Análisis Exploratorio de datos

Programa de Big Data y Ciencia de Datos

Carlos Daniel Jiménez
Uniempresarial

September 2, 2018

Observaciones sobre la base de datos

En está oprtunidad se trabajará con Celebrity death,la cual es es una base de datos creada por Hugo Darwood y se encuentra disponible en el siguiente link
https://www.kaggle.com/hugodarwood/celebrity-deaths

Descripción

Every dead celebrity, their reason for fame and their cause of death from 2006-2016. Nationality and fame score (number of citations) also included. [1]

[1] Tomado de https://www.kaggle.com/hugodarwood/celebrity-deaths

Inspección de los datos

Se procede a cargar la base de datos de la siguiente forma(recuerde usar la libreria readr)

```
celebrity_deaths_4<-read.csv("../Script_Uniempresarial/Base
```

Se procede a visaulizar los primeros cinco datos

```
## # A tibble: 5 x 9
##     age birth_year cause_o~ death~ death~ famous_for
## * <int>      <int> <chr>    <chr>   <int> <chr>
## 1    85       1921 " natur~ Janua~   2006 " businessma~
## 2    49       1957 " murde~ Janua~   2006 " musician (~
## 3    64       1942 " Alzhe~ Janua~   2006 " baseball p~
## 4    86       1920 " Alzhe~ Janua~   2006 " politician~
## 5    82       1924 " cance~ Janua~   2006 " nightclub ~
```

La función tbl df permite ver el mayor número de carácteres de una base de datos en una página, aunque como buena practica se recomienda el uso de la libreria knirt

La función glimpse sirve para ver la estructura de la base de datos de manera más eficiente

```
celebrity_deaths_4%>%
  glimpse()
```

```
## Observations: 21,458
## Variables: 9
## $ age             <int> 85, 49, 64, 86, 82, 52, 31, 31, 5
## $ birth_year      <int> 1921, 1957, 1942, 1920, 1924, 195
## $ cause_of_death <chr> " natural causes", " murdered", "
## $ death_month     <chr> "January", "January", "January",
## $ death_year      <int> 2006, 2006, 2006, 2006, 2006, 200
## $ famous_for      <chr> " businessman chairman of IBM (19
## $ name            <chr> "Frank Cary", "Bryan Harvey", "Pa
## $ nationality     <chr> "American", "American", "American
## $ fame_score      <int> 1, 2, 1, 2, NA, NA, 1, 4, 2, NA,
```

Lo anterior se complementa con el conteo de variables y datos que pose el data frame

```
BBDD<-rbind(dim(celebrity_deaths_4))
colnames(BBDD)<-c('Observaciones','Variables')
BBDD%>%
  cbind()%>%
  kable()
```

