B20102088-MuhammadHabibKhan-FinalLab

Muhammad Habib Khan

1/22/2022

Table of Contents

[Selecting a Dataset 2](#_Toc94203994)

[Data Set At A Glance 3](#_Toc94203995)

[Basic Plots 9](#_Toc94203996)

[Bar Plot: 10](#_Toc94203997)

[3D Pie Chart: 11](#_Toc94203998)

[Histogram: 12](#_Toc94203999)

[Scatter Plot: 12](#_Toc94204000)

[Box Plot: 13](#_Toc94204001)

[Correlation 14](#_Toc94204002)

[Correlation between Law & Casualties: 14](#_Toc94204003)

[Correlation of the Entire Dataset: 15](#_Toc94204004)

[Confidence Interval 17](#_Toc94204005)

[Based on Z-Distribution 17](#_Toc94204006)

[Based on t-Distribution 20](#_Toc94204007)

[Hypothesis Testing 23](#_Toc94204008)

[One Sample t-test: 24](#_Toc94204009)

[Two sample t-test for two independent samples 25](#_Toc94204010)

[Chi-Square Test 28](#_Toc94204011)

[Chi-Square Test of Independence: 28](#_Toc94204012)

[ANOVA 29](#_Toc94204013)

[One-Way ANOVA 30](#_Toc94204014)

[Two-Way ANOVA: 34](#_Toc94204015)

[Regression 40](#_Toc94204016)

[Linear Regression Model: 40](#_Toc94204017)

[Multiple Regression Model: 41](#_Toc94204018)

# Converting Time Series into Data Frame Structure  
  
data(Seatbelts)  
Seatbelts <- data.frame(Year=floor(time(Seatbelts)),  
Month=factor(cycle(Seatbelts),  
labels=month.abb), Seatbelts)  
  
knitr::opts\_chunk$set(echo = TRUE)  
  
library(knitr)

# Selecting a Dataset

Data Set Name: *Seatbelts*

Data Set From built-in R Data sets in package ‘datasets’

**Abstract**: Road Casualties in Great Britain 1969-84

**Data Set Information:**

The ‘Seatbelts’ data set in R is a multiple time-series data set that was commissioned by the Department of Transport in 1984 to measure differences in deaths before and after front seat belt legislation was introduced on 31st January 1983. It provides monthly total numerical data on a number of incidents including those related to death and injury in Road Traffic Accidents (RTA’s). The data set starts in January 1969 and observations run until December 1984.

**Attributes:**

DriversKilled: Car drivers killed.

drivers: Same as *UKDriverDeaths* deaths count.

*UKDriverDeaths is a time series giving the monthly totals of car drivers in Great Britain killed or seriously injured Jan 1969 to Dec 1984. Compulsory wearing of seat belts was introduced on 31 Jan 1983.* *Seat belts is more information on the same problem.*

front: Front-seat passengers killed or seriously injured.

rear: Rear-seat passengers killed or seriously injured.

kms: Distance driven.

PetrolPrice: Petrol price.

VanKilled: Number of van (‘light goods vehicle’) drivers killed.

law: 0/1: Was the law in effect that month?

# Data Set At A Glance

# Summary of the Entire Data  
summary(Seatbelts)

## Year Month DriversKilled drivers front   
## Min. :1969 Jan :16 Min. : 60.0 Min. :1057 Min. : 426.0   
## 1st Qu.:1973 Feb :16 1st Qu.:104.8 1st Qu.:1462 1st Qu.: 715.5   
## Median :1976 Mar :16 Median :118.5 Median :1631 Median : 828.5   
## Mean :1976 Apr :16 Mean :122.8 Mean :1670 Mean : 837.2   
## 3rd Qu.:1980 May :16 3rd Qu.:138.0 3rd Qu.:1851 3rd Qu.: 950.8   
## Max. :1984 Jun :16 Max. :198.0 Max. :2654 Max. :1299.0   
## (Other):96   
## rear kms PetrolPrice VanKilled   
## Min. :224.0 Min. : 7685 Min. :0.08118 Min. : 2.000   
## 1st Qu.:344.8 1st Qu.:12685 1st Qu.:0.09258 1st Qu.: 6.000   
## Median :401.5 Median :14987 Median :0.10448 Median : 8.000   
## Mean :401.2 Mean :14994 Mean :0.10362 Mean : 9.057   
## 3rd Qu.:456.2 3rd Qu.:17202 3rd Qu.:0.11406 3rd Qu.:12.000   
## Max. :646.0 Max. :21626 Max. :0.13303 Max. :17.000   
##   
## law   
## Min. :0.0000   
## 1st Qu.:0.0000   
## Median :0.0000   
## Mean :0.1198   
## 3rd Qu.:0.0000   
## Max. :1.0000   
##

# Gist of the Data including only few of the rows  
head(Seatbelts, 5)

## Year Month DriversKilled drivers front rear kms PetrolPrice VanKilled law  
## 1 1969 Jan 107 1687 867 269 9059 0.1029718 12 0  
## 2 1969 Feb 97 1508 825 265 7685 0.1023630 6 0  
## 3 1969 Mar 102 1507 806 319 9963 0.1020625 12 0  
## 4 1969 Apr 87 1385 814 407 10955 0.1008733 8 0  
## 5 1969 May 119 1632 991 454 11823 0.1010197 10 0

# Another useful command to look at the summarized data and skim it  
  
library(skimr)  
skim(Seatbelts)

Data summary

|  |  |
| --- | --- |
| Name | Seatbelts |
| Number of rows | 192 |
| Number of columns | 10 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Column type frequency: |  |
| factor | 1 |
| numeric | 8 |
| ts | 1 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Group variables | None |

**Variable type: factor**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| skim\_variable | n\_missing | complete\_rate | ordered | n\_unique | top\_counts |
| Month | 0 | 1 | FALSE | 12 | Jan: 16, Feb: 16, Mar: 16, Apr: 16 |

**Variable type: numeric**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| skim\_variable | n\_missing | complete\_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
| DriversKilled | 0 | 1 | 122.80 | 25.38 | 60.00 | 104.75 | 118.5 | 138.00 | 198.00 | ▁▇▆▃▁ |
| drivers | 0 | 1 | 1670.31 | 289.61 | 1057.00 | 1461.75 | 1631.0 | 1850.75 | 2654.00 | ▂▇▃▂▁ |
| front | 0 | 1 | 837.22 | 175.10 | 426.00 | 715.50 | 828.5 | 950.75 | 1299.00 | ▂▆▇▅▁ |
| rear | 0 | 1 | 401.21 | 83.10 | 224.00 | 344.75 | 401.5 | 456.25 | 646.00 | ▃▇▇▂▁ |
| kms | 0 | 1 | 14993.60 | 2938.05 | 7685.00 | 12685.00 | 14987.0 | 17202.50 | 21626.00 | ▂▆▇▇▂ |
| PetrolPrice | 0 | 1 | 0.10 | 0.01 | 0.08 | 0.09 | 0.1 | 0.11 | 0.13 | ▆▅▇▇▁ |
| VanKilled | 0 | 1 | 9.06 | 3.64 | 2.00 | 6.00 | 8.0 | 12.00 | 17.00 | ▅▇▅▆▂ |
| law | 0 | 1 | 0.12 | 0.33 | 0.00 | 0.00 | 0.0 | 0.00 | 1.00 | ▇▁▁▁▁ |

**Variable type: ts**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| skim\_variable | n\_missing | complete\_rate | start | end | frequency | deltat | mean | sd | min | max | median | line\_graph |
| Year | 0 | 1 | 1969 | 1984 | 12 | 0.08 | 1976.5 | 4.62 | 1969 | 1984 | 1976.5 | ⣀⣀⠤⠤⠒⠒⠉⠉ |

# Another command to displays the type and a preview of all columns as a row   
# so that it's very easy to take in and easily preview data type and sample data.  
  
library(dplyr)

