Proyecto Final

Equipo 13

08/2/2021

Contents

```
#librerias Necesarias para correr el script
library(data.table)
library(dplyr)
library(lubridate)
library(ggplot2)
library(tidyr)
library(purrr)
library(ggpubr)
library(forecast)
library(stringr)
```

Descarga del archivo del numero de accidentes desde la pagina de INEGI

```
url="https://www.inegi.org.mx/contenidos/programas/accidentes/datosabiertos/atus_anual_csv.zip"
zip_file="atus_anual_csv.zip"
if (!file.exists(zip_file)){
    download.file(url, destfile = zip_file , mode = 'wb')
}
```

Se descomprime el archivo con el numero de accidentes de 1997 a 2019

```
if (!dir.exists("atus_anual_1997_2019")) {
    unzip("atus_anual_csv.zip")
}
```

Se crea una lista con los nombres de los archivos (para 10 años)

```
names<-sort(dir("atus_anual_1997_2019/conjunto_de_datos/"),decreasing = TRUE)[1:10]

for(i in seq_along(names)) {
     names[i]<-(paste0("atus_anual_1997_2019/conjunto_de_datos/",names[i]))

}
```

Se extrae el nombre de las columnas

```
column_names<-fread("atus_anual_1997_2019/diccionario_de_datos/diccionario_de_datos_atus_anual_1997_2019
                    select = 1,data.table = F)[,1]
```

Se extraen los archivos y se almacenan en una lista

```
lista<-lapply(names,fread,select = (1:45),data.table = F,col.names = column_names, encoding = 'UTF-8')
```

El interes es por los datos relacionados con la CDMX por lo cual con base en el diccionario de datos, se sabe que el numero de entidad correspondiendte a la CDMX es 9

```
lista<-lapply(lista, function(x) filter(x,ID_ENTIDAD ==9))
```

Se combina la lista de datos en un solo dataframe

```
data <- do.call(rbind, lista)
```

Se agrego el nombre de la delegacion dependiendo el codigo Asignado primero se extraen los codigos de municipio:

```
Municipios<-fread("atus_anual_1997_2019/catalogos/tc_municipio.csv",
                  data.table = F,encoding = 'UTF-8',select = c(1:3),
                  col.names = c("Entidad","ID_MUNICIPIO","Municipio"))
```

Se filtran los municipios Pertenecientes a la CDMX ya que la Entidad 9 pertenece a la CDMX segun los registros del INEGI

```
(Municipios<-Municipios%>%filter(Entidad==9,ID_MUNICIPIO!=999)%>%select(ID_MUNICIPIO,Municipio))
```

Finalmente, se combina el nombre de los municipios y su codigo en el data.frama original creando una copia para preservar el set de datos originales.

```
data_clean<-data%>%left_join(Municipios,by="ID_MUNICIPIO")
```

importamos el diccionario de datos para colsultas acerca de variables

```
dic_datos<-fread("atus_anual_1997_2019/diccionario_de_datos/diccionario_de_datos_atus_anual_1997_2019.cs
head(dic_datos)
```

```
##         COLUMNA
## 1:    COBERTURA
## 2:   ID_ENTIDAD
## 3: ID_MUNICIPIO
## 4:         ANIO
## 5:          MES
## 6:      ID_HORA
##                                                                                                    
## 1:                                          Área geográfica a la que están referidos los indicadores es
## 2:            Clave de la entidad federativa según el Catálogo de Entidades, Municipios y Localidades
## 3:                       Clave del municipio según el Catálogo de Entidades, Municipios y Localidades
## 4:                                        Los cuatro dígitos correspondientes al año en que ocurrió el
```

```
## 5:                                              Correspondiente al mes de referencia en que ocurrió el
## 6: La hora (sin los minutos) en que ocurrió el accidente, con rango: 00-23 horas. Clave 99 Hora no es
##    TIPO_DATO LONGITUD COD_VALIDO
## 1:   varchar      200
## 2:   varchar        2     \t01-32
## 3:   varchar        3   \t001-999
## 4:       int       NA  1997-2019
## 5:   varchar        2     01-dic
## 6:       int       NA      \t0-23
```

Se agrega una columna con el nombre del mes de acuerdo al número de mes

```
data_clean$Nom_Mes<-recode(data_clean$MES,"1"="ENE","2"="FEB","3"="MAR",
       "4"="ABR","5"="MAY","6"="JUN","7"="JUL","8"="AGO","9"="SEP",
       "10"="OCT","11"="NOV","12"="DIC")
```

Se arreglo el nombre de los dias de la Semana.

```
data_clean$DIASEMANA<-recode(data_clean$DIASEMANA,"lunes"="Lunes","Miercoles"="Miércoles","Sabado"="Sába
```

Se agrega una columna de Fecha-hora

```
data_clean<-data_clean%>%mutate(Fecha=paste0(MES,"/",ID_DIA,"/",ANIO," ",ID_HORA,":",ID_MINUTO,":"))
```

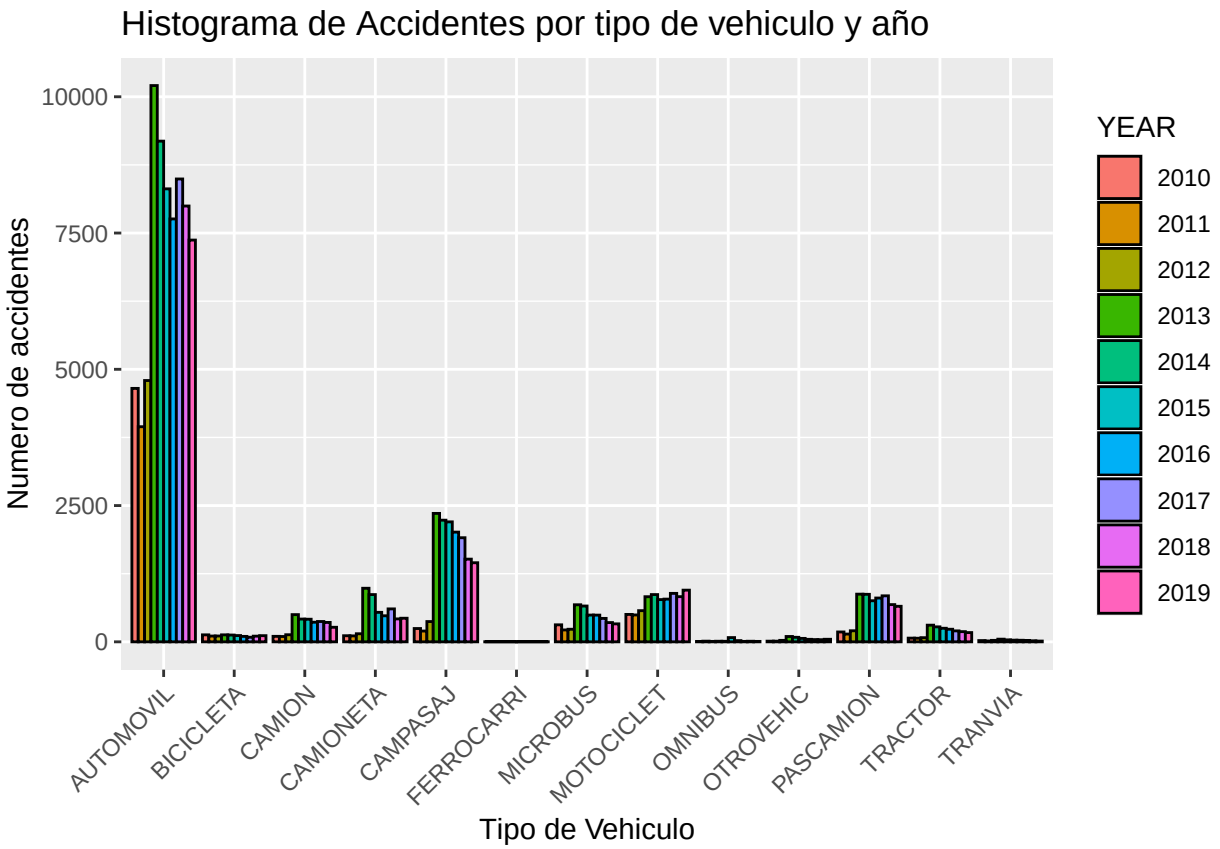
Se cambio a formato Fecha-hora utilizando la libreria lubridate

```
data_clean$Fecha<-mdy_hm(data_clean$Fecha)
```

Generacion de la grafica de histograma de Numero de Accidentes dependiendo el tipo de vehiculo por Año

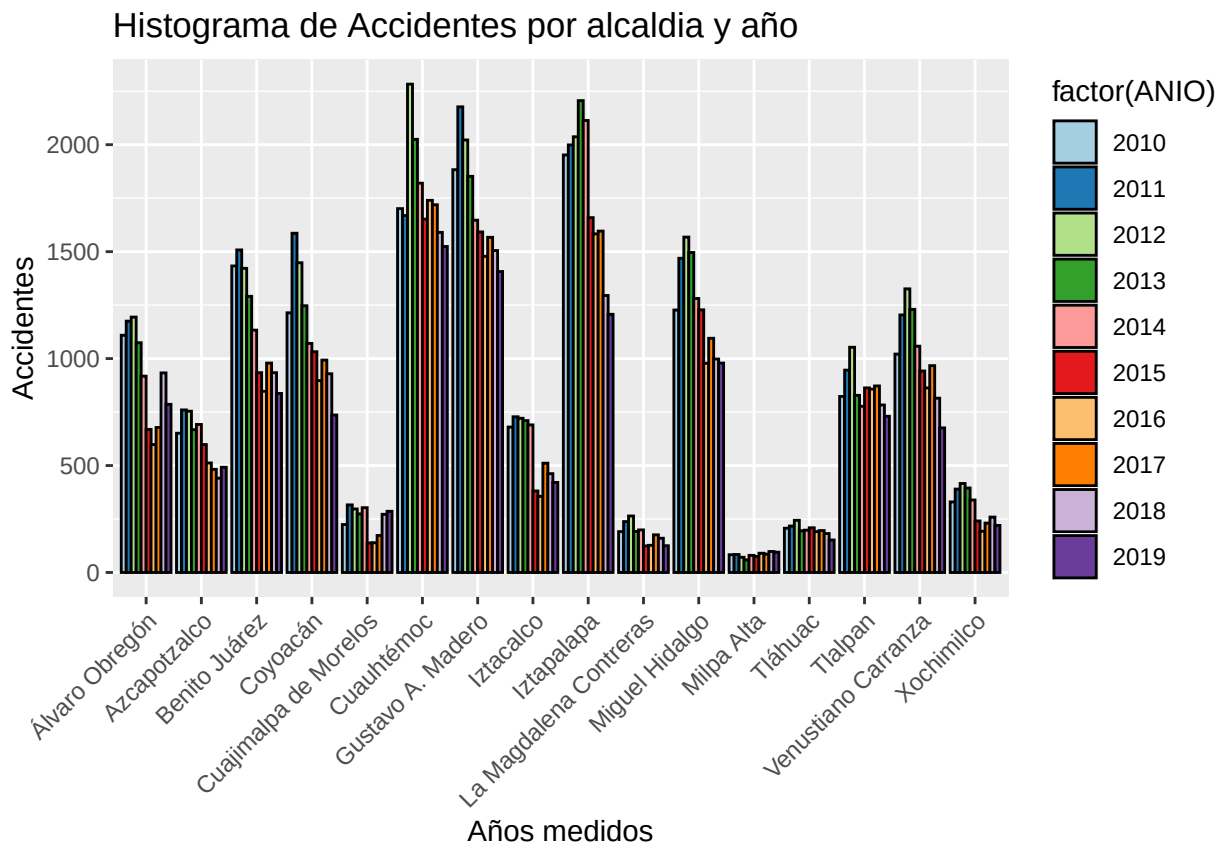
```
#nombre de las columnas de tipo de vehiculo
Modelo<-c("AUTOMOVIL","CAMPASAJ","MICROBUS","PASCAMION","OMNIBUS","TRANVIA","CAMIONETA","CAMION","TRACTO

#convierte las columnas en filas
md2<-gather(data_clean,Modelo,key = "Modelo",value = "Num_accidentes")
#filtra los datos para eliminar lo valores con 9 y las edades de 0  y 99
md2<-md2%>%filter(ID_EDAD!=0,Num_accidentes!=0,ID_EDAD!=99)
#agrupa por modelo y fecha y calcula el numero de elementos
md2<-md2%>%group_by(Fecha,Modelo)%>%select(Fecha,Modelo,Num_accidentes)%>%summarise(n=n())
#agrupa por modelo y fecha
md2<-md2%>%group_by(Modelo,year(Fecha))%>%summarise(n=n())
```

```
ggplot(data = md2, aes(factor(Modelo),n)) +
  geom_col(aes(fill=factor(`year(Fecha)`)),position ="dodge",col="black")+
  theme(axis.text.x = element_text(angle=45, hjust=1))+ ggtitle("Histograma de Accidentes por tipo de ve
  xlab("Tipo de Vehiculo") +labs(fill = "YEAR")+ylab("Numero de accidentes")
```



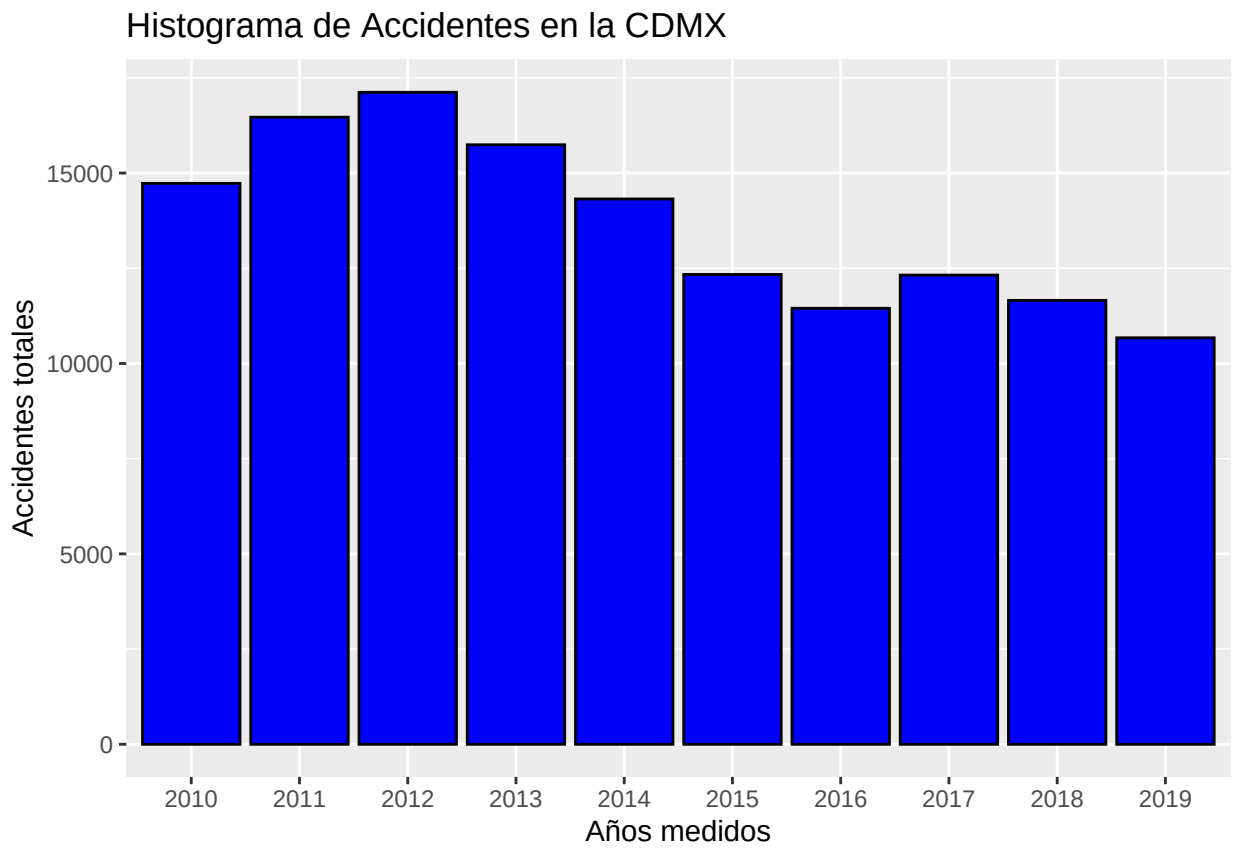
Generación de histograma de accidentes por alcaldia y año

```
g <- ggplot(data_clean, aes(Municipio))
g+geom_bar(aes(fill=factor(ANIO)),position = "dodge",col="black")+
  theme(axis.text.x = element_text(angle=45, hjust=1))+scale_fill_brewer(palette="Paired")+ggtitle("His
  ylab("Accidentes") +  xlab("Años medidos")
```



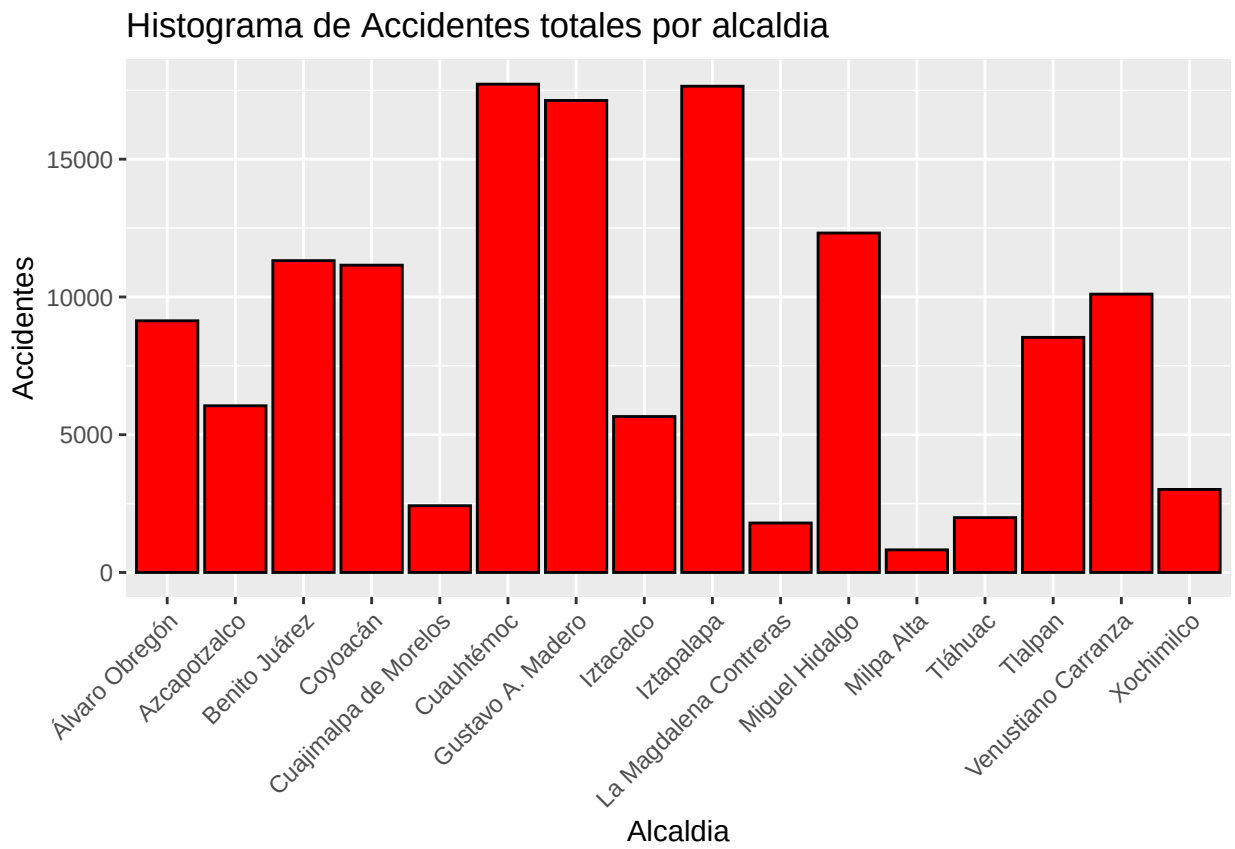
Generación de histograma de accidentes por año

```
data_clean %>%
  ggplot() +
  aes(x = factor(ANIO)) + geom_bar(col= "black", fill = "blue", stat = "count")+
  ggtitle("Histograma de Accidentes en la CDMX") +
  ylab("Accidentes totales") +
  xlab("Años medidos")
```



