, which is a non-parametric method and does not make any assumptions about the distribution of the data.

4.2.1.2 Removing Correlated variables

To avoid multicollinearity, we need to drop one of the correlated variables that the Cramer's V test revealed. We will use the AIC criterion to compare the full model and the models with each correlated variable omitted. The variable with the lowest AIC will be retained. If there is a tie, we will favor the variable with fewer levels for easier interpretation.

Table 7: AIC of models with Correlated Variables

	model	df	AICc
model_c2v2	- C_WTHR	56	55507.94
model_c1v1	- C_RSUR	51	55521.84
model_c1v2	- C_CONF_AGR	49	55671.87
model_c3v1	- C_VEHS_AGR	56	56075.67
model_c3v2	-C_RCFG_AGR	53	56175.44
model_c2v1	-C_TRAF_AGR	54	58780.18

```
#First pair of correlated variables
model_c1v1 <- glm(  C_SEV_FACTOR~. - C_WTHR, data = cars_u, family = binomial)
model_c1v2 <- glm(  C_SEV_FACTOR~. - C_RSUR, data = cars_u, family = binomial)
#Second pair of correlated variables
model_c2v1 <- glm(  C_SEV_FACTOR~. - C_CONF_AGR, data = cars_u, family = binomial)
model_c2v2 <- glm(  C_SEV_FACTOR~. -C_VEHS_AGR, data = cars_u, family = binomial)
#Third pair of correlated variables
model_c3v1 <- glm(  C_SEV_FACTOR~. -C_RCFG_AGR, data = cars_u, family = binomial)
model_c3v2 <- glm(  C_SEV_FACTOR~. -C_TRAF_AGR, data = cars_u, family = binomial)
```

As we can see from Table 7 following the AIC criteria, from the pair C_VEHS_AGR and C_CONF_AGR, it is clear that we have the model with the lowest AIC if we remove C_CONF_AGR. For the other two pairs the AIC criteria is not determinant, so we keep C_RCFG_AGR and C_WTHR because they have less levels, being easier to interpret.

After removing the correlated variables the model to be fit in R is as follows:

```
model <- glm(  C_SEV_FACTOR~. - C_RSUR - C_CONF_AGR -C_TRAF_AGR,
               data = cars_u, family = binomial)
```

We are going to interpret these results in the Results Section.

```
logistic.prob <- predict(model, type="response")
roc.out <- roc(cars_u$C_SEV_FACTOR, logistic.prob, levels=c(0, 1))
roc.coords <- coords(roc.out, x="best", input="threshold")
```

4.2.1.3 Accuracy, AUC and ROC

In order to identify the best threshold, we used the ROC curve depicted in the following figure. The chosen value is 0.5166719, which yields a specificity of 0.7506217 and a sensitivity of 0.5825705. The classifier has a fair performance with an AUC of 0.7293261.

```
##   threshold specificity sensitivity
## 1 0.5166719   0.7506217   0.5825705
```

4.2.1.4 Confusion matrix

The same results showed before can be assess by the confusion matrix. We can see that for the model is easier to assess correctly when a car accident has no fatalities, with a high specificity, but not that well when an car accident ends with fatalities, sensitivity of 0.5825705.

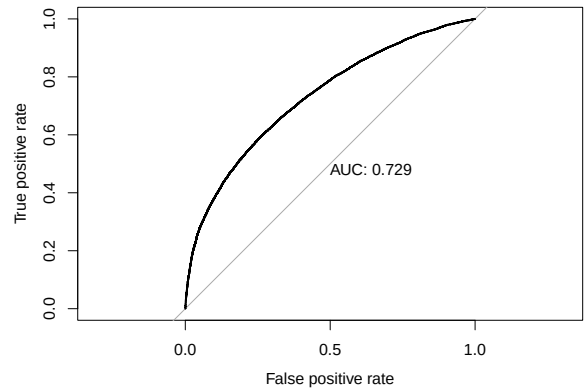


Figure 31: ROC

```
model_prob <- predict(model, newdata = cars_u, type = "response")
model_pred <- 1*(model_prob > roc.coords$threshold) + 0

conf_matrix <- table(pred = model_pred, true = cars_u$C_SEV_FACTOR)
cm <- confusionMatrix(data=factor(model_pred),reference=factor(cars_u$C_SEV_FACTOR))
```