Observaciones	Variables
21458	9

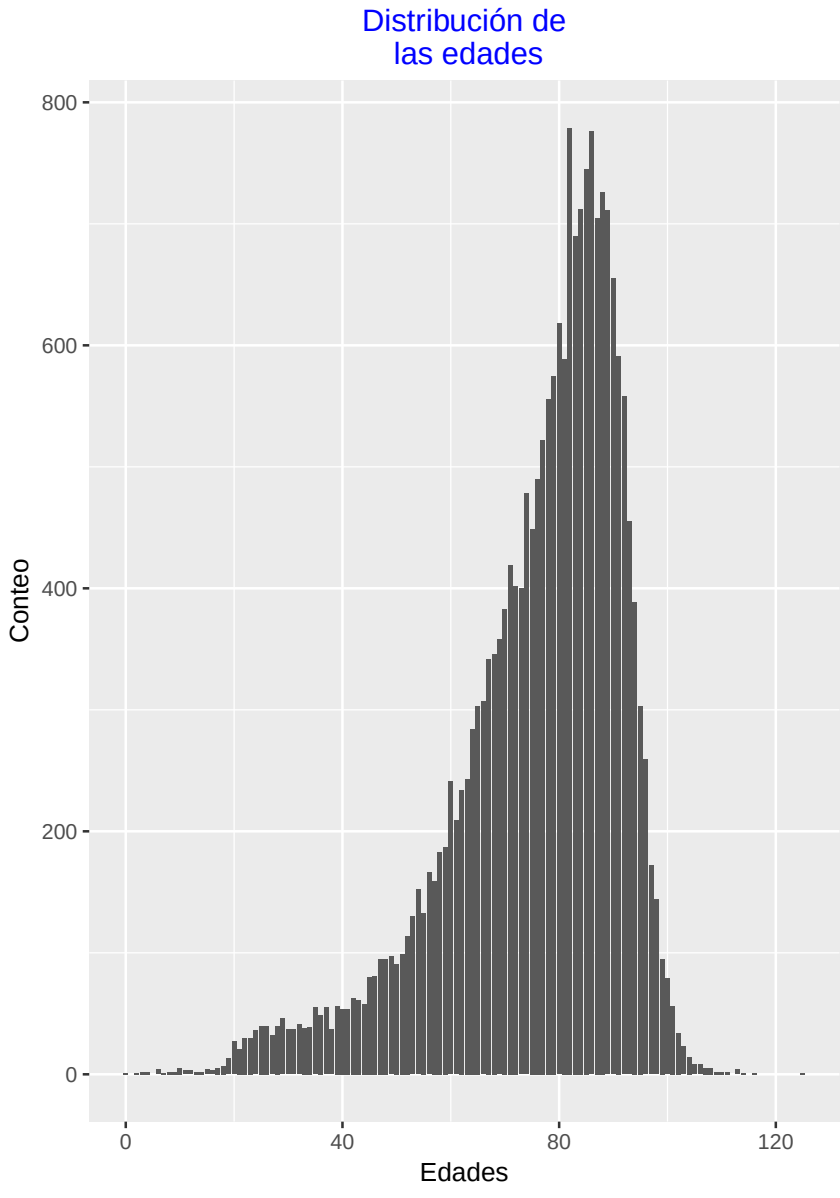
Algunas medidas de resumen por variable

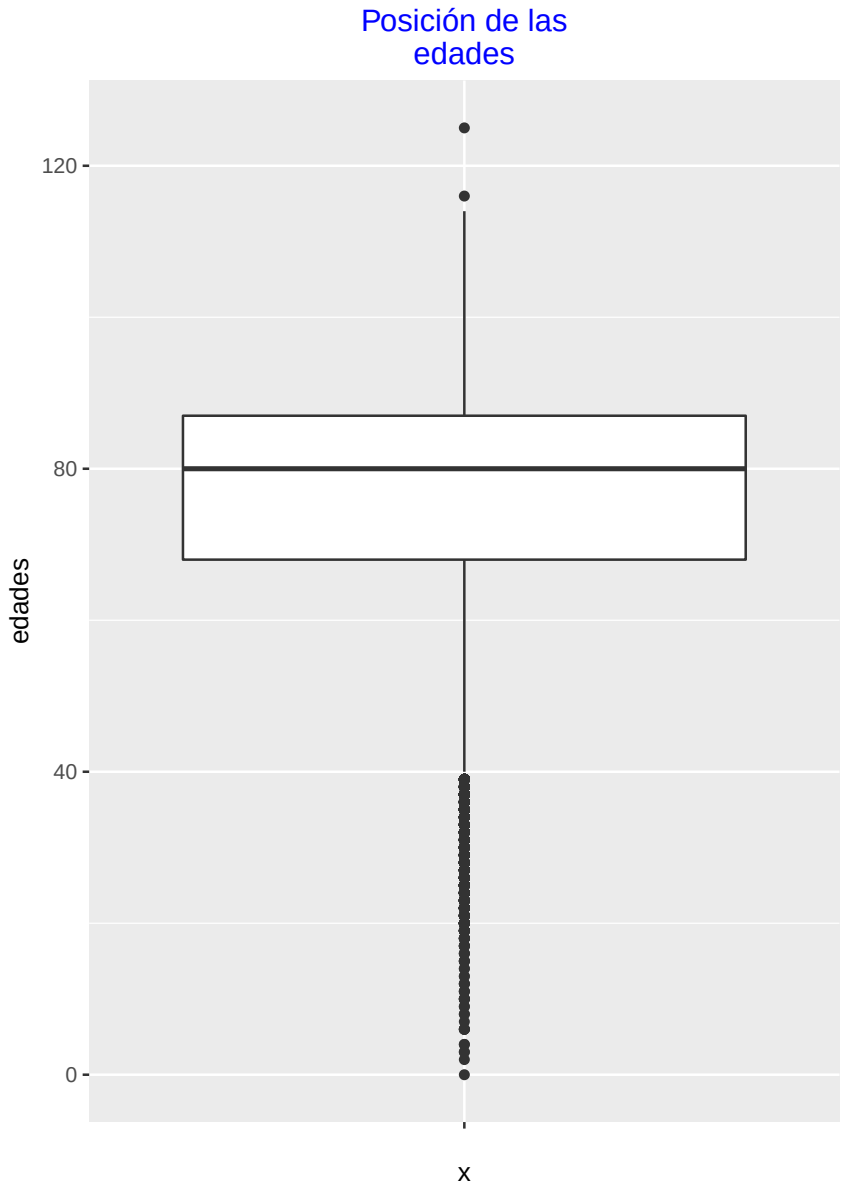
Variable edad

```
celebrity_deaths_4$age<-as.numeric(celebrity_deaths_4$age)
celebrity_deaths_4$age%>%
  summary()
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0      68      80      76      87     125
```