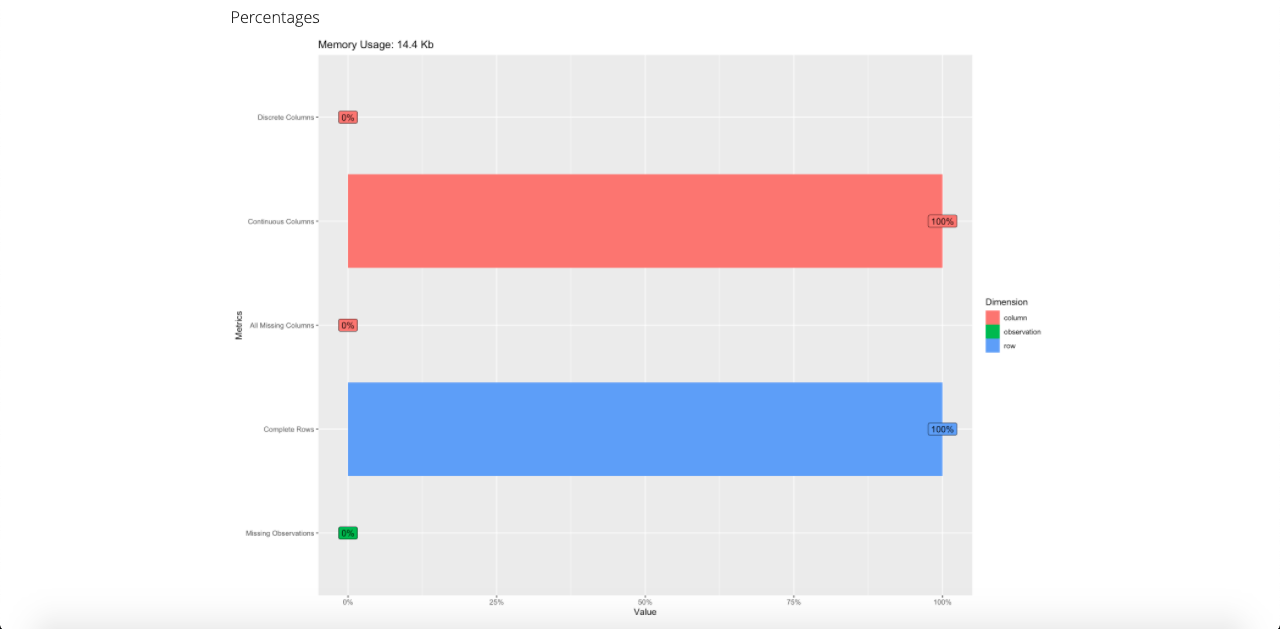
##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

glimpse(Seatbelts)

## Rows: 192  
## Columns: 10  
## $ Year <dbl> 1969, 1969, 1969, 1969, 1969, 1969, 1969, 1969, 1969, 19…  
## $ Month <fct> Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, D…  
## $ DriversKilled <dbl> 107, 97, 102, 87, 119, 106, 110, 106, 107, 134, 147, 180…  
## $ drivers <dbl> 1687, 1508, 1507, 1385, 1632, 1511, 1559, 1630, 1579, 16…  
## $ front <dbl> 867, 825, 806, 814, 991, 945, 1004, 1091, 958, 850, 1109…  
## $ rear <dbl> 269, 265, 319, 407, 454, 427, 522, 536, 405, 437, 434, 4…  
## $ kms <dbl> 9059, 7685, 9963, 10955, 11823, 12391, 13460, 14055, 121…  
## $ PetrolPrice <dbl> 0.10297181, 0.10236300, 0.10206249, 0.10087330, 0.101019…  
## $ VanKilled <dbl> 12, 6, 12, 8, 10, 13, 11, 6, 10, 16, 13, 14, 14, 6, 8, 1…  
## $ law <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,…

# A comprehensive look at the entire data set produced in html format   
# It includes a ton of information including the basic statistics, structure,   
# missing data, distribution visualizations, correlation matrix and principal   
# component analysis. The screenshots of the webpage are attached below and the  
# html file is also attached.   
  
library(DataExplorer)  
DataExplorer::create\_report(datasets::Seatbelts)

# 

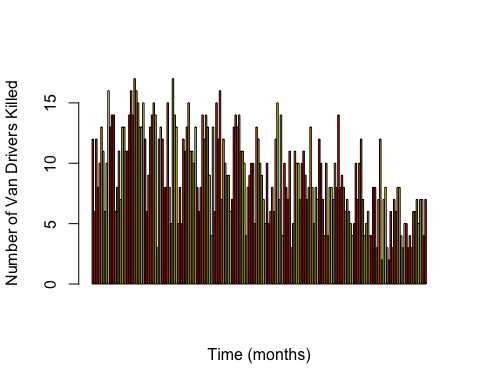
# Basic Plots

Some basic plots, charts and graphs of the data set and its classes are shown below which includes:

1. Bar Plot
2. 3D Pie Chart
3. Histogram
4. Scatter Plot
5. Box Plot

## Bar Plot:

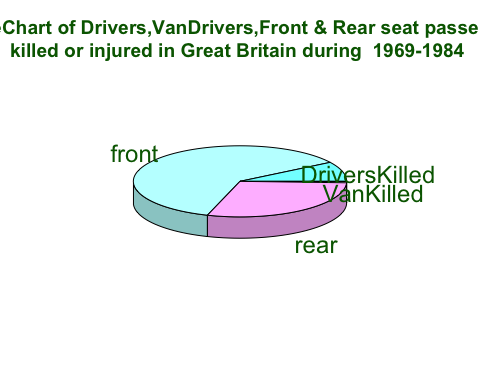
attach(Seatbelts)  
  
# Bar-plot of Number of Van Drivers killed through out the years  
  
barplot(Seatbelts$VanKilled, xlab='Time (months)', ylab='Number of Van Drivers Killed', col=heat.colors(10))



The bar plot above shows the number of Van Drivers killed during a car accident in Britain in the described time period. While the numbers were small to begin with, we can see the entire pattern going down on a negative slope as they month passes, indicating that when the law of seat belts being mandatory was passed, the amount of deaths for van drivers decreased too even though the time was beginning of Industrial Revolution and the cars on the roads increased as the days passed.

## 3D Pie Chart:

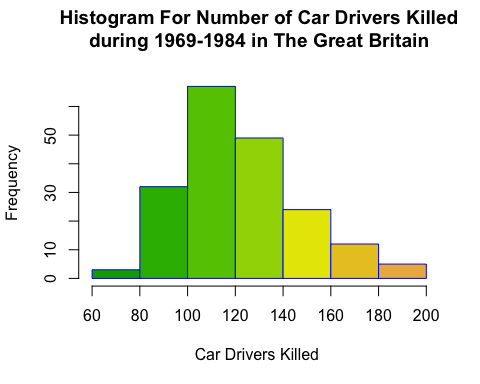
# A 3D pie chart of the mean of the number of car drivers, van drivers, front seat and rear seat passengers killed through out Jan, 1969 - Dec, 1984  
  
library(plotrix)  
x <- c(mean(Seatbelts$DriversKilled), mean(Seatbelts$front), mean(Seatbelts$rear), mean(Seatbelts$VanKilled))  
names(x) <- c("DriversKilled", "front", "rear", "VanKilled")  
pie3D(x, labels = names(x), col = cm.colors(4), main = "3DPieChart of Drivers,VanDrivers,Front & Rear seat passengers\nkilled or injured in Great Britain during 1969-1984 ", labelcol = "darkgreen", col.main = "darkgreen")



Above I have constructed a 3D pie chart using the plotrix library. The pie chart shows the proportion of casualties in car accidents. An average was taken of the classes through out the period to construct this chart. The least are van drivers. This must be because vans are usually less on the roads then sedan/coupe cars. Secondly, vans are often driven slowly then the other cars. The most portion is taken by the front seat passengers, followed by rear seat passengers and after that comes the drivers at 3rd place. Surprisingly, according to the data provided, passengers, front more than rear, are likely to meet the fate of death than the one behind the wheel. One reason may be the fact that most driver seats have airbags while the passengers seat do not.

## Histogram:

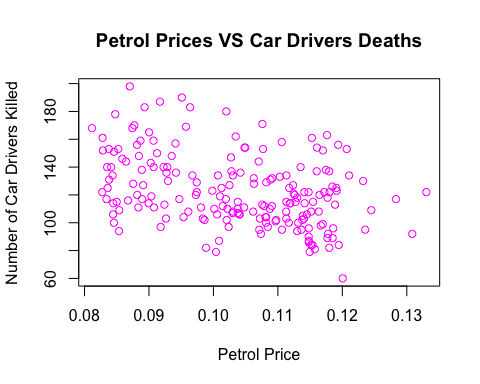
# Histogram for Drivers killed through out the period. Histogram is same as bar plot except that they depict the data in grouped form rather than discrete.   
  
hist(DriversKilled , col=terrain.colors(11) , border = 'blue', main='Histogram For Number of Car Drivers Killed\nduring 1969-1984 in The Great Britain\n ', xlab = 'Car Drivers Killed ' )



The histogram plotted demonstrates what was the usual toll of deaths of car drivers in the given time period. As we can see, at most times the casualties were between 100 and 120, peaking at 200 and hitting the lowest at 60.