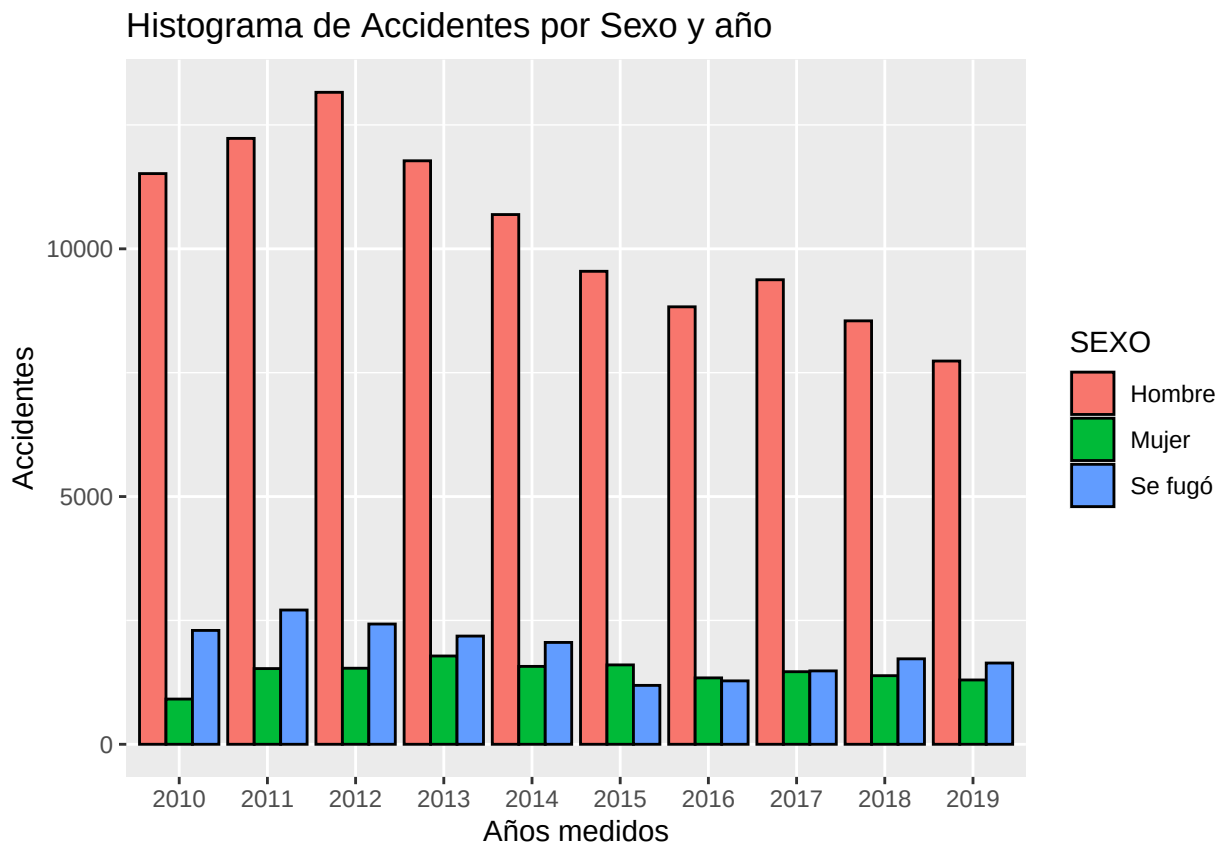
Generación de histograma de accidentes por alcaldia

```
data_clean %>%
  ggplot() +
  aes(x = Municipio) + geom_bar(col= "black", fill = "red", stat = "count")+
  ggtitle("Histograma de Accidentes totales por alcaldia") +
  ylab("Accidentes") +
  xlab("Alcaldia")+theme(axis.text.x = element_text(angle=45, hjust=1))
```



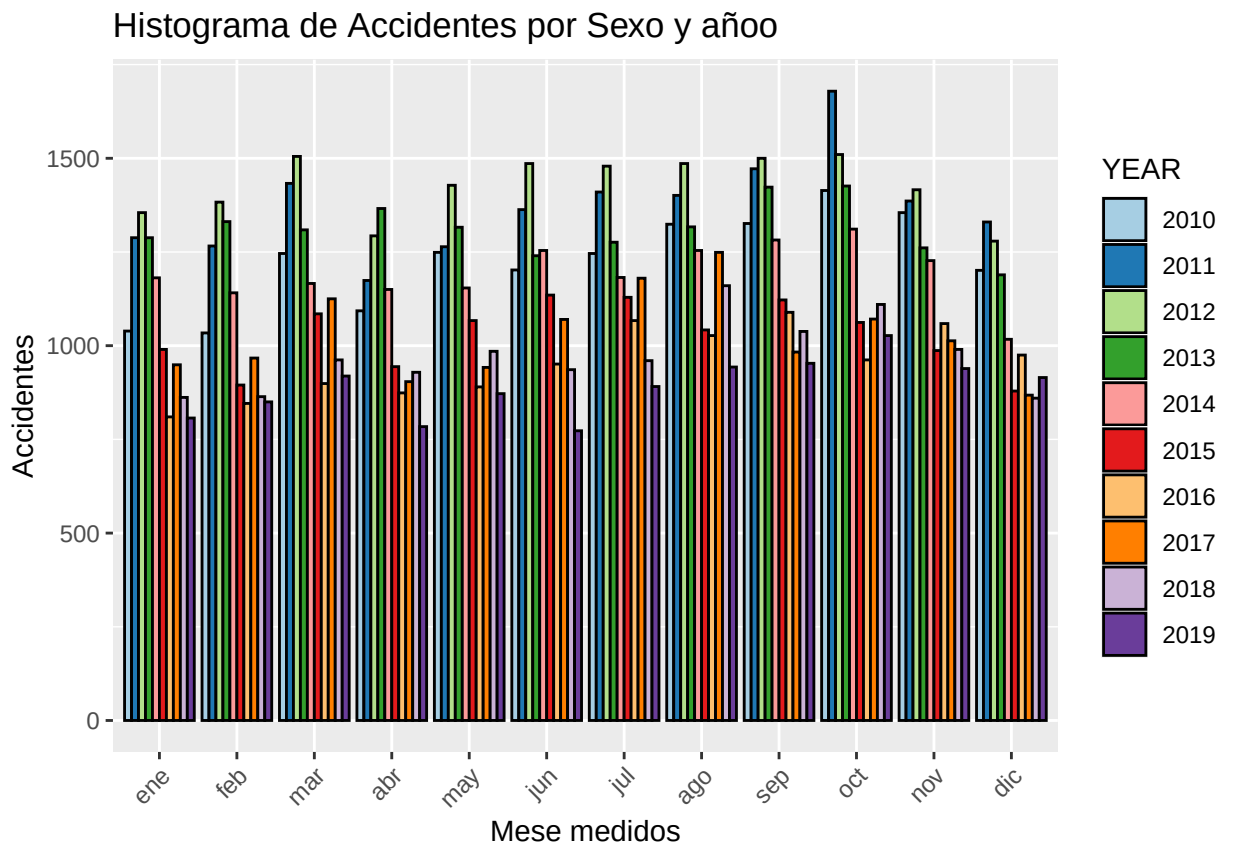
Generación de histograma de accidentes por sexo y año

```
data_clean %>%
  ggplot() +
  aes(x = factor(ANIO)) + geom_bar(aes(fill = SEXO),position = "dodge",col="black",stat = "count")+
  ggtitle("Histograma de Accidentes por Sexo y año") +
  ylab("Accidentes") +
  xlab("Años medidos")
```



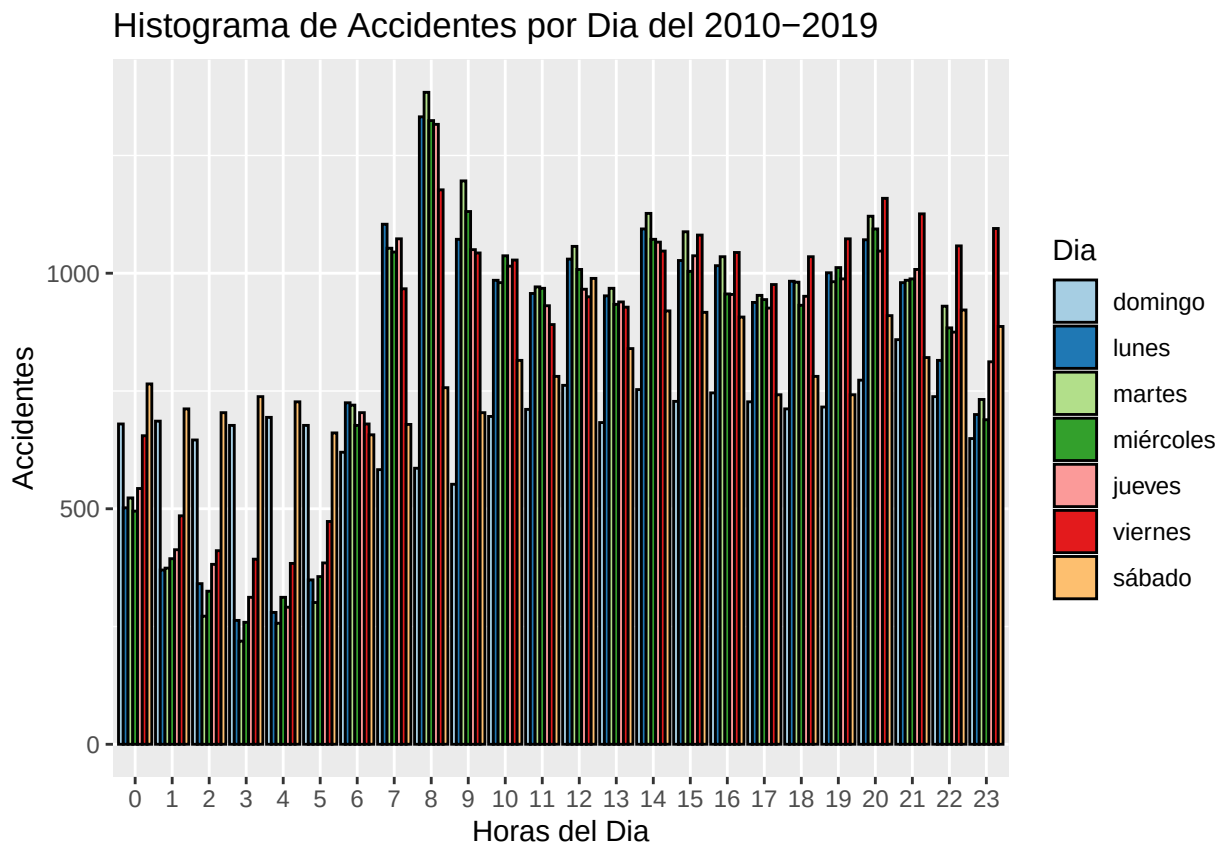
Generación de histograma de accidentes por mes y año

```
data_clean %>%
  ggplot() +
  aes(x = month(Fecha,label = TRUE)) + geom_bar(aes(fill = factor(ANIO)),position = "dodge",col="black"
  ggtitle("Histograma de Accidentes por Sexo y añoo") +
  ylab("Accidentes") +
  xlab("Mese medidos")+theme(axis.text.x = element_text(angle=45, hjust=1))+scale_fill_brewer(palette="
```



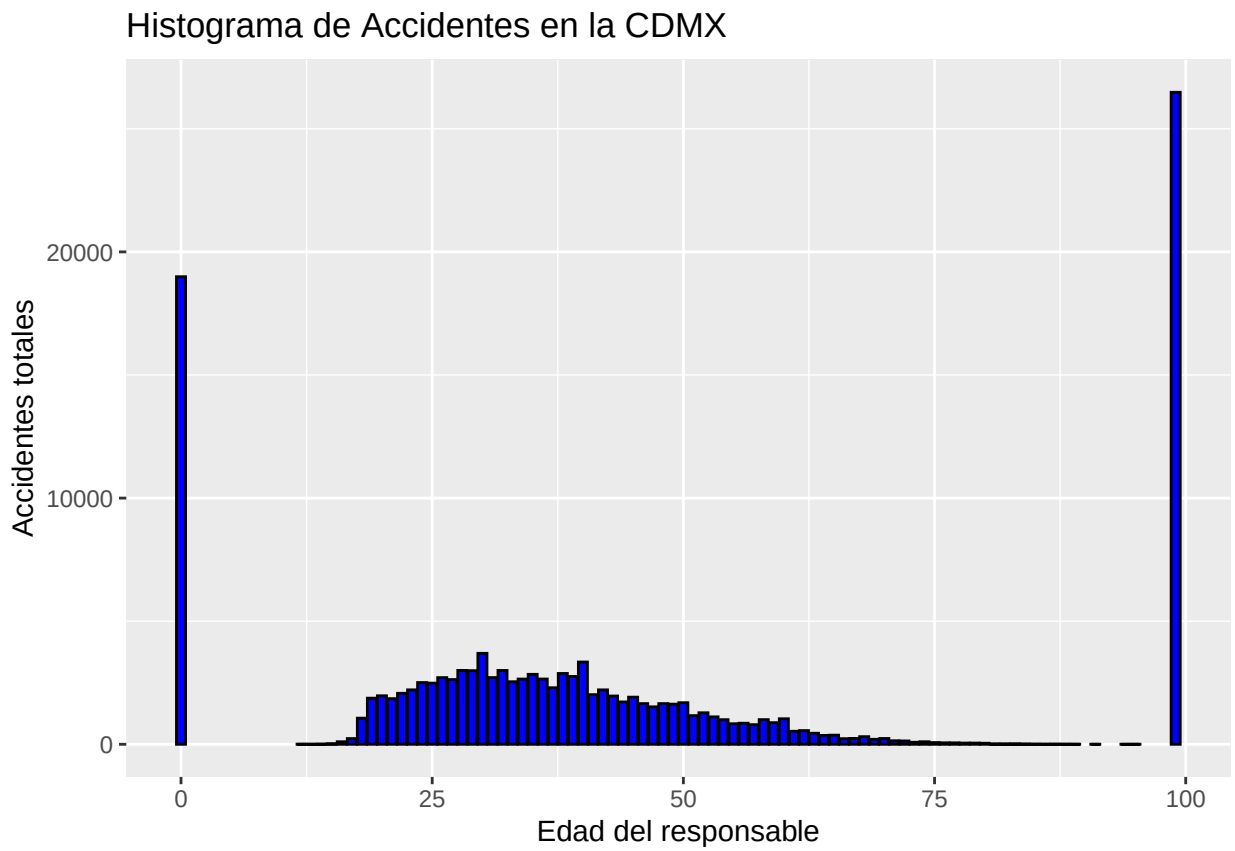
Generación de histograma de accidentes por hora y dia

```
data_clean %>%
  ggplot() +
  aes(x = factor(ID_HORA)) + geom_bar(aes(fill = wday(Fecha,label = T,abbr = F)),position = "dodge",col=
  ggtitle("Histograma de Accidentes por Dia del 2010-2019") +
  ylab("Accidentes") +
  xlab("Horas del Dia")+labs(fill = "Dia")+scale_fill_brewer(palette="Paired")#+facet_wrap(.~DIASEMANA,
```



Generación de histograma de accidentes por edad

```
data_clean %>%
  ggplot() +
  aes(x = ID_EDAD) + geom_bar(col= "black", fill = "blue", stat = "count")+
  ggtitle("Histograma de Accidentes en la CDMX") +
  ylab("Accidentes totales") +
  xlab("Edad del responsable")
```



ANALISis INFERENCIALES Apartir de la utima grafica obtenida de numero de accidentes en funcion de la edad se comienza a analizar los datos. Primero se observa que los valores de edad 0 y 100 albergan demasiados datos, de acuerdo a los registros del INEGI cuando el conductor se fugaba, los registros asignan el valor 0 a la edad. En cuanto a 99 significa que se desconoce la edad del conductor.

```
(data_clean%>%group_by(ID_EDAD)%>%summarise(num_acc=n())%>%filter(ID_EDAD==0|ID_EDAD==99)%>%mutate(prop
```

```
## # A tibble: 2 x 3
##   ID_EDAD num_acc proportion
##     <int>   <int>      <dbl>
## 1       0   18990      0.139
## 2      99   26480      0.194
```