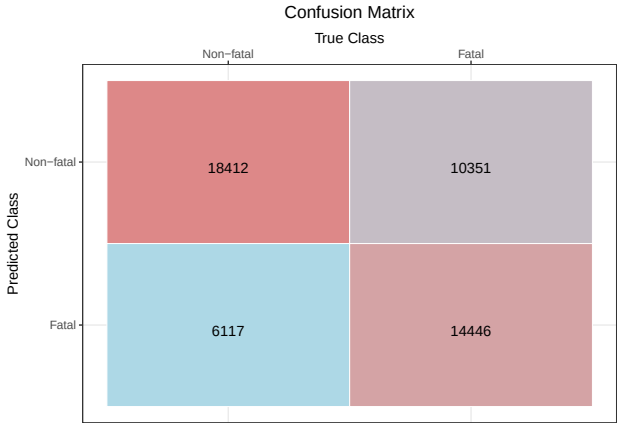


Figure 32: Confusion matrix of logistic regression model

(#fig:confmatrix.lr)

The accuracy of the model with the best threshold is of 0.6661396.

4.2.1.5 K-fold Cross-Validation

To assess the logistic regression model and detect issues such as overfitting or selection bias, we use K-fold cross-validation. This method divides the data into K groups and uses one group as a test set and the others as a training set. We repeat this procedure K times, each time with a different group as the test set. The final score of the model is the average of the K test scores. We set K to 10 for K-fold cross validation.

```r
set.seed(123)

#specify the cross-validation method
cv <- trainControl(method = "cv", number = 10, savePredictions=TRUE)

#fit a regression model and use k-fold CV to evaluate performance
model_cv <- train(C_SEV_FACTOR~. - C_RSUR - C_CONF_AGR -C_TRAF_AGR, data = cars_u,
                  method = "glm", family = binomial, trControl = cv)

#view summary of k-fold CV
print(model_cv)
```

```
## Generalized Linear Model
##
## 49326 samples
##    13 predictor
##     2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 44393, 44393, 44393, 44393, 44393, 44394, ...
## Resampling results:
##
##   Accuracy   Kappa
##   0.6643352  0.3290648
```

```r
pred <- model_cv$pred
pred$equal <- ifelse(pred$pred == pred$obs, 1,0)
eachfold <- pred %>%
  group_by(Resample) %>%
  summarise_at(vars(equal),
               list(Accuracy = mean))
```

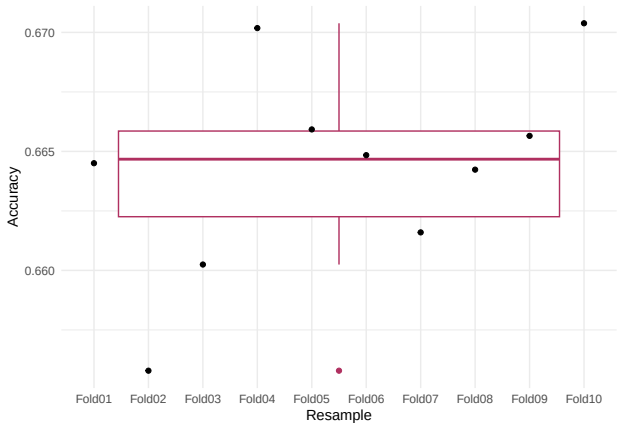


Figure 33: Boxplot K-fold

Our model performs well on the training set with k fold cross validation. It has an average accuracy of about 67% and a low variability as shown in Figure 33. This means that our model is not overfitting the data.

4.2.1.6 Dimensionality Reduction

One of the main steps for applying logistic regression is to reduce the number of variables, because not all them are important, and is possible that they can cause overfitting. We are going to apply variable selection, but for our model, as all the variables are significant and categorical, the model with all the parameters are preferred over the ones with less parameters.