## Scatter Plot:

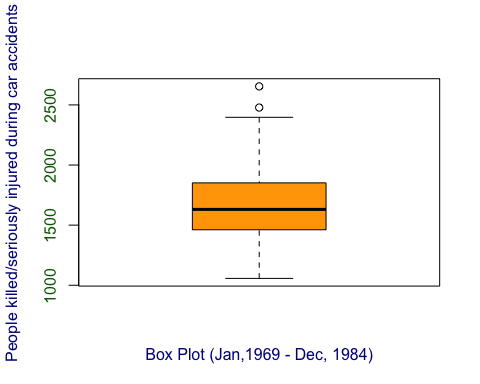
# Scatter Plot Between Increment in Petrol Prices & Number of Drivers Killed through out the period   
  
plot(PetrolPrice, DriversKilled, type='p', xlab = 'Petrol Price', ylab='Number of Car Drivers Killed', main='Petrol Prices VS Car Drivers Deaths', col='magenta')



Above us we can see a scatter plot between increment in petrol prices and number of drivers killed through out the period. If we try to look at the relationship between the two classes, and try to predict a trend based on this plot, we may find that the death of car drivers loosely depend on petrol prices much as they declined slowly at each price point except for at the last one which is the highest price where we see a sudden drop. This may have to do with the introduction of law of seat belts introduced and being implemented by then and also the fact that car safety features were becoming more and more common with airbags and other durability features added to withstand such situations. A better petrol price vs deaths plot will be when we consider only the months when law was not passed which we will look into in the regression section.

## Box Plot:

# Box plot of people killed/seriously injured during car accidents in Great Britain during 1969-1984. The box plots represents five values in the graph i.e., minimum, first quartile, second quartile (median), third quartile, the maximum value of the data vector.  
  
boxplot(drivers, col='orange', xlab = "Box Plot (Jan,1969 - Dec, 1984)", ylab = "People killed/seriously injured during car accidents", col.axis = "darkgreen", col.lab = "dark blue")



This box plot is based on the total amount of people killed or seriously injured during the given period. This is the same as the data set *UKDriverDeaths*. There are five values that this plot tells us. As we can see, the minimum amount of casualties was around 1000, then the first quartile is a little less than 1500, the median (second quartile; black line) is around 1700 while the third quartile is closing to 2000. The maximum here can be seen close to 2500.

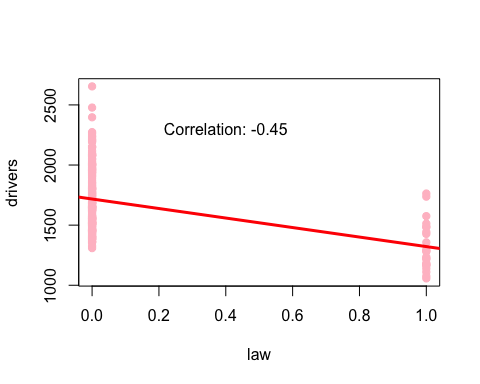
# Correlation

**Definition:** When a variable tends to change from one to another, whether direct or indirect, it is considered correlated. A correlation coefficient is applied to measure a degree of association in variables and is usually called Pearson’s correlation coefficient. The correlation coefficient is measured on a scale with values from +1 through 0 and -1. When both variables increase, the correlation is positive, and if one variable increases, and the other decreases, the correlation is negative.

## Correlation between Law & Casualties:

Lets have a look at the most important correlation of the data set:

plot(law, drivers, pch = 19, col = "pink")  
abline(lm(drivers ~ law), col = "red", lwd = 3)  
text(paste("Correlation:", round(cor(law, drivers), 2)), x = 0.4, y = 2300)

 The above scatter plot depicts the correlation between the total amount of people killed or seriously injured during the given period and if the law was passed then or not (0 = not passed, 1 = passed). It is clearly evident that the casualties decreases by a great deal after the law was passed. The *cor* commands tells us that the correlation coefficient is negative 0.45 (-0.45) which proves that deaths decreases as law is passed. The r-value in magnitude is 0.45 which tells that this is a fairly moderate correlation. If we were to consider other factors too, which includes the increased number of vehicles on the road or the increased population, then this correlation is pretty good even without those factors included. But to counter those extra factors, the safety of the vehicles increased/improved too.

## Correlation of the Entire Dataset:

graphics.off()  
library(corrplot)

## corrplot 0.92 loaded

# converting months into numeric data as cor command does not accept non-numeric data  
Seatbelts$Month <- as.numeric(Month)   
  
# This command prints the correlation between all of the classes of the data set  
cor(Seatbelts)

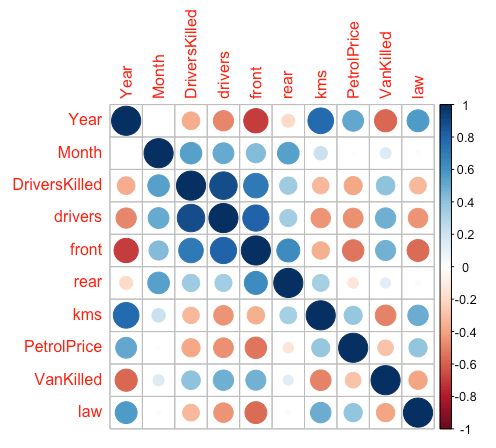
## Year Month DriversKilled drivers front  
## Year 1.0000000 0.00000000 -0.3657291 -0.4809857 -0.6932829  
## Month 0.0000000 1.00000000 0.5458411 0.5084645 0.4305932  
## DriversKilled -0.3657291 0.54584115 1.0000000 0.8888264 0.7067596  
## drivers -0.4809857 0.50846452 0.8888264 1.0000000 0.8084114  
## front -0.6932829 0.43059322 0.7067596 0.8084114 1.0000000  
## rear -0.1969329 0.54996694 0.3533510 0.3436685 0.6202248  
## kms 0.7739559 0.21251863 -0.3211016 -0.4447631 -0.3573823  
## PetrolPrice 0.5109097 0.01030832 -0.3866061 -0.4576675 -0.5392394  
## VanKilled -0.5773166 0.14495270 0.4070412 0.4853995 0.4724207  
## law 0.5619346 0.02555513 -0.3285051 -0.4452269 -0.5624455  
## rear kms PetrolPrice VanKilled law  
## Year -0.19693293 0.7739559 0.51090973 -0.5773166 0.56193462  
## Month 0.54996694 0.2125186 0.01030832 0.1449527 0.02555513  
## DriversKilled 0.35335102 -0.3211016 -0.38660609 0.4070412 -0.32850510  
## drivers 0.34366850 -0.4447631 -0.45766754 0.4853995 -0.44522689  
## front 0.62022476 -0.3573823 -0.53923937 0.4724207 -0.56244554  
## rear 1.00000000 0.3330069 -0.13262721 0.1217581 0.02906753  
## kms 0.33300689 1.0000000 0.38390038 -0.4980356 0.49054938  
## PetrolPrice -0.13262721 0.3839004 1.00000000 -0.2885584 0.39069323  
## VanKilled 0.12175808 -0.4980356 -0.28855841 1.0000000 -0.39494121  
## law 0.02906753 0.4905494 0.39069323 -0.3949412 1.00000000

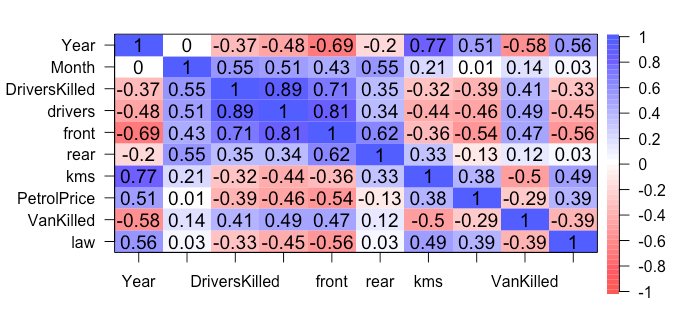
# Converting into matrix to use with command \*corrplot\*   
cor.mat.seatbelts = cor(Seatbelts)  
  
# Plotting the correlation between all of the classes of the data set visually  
corrplot(cor.mat.seatbelts)  
  
# Another way of displaying correlation plot between all the classes, almost the same as corrplot command but this one displays the Pearson's Coefficient written in numbers rather than fading colors  
library(psych)

##   
## Attaching package: 'psych'

## The following object is masked from 'package:plotrix':  
##   
## rescale

corPlot(cor.mat.seatbelts, cex = 1.2)





The cor command prints the correlation between all of the classes of the data set numerically in a tabular form. Using corrplot, we plot all that data into a visually appealing and easy to grasp graph. Some main selective insights from this correlation data:

* The petrol prices and years are moderately positively correlated, meaning that the petrol prices raised as the years passes
* Years passed and law passed are weakly but close to moderately correlated with the total amount of people killed or seriously injured in car accidents. However, they both are moderately correlated with the deaths of front seat passengers. The correlation is negative in all instances showing that years passing resulted in less deaths.
* Years passed and number of Van Drivers killed also has a moderate negative correlation.
* The kms (distance) also has a weak but close to moderate negative correlation with the total amount of people killed or seriously injured in car accidents.