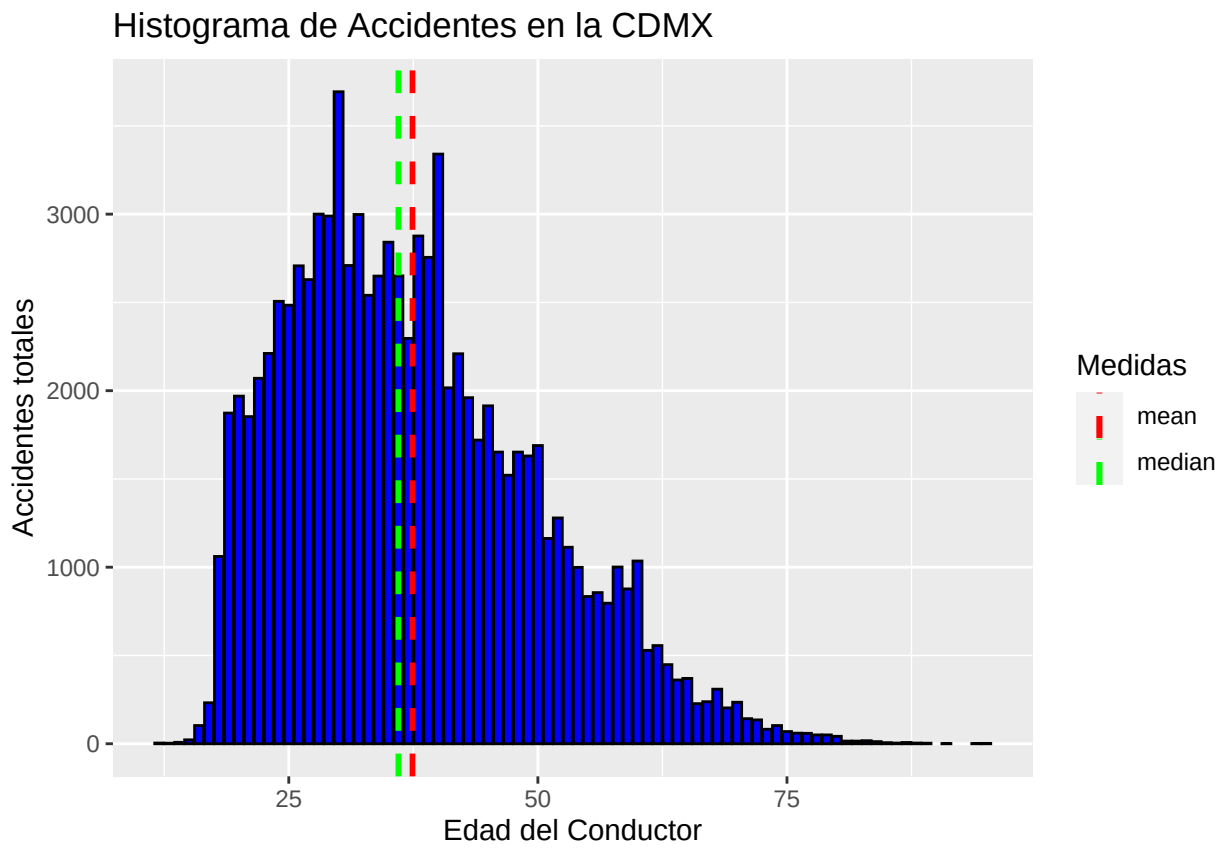
Son considerables la cantidad de datos y causan que la distribucion se ve afectada, para fines practicos se decidio eliminar estos datos extremos

```
df_edad<-data_clean %>% filter(ID_EDAD!=0,ID_EDAD!=99)#%>%summarise(mean=median(ID_EDAD))
```

ELiminando los valores la distribucion de edades queda de la siguiente forma:

```
g<-data_clean %>% filter(ID_EDAD!=0,ID_EDAD!=99)%>%
  ggplot() +
  aes(x = ID_EDAD) + geom_bar(col= "black", fill = "blue", stat = "count")+
  ggtitle("Histograma de Accidentes en la CDMX") +
```

```
  ylab("Accidentes totales") +
    xlab("Edad del Conductor")
g+geom_vline(aes(xintercept=median(ID_EDAD),color="median"),linetype="dashed",size=1) +
  geom_vline(aes(xintercept=mean(ID_EDAD),color="mean"),linetype="dashed", size=1) +
  scale_color_manual(name = "Medidas", values = c(median = "green", mean = "red"))
```



teniendo los siguientes valores de media y mediana

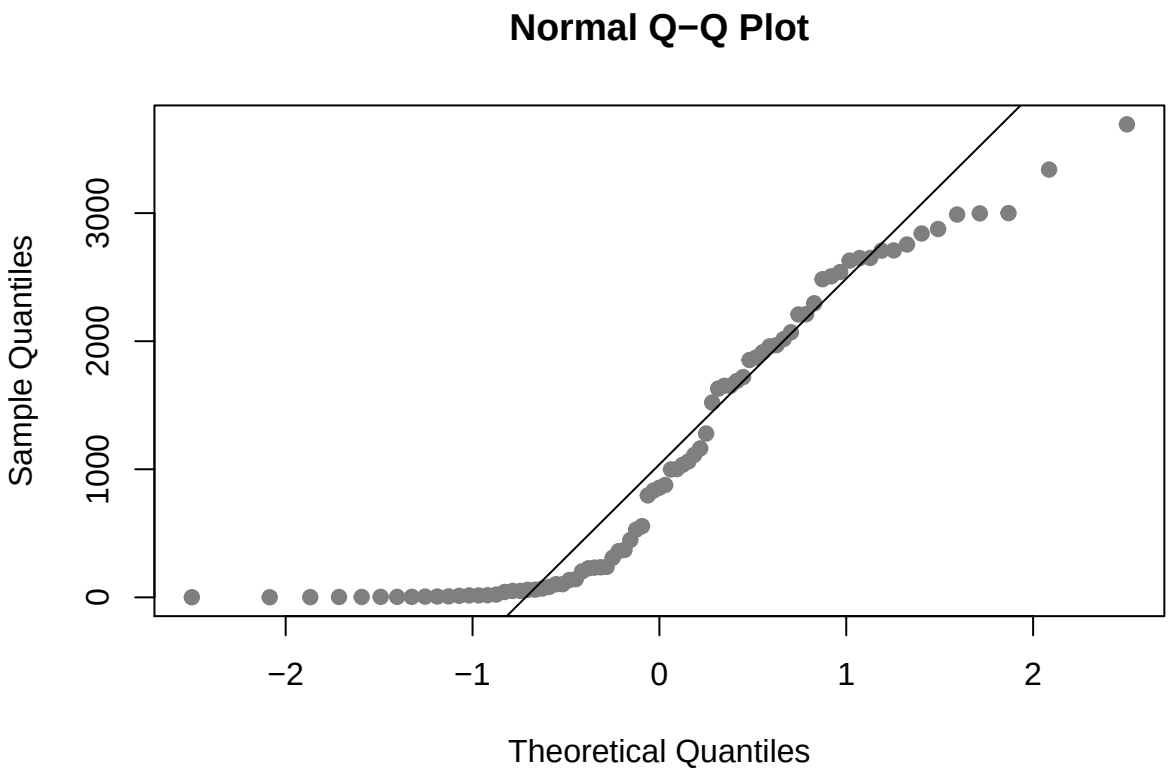
```
mean(df_edad$ID_EDAD);median(df_edad$ID_EDAD)
```

```
## [1] 37.41633
```

```
## [1] 36
```

Para poder realizar algunas hipotesis primero determinamos si la distribucion de las edades se comporta como una distribucion normal, primero realizamos un grafico Q-Q plot Para ver si los datos se ajustan

```
df_edad<-data_clean%>%filter(ID_EDAD!=0,ID_EDAD!=99)%>%group_by(ID_EDAD)%>%summarise(n=n())

qqnorm(df_edad$n, pch = 19, col = "gray50")
qqline(df_edad$n)
```



de acuerdo a los graficos la distribucion no se comporta como una normal, podemos realizar un Shapiro test para comprobarlo

```
(shapiro.test(df_edad$n))
```

tal y como lo supusimos no se comporta de manera normal.Por esta razon se decidio hacer pruebas No parametricas entre estas se encuentra el test de Mann–Whitney–Wilcoxon la cual contrasta que la probabilidad de que una observación de la población X supere a una observación de la población Y es igual a la probabilidad de que una observación de la población "Y" supere a una de la población X. Es decir, que los valores de una población no tienden a ser mayores que los de otra.las hipotesis por lo tanto serian **H0:P(X>Y)=P(Y>X) Ha:P(X>Y) P(Y>X)**

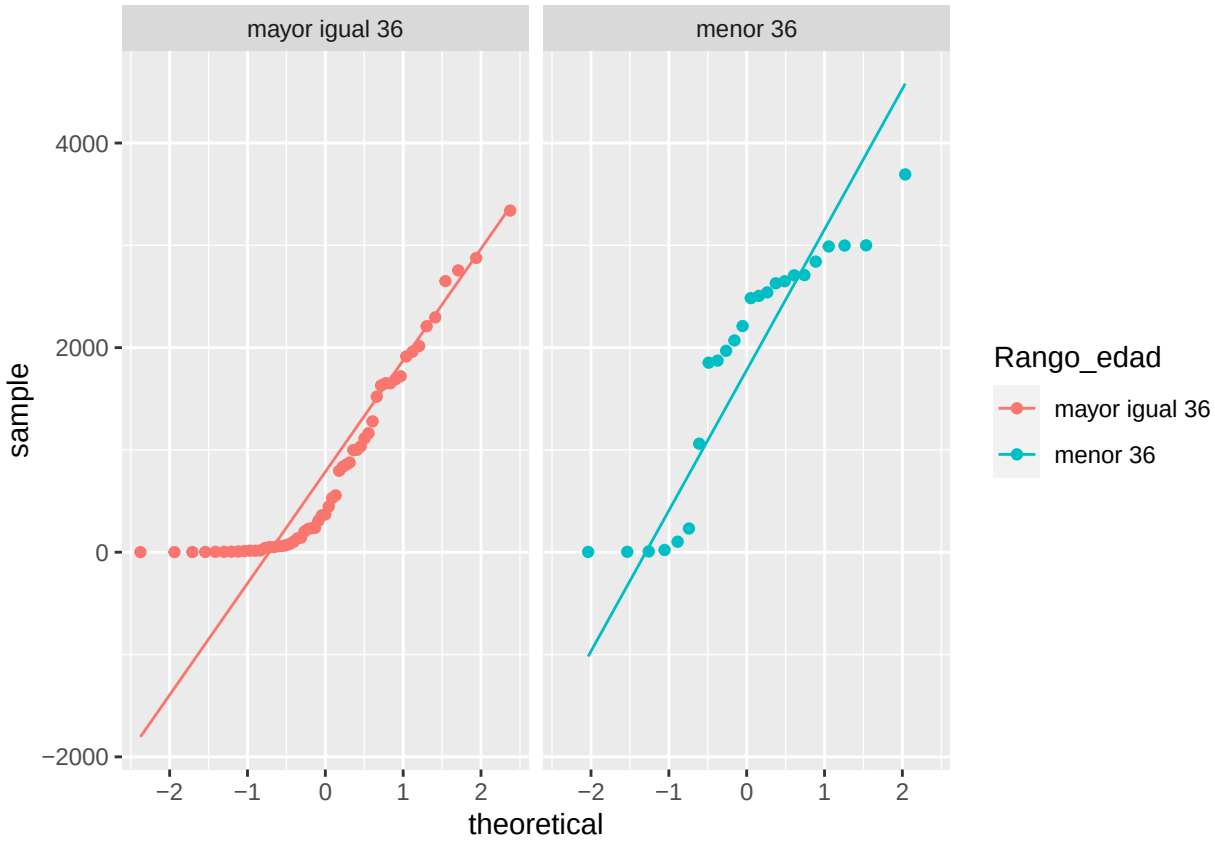
Es común encontrar mencionado que el test de Mann–Whitney–Wilcoxon compara medianas arreglamos el data set con las distribuciones en funcion de la edad y Dividimos la muestra en dos grupos el primer grupo pertenece a las personas mayores o iguales a 36 y el segundo grupo pertence al grupo menor de 36. se decidio usar estos valores debido a que la media y la mediana de la poblacion estaba entre 36 y 37.

por lo Tanto en esta prueba se compararan estos dos grupo para saber si existe una diferencia entre los dos grupos.

```
df_edad<-data_clean%>%filter(ID_EDAD!=0,ID_EDAD!=99)%>%group_by(ID_EDAD)%>%summarise(n=n())%>%mutate(Ra
```

si observamos los dos grupos recien creados en un grafico Q-Q podemos ver que su distribucion es muy paracida.por lo cual podemos aplicar esta prueba, que no requiere que los grupos tengan distribucion normal, pero si que tengan una asimetrica o parecida

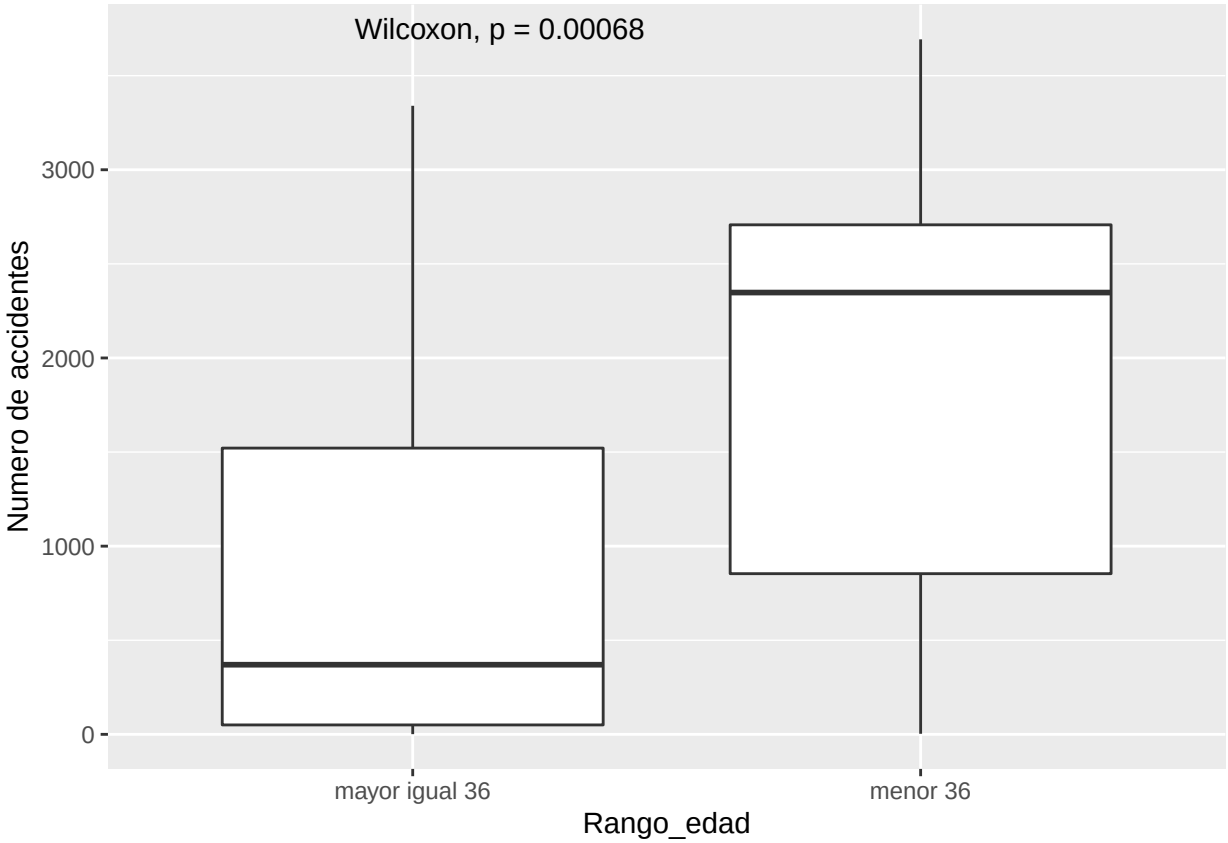
```
ggplot(df_edad, aes(sample = n, col = Rango_edad))+
  stat_qq()+
  stat_qq_line()+
  facet_grid(.~ Rango_edad)
```




```
#aplicando el test wilcox

(wilcox.test(n ~ Rango_edad, data = df_edad))
```

de acuerdo al p-value ibtenido por la prueba podemos rechaza la hipotesis nula la cual nos dice que no existe diferencia entre los dos grupos. por lo tanto se apoya la hipotesis alternativa y podemoss afirmar que existe una diferencia entre los grupos

```
#Utilizando la libreria ggpur podemos visulizar los resultados

ggplot(df_edad, aes(x = Rango_edad, y = n))+
  geom_boxplot()+
  stat_compare_means()+ylab("Numero de accidentes")
```



por los resultados obtenidos podemos observar que las medianas son diferentes y existe una diferencia considerebale entre los dos grupos de acuerdo al numero de accidentes siendo mayor en grupos de menos de 36 años. Este tipo de prueba es analoga al t_test con la diferencia que el t_tes solo se puede aplicar a distribuciones normales

- ¿Existe una diferencia entre grupos de edad en cuanto a la probabilidad de sufrir un accidente?

para probar esto utilizaremos una prueba Chi-Squared Goodness of Fit para determinar la proporcion entre los grupos de edades

```
#filtramos los datos eliminando las edades 0 y 99
mult<-data_clean%>%filter(ID_EDAD!=0,ID_EDAD!=99)%>%select(ID_EDAD)

#calculamos los quatiles de las edades para crear los grupos
(quantile(mult$ID_EDAD, prob=c(0,0.25,0.5,0.75,1)))
```

```
##   0%  25%  50%  75% 100%
##   12   28   36   45   95
```

```
#dividimos la muestra en grupos de acuerdo a los quantiles
mult<-mult%>%mutate(Rango_edad = case_when((ID_EDAD==12 | ID_EDAD<28) ~ "12 a <28",
                                           (ID_EDAD==28 | ID_EDAD<36) ~ "28 a <36",
                                           (ID_EDAD==36 | ID_EDAD<45) ~ "36 a <45",
                                           (ID_EDAD==45 | ID_EDAD<=95) ~ "45 a 95",
                                           TRUE ~ "other"))
```

```r
#filtramos los datos eliminando las edades 0 y 99
mult<-data_clean%>%filter(ID_EDAD!=0,ID_EDAD!=99)%>%select(ID_EDAD)

#calculamos los quatiles de las edades para crear los grupos
(quantile(mult$ID_EDAD, prob=c(0,0.25,0.5,0.75,1)))
```

```
##   0%  25%  50%  75% 100%
##   12   28   36   45   95
```

```r
#dividimos la muestra en grupos de acuerdo a los quantiles
mult<-mult%>%mutate(Rango_edad = case_when((ID_EDAD==12 | ID_EDAD<28) ~ "12 a <28",
                                           (ID_EDAD==28 | ID_EDAD<36) ~ "28 a <36",
                                           (ID_EDAD==36 | ID_EDAD<45) ~ "36 a <45",
                                           (ID_EDAD==45 | ID_EDAD<=95) ~ "45 a 95",
                                           TRUE ~ "other"))

#Conteo de los diferentes grupos
table(mult$Rango_edad)/dim(mult)[1]
```

```
##
##  12 a <28  28 a <36  36 a <45   45 a 95
## 0.2379519 0.2563881 0.2389043 0.2667557
```

de acuerdo a los conteos para ser muy cercanas los conteos no se aprecia una diferencia tan marcada para comprobar si los accidentes se dan en la misma proporcion no importando el rango de edad realizamos un Chi-Squared Goodness of Fit las hipotesis de la prueba son las siguientes: **ho todos los grupos tienen la misma probabilidad de sufrir accidentes ha los grupos tienen valores diferentes**

```r
(chisq.test( x = table(mult$Rango_edad) ))
```

```
##
##  Chi-squared test for given probabilities
##
## data:  table(mult$Rango_edad)
## X-squared = 215.51, df = 3, p-value < 2.2e-16
```

- Test binomial Basado en sexo: ¿Hombres o mujeres son mas propensos a sufrir accidentes?