1. Stepwise Selection: We apply the function step in R with the AIC criteria using forward and backward stepwise selection. In the case of Forward selection, it starts with the model with only the intercept and considers one variable addition of the model and adds the one with the lowest AIC. The backward stepwise selection begins with the saturated model, and eliminate variables using the same criteria explained before.

```
#Feature selection AIC
nothing <- glm(  C_SEV_FACTOR~1, data = cars_u, family = binomial)
backwards = step(model , criteria = "AIC", trace=0)
forwards = step(nothing, criteria = "AIC", trace = 0,
                scope=list(lower=formula(nothing),upper=formula(model), direction="forward"))
```

The final model with Backwards elimination using AIC criteria is:

```
## C_SEV_FACTOR ~ (C_MNTH + C_WDAY + C_HOUR_AGR + C_VEHS_AGR + C_CONF_AGR +
##     C_TRAF_AGR + C_RSUR + C_WTHR + C_RALN + C_RCFG_AGR + P_CHILD +
##     P_ELD + C_SAFE) - C_RSUR - C_CONF_AGR - C_TRAF_AGR
```

The final model with Forward elimination using AIC criteria is:

```
## C_SEV_FACTOR ~ C_SAFE + C_RCFG_AGR + C_HOUR_AGR + P_ELD + C_RALN +
##     C_MNTH + P_CHILD + C_WTHR + C_WDAY + C_VEHS_AGR
```

As we can see both, backward and forward stepwise selection select the model with all the variables.

So, in order to control the possible overfitting, we apply shrinkage regression to reduce the effect of each parameter in the final model.

4.2.1.7 Shrinkage Regression

One of the challenges of logistic regression is to avoid overfitting the data, which can lead to poor generalization and high variance. A common technique to address this issue is regularization, which adds a penalty term to the cost function that depends on the magnitude of the coefficients. In this report, we will use L1 regularization, also known as Lasso, which penalizes the absolute value of the coefficients. This has the effect of shrinking some coefficients to zero, effectively performing feature selection and reducing the complexity of the model.

```
#Define train and test set
trainIndex <- createDataPartition(cars_u$C_SEV_FACTOR, p = .7,
                                  list = FALSE,
                                  times = 1)
cars.train <- cars_u[ trainIndex,]
cars.val  <- cars_u[-trainIndex,]

y.train <-  cars.train[,1]
```

```r
# Searching for the optimal value of lambda
set.seed(123)
cars.train.matrix <- model.matrix( ~., cars.train[, -1])
cv.lasso <- cv.glmnet(cars.train.matrix, y.train, alpha = 1, family = "binomial")
```

```r
# Optimal value of lambda that minimizes the cross-validation error
lambda.min <- cv.lasso$lambda.min
```

We obtained a cross-validation error minimum at lambda 7.8144165×10^{-4}, which improves the accuracy by about 2 percentage points. It is possible to get better results by changing the logistic regression threshold, but we used the same one as in the non-regularized case for a fair comparison.

```r
# Model with best lambda value
lasso.model <- glmnet(cars.train.matrix , y.train, alpha = 1, family = "binomial",
                      lambda = cv.lasso$lambda.min)
```

```r
# Accuracy of regularized model with best lambda value
cars_u.matrix <- model.matrix( ~., cars_u[, -1])

model_prob <- lasso.model %>% predict(newx = cars_u.matrix)

model_pred <- 1*(model_prob > roc.coords$threshold) + 0
accuracy_lasso <- sum(model_pred == cars_u$C_SEV_FACTOR) /nrow(cars_u)
```

The accuracy with a L1 regularization with the best lambda is 0.6875887.