# Confidence Interval

**Definition:** Confidence intervals (CI) are part of inferential statistics that help in making inference about a population from a sample. Based on the confidence level, a true population mean is likely covered by a range of values called confidence interval.

Here, I have constructed two confidence intervals. One on *VanDrivers* and second on *front* based on z-distribution and t-distribution respectively.

## Based on Z-Distribution

library(Rmisc)

## Loading required package: lattice

## Loading required package: plyr

## ------------------------------------------------------------------------------

## You have loaded plyr after dplyr - this is likely to cause problems.  
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:  
## library(plyr); library(dplyr)

## ------------------------------------------------------------------------------

##   
## Attaching package: 'plyr'

## The following objects are masked from 'package:dplyr':  
##   
## arrange, count, desc, failwith, id, mutate, rename, summarise,  
## summarize

# This function computes Confidence intervals based on Z-Distribution  
CI\_z <- function (x, ci = 0.95)  
{  
`%>%` <- magrittr::`%>%`  
standard\_deviation <- sd(x)  
sample\_size <- length(x)  
Margin\_Error <- abs(qnorm((1-ci)/2))\* standard\_deviation/sqrt(sample\_size)  
df\_out <- data.frame( sample\_size=length(x), Mean=mean(x), sd=sd(x),  
Margin\_Error=Margin\_Error,  
'CI lower limit'=(mean(x) - Margin\_Error),  
'CI Upper limit'=(mean(x) + Margin\_Error)) %>%  
tidyr::pivot\_longer(names\_to = "Measurements", values\_to ="values", 1:6 )  
return(df\_out)  
}  
  
  
# Selecting a random sample of 30 from VanKilled (no. of van drivers killed)  
v\_killed = sample(VanKilled, 30)  
  
# Checking normality to apply z-distribution or not  
shapiro.test(v\_killed)

##   
## Shapiro-Wilk normality test  
##   
## data: v\_killed  
## W = 0.95513, p-value = 0.2315

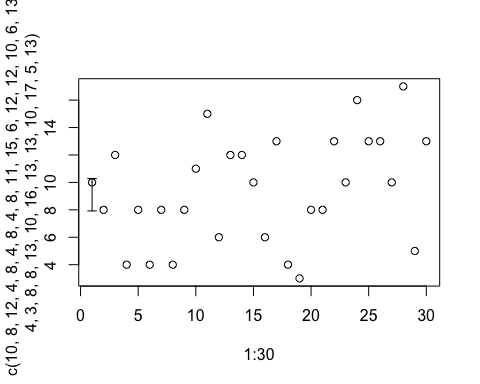
# As p > 0.05 in Shapiro normality test , hence we can apply z-distribution and the sample size is also 30.  
CI\_z(v\_killed, ci= 0.95)

## # A tibble: 6 × 2  
## Measurements values  
## <chr> <dbl>  
## 1 sample\_size 30   
## 2 Mean 9.47  
## 3 sd 3.87  
## 4 Margin\_Error 1.38  
## 5 CI.lower.limit 8.08  
## 6 CI.Upper.limit 10.9

# Population mean of VanKilled  
mean(VanKilled)

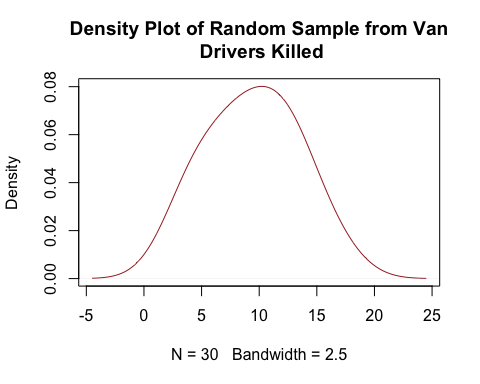
## [1] 9.057292

# Confidence Interval Plot  
library(plotrix)  
plotCI(v\_killed, li = 7.921209 , ui = 10.278791 )



# Density Plot showing the VanKilled Distribution  
plot(density(v\_killed, width=10), col = 'brown', main= 'Density Plot of Random Sample from Van\n Drivers Killed')

**NOTE**: Because random sample was taken, the values changed when the document was knitted causing the upper and lower limits to vary negligibly. The CI interval plot was plotted with the values from previous randomly generated sample. The difference is negligible.



**Explanation:** Using the sample command, we select a random sample of size 30 (n=30) from the VanKilled class. Then using Shapiro test, we test for normality and as the p value is found to be greater than 0.05, the normality checks out. Because the sample size is 30, we opt for z-distribution. The function calculates the upper and lower limit and our population mean *9.057292* falls in the range. As the interval is set at 0.95 hence there is a 95% chance of the population mean to be in between this range every time the result is calculated and 5% chance of falling out of the range.

**Graphs:** Using the upper and lower limits, we plot the Confidence Interval Scatter Plot showing the interval. Secondly, a density plot of the distribution of VanKilled is displayed, showing its normal nature.

## Based on t-Distribution

# This function computes Confidence intervals based on t-Distribution  
CI\_t <- function (x, ci = 0.95)  
{  
`%>%` <- magrittr::`%>%`  
Margin\_Error <- qt(ci + (1 - ci)/2, df = length(x) - 1) \* sd(x)/sqrt(length(x))  
df\_out <- data.frame( sample\_size=length(x), Mean=mean(x), sd=sd(x),  
Margin\_Error=Margin\_Error,  
'CI lower limit'=(mean(x) - Margin\_Error),  
'CI Upper limit'=(mean(x) + Margin\_Error)) %>%  
tidyr::pivot\_longer(names\_to = "Measurements", values\_to ="values", 1:6 )  
return(df\_out)  
}  
  
# Selecting a random sample of 30 from VanKilled (no. of van drivers killed)  
f\_killed = sample(front, 20)  
  
# Checking normality   
shapiro.test(f\_killed)

##   
## Shapiro-Wilk normality test  
##   
## data: f\_killed  
## W = 0.94827, p-value = 0.3416

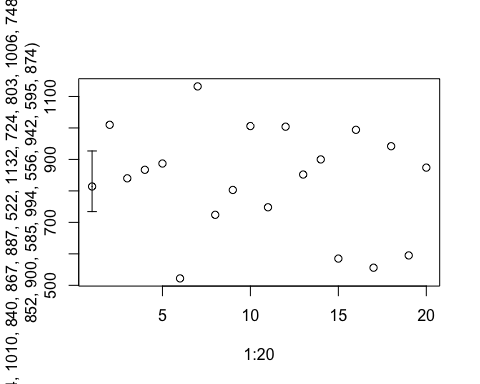
# As the p > 0.05 in Shapiro normality test , hence we can apply t-distribution as the sample size is also less than 30.  
CI\_t(f\_killed, ci= 0.95)

## # A tibble: 6 × 2  
## Measurements values  
## <chr> <dbl>  
## 1 sample\_size 20   
## 2 Mean 833.   
## 3 sd 169.   
## 4 Margin\_Error 78.9  
## 5 CI.lower.limit 754.   
## 6 CI.Upper.limit 912.

# Population mean of VanKilled  
mean(front)

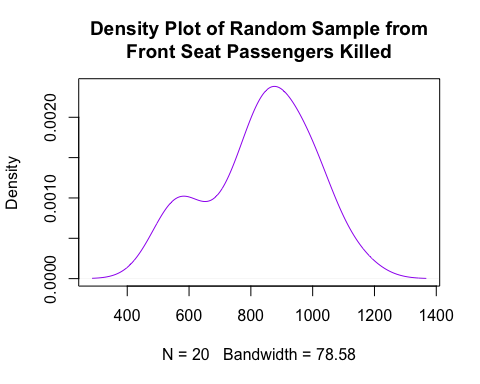
## [1] 837.2188

# Confidence Interval Plot  
library(plotrix)  
plotCI(f\_killed, li = 734.24234 , ui = 926.85766 )



# Density Plot showing the 'front' Distribution  
plot(density(f\_killed), col = 'purple', main= 'Density Plot of Random Sample from\nFront Seat Passengers Killed')

**NOTE**: Because random sample was taken, the values changed when the document was knitted causing the upper and lower limits to vary negligibly. The CI interval plot was plotted with the values from previous randomly generated sample. The difference is negligible.