Debido a que el sexo es una variable independiente ya que puede tomar unicamente el valor de hombre o mujer, esta puede ser analizada por medio de una distribucion binomial primero recordando los graficos se accidentes por tipo de sexo. hay varios registros que reportan “se fugo” como valor a la variable sexo, analizando la proporcion de estos

```r
# valores tenemos:

(data_clean%>%group_by(SEXO)%>%summarise(Accidentes=n())%>%mutate(Proporcion=(Accidentes/sum(Accidentes)
```

```
## # A tibble: 3 x 3
##   SEXO    Accidentes Proporcion
##   <chr>        <int>      <dbl>
## 1 Hombre      103402       75.6
## 2 Mujer        14420       10.5
## 3 Se fugó      18990       13.9
```

hay una clara diferencia en las proporciones entre hombres y mujeres, de igual forma existe muchos registros de Fuga, sin embargo para fines de la hipotesis se excluiran estos valores.

```
#Extraemos los datos necesarios exluyendo el valor "se fugo"
h_m<-data_clean%>%filter(SEXO!="Se fugó")%>%select(SEXO)
```

Primero empezaremos por analizar las proporciones Hipótesis para la primer prueba

H0: la proporción de Accidentes es igual (50%) para ambos sexos, por lo tanto p=0.5 Ha: la proporción de Accidentes no es igual entre ambos sexos p 0.5

```
#revisando la proporcion de datos nuevamente
(tabla <- table(h_m))
```

```
## h_m
## Hombre  Mujer
## 103402  14420
```

```
#realizando el test binomial a dos colas obtenemos :
(binom.test(x = tabla, alternative = "two.sided", conf.level = 0.95))
```

```
##
##  Exact binomial test
##
## data:  tabla
## number of successes = 103402, number of trials = 117822, p-value <
## 2.2e-16
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
##  0.8757271 0.8794783
## sample estimates:
## probability of success
##               0.877612
```

El test binomial rechaza la hipotesis nula en funcion del valor p, por lo tanto las distribuciones entre los dos grupos no son iguales

Realizando otra prueba binomial ahora considerando las siguientes hipotesis **H0: la proporción de Accidentes es igual (90%) para ambos sexos, por lo tanto p=0.9 Ha: la proporción de Accidentes no es igual entre ambos sexos p 0.9**

```
(binom.test(x=tabla,p=0.9, alternative="less"))
```

```
##
##  Exact binomial test
##
## data:  tabla
## number of successes = 103402, number of trials = 117822, p-value <
## 2.2e-16
## alternative hypothesis: true probability of success is less than 0.9
## 95 percent confidence interval:
##  0.0000000 0.8791798
## sample estimates:
## probability of success
##               0.877612
```

podriamos afirmar que la proporcion de hombres y mujeres que sufren accidentes es menor al 90% por lo tanto se rechaza la hipotesis nula

Podemos realizar otra prueba binomial por medio de otros tests como lo es la Prueba de hipótesis para la proporción p de Wald

Ahora que sabemos que las proporciones no son iguales podemos proceder a realizar una prueba de proporcion considerando las siguientes hipotesis **H0:p=0.90 H1:p<0.90** Con un nivel =0.05.

```
dim(h_m)[1]
```

```
## [1] 117822
```

```
z <- (103402/dim(h_m)[1] - 0.90) / sqrt(0.90 * (1 - 0.90) / dim(h_m)[1])
z  # Para obtener el valor del estadístico
```

```
## [1] -25.61577
```

```
#Para obtener el valor-P de la prueba debemos tener en cuenta el sentido en la hipótesis alternativa H1
(pnorm(q=z, lower.tail=T))  # Para obtener el valor-P
```

```
## [1] 5.090311e-145
```

en base a este valor podemos rechazar la hipotesis nula y decir que la proporcion es en efecto menor a 90

Prueba 2 de Pearson Para realizar la prueba 2 de Pearson se usa la función prop.test **H0:p=0.85 H1:p not equal 0.85** Con un nivel =0.05.

```
(prop.test(tabla, p=0.85, alternative="two.sided",
          conf.level=0.95, correct=FALSE))
```

```
##
##  1-sample proportions test without continuity correction
##
## data:  tabla, null probability 0.85
## X-squared = 704.55, df = 1, p-value < 2.2e-16
## alternative hypothesis: true p is not equal to 0.85
## 95 percent confidence interval:
##  0.8757283 0.8794710
## sample estimates:
##        p
## 0.877612
```

por lo tanto nuestra proporcion estra en tre 85 y 90%

- ¿Como ha cambiado la proprocion a lo largo de 10 años?

para saber como ha cambiado esta proporcion a lo largo de estos 10 años para esto utilizamos la funcion binom.test para realizar la prueba a las muestras filtradas por año

```r
tt<-data_clean%>%group_by(Fecha,ANIO)%>%filter(SEXO!="Se fugó")%>%select(SEXO,ANIO)
tt<-tt%>%group_by(ANIO)%>%select(SEXO)
#hacemos data set de acuerdo al año
tt<-group_split(tt)
#creamos una tabla para diferencial a hombres y mujeres
tt<-map(tt,table)
#aplicamos la prueba binomial a todos los años
(tt<-map(tt,binom.test,p=0.8, alternative="greater"))
```

```
## [[1]]
##
##  Exact binomial test
##
## data:  .x[[i]]
## number of successes = 11518, number of trials = 12431, p-value <
## 2.2e-16
## alternative hypothesis: true probability of success is greater than 0.8
## 95 percent confidence interval:
##  0.9225914 1.0000000
## sample estimates:
## probability of success
##              0.9265546
##
##
## [[2]]
##
##  Exact binomial test
##
## data:  .x[[i]]
## number of successes = 12228, number of trials = 13756, p-value <
## 2.2e-16
## alternative hypothesis: true probability of success is greater than 0.8
## 95 percent confidence interval:
##  0.884417 1.000000
## sample estimates:
## probability of success
##              0.8889212
##
##
## [[3]]
##
##  Exact binomial test
##
## data:  .x[[i]]
## number of successes = 13158, number of trials = 14693, p-value <
## 2.2e-16
## alternative hypothesis: true probability of success is greater than 0.8
## 95 percent confidence interval:
##  0.8912858 1.0000000
## sample estimates:
## probability of success
##              0.8955285
##
```

```
##
## [[4]]
##
##  Exact binomial test
##
## data:  .x[[i]]
## number of successes = 11776, number of trials = 13558, p-value <
## 2.2e-16
## alternative hypothesis: true probability of success is greater than 0.8
## 95 percent confidence interval:
##  0.8636963 1.0000000
## sample estimates:
## probability of success
##              0.8685647
##
##
## [[5]]
##
##  Exact binomial test
##
## data:  .x[[i]]
## number of successes = 10691, number of trials = 12263, p-value <
## 2.2e-16
## alternative hypothesis: true probability of success is greater than 0.8
## 95 percent confidence interval:
##  0.8667378 1.0000000
## sample estimates:
## probability of success
##              0.8718095
##
##
## [[6]]
##
##  Exact binomial test
##
## data:  .x[[i]]
## number of successes = 9546, number of trials = 11148, p-value < 2.2e-16
## alternative hypothesis: true probability of success is greater than 0.8
## 95 percent confidence interval:
##  0.8507186 1.0000000
## sample estimates:
## probability of success
##              0.8562971
##
##
## [[7]]
##
##  Exact binomial test
##
## data:  .x[[i]]
## number of successes = 8829, number of trials = 10169, p-value < 2.2e-16
## alternative hypothesis: true probability of success is greater than 0.8
## 95 percent confidence interval:
##  0.8625825 1.0000000
```

```
## sample estimates:
## probability of success
##               0.868227
##
##
## [[8]]
##
##  Exact binomial test
##
## data:  .x[[i]]
## number of successes = 9375, number of trials = 10840, p-value < 2.2e-16
## alternative hypothesis: true probability of success is greater than 0.8
## 95 percent confidence interval:
##  0.8593326 1.0000000
## sample estimates:
## probability of success
##              0.8648524
##
##
## [[9]]
##
##  Exact binomial test
##
## data:  .x[[i]]
## number of successes = 8546, number of trials = 9931, p-value < 2.2e-16
## alternative hypothesis: true probability of success is greater than 0.8
## 95 percent confidence interval:
##  0.8546911 1.0000000
## sample estimates:
## probability of success
##              0.8605377
##
##
## [[10]]
##
##  Exact binomial test
##
## data:  .x[[i]]
## number of successes = 7735, number of trials = 9033, p-value < 2.2e-16
## alternative hypothesis: true probability of success is greater than 0.8
## 95 percent confidence interval:
##  0.8500935 1.0000000
## sample estimates:
## probability of success
##              0.8563047
```

en todos los casos se rechaza la hipotesis nula de la probabilidad igual a 0.8 y se apoya la hipotesis de la proporcion mayo a 0.8 (hipotesus nula)

```
paste((map(tt,`[[`,3)),"p-value")
```

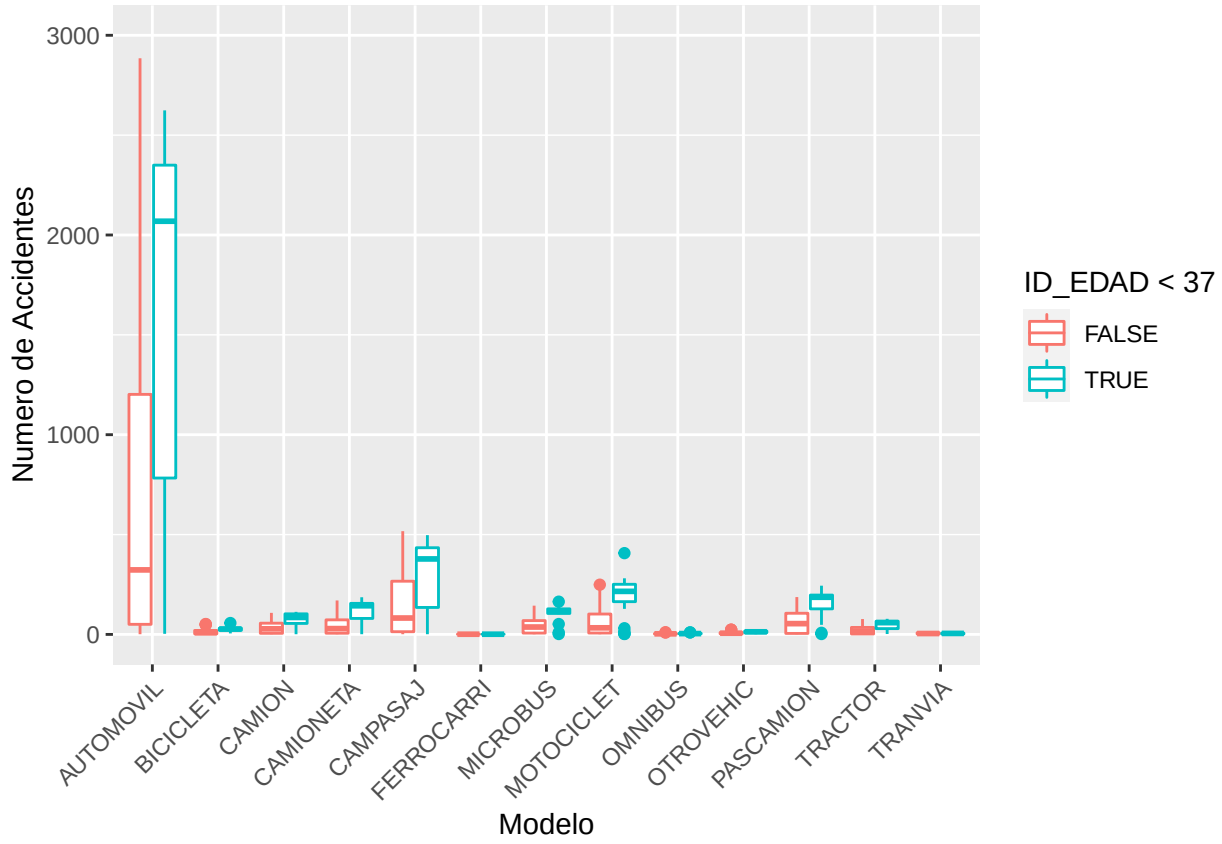
```
##  [1] "0 p-value"                     "7.39075435253127e-172 p-value"
##  [3] "4.17851455021987e-214 p-value" "5.6618781652738e-98 p-value"
```

```
##  [5] "9.43072698132542e-98 p-value"  "3.69895120095109e-54 p-value"
##  [7] "2.97670617285384e-73 p-value"  "3.06671927961991e-70 p-value"
##  [9] "4.29066753886047e-56 p-value"  "2.86249686096671e-44 p-value"
```

- ¿Existe una relacion entre Modelo del vehiculo y Edad?

```
#En el analis de datos se genero la siguiente grafica
md2<-gather(data_clean,Modelo,key = "Modelo",value = "Num_accidentes")
md2<-md2%>%filter(ID_EDAD!=0,Num_accidentes!=0,ID_EDAD!=99)
md2<-md2%>%group_by(Fecha,Modelo,ID_EDAD)%>%summarise(n=n())
md2<-md2%>%group_by(Modelo,ID_EDAD)%>%summarise(n=n())
```

```
ggplot(data = md2, aes(Modelo,n)) +
  geom_boxplot(aes(colour = ID_EDAD < 37))+ ylim(range(0:3000))+ylab("Numero de Accidentes")+
  theme(axis.text.x = element_text(angle=45, hjust=1))
```



En el grafico podemos ver como existe una pequeña relacion entre el tipo de vehiculo y el sexo del coductor por lo cual podremos a prueba esta relacion. Por lo cual realizaremos una prueba independencia **H0: The two variables are independent. H1: The two variables relate to each other.**

```
#nombre de los vehiculos para la variable modelos
md2<-md2%>%mutate(Rango_edad = case_when((ID_EDAD==12 | ID_EDAD<28) ~ "12 a <28",
                                         (ID_EDAD==28 | ID_EDAD<36) ~ "28 a <36",
                                         (ID_EDAD==36 | ID_EDAD<45) ~ "36 a <45",
                                         (ID_EDAD==45 | ID_EDAD<=95) ~ "45 a 95",
```

```
                                          TRUE ~ "other"))
(table(md2$Modelo, md2$Rango_edad))

(chisq.test(table(md2$Modelo, md2$Rango_edad), simulate.p.value = TRUE))
```

De acuerdo a lo obtenido no se puede rechazar la hipotesis nula por lo cual podemos decir que es probable que exista una independencia entre las variables edad y modelo de carro

ANALISIS DE REGRESION

Se analiza el dataset para empezar con la construcción de los análisis de regresión que nos permitan evaluar los diferentes predictores del número de accidentes y otras variables de respuesta.

```
#View(data_clean)
count(data_clean)
```

```
##        n
## 1 136812
```

Se utilizo el método de selección Forward, para la introducción de las variables dentro de los ajustes multivariables. Debido a que existe un sesgo importante de los datos (SEXO y EDAD) se considero evaluar SEXO y EDAD en la mayoría de nuestros análisis de regresión por separado.

La metodología para todos y cada uno de los análisis de regresión fue la limpieza y estandarización de los data sets. Al tener varias variables categóricas y dada la naturaleza del data set original se requirió agrupar los datos con el fin de crear una variable continua que permitiera llevar a cabo el análisis de regresión multivariable

```
V_regresion1 <- data_clean %>%
  group_by(Municipio,ID_HORA, DIASEMANA) %>%
  summarize(Municipio,ID_HORA, DIASEMANA)

gripi <- count(V_regresion1) %>%
  mutate(DIASEMANA = str_to_upper(DIASEMANA)) %>%
  mutate(DIASEMANA = str_replace(DIASEMANA,"MIÉRCOLES","MIERCOLES")) %>%
  mutate(DIASEMANA = str_replace(DIASEMANA,"SÁBADO","SABADO"))

modu <- lm(n ~ Municipio + ID_HORA + DIASEMANA , data=gripi)
modul <- lm(n ~ Municipio*ID_HORA*DIASEMANA, data=gripi)

(summary(modu))
```

```
##
## Call:
## lm(formula = n ~ Municipio + ID_HORA + DIASEMANA, data = gripi)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -70.014  -9.716   0.471   9.552  82.821
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     31.61080    1.79897  17.572  < 2e-16 ***
```

```
## MunicipioAzcapotzalco           -18.35714    2.04479  -8.978  < 2e-16 ***
## MunicipioBenito Juárez           13.00000    2.04479   6.358 2.40e-10 ***
## MunicipioCoyoacán                12.01786    2.04479   5.877 4.69e-09 ***
## MunicipioCuajimalpa de Morelos  -39.92654    2.04786 -19.497  < 2e-16 ***
## MunicipioCuauhtémoc              51.12500    2.04479  25.003  < 2e-16 ***
## MunicipioGustavo A. Madero       47.59524    2.04479  23.276  < 2e-16 ***
## MunicipioIztacalco              -20.67857    2.04479 -10.113  < 2e-16 ***
## MunicipioIztapalapa              50.67262    2.04479  24.781  < 2e-16 ***
## MunicipioLa Magdalena Contreras -43.65649    2.04786 -21.318  < 2e-16 ***
## MunicipioMiguel Hidalgo          18.95833    2.04479   9.272  < 2e-16 ***
## MunicipioMilpa Alta             -49.36493    2.05724 -23.996  < 2e-16 ***
## MunicipioTláhuac                -42.51159    2.04786 -20.759  < 2e-16 ***
## MunicipioTlalpan                 -3.58333    2.04479  -1.752  0.07982 .
## MunicipioVenustiano Carranza      5.76190    2.04479   2.818  0.00487 **
## MunicipioXochimilco             -36.42857    2.04479 -17.815  < 2e-16 ***
## ID_HORA                           1.32348    0.05233  25.291  < 2e-16 ***
## DIASEMANAJUEVES                   8.68578    1.35339   6.418 1.63e-10 ***
## DIASEMANALUNES                    8.39490    1.35339   6.203 6.41e-10 ***
## DIASEMANAMARTES                   9.30749    1.35428   6.873 7.82e-12 ***
## DIASEMANAMIERCOLES                8.28255    1.35339   6.120 1.07e-09 ***
## DIASEMANASABADO                   6.31250    1.35250   4.667 3.20e-06 ***
## DIASEMANAVIERNES                 11.78448    1.35428   8.702  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18.74 on 2658 degrees of freedom
## Multiple R-squared:  0.776,  Adjusted R-squared:  0.7741
## F-statistic: 418.4 on 22 and 2658 DF,  p-value: < 2.2e-16
```

```
(anova(modu,modul))
```

```
## Analysis of Variance Table
##
## Model 1: n ~ Municipio + ID_HORA + DIASEMANA
## Model 2: n ~ Municipio * ID_HORA * DIASEMANA
##   Res.Df    RSS  Df Sum of Sq      F    Pr(>F)
## 1   2658 933541
## 2   2457 734847 201    198694 3.3052 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Análisis de Coeficientes: Se toma como referencia por default a la alcaldía Álvaro Obregón, Observamos que los coeficientes de los factores de predicción del número de accidente durante los años 2010 - 2019 indican que la Alcaldía Cuauhtemóc tendrá más accidente que cualquiera de las otras alcaldías, de la misma manera observamos que entre más tarde sea contemplando un horario de 0-24 horas, existirá un aumento en el número de choques, finalmente el día viernes se infiere será el día con mayor número de choques.

Significancia: Unicamente los coeficientes calculados para los factores de Alcaldía Tlalpán y Venustiano Carranza no fueron significativos. Valor de R^2 por encima de 0.5, y valor de P-Value del ajuste menor a alpha.

Interacción: Calculamos la interacción entre los factores y encontramos que la interacción si es significativa por lo que la combinación de la Alcaldía, la hora y el día de la semana si tendrá un impacto en la cantidad de accidentes reportados.