4.2.1.8 Stability of the parameters with undersampling

Due to the great difference between the majority and minority class, we checked the stability of parameters of the models for different undersampled data. To do this, we undersampled the data set using 10 different seeds, fit each model using glm, and calculate the mean and standard deviation using each of the ten values for every level. The obtained values are displayed in the table, which does not show any level that varies too much from its mean value (in the case of the levels that have higher mean values), thus ensuring representativeness of the sample of the majority class used in the undersampling process.

```r
options(width = 100)
set.seed(123)
list_seed = c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
mod_summaries <- list()
for(i in 1:length(list_seed)) {                  # Head of for-loop
  cars_u <- ovun.sample(C_SEV_FACTOR ~ ., data = accidents,
                        seed = list_seed[i], method = "under")$data

  model_loop <- glm(C_SEV_FACTOR~. - C_RSUR - C_CONF_AGR -C_TRAF_AGR,
                    data = cars_u, family = binomial)
  assign(gsub(" ", "", paste('model_', toString(i), collapse = ""),
              fixed = TRUE), model_loop)
  }

compare_glm <- compareGLM(model_1, model_2, model_3, model_4, model_5,
           model_6, model_7, model_8, model_9, model_10)

compare_glm$Fit.criteria
```

```
##    Rank Df.res   AIC  AICc   BIC McFadden Cox.and.Snell Nagelkerke p.value
## 1    40  49580 60210 60210 60570   0.1237        0.1572     0.2099       0
## 2    40  49580 60150 60150 60510   0.1246        0.1582     0.2112       0
## 3    40  49420 60010 60010 60370   0.1266        0.1610     0.2146       0
## 4    40  49460 59990 59990 60350   0.1269        0.1613     0.2150       0
## 5    40  49290 59980 59980 60340   0.1271        0.1620     0.2157       0
## 6    40  49610 60320 60320 60680   0.1221        0.1552     0.2073       0
## 7    40  49540 59960 59960 60320   0.1273        0.1615     0.2155       0
## 8    40  49600 60010 60010 60370   0.1266        0.1606     0.2144       0
## 9    40  49680 60260 60260 60620   0.1229        0.1560     0.2085       0
## 10   40  49460 59960 59960 60320   0.1273        0.1618     0.2157       0
```

```r
# Cofficients variability:
coef_mean = (model_1$coefficients[2:length(model_1$coefficients)] +
               model_2$coefficients[2:length(model_2$coefficients)]
             + model_3$coefficients[2:length(model_3$coefficients)]
             + model_4$coefficients[2:length(model_4$coefficients)]
             + model_5$coefficients[2:length(model_5$coefficients)]
             + model_6$coefficients[2:length(model_6$coefficients)]
             + model_7$coefficients[2:length(model_7$coefficients)]
             + model_8$coefficients[2:length(model_8$coefficients)]
             + model_9$coefficients[2:length(model_9$coefficients)]
             + model_10$coefficients[2:length(model_10$coefficients)]) / 10

aux = list(model_1$coefficients[2:length(model_1$coefficients)],
           model_2$coefficients[2:length(model_2$coefficients)],
           model_3$coefficients[2:length(model_3$coefficients)],
           model_4$coefficients[2:length(model_4$coefficients)],
           model_5$coefficients[2:length(model_5$coefficients)],
           model_6$coefficients[2:length(model_6$coefficients)],
           model_7$coefficients[2:length(model_7$coefficients)],
           model_8$coefficients[2:length(model_8$coefficients)],
           model_9$coefficients[2:length(model_9$coefficients)],
           model_10$coefficients[2:length(model_10$coefficients)])

coef_sd <- c()
for (j in 1:length(aux[[c(1)]])) {
  list_aux = c()
  for (i in 1:10){
    list_aux <- append(list_aux, aux[[c(i, j)]])
  }
  coef_sd <- append(coef_sd, sd(list_aux))
}
cbind(coef_mean, coef_sd)
```

```
##                              coef_mean    coef_sd
## C_MNTHfeb                   0.01456178 0.03550002
## C_MNTHmar                   0.08454909 0.03608395
## C_MNTHapr                   0.21182099 0.02307641
## C_MNTHmay                   0.33358703 0.03147296
## C_MNTHjun                   0.34846328 0.03063728
## C_MNTHjul                   0.49792401 0.03378087
## C_MNTHaug                   0.48369190 0.03333728
## C_MNTHsep                   0.40551947 0.03229313
```

```
## C_MNTHoct                      0.35285223 0.02420346
## C_MNTHnov                      0.22460281 0.02777406
## C_MNTHdic                      0.16174686 0.02699064
## C_WDAYtue                     -0.02002679 0.02046616
## C_WDAYwed                     -0.04656513 0.03196041
## C_WDAYthu                      0.02757900 0.02369369
## C_WDAYfri                      0.05405570 0.02728993
## C_WDAYsat                      0.14291295 0.02183023
## C_WDAYsun                      0.17166163 0.02678747
## C_HOUR_AGREarly Morning        0.47416257 0.02458765
## C_HOUR_AGREvening              0.26023276 0.02409823
## C_HOUR_AGRMorning             -0.01382929 0.02224548
## C_VEHS_AGR2                   -0.04520085 0.01101226
## C_VEHS_AGR+3                  -0.15988112 0.02380178
## C_WTHRovercast                -0.12893374 0.02409519
## C_WTHRraining                 -0.35524801 0.02570658
## C_WTHRsnowing                 -0.13347097 0.02200319
## C_WTHRfreezing rain           -0.03538880 0.10768048
## C_WTHRvisibility limitation    0.34902876 0.04846135
## C_WTHRstring wind              0.27064156 0.10254822
## C_WTHRother                    0.10123032 0.14337845
## C_RALN2                        0.16134667 0.02453855
## C_RALN3                        0.35514021 0.03183960
## C_RALN4                        0.45848616 0.04286980
## C_RALN5                        0.26350717 0.08304520
## C_RALN6                        0.43129228 0.07139993
## C_RCFG_AGRat an intersection  -0.77618699 0.01783414
## C_RCFG_AGRother               -0.69580873 0.03610195
## P_CHILD1                      -0.35726537 0.01608746
## P_ELD1                         0.49071552 0.02325321
## C_SAFE1                        1.59167339 0.01409738
```