**Explanation:** Using the sample command, we select a random sample of size 20 (n=20) from the VanKilled class. Then using Shapiro test, we test for normality and as the p value is found to be greater than 0.05, the normality checks out. Because the sample size is 20 (less than 30), we opt for t-distribution. The function calculates the upper and lower limit and our population mean *837.2188* falls in the range. As the interval is set at 0.95 hence there is a 95% chance of the population mean to be in between this range every time the result is calculated and 5% chance of falling out of the range.

**Graphs:** Using the upper and lower limits, we plot the Confidence Interval Scatter Plot showing the interval. Secondly, a density plot of the distribution of *front* is displayed, showing its normal nature.

# Hypothesis Testing

**Definition:** A hypothesis is made by the researchers about the data collected for any experiment or data set. A hypothesis is an assumption made by the researchers that are not mandatory true. In simple words, a hypothesis is a decision taken by the researchers based on the data of the population collected.

## One Sample t-test:

library(stats)  
# taking a sample of total deaths and serious injuries during car accidents in the given period  
tdrivers = sample(drivers, 20)  
  
# Shapiro Test to check the normality  
shapiro.test(tdrivers)

##   
## Shapiro-Wilk normality test  
##   
## data: tdrivers  
## W = 0.95846, p-value = 0.5135

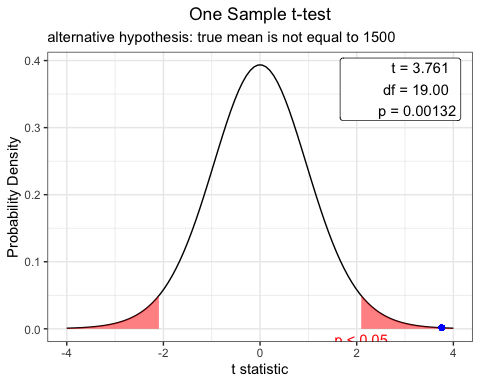
# using t test command to get the hypothesis test with Ho = mean equals 1500 and H1 = mean not equals 1500  
t\_test\_drivers <- t.test(tdrivers, mu = 1500)  
t\_test\_drivers

##   
## One Sample t-test  
##   
## data: tdrivers  
## t = 3.7614, df = 19, p-value = 0.001321  
## alternative hypothesis: true mean is not equal to 1500  
## 95 percent confidence interval:  
## 1590.594 1817.906  
## sample estimates:  
## mean of x   
## 1704.25

# Using the \*webr\* package we plot the one sample t-test  
require(webr)

## Loading required package: webr

plot(t\_test\_drivers)



**Explanation:** For this one sample hypothesis test using t-distribution, we take a random sample of size 20 from the class *drivers*. Using Shapiro test, we confirm the normality as to whether it is right to use t-distribution here and as the p-value was > 0.05, hence normality is true. Then using the t-test command from the library *stats* we generate a hypothesis test against the sample with;

Ho (null hypothesis) = mean equals 1500

H1 (alternative hypothesis) = mean not equals 1500

The P-Value from this test is found to be *0.001321*. If we were to make a decision whether to reject or accept null hypothesis on the basis of level of significance (alpha) = 0.05 (5%), then our null hypothesis will be **rejected here** here as the alpha-value > p-value & we accept the **alternative hypothesis**

Using the *webr* package, we plot the one-sample t-test which shows us that our value resides in the critical region hence accepting the alternative hypothesis.

## Two sample t-test for two independent samples

library(stats)  
# taking a sample of 20 from class \*front\* (front seat passenger casualties)  
tfront = sample(front, 20)  
  
# taking a sample of 20 from class \*rear\* (rear seat passenger casualties)  
trear = sample(rear, 20)  
  
# Shapiro Test to check the normality  
shapiro.test(tfront)

##   
## Shapiro-Wilk normality test  
##   
## data: tfront  
## W = 0.97589, p-value = 0.8709

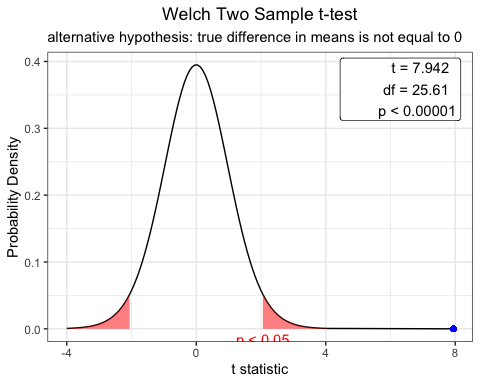
shapiro.test(trear)

##   
## Shapiro-Wilk normality test  
##   
## data: trear  
## W = 0.97747, p-value = 0.8974

# using t test command to get the hypothesis test with Ho = mean 1 equals mean 2 and H1 = mean 1 not equals mean 2  
t\_test\_passengers <- t.test(tfront, trear)  
t\_test\_passengers

##   
## Welch Two Sample t-test  
##   
## data: tfront and trear  
## t = 7.9421, df = 25.613, p-value = 2.256e-08  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## 291.8777 495.9223  
## sample estimates:  
## mean of x mean of y   
## 808.75 414.85

# Using the \*webr\* package we plot the two sample t-test  
require(webr)  
plot(t\_test\_passengers)

 **Explanation:** For this two sample hypothesis test using t-distribution, we take a random sample of size 20 from the class *front* and *rear* and hope to see if the average number of deaths of rear and front seat passengers are the same. Using Shapiro test, we confirm the normality as to whether it is right to use t-distribution here and as the p-value was > 0.05 for both the random samples, hence normality is true. Then using the t-test command from the library *stats* we generate a hypothesis test against the samples with;

Ho (null hypothesis) = mean 1 equals mean 2 (front avg. = rear avg.)

H1 (alternative hypothesis) = mean 1 not equals mean 2 (front avg. != rear avg.)

The P-Value from this test is found to be *0.00000002256*. If we were to make a decision whether to reject or accept null hypothesis on the basis of level of significance (alpha) = 0.01 (1%), then our null hypothesis will be **rejected here** here as the alpha-value > p-value & we accept the **alternative hypothesis**

The t-test tells that the mean of front is *848.20* and mean of rear is *409.05* and our p-value also confirm that there’s a highly significant and overwhelming evidence that both the means are not same. This consequently proves that the front seat casualties are higher than the rear seat deaths and statistically, one is more likely to die while sitting at the front during a car accident.

Using the *webr* package, we plot the two-sample t-test which shows us that our value does not reside in the acceptable region hence accepting the alternative hypothesis and rejecting null.

# Chi-Square Test

Chi-Square test in R is a statistical method which used to determine if two categorical variables have a significant correlation between them. The two variables are selected from the same population.

## Chi-Square Test of Independence:

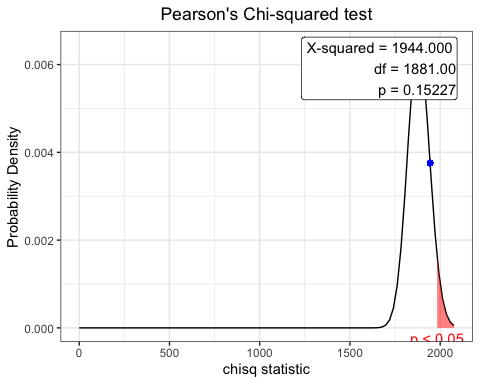
# Chi-Square test  
chsq\_test = chisq.test(Month, drivers, correct=TRUE)

## Warning in chisq.test(Month, drivers, correct = TRUE): Chi-squared approximation  
## may be incorrect

chsq\_test

##   
## Pearson's Chi-squared test  
##   
## data: Month and drivers  
## X-squared = 1944, df = 1881, p-value = 0.1523

# Using the \*webr\* package we plot the chi-square test of independence  
require(webr)  
plot(chsq\_test)



**Explanation:** We run the chi-square test of independence on *Months* and *drivers* (total people injured/seriously injured during accidents). Here the hypothesis is;

Ho = Months and Drivers are independent

H1 = Months and Drivers are dependent

After running the test using *chisq.test* command, we find the p-value to be *0.1523*. If we choose the level of significance to be 5% (alpha = 0.05), then p-value > alpha and we must reject the null hypothesis and **accept the alternative hypothesis** that *Months* and *drivers* are related.