```r
v_regresion2 <- data_clean %>%
  group_by(ID_EDAD, SEXO, AUTOMOVIL) %>%
  summarize(ID_EDAD,SEXO, AUTOMOVIL)

grip <- count(v_regresion2) %>%
  filter(ID_EDAD<90, ID_EDAD > 15, SEXO != "Se fugó")
automovil <- grip %>%
  group_by(ID_EDAD,SEXO) %>%
  filter(AUTOMOVIL>0) %>%
  summarize(AUTOMOVIL = sum(n))

#View(automovil)

modi <- lm(AUTOMOVIL ~ ID_EDAD + SEXO, data=automovil)
modil <- lm(AUTOMOVIL ~ ID_EDAD * SEXO, data=automovil)
(summary(modi))
```

```
##
## Call:
## lm(formula = AUTOMOVIL ~ ID_EDAD + SEXO, data = automovil)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1535.64  -343.14   -26.58   255.61  1477.49
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1910.78     107.93   17.70   <2e-16 ***
## ID_EDAD       -18.51       1.79  -10.34   <2e-16 ***
## SEXOMujer    -807.27      75.57  -10.68   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 456.4 on 143 degrees of freedom
## Multiple R-squared:  0.6017, Adjusted R-squared:  0.5961
## F-statistic:   108 on 2 and 143 DF,  p-value: < 2.2e-16
```

```r
(anova(modi,modil))
```

```
## Analysis of Variance Table
##
## Model 1: AUTOMOVIL ~ ID_EDAD + SEXO
## Model 2: AUTOMOVIL ~ ID_EDAD * SEXO
##   Res.Df      RSS Df Sum of Sq      F    Pr(>F)
## 1    143 29789534
## 2    142 18301286  1  11488248 89.138 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Análisis de Coeficientes:

Encontramos tras el ajuste lineal que a medida que aumenta la edad la cantidad de choques se reduce, el sexo como dicho anteriormente es un factor determinante para la cantidad de choques reportados.

Significancia:

Todos los valores fueron significativos con valor de p-value muy por debajo del nivel de alpha, El p-value del modelo también nos indica que es significativo

Interacción:

Si existe significancia, por lo que si existe una interacción entre el valor de la edad y el sexo, para la determinación de choques con automóvil

```
v_regresion3 <- data_clean %>%
  group_by(ID_EDAD, SEXO, MOTOCICLET) %>%
  summarize(ID_EDAD,SEXO, MOTOCICLET)

gript <- count(v_regresion3) %>%
  filter(ID_EDAD<90, ID_EDAD > 15, SEXO != "Se fugó")


MOTORAD <- gript %>%
  group_by(ID_EDAD,SEXO) %>%
  filter(MOTOCICLET>0) %>%
  summarize(MOTOCICLET = sum(n))

#View(MOTORAD)


modiN <- lm(MOTOCICLET ~ ID_EDAD + SEXO, data=MOTORAD)
modilN <- lm(MOTOCICLET ~ ID_EDAD * SEXO, data=MOTORAD)

(summary(modiN))
```

```
##
## Call:
## lm(formula = MOTOCICLET ~ ID_EDAD + SEXO, data = MOTORAD)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -165.668  -35.143   -8.748   29.738  223.728
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 215.6917    13.1819   16.36   <2e-16 ***
## ID_EDAD      -2.3140     0.2312  -10.01   <2e-16 ***
## SEXOMujer   -98.9634     9.0482  -10.94   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 50.77 on 126 degrees of freedom
## Multiple R-squared:  0.6072, Adjusted R-squared:  0.601
## F-statistic: 97.38 on 2 and 126 DF,  p-value: < 2.2e-16
```

```
(anova(modiN,modilN))
```

```
## Analysis of Variance Table
```

```
##
## Model 1: MOTOCICLET ~ ID_EDAD + SEXO
## Model 2: MOTOCICLET ~ ID_EDAD * SEXO
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    126 324747
## 2    125 194759  1    129988 83.429 1.474e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Análisis de Coeficientes:

Corroboramos tras el ajuste lineal que a medida que aumenta la edad, la cantidad de choques se reduce, el sexo como dicho anteriormente es un factor determinante para la cantidad de choques reportados.

Significancia:

Todos los valores fueron significativos con valor de p-value muy por debajo del nivel de alpha, El p-value del modelo también nos indica que es significativo

Interacción:

Si existe significancia, por lo que si existe una interacción entre el valor de la edad y el sexo, para la determinación de choques con motocicleta

#Prueba de Hipótesis

Se efectúo una prueba de hipótesis con la finalidad de mitigar el sesgo relacionado al tipo de vehículo. Para ello se utilizo la información de otro data set del INEGI, con lo cual se encontró la cantidad de parque vehicular de las variables de Autos y Motocicletas durante los años 2010-2019. De esta manera se midió la diferencia de proporciones entre los dos vehículos con mayor número de reportes de accidentes registrados. "https://www.inegi.org.mx/contenidos/programas/vehiculosmotor/datosabiertos/vmrc_anual_csv.zip"

```
Prueba_de_motosautos <- prop.test(x = c(122172, 9703), n = c(4714909, 282450), alternative="greater")
Prueba_de_motosautos
```

```
##
##  2-sample test for equality of proportions with continuity correction
##
## data:  c(122172, 9703) out of c(4714909, 282450)
## X-squared = 738.71, df = 1, p-value = 1
## alternative hypothesis: greater
## 95 percent confidence interval:
##  -0.009019416  1.000000000
## sample estimates:
##     prop 1     prop 2
## 0.02591185 0.03435298
```

Los resultados de la prueba de hipótesis nos arrojan que la proporción de choques de autos comparada con el parque vehicular de este tipo de vehículo es menor que la proporción de choques en moto comparada con el parque vehicular de este. De esta manera podemos inferir que efectivamente el tipo de vehiculo tendrá un impacto en la ocurrencia o probalidad de colisión siendo de acuerdo a esta hipótesis las motos un vehículo mas propicio a choque que los autos.

#Regresión Lineal

Finalmente se efectuaron dos regresiones lineales más con el fin de buscar los fcatores más determinantes para los casos de accidentes con fuga de conductor y los accidentes de tipacidad fatal (Algún involucrado fallecido), a continuación se evalúan los resultados

```r
v_regresion4 <- data_clean %>%
  group_by(Municipio,SEXO,DIASEMANA,ID_HORA, CLASACC, CINTURON) %>%
  summarize(Municipio,SEXO,DIASEMANA,ID_HORA, CLASACC, CINTURON)

gripig <- count(v_regresion4) %>%
  mutate(DIASEMANA = str_to_upper(DIASEMANA)) %>%
  mutate(DIASEMANA = str_replace(DIASEMANA,"MIÉRCOLES","MIERCOLES")) %>%
  mutate(DIASEMANA = str_replace(DIASEMANA,"SÁBADO","SABADO")) %>%
  filter(SEXO == "Se fugó") %>%
  group_by(Municipio)

#View(gripig)

modulik <- lm(n ~ Municipio + ID_HORA + DIASEMANA , data=gripig)
modulilkt <- lm(n ~ Municipio*ID_HORA*DIASEMANA, data=gripig)
(summary(modulik))
```

```
##
## Call:
## lm(formula = n ~ Municipio + ID_HORA + DIASEMANA, data = gripig)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.9513 -2.2451 -0.5606  1.5037 21.9195
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     2.937937   0.263992  11.129  < 2e-16 ***
## MunicipioAzcapotzalco          -0.157926   0.302438  -0.522 0.601574
## MunicipioBenito Juárez          1.139180   0.296622   3.841 0.000124 ***
## MunicipioCoyoacán               0.838780   0.289190   2.900 0.003744 **
## MunicipioCuajimalpa de Morelos -1.512195   0.322910  -4.683 2.91e-06 ***
## MunicipioCuauhtémoc             3.935574   0.276915  14.212  < 2e-16 ***
## MunicipioGustavo A. Madero      3.228063   0.272049  11.866  < 2e-16 ***
## MunicipioIztacalco             -1.190834   0.317469  -3.751 0.000178 ***
## MunicipioIztapalapa             3.232402   0.276521  11.690  < 2e-16 ***
## MunicipioLa Magdalena Contreras -2.171562   0.456154  -4.761 1.99e-06 ***
## MunicipioMiguel Hidalgo         1.376649   0.298515   4.612 4.10e-06 ***
## MunicipioMilpa Alta            -1.683421   0.343456  -4.901 9.85e-07 ***
## MunicipioTláhuac               -1.863674   0.362417  -5.142 2.83e-07 ***
## MunicipioTlalpan               -0.675595   0.295916  -2.283 0.022473 *
## MunicipioVenustiano Carranza    0.053343   0.289607   0.184 0.853872
## MunicipioXochimilco            -1.496418   0.333797  -4.483 7.54e-06 ***
## ID_HORA                         0.036032   0.007902   4.560 5.25e-06 ***
## DIASEMANAJUEVES                -0.050059   0.199860  -0.250 0.802234
## DIASEMANALUNES                 -0.225438   0.198808  -1.134 0.256875
## DIASEMANAMARTES                -0.059447   0.202840  -0.293 0.769480
## DIASEMANAMIERCOLES             -0.181182   0.200485  -0.904 0.366193
## DIASEMANASABADO                 0.285091   0.194019   1.469 0.141795
## DIASEMANAVIERNES                0.272986   0.197565   1.382 0.167116
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 3.624 on 4528 degrees of freedom
## Multiple R-squared:  0.2228, Adjusted R-squared:  0.219
## F-statistic:    59 on 22 and 4528 DF,  p-value: < 2.2e-16
```

```
(anova(modulik,modulilkt))
```

```
## Analysis of Variance Table
##
## Model 1: n ~ Municipio + ID_HORA + DIASEMANA
## Model 2: n ~ Municipio * ID_HORA * DIASEMANA
##   Res.Df   RSS  Df Sum of Sq    F Pr(>F)
## 1   4528 59483
## 2   4327 57818 201    1665.1 0.62      1
```

Análisis de Coeficientes:

Se reportó a la Alcaldía de Iztapalapa, seguida por Alcaldía Cuahtémoc y Gustavo A. Madero como las alcaldías con mayor tendencia a reporte de fuga de conductor. La hora es significativa con un valor de pvalue muy por debajo del valor de alpha y presenta una correlación directa, es decir que conforme las horas del día avanzan incrementa el número de choques con conductor que se fuga. El día Miércoles se infiere y espera sea el día con menor número de accidentes en el que conductar se fuga

Significancia:

Unicamente los valores de la alcaldía de Azcapotzalco y Tlalpan, aunados al día Viernes y Jueves no son significativos. El Modelo con una valor muy pequeño de p-value rectifica que es significativo

Interacción:

Se comprueba la interacción entre las variables ajustadas.

```
v_regresion5 <- data_clean %>%
  group_by(Municipio,ALIENTO, CINTURON, CLASACC) %>%
  summarize(Municipio, ALIENTO, CINTURON, CLASACC)

gripigb <- count(v_regresion5) %>%
  filter(CLASACC == "Fatal")  %>%
  filter(CINTURON != "Se ignora" ) %>%
  filter(ALIENTO != "Se ignora")

#View(gripigb)

modulikur <- lm(n ~ Municipio + ALIENTO + CINTURON, data=gripigb)
modulilktr <- lm(n ~ Municipio*ALIENTO*CINTURON, data=gripigb)
(summary(modulikur))
```

```
##
## Call:
## lm(formula = n ~ Municipio + ALIENTO + CINTURON, data = gripigb)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -22.0864  -2.5021  -0.9928   3.6522  17.5864
##
## Coefficients:
```

```
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     24.1636     4.3518   5.553 3.29e-06 ***
## MunicipioAzcapotzalco          -13.7212     6.0540  -2.266  0.02990 *
## MunicipioBenito Juárez          -4.5000     5.5761  -0.807  0.42526
## MunicipioCoyoacán               -4.5000     5.5761  -0.807  0.42526
## MunicipioCuajimalpa de Morelos -12.3879     6.0540  -2.046  0.04853 *
## MunicipioCuauhtémoc             -6.7500     5.5761  -1.211  0.23442
## MunicipioGustavo A. Madero      19.2500     5.5761   3.452  0.00151 **
## MunicipioIztacalco              -4.3879     6.0540  -0.725  0.47354
## MunicipioIztapalapa              0.2788     6.0540   0.046  0.96354
## MunicipioLa Magdalena Contreras -18.0473     6.9298  -2.604  0.01355 *
## MunicipioMiguel Hidalgo         -12.7212     6.0540  -2.101  0.04310 *
## MunicipioMilpa Alta             -11.7212     6.0540  -1.936  0.06121 .
## MunicipioTláhuac                -15.1163     6.9298  -2.181  0.03617 *
## MunicipioTlalpan                 -1.2500     5.5761  -0.224  0.82396
## MunicipioVenustiano Carranza    -11.0545     6.0540  -1.826  0.07665 .
## MunicipioXochimilco             -13.7212     6.0540  -2.266  0.02990 *
## ALIENTOSí                       -13.0947     2.3529  -5.565 3.16e-06 ***
## CINTURONSí                       -7.2326     2.3529  -3.074  0.00415 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.886 on 34 degrees of freedom
## Multiple R-squared:  0.7169, Adjusted R-squared:  0.5753
## F-statistic: 5.065 on 17 and 34 DF,  p-value: 2.921e-05
```

```
(anova(modulikur,modulilktr))
```

```
## Analysis of Variance Table
##
## Model 1: n ~ Municipio + ALIENTO + CINTURON
## Model 2: n ~ Municipio * ALIENTO * CINTURON
##   Res.Df    RSS Df Sum of Sq F Pr(>F)
## 1     34 2114.3
## 2      0    0.0 34    2114.3
```

Análisis de Coeficientes:

La alcaldía con menor proyección a albergar accidentes fatales es Magdalena Contreras, mientras que la alcaldía Gustavo Madero es la de mayor proyección de accidentes fatales. La variable de aliento alcohólico presenta una correlación inversamente proporcional (Esto debido a un sesgo en la información) Finalmente la variable de uso del cinturón si es significativa con un valor de Pvalue de .0004 y será también efectiva para la reducción del incremento en el número de choques fatales

Significancia:

EL valor de R2 es alto (>.7) por lo que el ajuste es bueno, y las variables Cinturónsí y alientosí (variables dummy creeadas por R) fueron significativas

Interacción:

La interacción es significativa entre las variables utilizadas para el ajuste

Analisis de Tiempo

Se realizo la prediccion del numero de acccidentes para dos años, mediante el uso de series de tiempo. Para lo cual se ocupo un modelo ARIMA

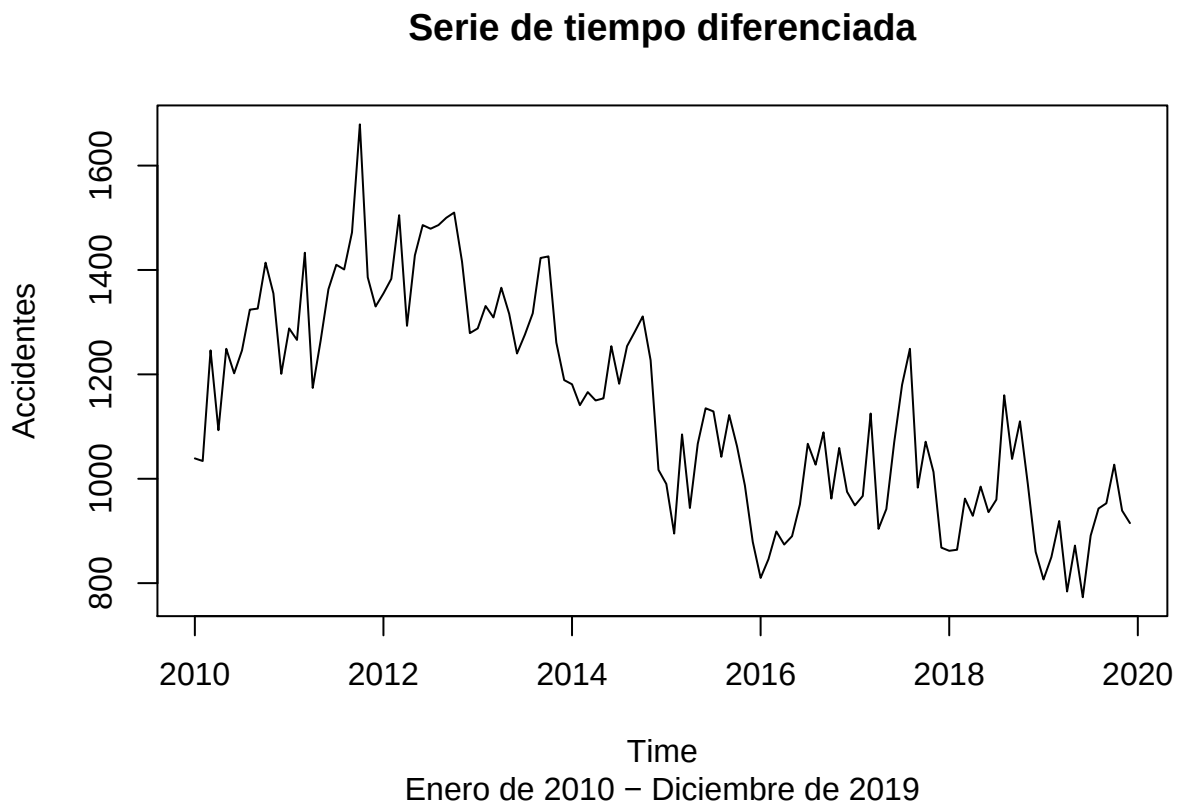
Prediccion de accidentes para el 2020 al 2022

```
#se utilizo la paqueteria Forecast para comparar modelos
#importamos los datos  de tiempo
#Agrupamos el numero de accidentes en funcion del mes
time<-data_clean%>%group_by(year(Fecha),month(Fecha))%>%summarise(Num_acc=n())
#creando la serie de tiempo en funcion del mes
time.ts <- ts(time$Num_acc, start =2010,freq = 12)
(time.ts)
```