4.3 KNN

We applied the KNN algorithm to our data and experimented with different values of K to find the optimal one.

```
#Define train and test set
trainIndex <- createDataPartition(cars_u$C_SEV_FACTOR, p = .7,
                                  list = FALSE,
                                  times = 1)
cars.train <- cars_u[ trainIndex,]
cars.val  <- cars_u[-trainIndex,]

y.train <-  cars.train[,1]

cars.knn.train <- cars.train
cars.knn.val <- cars.val
columns <- names(cars.knn.train)

for (var in columns) {
  cars.knn.train[[var]] <- as.integer(cars.knn.train[[var]])
  cars.knn.val[[var]] <- as.integer(cars.knn.val[[var]])
```

```
}

y.train <-  cars.knn.train[,1]
cars.knn.train <- cars.knn.train[,-1]
y.val <- cars.knn.val[,1]
cars.knn.val <- cars.knn.val[,-1]
```

```
k.values <- 1:40
accuracy <- vector()

set.seed(123)
for (k in k.values) {
  knn.pred <- knn(cars.knn.train , cars.knn.val, y.train, k=k)
  acc <- mean(knn.pred==y.val)
  accuracy <- c(accuracy, acc)
}
```

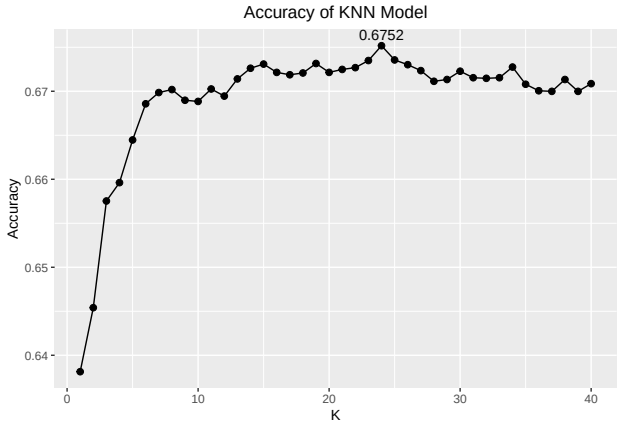


Figure 34: Accuracy of KNN model in the validation set

```
set.seed(123)
knn.pred <- knn(cars.knn.train , cars.knn.val, y.train, k=top.k)
accuracy_knn_val <- mean(knn.pred==y.val)
```

The mean accuracy applying KNN in the validation set with K between 1 and 40 is 0.6747037 with a top accuracy of 0.6751751. The best K turned out to be 24, which gave us an accuracy of 0.6751751.