Plotting the chi-square distribution using *webr* package, we can clearly see that the calculated value lies long before the critical region and hence the we accept alternative hypothesis of dependency between two variables.

# ANOVA

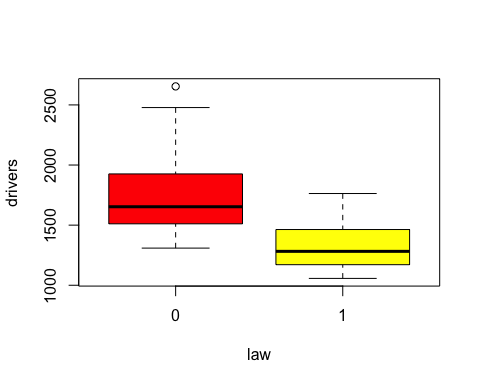
Like the t-test, Analysis of Variance (ANOVA) helps you find out whether the differences between groups of data are statistically significant. Difference is t-test is limited for two groups only while ANOVA can be applied on more than two groups. It works by analyzing the levels of variance within the groups through samples taken from each of them.

## One-Way ANOVA

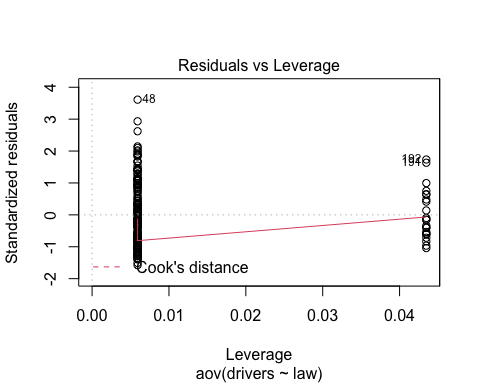
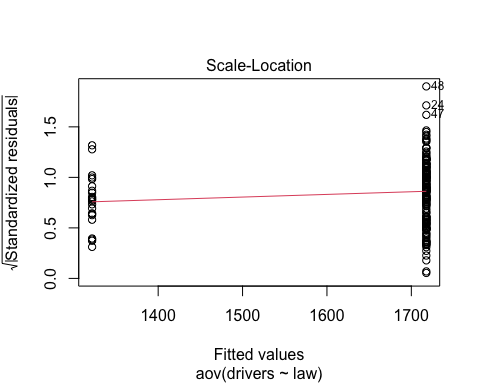
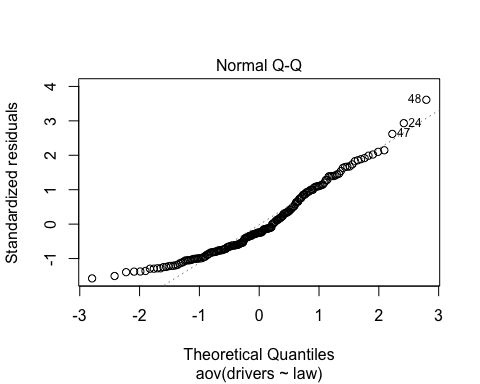
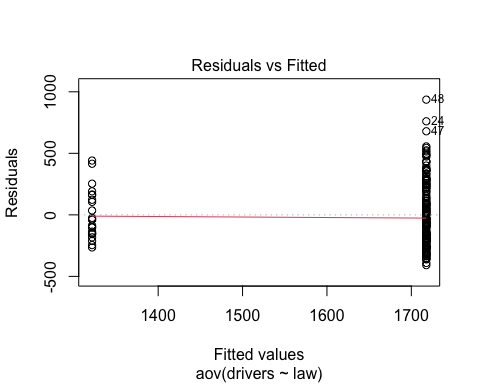
# First we use aov() to run the model  
a\_oneway <- aov(drivers ~ law)  
  
# then we use summary() to print the summary of the model  
summary(a\_oneway)

## Df Sum Sq Mean Sq F value Pr(>F)   
## law 1 3175602 3175602 46.98 9.76e-11 \*\*\*  
## Residuals 190 12844428 67602   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Box plot  
boxplot(drivers ~ law, col = heat.colors(2))



# Multiple plots from webr  
plot(a\_oneway)



**Explanation:** Here we test the total deaths and serious injuries in car accidents (drivers: quantitative variable) against the categorical variable law passed or not (passed:1, not passed:0). Here the hypothesis is;

Null hypothesis (H0) : The average mean of two groups are not different.

Alternative hypothesis (H1): The average mean of two groups are different.

Using the aov and summary command, we find that p-value is 9.76e-11. If we test it against the significance level of 5% (alpha = 0.05) then as p-value < alpha, we **reject the null hypothesis** in favor of alternative hypothesis. Hence the mean of two groups are different which proves that less people died when law of Seat belts was passed.

The box plot of both classes also shows that the mean value of both groups differ significantly.

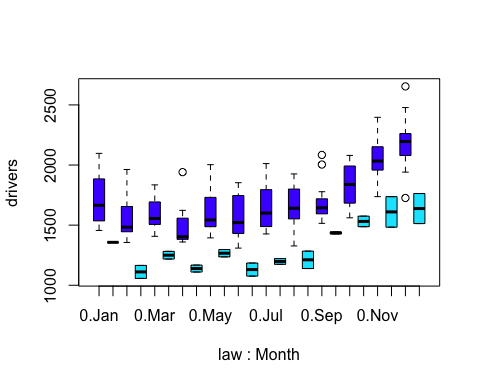
Using webr package, we can plots and get more insight on the data. We can see from Q-Q plot that the data is overall fairly normally distributed. The fitted vs Residual plots shows the difference between the expected and actual values.

## Two-Way ANOVA:

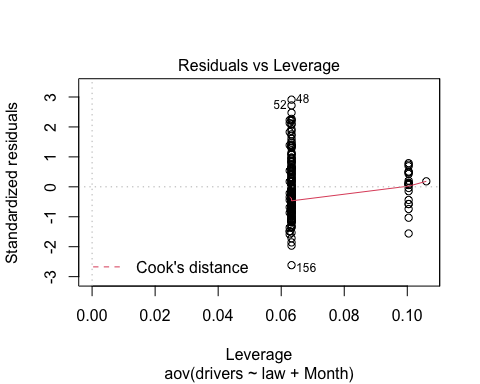
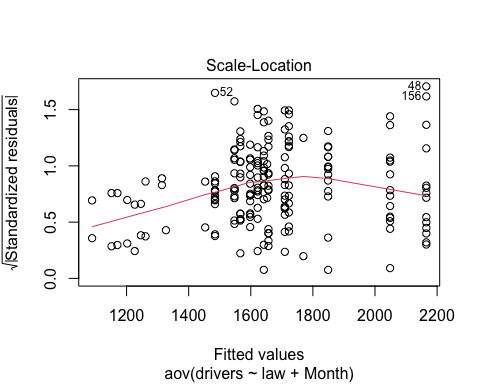
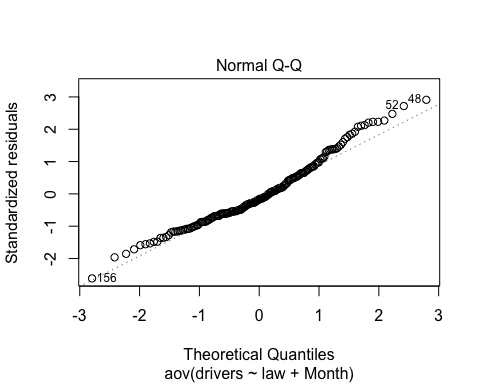
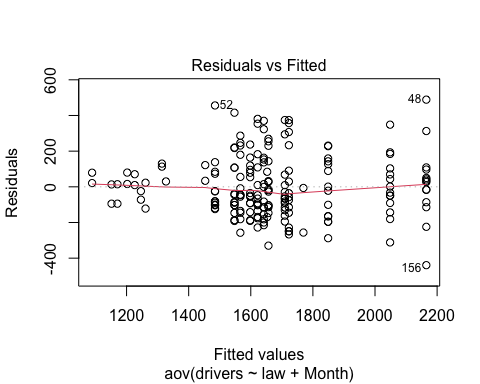
# First we use aov() to run the model  
a\_twoway <- aov(drivers ~ law+Month)  
  
# then we use summary() to print the summary of the model  
summary(a\_twoway)

## Df Sum Sq Mean Sq F value Pr(>F)   
## law 1 3175602 3175602 105.55 <2e-16 \*\*\*  
## Month 11 7458809 678074 22.54 <2e-16 \*\*\*  
## Residuals 179 5385619 30087   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Box plot  
boxplot(drivers ~ law+Month, col = topo.colors(2))



# Multiple plots from webr  
plot(a\_twoway)



**Explanation:** Here we test the total deaths and serious injuries in car accidents (drivers: quantitative variable) against the categorical variables law passed or not (passed:1, not passed:0) and Months. Here the hypothesis is;

Null hypothesis (H0) : The average mean of two groups are not different.