```
##       Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec
## 2010 1039 1034 1246 1093 1249 1202 1246 1324 1326 1414 1355 1201
## 2011 1288 1266 1433 1174 1264 1363 1410 1401 1472 1679 1386 1330
## 2012 1355 1383 1505 1293 1428 1486 1479 1486 1500 1510 1416 1279
## 2013 1288 1331 1309 1366 1316 1240 1276 1317 1423 1426 1261 1189
## 2014 1181 1141 1166 1150 1154 1254 1182 1254 1282 1311 1227 1017
## 2015  990  895 1085  944 1067 1135 1129 1042 1122 1062  987  879
## 2016  810  846  899  874  890  951 1067 1027 1089  962 1059  975
## 2017  949  967 1125  904  942 1070 1180 1249  983 1071 1013  868
## 2018  862  864  962  929  985  936  960 1160 1038 1110  990  860
## 2019  807  850  919  784  872  773  891  943  953 1027  939  915
```

Visulizamos la serie de tiempo obtenida

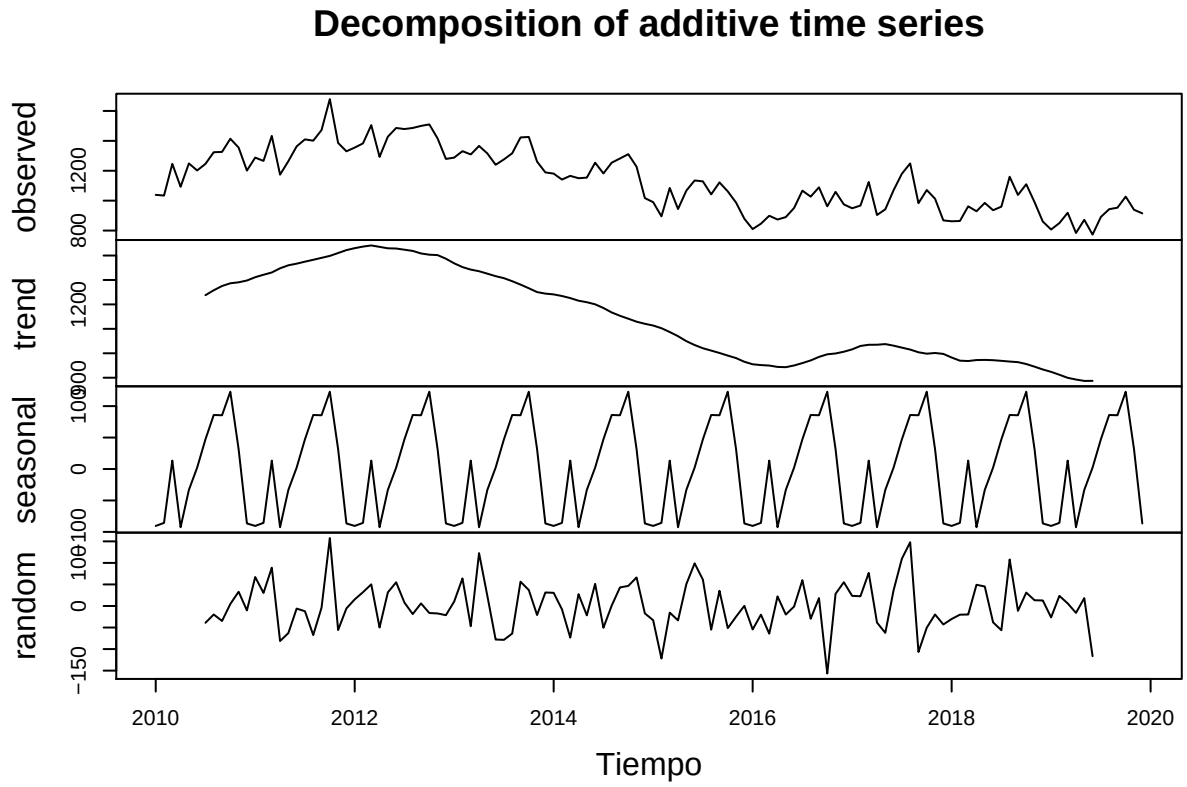
```
#grafico de la serie de tiempo
plot(time.ts,
     main = "Serie de tiempo diferenciada",
     ylab = "Accidentes",
     sub = "Enero de 2010 - Diciembre de 2019")
```



Analizamos la serie de tiempo de acuerdo al modelo Additivo y Multiplicativo para buscar estacionalidad

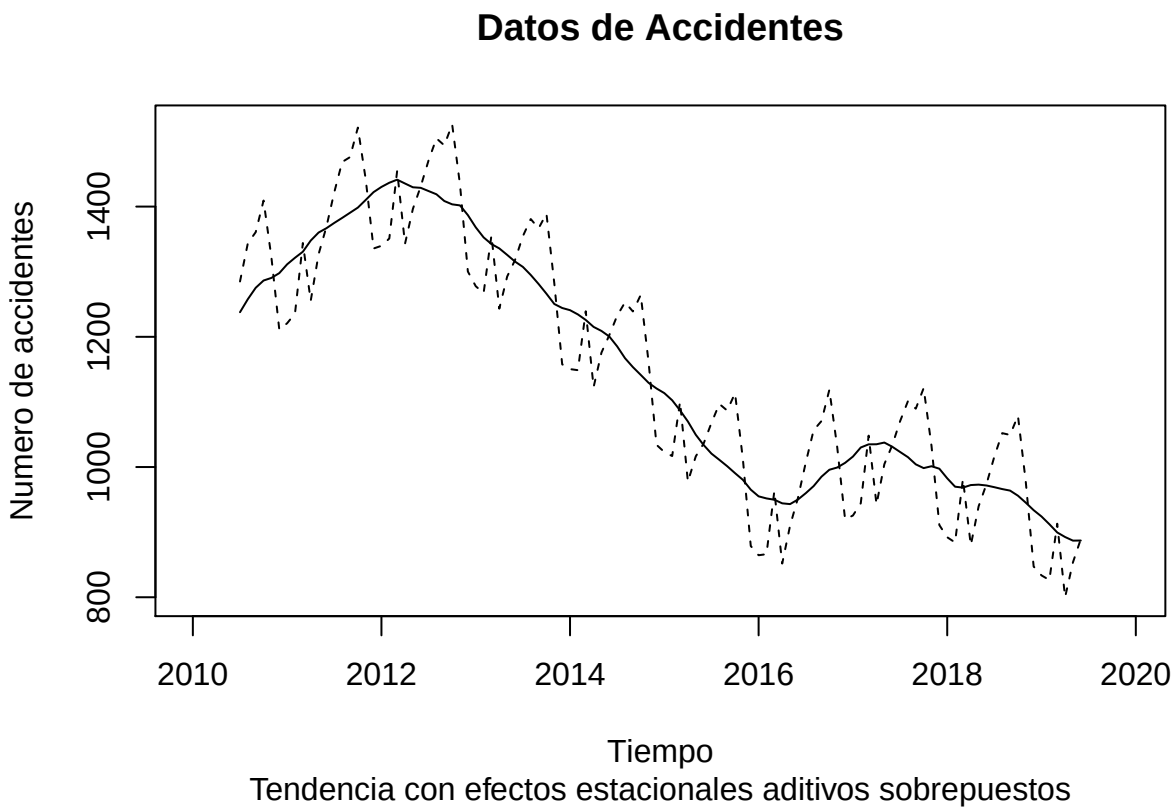
Modelo aditivo

```
acc.decom.A <- decompose(time.ts)
plot(acc.decom.A, xlab = "Tiempo",
     sub = "Descomposición de los datos de accidentes")
```



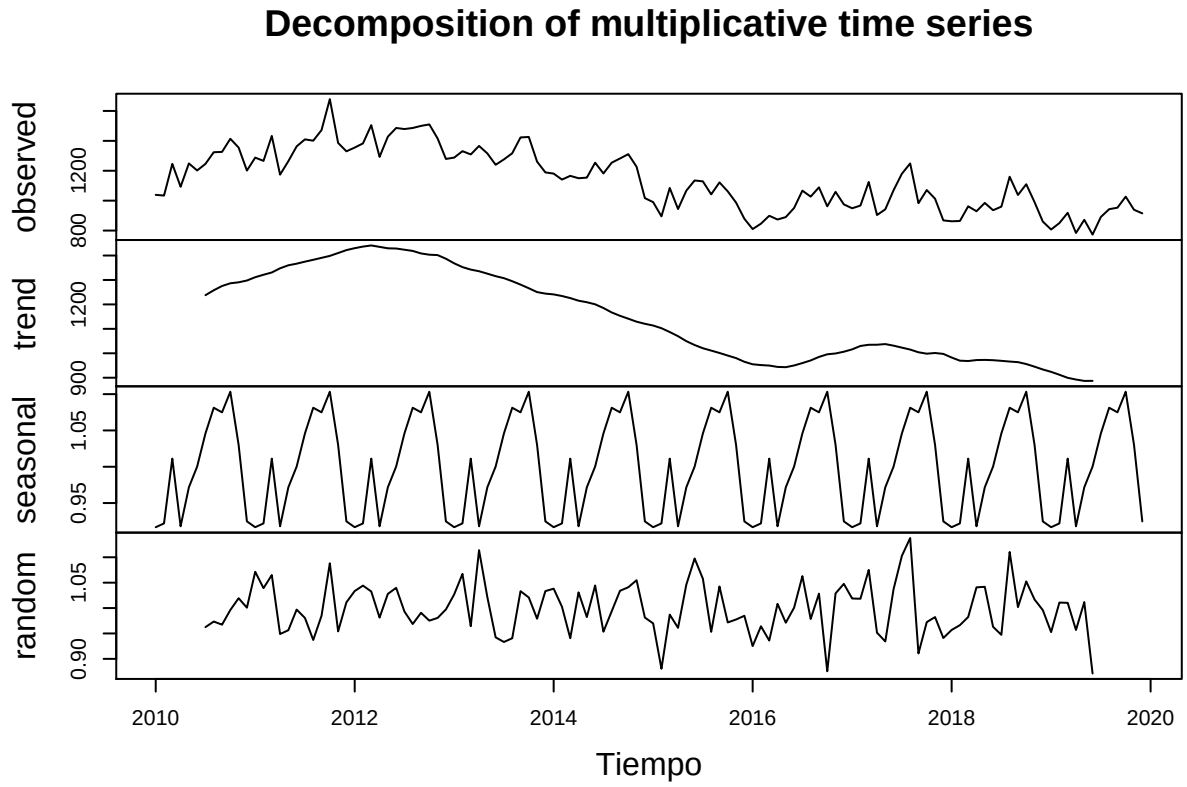

```
#Componentes
Tendencia <-acc.decom.A$trend
Estacionalidad <- acc.decom.A$seasonal
Aleatorio <- acc.decom.A$random
```

```
ts.plot(cbind(Tendencia, Tendencia + Estacionalidad),
        xlab = "Tiempo", main = "Datos de Accidentes",
        ylab = "Numero de accidentes", lty = 1:2,
        sub = "Tendencia con efectos estacionales aditivos sobrepuestos")
```



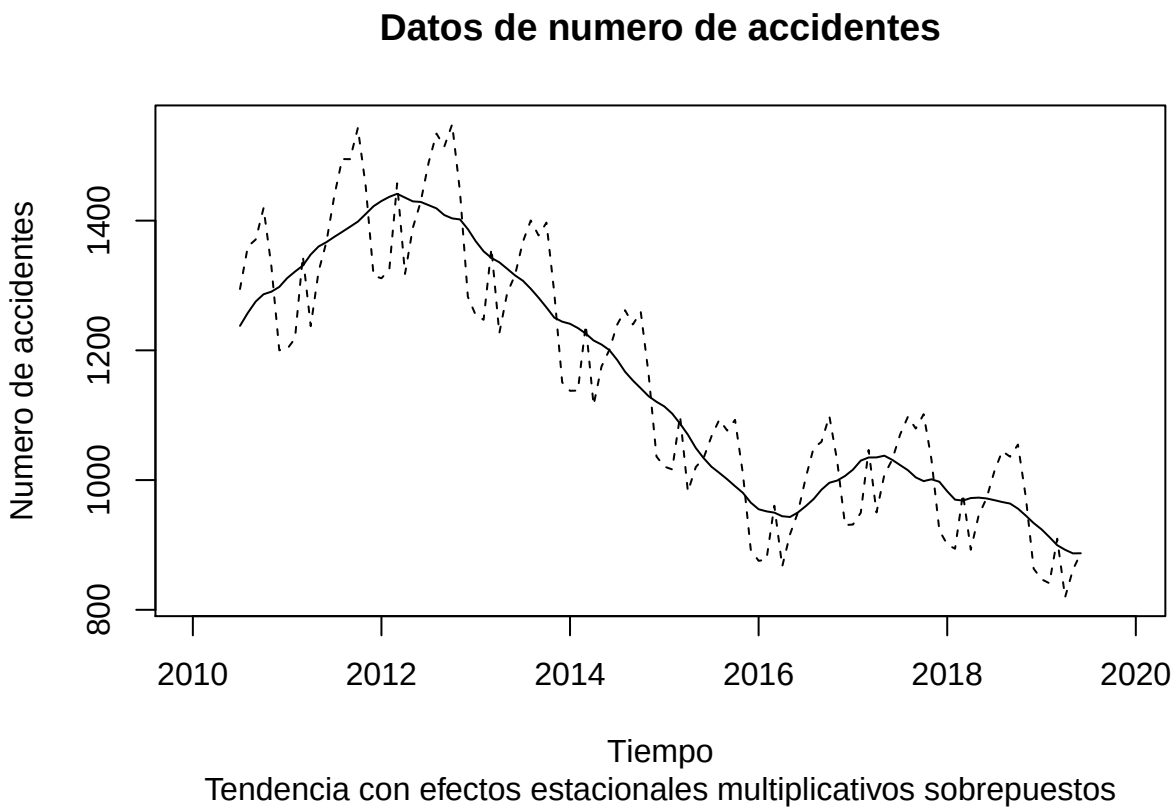
Modelo Multiplicativo

```
acc.decom.M <- decompose(time.ts, type = "mult")

plot(acc.decom.M, xlab = "Tiempo",
     sub = "Descomposición de los datos de accidentes")
```



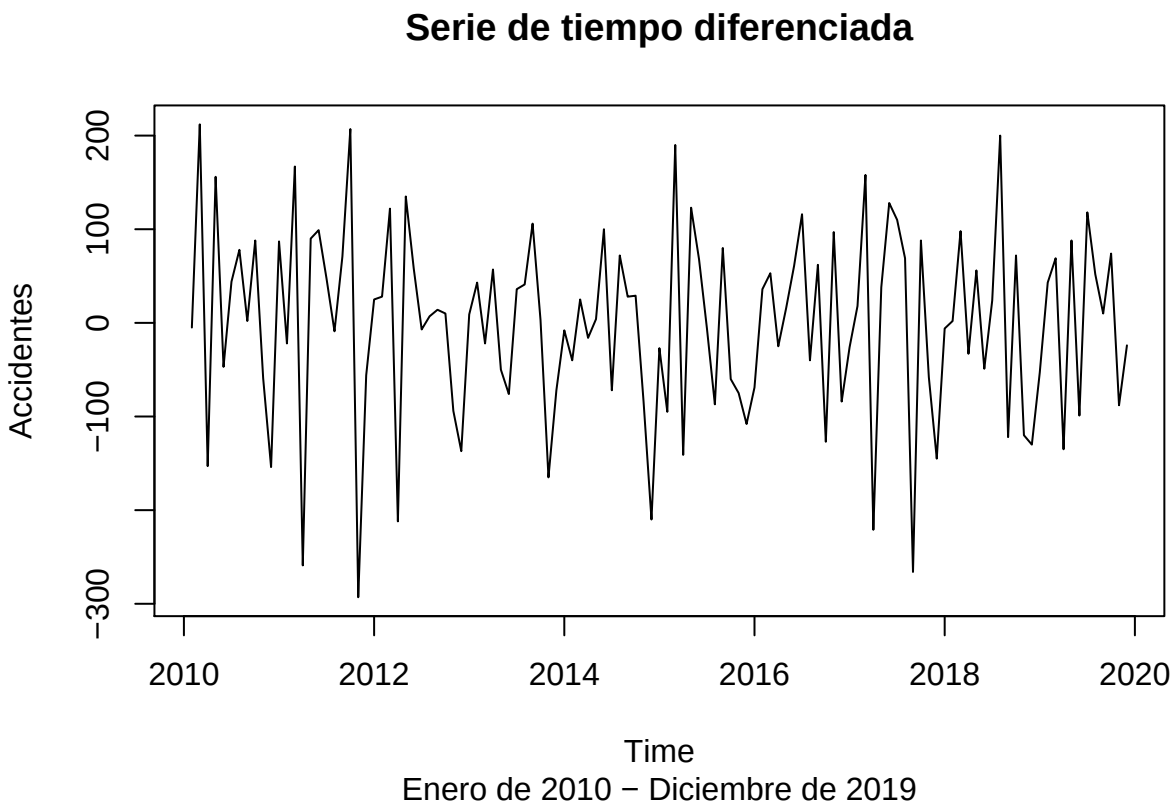

```
#Componentes
Trend <- acc.decom.M$trend
Seasonal <- acc.decom.M$seasonal
Random <- acc.decom.M$random
```

```
ts.plot(cbind(Trend, Trend*Seasonal), xlab = "Tiempo", main = "Datos de numero de accidentes",
        ylab = "Numero de accidentes", lty = 1:2,
        sub = "Tendencia con efectos estacionales multiplicativos sobrepuestos")
```



Como podemos notar esta serie de tiempo no tiene estacionalidad y una varianza constante por lo cual es necesario hacer una transformacion.

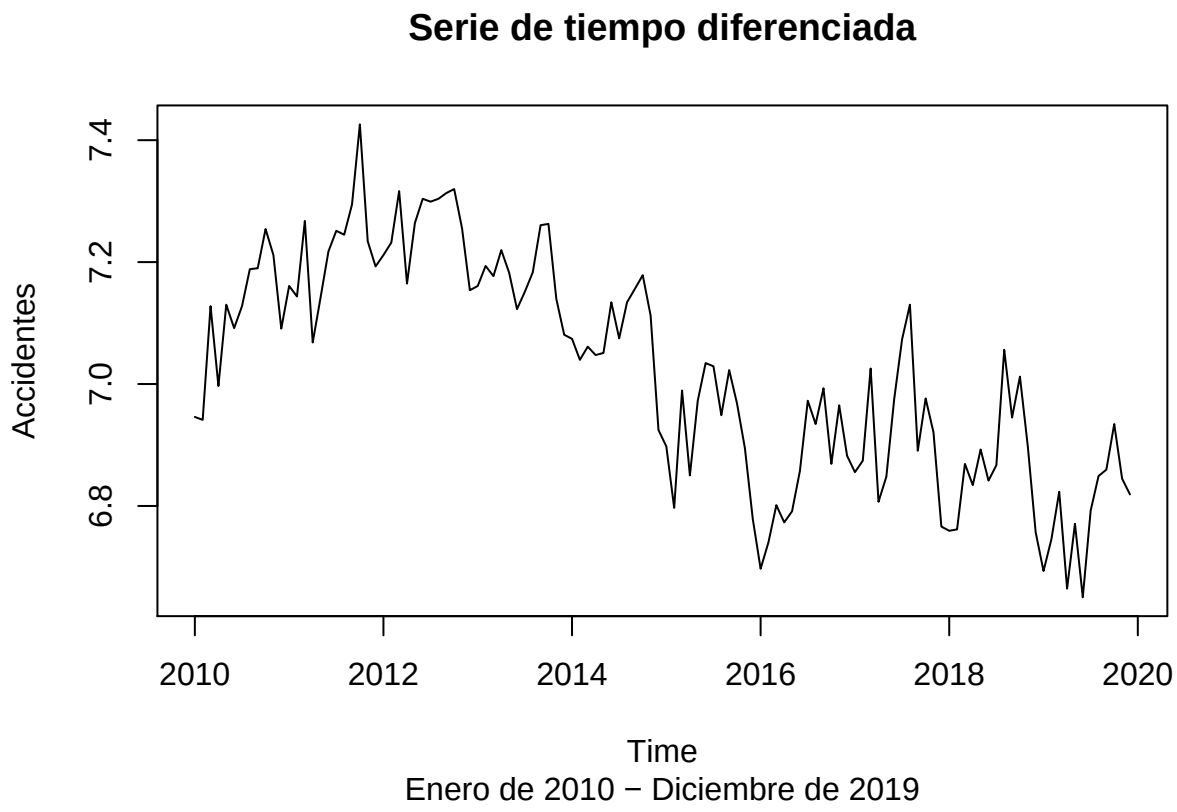
```
plot(diff(time.ts),
     main = "Serie de tiempo diferenciada",
     ylab = "Accidentes",
     sub = "Enero de 2010 - Diciembre de 2019")
```



Se realizo una transformacion por diferencias y se obtuvo una serie de tiempo con una varianza mas constante buescando obteniendo estacionalidad, por lo cual es posible hacer nuestras prediccion de manera mas facil

Se intento con una tranformacion logaritmica sin embargo no hubo una buena aproximacion.