4.3.1 Accuracy on full dataset with undersampling

```
cars_u_knn <- rbind(cars.knn.train, cars.knn.val)
cars_u_knn.pred <- knn(cars.knn.train , cars_u_knn, y.train, k=top.k)
cars_u.y <- c(y.train, y.val)
cm <- confusionMatrix(data=factor(cars_u_knn.pred),reference=factor(cars_u.y ))
accuracy_knn <- cm[["overall"]][["Accuracy"]]
knn_specificity <- cm[["byClass"]][["Specificity"]]
knn_sensitivity <- cm[["byClass"]][["Sensitivity"]]
```

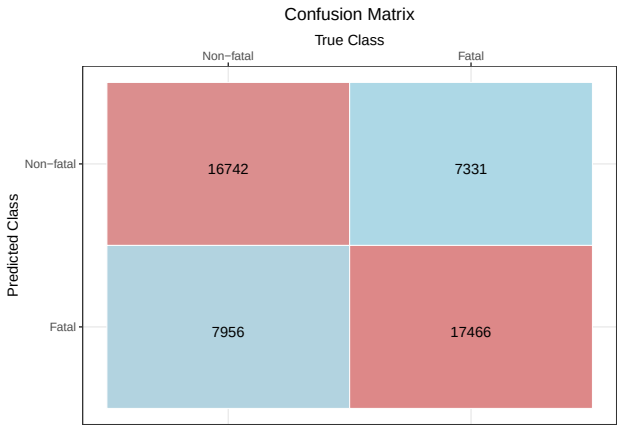


Figure 35: Confusion matrix of KNN model

Finally we applied the best K obtained in the validation set to the complete data set with undersampling. We obtained an accuracy of 0.6911405 . This is a slight improvement over the non regularized logistic regression model. We also showed the confusion matrix (Figure 35). The specifity and sensitivity of our model were 0.7043594 and 0.6778687, respectively.

5 Results

In this section we are going to interpret the results of the Logistic regression and KNN model and compare both models. The final Logistic Regression model after eliminating correlated variables is as follows:

```
##
## Call:
## glm(formula = C_SEV_FACTOR ~ . - C_RSUR - C_CONF_AGR - C_TRAF_AGR,
##     family = binomial, data = cars_u)
##
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                -0.3489069  0.0499380  -6.987 2.81e-12 ***
## C_MNTHfeb                  -0.0167944  0.0498603  -0.337  0.73625
## C_MNTHmar                   0.1007544  0.0506006   1.991  0.04646 *
## C_MNTHapr                   0.2765150  0.0518147   5.337 9.47e-08 ***
## C_MNTHmay                   0.2588292  0.0494840   5.231 1.69e-07 ***
## C_MNTHjun                   0.2906687  0.0484360   6.001 1.96e-09 ***
## C_MNTHjul                   0.5157635  0.0481703  10.707  < 2e-16 ***
## C_MNTHaug                   0.4648576  0.0477709   9.731  < 2e-16 ***
## C_MNTHsep                   0.3882565  0.0483797   8.025 1.01e-15 ***
## C_MNTHoct                   0.3569007  0.0478957   7.452 9.22e-14 ***
## C_MNTHnov                   0.2245486  0.0473249   4.745 2.09e-06 ***
## C_MNTHdic                   0.1456758  0.0462680   3.149  0.00164 **
## C_WDAYtue                  -0.0513653  0.0382544  -1.343  0.17936
## C_WDAYwed                  -0.0955099  0.0381860  -2.501  0.01238 *
## C_WDAYthu                   0.0005893  0.0375029   0.016  0.98746
## C_WDAYfri                   0.0185967  0.0365487   0.509  0.61088
## C_WDAYsat                   0.1034255  0.0372750   2.775  0.00553 **
## C_WDAYsun                   0.1263924  0.0387329   3.263  0.00110 **
```

```
## C_HOUR_AGREarly Morning       0.4879985  0.0349231  13.974  < 2e-16 ***
## C_HOUR_AGREvening             0.2505431  0.0256947   9.751  < 2e-16 ***
## C_HOUR_AGRMorning            -0.0403698  0.0254872  -1.584  0.11321
## C_VEHS_AGR2                  -0.0454548  0.0229400  -1.981  0.04754 *
## C_VEHS_AGR+3                 -0.1442971  0.0347815  -4.149 3.34e-05 ***
## C_WTHRovercast               -0.0928928  0.0308044  -3.016  0.00256 **
## C_WTHRraining                -0.3439641  0.0336176 -10.232  < 2e-16 ***
## C_WTHRsnowing                -0.1610277  0.0400424  -4.021 5.78e-05 ***
## C_WTHRfreezing rain          -0.0383710  0.1141616  -0.336  0.73679
## C_WTHRvisibility limitation   0.4178952  0.0663857   6.295 3.07e-10 ***
## C_WTHRstring wind             0.4016234  0.1441412   2.786  0.00533 **
## C_WTHRother                   0.0527745  0.1921214   0.275  0.78355
## C_RALN2                       0.1330070  0.0302082   4.403 1.07e-05 ***
## C_RALN3                       0.3784669  0.0317614  11.916  < 2e-16 ***
## C_RALN4                       0.4826748  0.0394389  12.239  < 2e-16 ***
## C_RALN5                       0.1450164  0.0977812   1.483  0.13806
## C_RALN6                       0.3253859  0.1116216   2.915  0.00356 **
## C_RCFG_AGRat an intersection -0.7767099  0.0222033 -34.982  < 2e-16 ***
## C_RCFG_AGRother              -0.6700910  0.0398280 -16.825  < 2e-16 ***
## P_CHILD1                     -0.3585545  0.0237212 -15.115  < 2e-16 ***
## P_ELD1                        0.4707591  0.0259067  18.171  < 2e-16 ***
## C_SAFE1                       1.6135108  0.0285243  56.566  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 68379  on 49325  degrees of freedom
## Residual deviance: 59638  on 49286  degrees of freedom
## AIC: 59718
##
## Number of Fisher Scoring iterations: 4
```