Alternative hypothesis (H1): The average mean of two groups are different.

Using the aov and summary command, we find that p-value is less than 2e-16 for both *law* & *Month.* If we test it against the significance level of 5% (alpha = 0.05) then as p-value < alpha, we **reject the null hypothesis** in favor of alternative hypothesis. Hence the mean of two groups are different which proves that less people died when law of Seat belts was passed.

The box plot of both classes also shows that the mean value of both groups differ significantly.

Using webr package, we can plots and get more insight on the data. We can see from Q-Q plot that there are *outliers* on both ends of the chart, but those on the upper end look more severe than those on the bottom and might be slightly right-skewed. But the data is overall fairly normally distributed. The fitted vs Residual plots shows the difference between the expected and actual values.

Adding Month to the model has made the model better as the residuals are now way less and the p-value has also decreased a good amount from one-way anova. This may also explain the seasonal effect in the number of deaths with higher deaths occurring in the Fall/Winter compared to Spring/Summer.

# Regression

Regression can be defined as the parameter to explain the relationship between two separate variables. It is more of a dependent feature where the action of one variable affects the outcome of the other variable. To put in the simplest terms, regression helps identify how variables affect each other.

## Linear Regression Model:

# Data set only with entries where law was not passed to test the effect of petrol prices without law passed/not passed affecting the test  
Law\_notpassed <-filter(Seatbelts,law ==0)  
  
# lm model   
lr <- lm(DriversKilled ~ PetrolPrice, data = Law\_notpassed)  
summary(lr)

##   
## Call:  
## lm(formula = DriversKilled ~ PetrolPrice, data = Law\_notpassed)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -47.804 -17.664 -4.083 14.058 62.712   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 190.53 15.33 12.425 < 2e-16 \*\*\*  
## PetrolPrice -634.69 149.51 -4.245 3.62e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 23.12 on 167 degrees of freedom  
## Multiple R-squared: 0.0974, Adjusted R-squared: 0.092   
## F-statistic: 18.02 on 1 and 167 DF, p-value: 3.616e-05

# ggplot of regression   
library(ggplot2)

##   
## Attaching package: 'ggplot2'

## The following objects are masked from 'package:psych':  
##   
## %+%, alpha

pvd <- ggplot (Law\_notpassed, aes(x = PetrolPrice, y= DriversKilled))+geom\_point(size = 2, shape = 18, col="darkgreen") +stat\_smooth (method = lm)+ xlab("Petrol Price") + ylab (" Monthly Driver Deaths")  
pvd

## `geom\_smooth()` using formula 'y ~ x'

**Explanation**: Here, the data is only from when law was not passed to measure a better affect of Petrol Prices on the number of car drivers killed. Petrol Prices here serve as the independent variable while the Driverskilled serves as the dependent variable. Using the lm command, we generate a linear regression model. Lets interpret the results;

Residuals: The residuals are the difference between the actual values and the predicted values. We want our median to be zero. From the data we have, our distribution is slightly right skewed.

Coefficients — Estimate: In the plot below, the line where it meets the y-axis is the y-intercept (b) written next to Intercept and the slope of the line (m) is written next to Petrol Prices . Because regression model finds the line that fits the points in such a way that it minimizes the distance between each point and the line, we have now the values to formulate an equation.

*y = -634.69 (x) + 190.53*

The negative slope denotes a negative relation between Petrol Prices and DriversKilled, meaning one would go up if other one goes down and vice-versa.

Coefficients – Std. Error: The standard error of the coefficient is an estimate of the standard deviation of the coefficient. In effect, it is telling us how much uncertainty there is with our coefficient. Using standard error, we can create Confidence Intervals. Lets say we create an interval of 95% then it would be as following:

-634.69 +- 1.96 (149.51) = -341.65 , -927.73

Hence we are 95% sure the slope falls in this range.

Coefficients – t value: The t-statistic is simply the coefficient divided by the standard error. In general, we want our coefficients to have large t-statistics, because it indicates that our standard error is small in comparison to our coefficient. The larger our t-statistic is, the more certain we can be that the coefficient is **not** zero. The t-statistic is then used to find the p-value.

Coefficients – Pr(>|t|) [p values] and Signif. Codes: The p-value is calculated using the t-statistic from the T distribution. The p-value, in association with the t-statistic, help us to understand how *significant* our coefficient is to the model. If p-value is below the significance level (usually 0.05), then it is significant meaning the coefficient is **not zero.** As we can see in the data, the p-value for both the intercept and the independent variable Petrol Price is very low denoting that both are significant and **non-zero.** However, the significance of the y-intercept or otherwise does not make much of a difference but excluding it when it is not significant will lead to a model where the dependent variable is zero when the dependent variable is zero hence including it is always vital even though its meaningless for the most part.

This is the same as running t-test on parametres of regression. Let us run it on slope.

Ho = null hypothesis: slope is zero

H1 = alternative hypothesis: slope is not zero

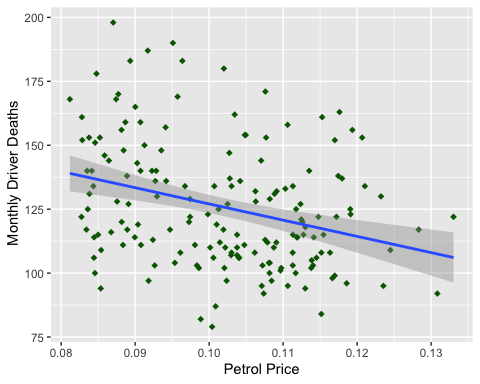
Running t-test on it will yield the t and p value already given in the linear model. Testing it against 95% confidence interval will make us reject null hypothesis in favor of alternative hypothesis as alpha value > p value. This is the same conclusion we reached above.

The coefficient codes give us a quick way to visually see which coefficients are significant to the model. To the right of the p-values you’ll see several asterisks (or none if the coefficient is not significant to the model). The number of asterisks corresponds with the significance of the coefficient as described in the legend just under the coefficients section. The more asterisks, the more significant.

Residual Standard Error: The residual standard error is a measure of how well the model fits the data. If we look at the plot below, we notice that the line doesn’t perfectly flow through each of the points and that there is a “residual” between the Deaths and the line (shown as a blue line). The residual standard error tells us the *average* amount that the actual values of Y (the dots) differ from the predictions (the line) in units of Y. In general, we want the smallest residual standard error possible, because that means our model’s prediction line is very close to the actual values, on average.

For our current model, we can see that on average, the actual values are around 23 deaths away from the predicted values (regression line).

Multiple R-squared and Adjusted R-squared: The Multiple R-squared value is most often used for simple linear regression (one predictor). It tells us what percentage of the variation within our dependent variable that the independent variable is explaining. In other words, it’s another method to determine how well our model is fitting the data. In the example above, Petrol Prices explains **9.74%** of the variation within Monthly Deaths, our dependent variable. This means that Petrol Prices helps to explains some of the variation within Deaths, but not as much as we would like. Ultimately, our model isn’t fitting the data very well.



# f-statistic:  
anova(lr)

## Analysis of Variance Table  
##   
## Response: DriversKilled  
## Df Sum Sq Mean Sq F value Pr(>F)   
## PetrolPrice 1 9631 9631.2 18.021 3.616e-05 \*\*\*  
## Residuals 167 89252 534.4   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

When running a regression model, either simple or multiple, a hypothesis test is being run on the global model.

Ho = Null hypothesis: no relationship between the dependent variable and the independent variable(s)

H1 = Alternative hypothesis: there is a relationship.

Said another way, the null hypothesis is that the coefficients for all of the variables in your model are zero. The alternative hypothesis is that at least one of them is not zero. The F-statistic and overall p-value help us determine the result of this test. Looking at the F-statistic alone can be a little misleading depending on how many variables are in your test. If you have a lot of independent variables, it’s common for an F-statistic to be close to one and to still produce a p-value where we would reject the null hypothesis. However, for smaller models, a larger F-statistic generally indicates that the null hypothesis should be rejected. A better approach is to utilize the p-value that is associated with the F-statistic.