```
plot(log(time.ts),
     main = "Serie de tiempo diferenciada",
     ylab = "Accidentes",
     sub = "Enero de 2010 - Diciembre de 2019")
```



Mediante la siguiente funcion buscaremos obtener los mejores ordenes y modelo para realizar el pronostico de la serie de tiempo

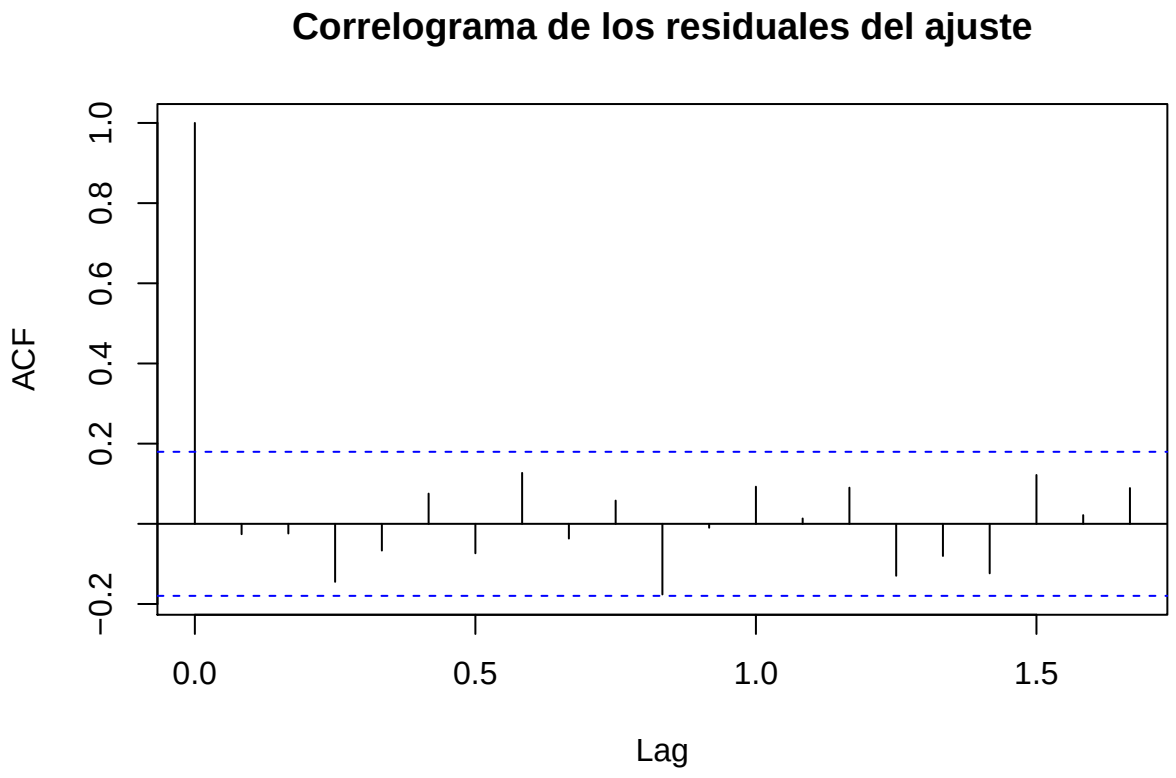
```
get.best.arima <- function(x.ts, maxord = c(1, 1, 1, 1, 1, 1)){
  best.aic <- 1e8
  n <- length(x.ts)
  for(p in 0:maxord[1])for(d in 0:maxord[2])for(q in 0:maxord[3])
    for(P in 0:maxord[4])for(D in 0:maxord[5])for(Q in 0:maxord[6])
    {
      fit <- arima(x.ts, order = c(p, d, q),
                   seas = list(order = c(P, D, Q),
                               frequency(x.ts)), method = "CSS")
      fit.aic <- -2*fit$loglik + (log(n) + 1)*length(fit$coef)
      if(fit.aic < best.aic){
        best.aic <- fit.aic
        best.fit <- fit
        best.model <- c(p, d, q, P, D, Q)
      }
    }
  list(best.aic, best.fit, best.model)
}
```

Ocupando la funcion recien declarada buscamos el mejor modelo y orden

```
best.arima.elec <- get.best.arima(diff(time.ts),
                                  maxord = c(2, 2, 2, 2, 2, 2))

best.fit.elec <- best.arima.elec[[2]]  # Modelo
best.arima.elec[[3]] # Tipo de modelo (órdenes)
best.fit.elec
best.arima.elec[[1]] # AIC
```

Por ultimo analizamos los residuales en graficos de correlogramas de la serie residual

```
#ACF para residuales del ajuste
acf(resid(best.fit.elec), main = "")
title(main = "Correlograma de los residuales del ajuste")
```

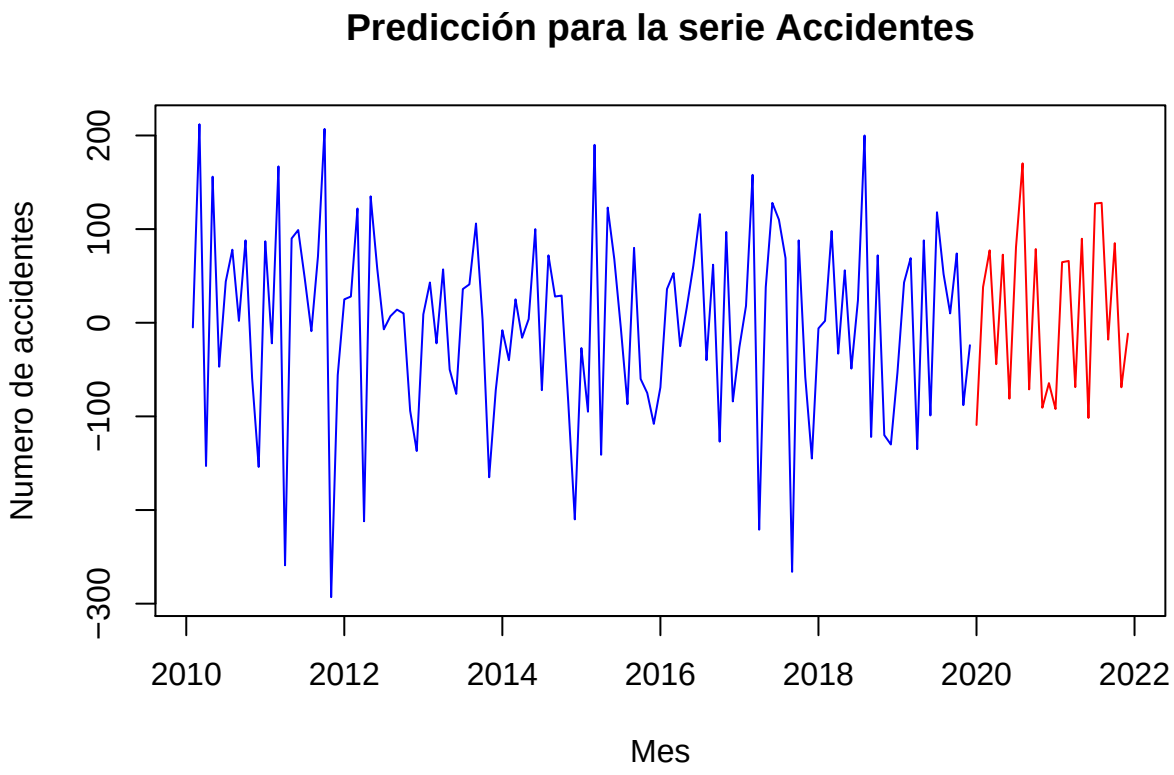


El grafico del correlograma luce bien, ya que se ve una distrubucion gausiana de ruido blanco Podria mejorarse el modelo realizando otros ajustes en los ordenes, pero se continuara con este modelo para generar valores predecidos de accidentes para dos años

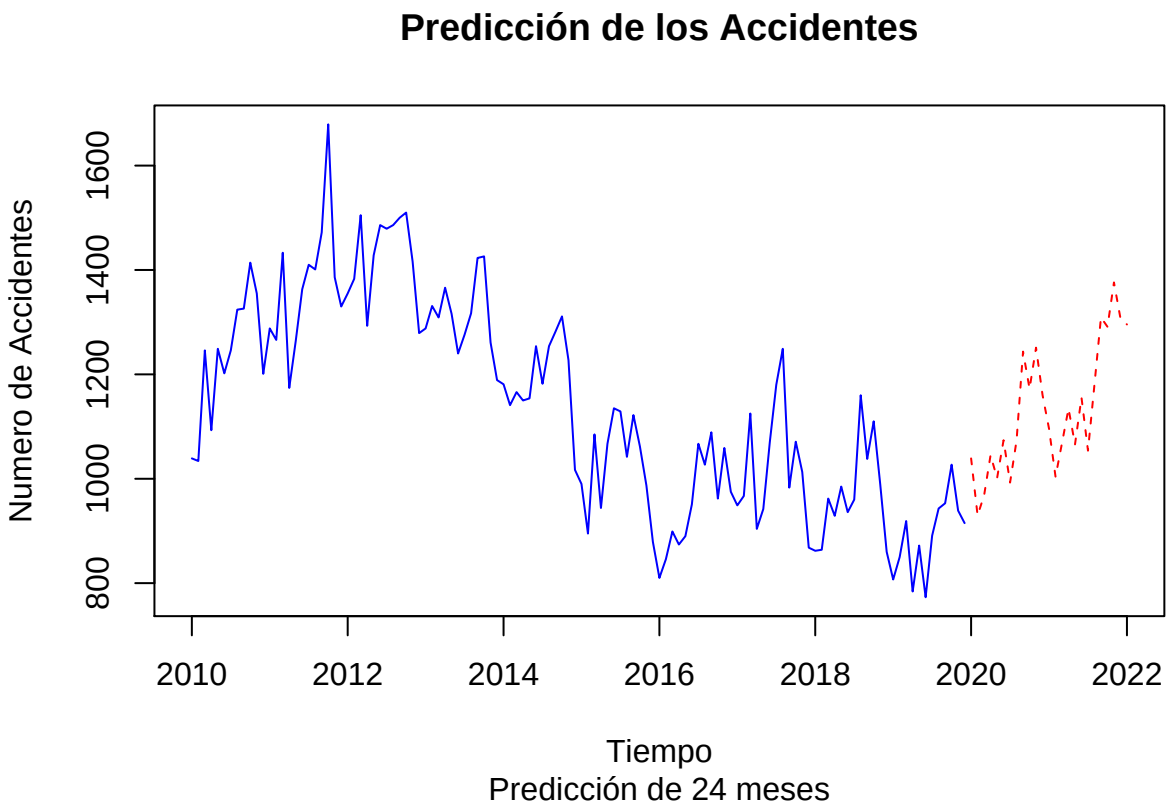
```
#ts(cumsum(c(time.ts[1],pr)), start =2010,freq = 12)
#prediccion
pr <- predict(best.fit.elec, 24)$pred
```

```
ts.plot(cbind(window(diff(time.ts), start = 2010),
              pr), col = c("blue", "red"), xlab = "")
title(main = "Predicción para la serie Accidentes",
```

```
      xlab = "Mes",
      ylab = "Numero de accidentes")
```




```
ssd<-diff(time.ts)

ts.plot(cbind(ts(cumsum(c(time.ts[1],ssd)), start =2010,freq = 12), ts(cumsum(c(time.ts[1],pr)), start =
        col = c("blue", "red"), xlab = "Tiempo",
        ylab = "Numero de Accidentes",
        main = "Predicción de los Accidentes",
        sub = "Predicción de 24 meses")
```



Utilizando el modelo ARIMA utilizamos la funcion auto.arima de la libreria forecast para obtener el mejor modelo

```
auto1=auto.arima(time.ts,D=1,approximation = F,allowdrift = T,allowmean = T)
```

##este tipo de modelo ya tiene inlcuido la transformacion en el valor D que #significa una diferenciacion

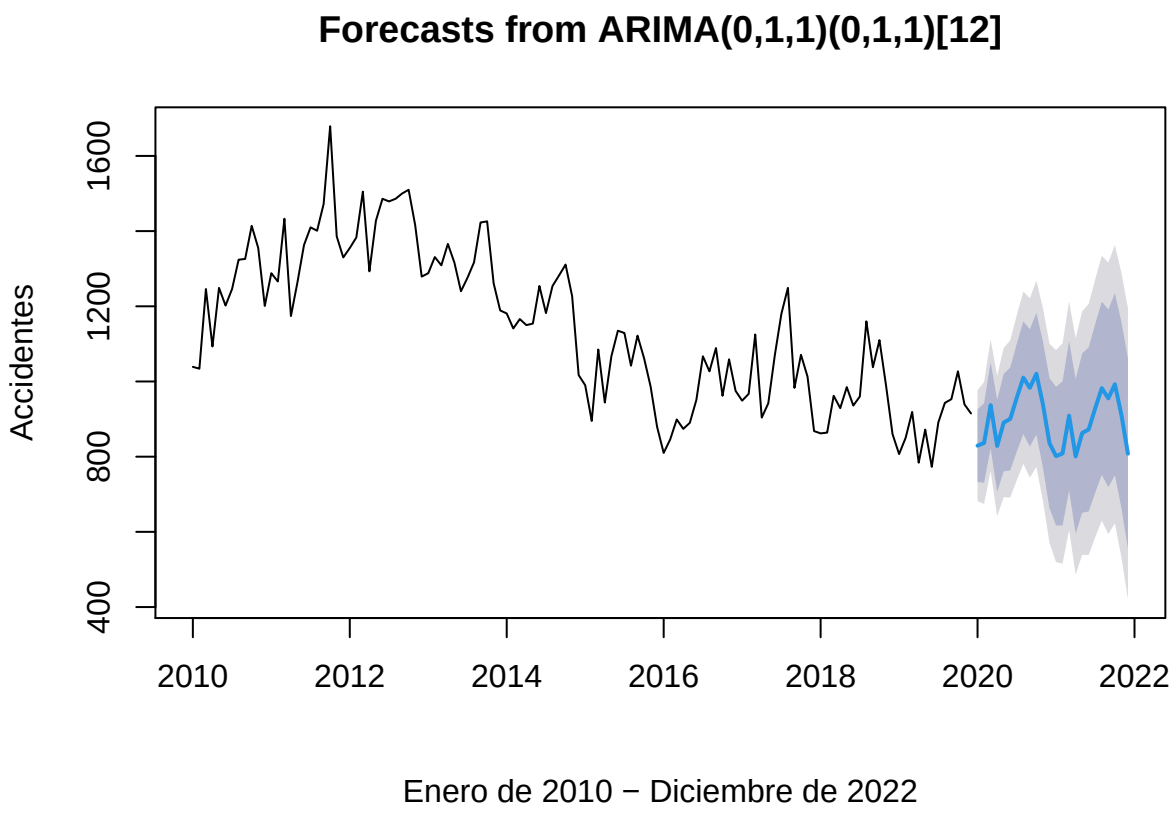
```
#se hace la prediccion el funcion de 24 meses (2 años)
forecast1<-forecast(auto1,h=24)
#Valores generados
head(forecast1)
```

```
## $method
## [1] "ARIMA(0,1,1)(0,1,1)[12]"
##
## $model
## Series: time.ts
## ARIMA(0,1,1)(0,1,1)[12]
##
## Coefficients:
##           ma1     sma1
##       -0.5517  -0.7961
## s.e.   0.0787   0.1545
##
## sigma^2 estimated as 5656:  log likelihood=-619.22
```