As we want to interpret the parameters of the model as odds ratios we calculate the exponential of each parameter. If the the result is greater than 1, this means that the parameter increases the odds of the occurrence of a fatality in the accident. If the result is less than 1, the parameter decreases the odds of a casualty in the accident.

```
## C_RCFG_AGRat an intersection             C_RCFG_AGRother                   P_CHILD1
##                    0.4599167                   0.5116620                  0.6986856
##                  (Intercept)               C_WTHRraining               C_WTHRsnowing
##                    0.7054588                   0.7089544                  0.8512685
##                C_VEHS_AGR+3                   C_WDAYwed              C_WTHRovercast
##                    0.8656306                   0.9089094                  0.9112912
##                    C_WDAYtue                C_VEHS_AGR2           C_HOUR_AGRMorning
##                    0.9499316                   0.9555628                  0.9604342
##          C_WTHRfreezing rain                   C_MNTHfeb                  C_WDAYthu
##                    0.9623558                   0.9833459                  1.0005895
##                    C_WDAYfri                C_WTHRother                  C_MNTHmar
##                    1.0187707                   1.0541919                  1.1060050
##                    C_WDAYsat                   C_WDAYsun                    C_RALN2
##                    1.1089632                   1.1347273                  1.1422580
##                      C_RALN5                   C_MNTHdic                  C_MNTHnov
##                    1.1560585                   1.1568211                  1.2517575
```

```
##            C_HOUR_AGREvening                    C_MNTHmay                    C_MNTHapr
##                    1.2847230                    1.2954126                    1.3185267
##                    C_MNTHjun                      C_RALN6                    C_MNTHoct
##                    1.3373214                    1.3845649                    1.4288940
##                      C_RALN3                    C_MNTHsep          C_WTHRstring wind
##                    1.4600444                    1.4744079                    1.4942485
##  C_WTHRvisibility limitation                    C_MNTHaug                       P_ELD1
##                    1.5187615                    1.5917875                    1.6012092
##                      C_RALN4      C_HOUR_AGREarly Morning                    C_MNTHjul
##                    1.6204028                    1.6290525                    1.6749169
##                      C_SAFE1
##                    5.0204062
```

With these results we focus on answering the Data Science questions:

1. Which time and day of the week is more probable that an accident with fatalities occurs?

The statistical analysis shows that the time of the day, the day of the week and the month of the year are all associated with the likelihood of fatal accidents. Specifically, compared to the Afternoon, Early morning and Evening have higher odds of fatal accidents by 1.63 and 1.28 times respectively. Similarly, compared to Mondays, Saturdays and Sundays have higher odds by 1.11 and 1.13 times respectively. Moreover, compared to January, July, August and September have the highest odds increases by 1.67, 1.59 and 1.47 times respectively. Therefore, a prevention campaign should target the summer months, the weekends and the early morning hours.