We can see from the anova test, the F-statistic is large and our p-value is very small. This would lead us to **reject the null hypothesis** and conclude that there is strong evidence that a relationship does exist between *Monthly Deaths* and *Petrol Prices*.

## Multiple Regression Model:

# Data set only with entries where law was not passed to test the effect of petrol prices and kms (distance) without law passed/not passed affecting the test  
Law\_notpassed <-filter(Seatbelts,law ==0)  
  
# linear model   
mr <- lm(DriversKilled ~ PetrolPrice + kms, data = Law\_notpassed)  
summary(mr)

##   
## Call:  
## lm(formula = DriversKilled ~ PetrolPrice + kms, data = Law\_notpassed)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -50.425 -17.921 -2.416 14.042 60.903   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.004e+02 1.644e+01 12.193 < 2e-16 \*\*\*  
## PetrolPrice -5.737e+02 1.535e+02 -3.738 0.000255 \*\*\*  
## kms -1.113e-03 6.875e-04 -1.619 0.107331   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 23.01 on 166 degrees of freedom  
## Multiple R-squared: 0.1114, Adjusted R-squared: 0.1007   
## F-statistic: 10.41 on 2 and 166 DF, p-value: 5.512e-05

# save predictions of the model in the new data frame together with variable you want to plot against  
predicted\_df <- data.frame(Law\_notpassed$kms, car\_drivers\_pred = predict(mr, Law\_notpassed))  
  
# predicted line of multiple linear regression  
library(ggplot2)  
ggplot(data = Law\_notpassed, aes(x = kms, y = DriversKilled)) +   
 geom\_point(color='blue') +  
 stat\_smooth(method = lm, color='red',data = predicted\_df, aes(x=Law\_notpassed$kms, y=car\_drivers\_pred))

## `geom\_smooth()` using formula 'y ~ x'

**Explanation**: Here, the data is only from when law was not passed to measure a better affect of Petrol Prices and kms on the number of car drivers killed. Petrol Prices and kms here serve as the independent variables while the Driverskilled serves as the dependent variable. Using the lm command, we generate a linear regression model. Lets interpret the results;

Residuals: The residuals are the difference between the actual values and the predicted values. We want our median to be zero. From the data we have, our distribution is slightly right skewed.

Coefficients — Estimate: In the plot below, the line where it meets the y-axis is the y-intercept (b) written next to Intercept and the slope of first variable (PetrolPrices) (m1) is written next to Petrol Prices while slope of second variable (kms) (m2) is written next to kms. Because regression model finds the line that fits the points in such a way that it minimizes the distance between each point and the line, we have now the values to formulate an equation.

*y = -573.7 (x1) -1113 (x2) + 190.53*

The negative slopes denotes a negative relation between Petrol Prices, kms and DriversKilled, meaning if independents would go up, dependent goes down and vice-versa.

Coefficients – Std. Error: The standard error of the coefficient is an estimate of the standard deviation of the coefficient. In effect, it is telling us how much uncertainty there is with our coefficient. Using standard error, we can create Confidence Intervals. Lets say we create an interval of 95% then it would be as following:

-573.7 +- 1.96 (153.5) = -272.84 , -874.56

-1113 +- 1.96 (0.0006875) = -1112.99, -1113.00

Hence we are 95% sure the slopes fall in these ranges.

Coefficients – t value: The t-statistic is simply the coefficient divided by the standard error. In general, we want our coefficients to have large t-statistics, because it indicates that our standard error is small in comparison to our coefficient. The larger our t-statistic is, the more certain we can be that the coefficient is **not** zero. The t-statistic is then used to find the p-value.

Coefficients – Pr(>|t|) [p values] and Signif. Codes: The p-value is calculated using the t-statistic from the T distribution. The p-value, in association with the t-statistic, help us to understand how *significant* our coefficient is to the model. If p-value is below the significance level (usually 0.05), then it is significant meaning the coefficient is **not zero.** As we can see in the data, the p-value for both the intercept and the independent variable Petrol Price is very low denoting that both are significant and **non-zero.** But the p-value for kms is greater than the significance level (0.1 > 0.05) and is **not significant and zero.** However, the significance of the y-intercept or otherwise does not make much of a difference but excluding it when it is not significant will lead to a model where the dependent variable is zero when the dependent variable is zero hence including it is always vital even though its meaningless for the most part.

This is the same as running t-test on parametres of regression. Let us run it on slopes.

Ho = null hypothesis: slope is zero

H1 = alternative hypothesis: slope is not zero

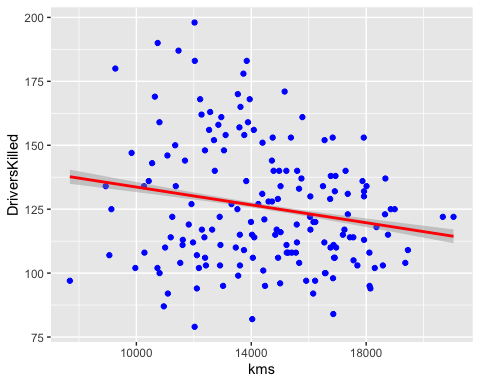
Running t-test on it will yield the t and p value already given in the linear model. Testing it against 95% confidence interval will make us reject null hypothesis in favor of alternative hypothesis as alpha value > p value. This is the same conclusion we reached above.

The coefficient codes give us a quick way to visually see which coefficients are significant to the model. To the right of the p-values you’ll see several asterisks (or none if the coefficient is not significant to the model). The number of asterisks corresponds with the significance of the coefficient as described in the legend just under the coefficients section. The more asterisks, the more significant. As you can see, there are no asterisks on kms indicating it is not significant.

Residual Standard Error: The residual standard error is a measure of how well the model fits the data. If we look at the plot below, we notice that the line doesn’t perfectly flow through each of the points and that there is a “residual” between the Deaths and the line (shown as a red line). The residual standard error tells us the *average* amount that the actual values of Y (the dots) differ from the predictions (the line) in units of Y. In general, we want the smallest residual standard error possible, because that means our model’s prediction line is very close to the actual values, on average.

For our current model, we can see that on average, the actual values are around 23 deaths away from the predicted values (regression line).

Multiple R-squared and Adjusted R-squared: The Adjusted R-squared value is most often used for multiple linear regression (more than one predictor). It tells us what percentage of the variation within our dependent variable that the independent variables is explaining. In other words, it’s another method to determine how well our model is fitting the data. In the example above, Petrol Prices and kms explains **10.07%** of the variation within Monthly Deaths, our dependent variable. As from the SLR, we know most of it is from Petrol prices and we saw above that kms is not significant. This means that Petrol Prices and a minuscule proportion of kms helps to explains some of the variation within Deaths, but not as much as we would like. Ultimately, our model isn’t fitting the data very well.



# F-statistic  
anova(mr)

## Analysis of Variance Table  
##   
## Response: DriversKilled  
## Df Sum Sq Mean Sq F value Pr(>F)   
## PetrolPrice 1 9631 9631.2 18.1960 3.339e-05 \*\*\*  
## kms 1 1388 1387.5 2.6214 0.1073   
## Residuals 166 87864 529.3   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

When running a regression model, either simple or multiple, a hypothesis test is being run on the global model.

Ho = Null hypothesis: no relationship between the dependent variable and the independent variable(s)

H1 = Alternative hypothesis: there is a relationship.

Said another way, the null hypothesis is that the coefficients for all of the variables in your model are zero. The alternative hypothesis is that at least one of them is not zero. The F-statistic and overall p-value help us determine the result of this test. Looking at the F-statistic alone can be a little misleading depending on how many variables are in your test. If you have a lot of independent variables, it’s common for an F-statistic to be close to one and to still produce a p-value where we would reject the null hypothesis. However, for smaller models, a larger F-statistic generally indicates that the null hypothesis should be rejected. A better approach is to utilize the p-value that is associated with the F-statistic.

We can see from the anova test, the F-statistic , for Petrol Prices, is large and our p-value is very small. For kms, the F-statistic is small and the p-value is also larger than the level of significance.

Independent Variable Petrol Prices:

This would lead us to **reject the null hypothesis** and conclude that there is strong evidence that a relationship does exist between *Monthly Deaths* and *Petrol Prices*.

Independent Variable kms:

This would lead us to **accept the null hypothesis** and reject alternative hypothesis and conclude that there is strong evidence that a relationship does **not** exist between *Monthly Deaths* and *kms*.