```
## AIC=1244.44   AICc=1244.68   BIC=1252.46
##
## $level
## [1] 80 95
##
## $mean
##            Jan       Feb       Mar       Apr       May       Jun       Jul
## 2020  829.2939  836.4096  936.9569  828.5855  890.7167  900.4177  957.0241
## 2021  801.3947  808.5103  909.0576  800.6862  862.8174  872.5184  929.1248
##            Aug       Sep       Oct       Nov       Dec
## 2020 1010.0963  983.0089 1020.5594  938.8704  835.8597
## 2021  982.1970  955.1096  992.6601  910.9711  807.9604
##
## $lower
##               80%      95%
## Jan 2020 732.6528 681.4940
## Feb 2020 730.5150 674.4578
## Mar 2020 822.5549 761.9941
## Apr 2020 706.2663 641.5145
## May 2020 760.9625 692.2749
## Jun 2020 763.6321 691.2222
## Jul 2020 813.5513 737.6014
## Aug 2020 860.2344 780.9023
## Sep 2020 827.0193 744.4434
## Oct 2020 858.6739 772.9769
## Nov 2020 771.2963 682.5880
## Dec 2020 662.7839 571.1631
## Jan 2021 617.0306 519.4342
## Feb 2021 616.8918 515.4551
## Mar 2021 710.4495 605.3126
## Apr 2021 595.3262 486.6151
## May 2021 650.9205 538.7490
## Jun 2021 654.2803 538.7521
## Jul 2021 704.7247 585.9344
## Aug 2021 751.7996 629.8346
## Sep 2021 718.8671 593.8079
## Oct 2021 750.7137 622.6351
## Nov 2021 663.4522 532.4236
## Dec 2021 554.9917 421.0781
##
## $upper
##               80%       95%
## Jan 2020  925.9351  977.0938
## Feb 2020  942.3042  998.3614
## Mar 2020 1051.3589 1111.9197
## Apr 2020  950.9047 1015.6565
## May 2020 1020.4708 1089.1585
## Jun 2020 1037.2032 1109.6131
## Jul 2020 1100.4969 1176.4468
## Aug 2020 1159.9583 1239.2904
## Sep 2020 1138.9985 1221.5744
## Oct 2020 1182.4449 1268.1419
## Nov 2020 1106.4444 1195.1528
## Dec 2020 1008.9354 1100.5562
```

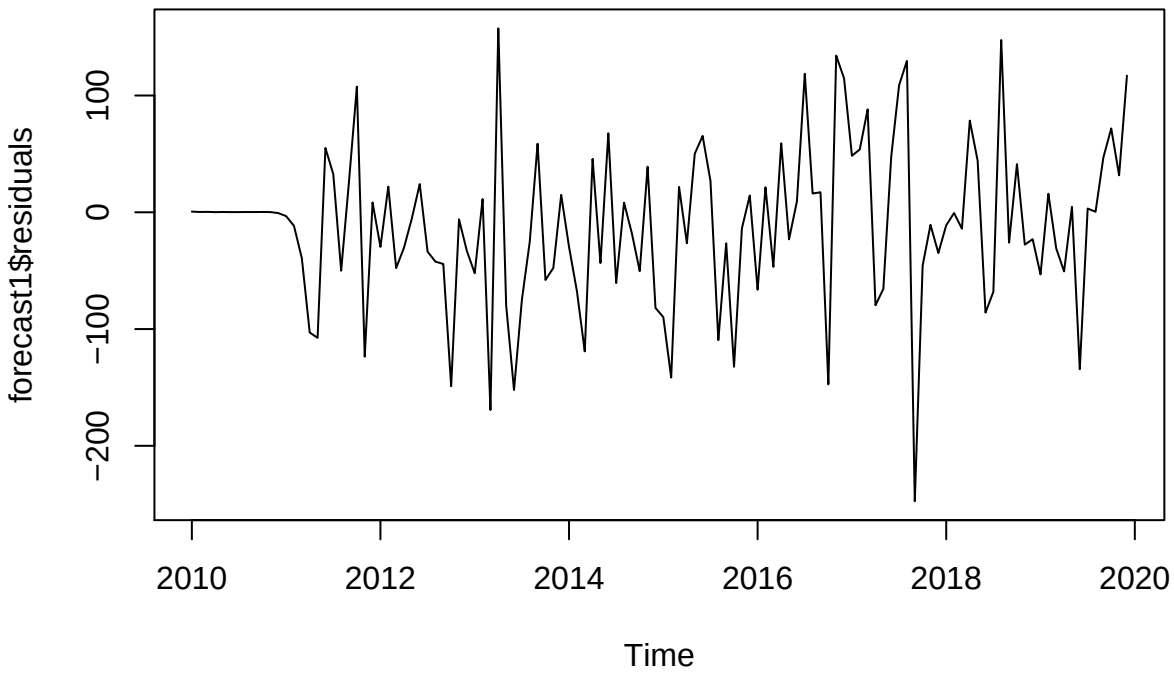
```
## Jan 2021  985.7587 1083.3551
## Feb 2021 1000.1288 1101.5655
## Mar 2021 1107.6658 1212.8026
## Apr 2021 1006.0463 1114.7574
## May 2021 1074.7143 1186.8858
## Jun 2021 1090.7565 1206.2847
## Jul 2021 1153.5249 1272.3151
## Aug 2021 1212.5944 1334.5594
## Sep 2021 1191.3521 1316.4113
## Oct 2021 1234.6065 1362.6852
## Nov 2021 1158.4900 1289.5186
## Dec 2021 1060.9291 1194.8426
```

```r
#graficamos los resultados
plot(forecast1,ylab = "Accidentes",
     sub = "Enero de 2010 - Diciembre de 2022")
```



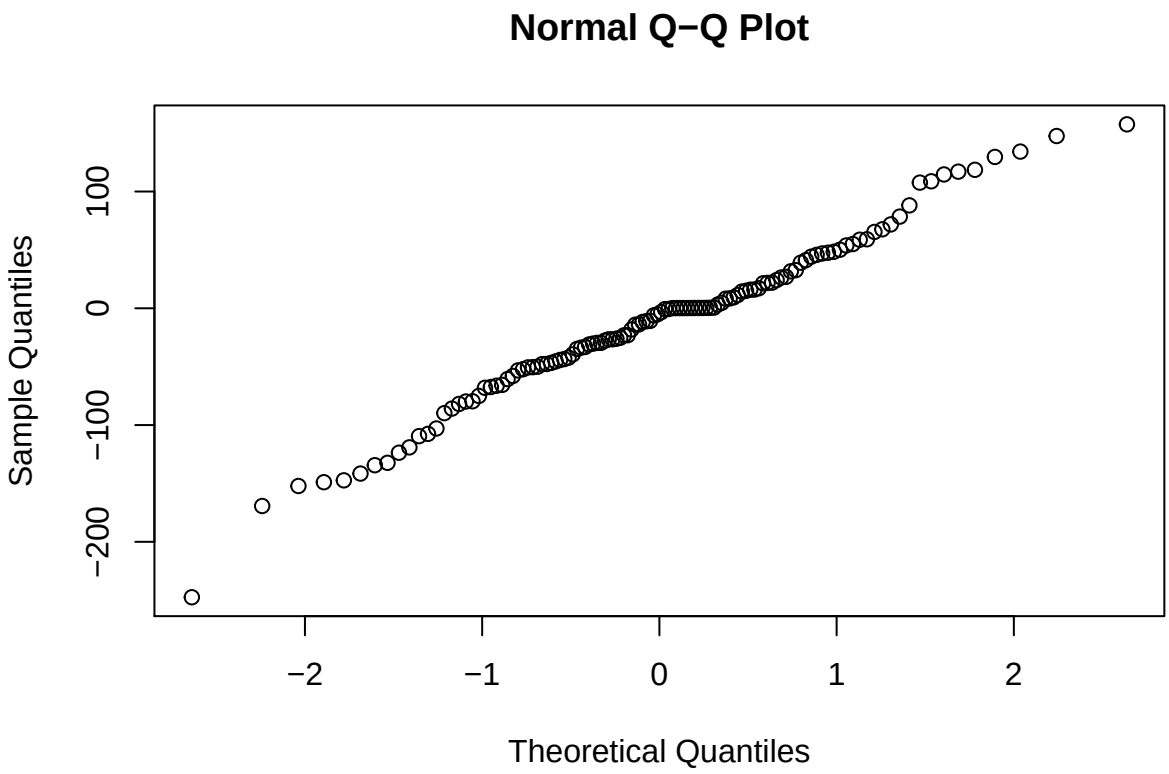
Analizamos los residuales generados

```r
plot(forecast1$residuals)
```



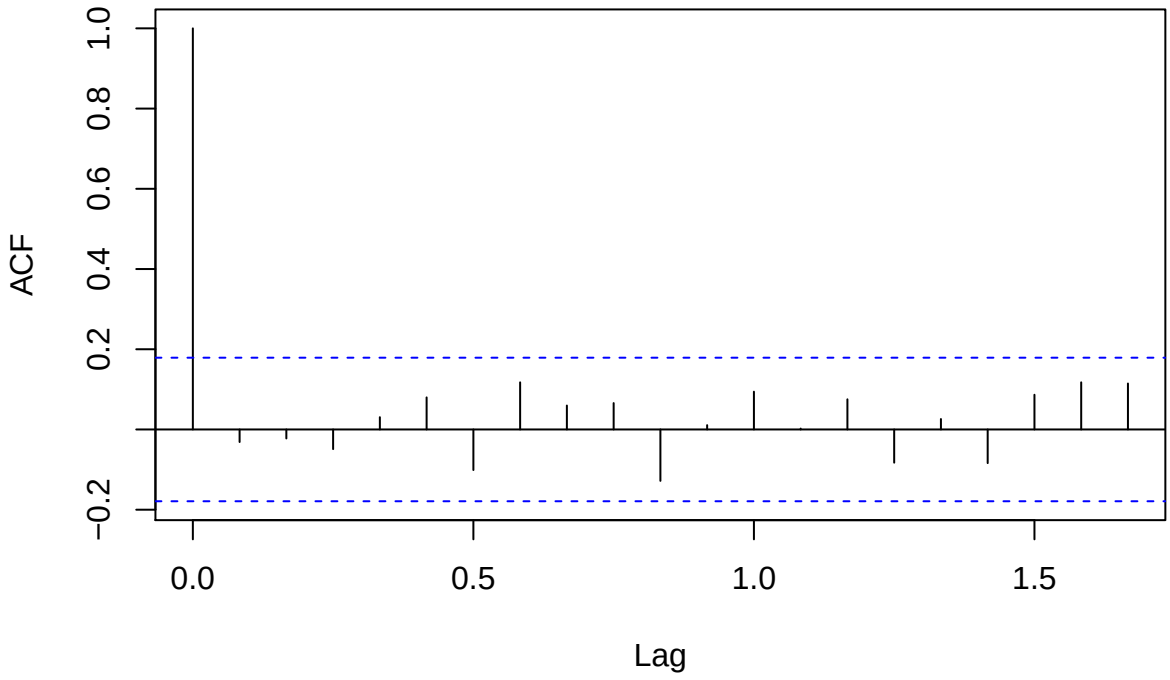
analizamos las tendencias de los residuales

```
qqnorm(forecast1$residuals)
```

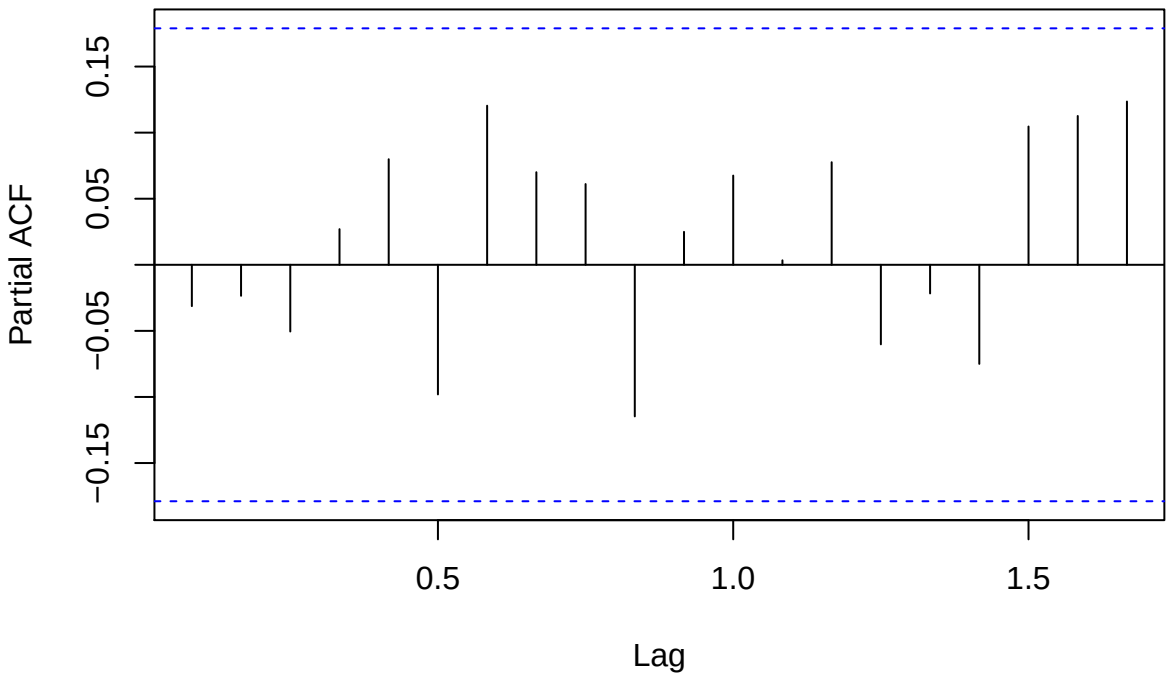


por ultimo analizamos los residuales en graficos de correlogramas de la serie residual acf y pacf

```
acf(forecast1$residuals, main = "")
```

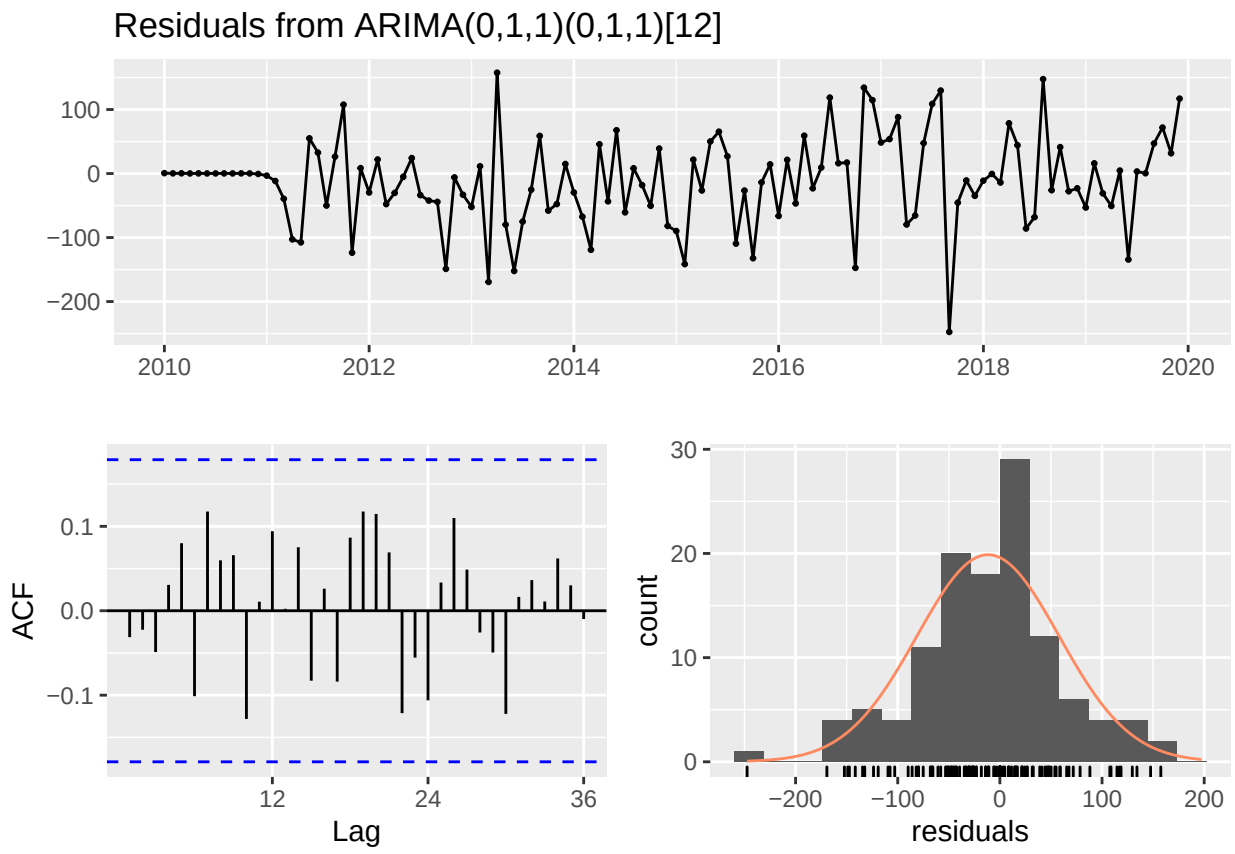



```
pacf(forecast1$residuals, main = "")
```



como sumary podemos analizar varios graficos a la vez

```
checkresiduals(forecast1)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(0,1,1)(0,1,1)[12]
## Q* = 21.89, df = 22, p-value = 0.4665
##
## Model df: 2.   Total lags used: 24
```

De igual forma podemos comprobar la precision del modelo

```
summary(forecast1)
```

```
##
## Forecast method: ARIMA(0,1,1)(0,1,1)[12]
##
## Model Information:
## Series: time.ts
## ARIMA(0,1,1)(0,1,1)[12]
##
## Coefficients:
##           ma1     sma1
##       -0.5517  -0.7961
## s.e.   0.0787   0.1545
##
## sigma^2 estimated as 5656:  log likelihood=-619.22
```

```
## AIC=1244.44   AICc=1244.68   BIC=1252.46
##
## Error measures:
##                       ME     RMSE     MAE       MPE    MAPE      MASE        ACF1
## Training set -11.60971 70.34752 52.1239 -1.113542 4.74473 0.4509276 -0.03126894
##
## Forecasts:
##          Point Forecast    Lo 80     Hi 80    Lo 95     Hi 95
## Jan 2020       829.2939 732.6528  925.9351 681.4940  977.0938
## Feb 2020       836.4096 730.5150  942.3042 674.4578  998.3614
## Mar 2020       936.9569 822.5549 1051.3589 761.9941 1111.9197
## Apr 2020       828.5855 706.2663  950.9047 641.5145 1015.6565
## May 2020       890.7167 760.9625 1020.4708 692.2749 1089.1585
## Jun 2020       900.4177 763.6321 1037.2032 691.2222 1109.6131
## Jul 2020       957.0241 813.5513 1100.4969 737.6014 1176.4468
## Aug 2020      1010.0963 860.2344 1159.9583 780.9023 1239.2904
## Sep 2020       983.0089 827.0193 1138.9985 744.4434 1221.5744
## Oct 2020      1020.5594 858.6739 1182.4449 772.9769 1268.1419
## Nov 2020       938.8704 771.2963 1106.4444 682.5880 1195.1528
## Dec 2020       835.8597 662.7839 1008.9354 571.1631 1100.5562
## Jan 2021       801.3947 617.0306  985.7587 519.4342 1083.3551
## Feb 2021       808.5103 616.8918 1000.1288 515.4551 1101.5655
## Mar 2021       909.0576 710.4495 1107.6658 605.3126 1212.8026
## Apr 2021       800.6862 595.3262 1006.0463 486.6151 1114.7574
## May 2021       862.8174 650.9205 1074.7143 538.7490 1186.8858
## Jun 2021       872.5184 654.2803 1090.7565 538.7521 1206.2847
## Jul 2021       929.1248 704.7247 1153.5249 585.9344 1272.3151
## Aug 2021       982.1970 751.7996 1212.5944 629.8346 1334.5594
## Sep 2021       955.1096 718.8671 1191.3521 593.8079 1316.4113
## Oct 2021       992.6601 750.7137 1234.6065 622.6351 1362.6852
## Nov 2021       910.9711 663.4522 1158.4900 532.4236 1289.5186
## Dec 2021       807.9604 554.9917 1060.9291 421.0781 1194.8426
```

Como notamos esta funcion ajusta mejor el modelo d e prediccion ya que nos resultaron residulaes que no necesitan la aplicacion de otro modelo para su analisis, por lo tanto nos quedaremos con el resultado de este ultimo y como podemos ver, las tendencias de accidentes en la CDMX se mantendran constantes segun la prediccion aunque tienen cierta tendencia a disminuir, aunque de acuerdo a nuestros intervalos de confidencia geenrados por el modelo este tambien podria aumentar.