2. What kind of weather condition is most prevalent during car accidents with fatalities?

The C_WTHR variable has statistically significant effects on the likelihood of a fatal outcome in a crash. Compared to clear and sunny days, the odds of at least one fatal in an accident are 1.52 and 1.49 times higher when there is visibility limitation or strong wind, respectively. Conversely, the odds of a fatal crash are lower when the weather is overcast, rainy, or snowy, by factors of 0.91, 0.71, and 0.85 respectively. A useful campaign could educate drivers on how to cope with low visibility or high wind situations.

3. Is there a correlation between the road configuration and the severity of the accident?

Compared to non-intersection accidents, the odds of having at least one fatality are 0.46 times lower for intersection accidents, and 0.51 times lower for other road configurations such as bridges, tunnels or traffic circles. These results suggest that more traffic signals should be installed in non-intersection roads.

4. Is there a correlation between the road alignment and the severity of the accident?

The risk of a having at least one fatal is higher on curved roads than on straight flat ones. Curved leveled roads have a 1.46 times higher risk, while curved roads with gradient have a 1.62 times higher risk. Based on these findings and the previous ones on road configuration, we suggest launching a campaign to raise awareness of the dangers of these roads and to install more traffic signs on them.

5. How does the safety device affect the likelihood of casualties in traffic accidents?

The variable C_SAFE indicates how likely it is for someone to die in a crash when no safety device is used. The result is 5.02, which means the risk of death is more than five times higher without any safety device. This is a very alarming number, and it suggests that we need to promote the use of safety devices among drivers and passengers, and also make sure that companies follow the safety standards, as this factor has a great influence on the survival rate of an accident.

6. When there is elderly and underage people involved in a car accident, is it more probable to have fatalities reported?

The analysis shows that the presence of underage people reduces the odds of a fatal outcome by a factor of 0.70. However, this finding is puzzling, as there is no obvious reason why underage people would be safer in transportation. Therefore, the age criterion for defining underage people may need to be revised. Conversely, the presence of elderly people increases the odds of a fatal outcome by a factor of 1.60. This suggests that more attention and care should be given to elderly people when they use any mode of transportation.

Regarding how well the model fit the data, with a threshold of 0.5166719 we obtained an accuracy of 0.6661396. But the sensitivity of the model is not very high, which can be related to the limitation of the logistic regression to capture non-linear relation between variables. Regarding dimensionality reduction, we do not remove variables because all of them are categorical, but we applied shrinkage regression with an L1 regularization. With this method, the accuracy increases in 2 percentage points. On the other hand, we applied KNN that with a K = 24, we obtained an accuracy of 0.6911405, similar to Logistic regression with L1 regularization. Also, it is possible to see an improvement in the sensitivity compare to the logistic regression. The main drawback of this model is that we can't interpret the parameters.

6 Conclusions

In this report, we have the aim to identify the factors that influence the likelihood and severity of car accidents. For this we applied a Logistic Regression and also a KNN. As the data set was severely imbalanced, we applied undersampling. By appliying Logistic regression, we conclude that the main variable that contributes to a casualty in a car accident is the use of Safety Device. In this line, the promotion of safety devices, as seat belts, child restrains, helmet, among other, is very important to decrease the number of casualties in car accidents. On the other hand, casualties in accidents mainly occur on low visibility or high wind conditions. The results also points at the weekends, early mornings and summer months as the ones with more odds of at least one fatality given that an accident has happened.

The accuracy of KNN (0.6911405) was better than logistic regression (0.6661396), and particularly it presents a better sensitivity. But the main advantage of Logistic regression is that we can deliver an interpretation of the variables, allowing us to identify the main factors behind car accidents with casualties and propose public policies to decrease them.