Se visualiza la edad a través de dos gráficos





Variable fame_score

```
celebrity_deaths_4$fame_score%>%
  summary()
```

```
##     Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    1.000   2.000   4.000   8.624   8.000 695.000    1606
```

- Se evidencia una fuerte tendencia en valores ausentes , por ello se imputa de la siguiente manera

```
celebrity_deaths_4$fame_score[is.na(celebrity_deaths_4$fame
  median(celebrity_deaths_4$fame_score,na.rm = TRUE)
```

Lo que resulta en lo siguiente

```
##                     .
## Min.         1.000000
## 1st Qu.      2.000000
## Median       4.000000
## Mean         8.277565
## 3rd Qu.      8.000000
## Max.       695.000000
```

Desde acá se puede hacer algunas conclusiones importantes

El total del resumen

```
##       age          birth_year   cause_of_death
##  Min.   :  0   Min.   :1889   Length:21458
##  1st Qu.: 68   1st Qu.:1925   Class :character
##  Median : 80   Median :1933   Mode  :character
##  Mean   : 76   Mean   :1936
##  3rd Qu.: 87   3rd Qu.:1944
##  Max.   :125   Max.   :2011
```

```
##  death_month          death_year    famous_for
##  Length:21458       Min.   :2006   Length:21458
##  Class :character   1st Qu.:2010   Class :character
##  Mode  :character   Median :2013   Mode  :character
##                     Mean   :2012
##                     3rd Qu.:2015
##                     Max.   :2016
```

```
##      name           nationality          fame_score     
##  Length:21458       Length:21458       Min.   :  1.000  
##  Class :character   Class :character   1st Qu.:  2.000  
##  Mode  :character   Mode  :character   Median :  4.000  
##                                        Mean   :  8.278  
##                                        3rd Qu.:  8.000  
##                                        Max.   :695.000
```

Un poco de SQL

```
celebrity_deaths_4<- sqldf("SELECT *,
CASE
                                WHEN cause_of_death LIKE '%cance
                                WHEN cause_of_death LIKE 'natura
                                WHEN cause_of_death LIKE '%murde
                                WHEN cause_of_death LIKE '%Alzhe
                                WHEN cause_of_death LIKE '%heart
                                WHEN cause_of_death LIKE '%suici
                                WHEN cause_of_death LIKE '%pneum
                                WHEN cause_of_death LIKE '%crash
                                WHEN cause_of_death IS NULL THEN
                                ELSE 'Otras' END AS cause_group
                                FROM celebrity_deaths_4")
```

Qué se hizo?

Se categorizo el tipo de enfermedades de la siguiente manera, tenga en cuenta lo siguiente

```
celebrity_deaths_4$cause_of_death%>%
  table%>%
  head(10)%>%
  tbl_df()
```

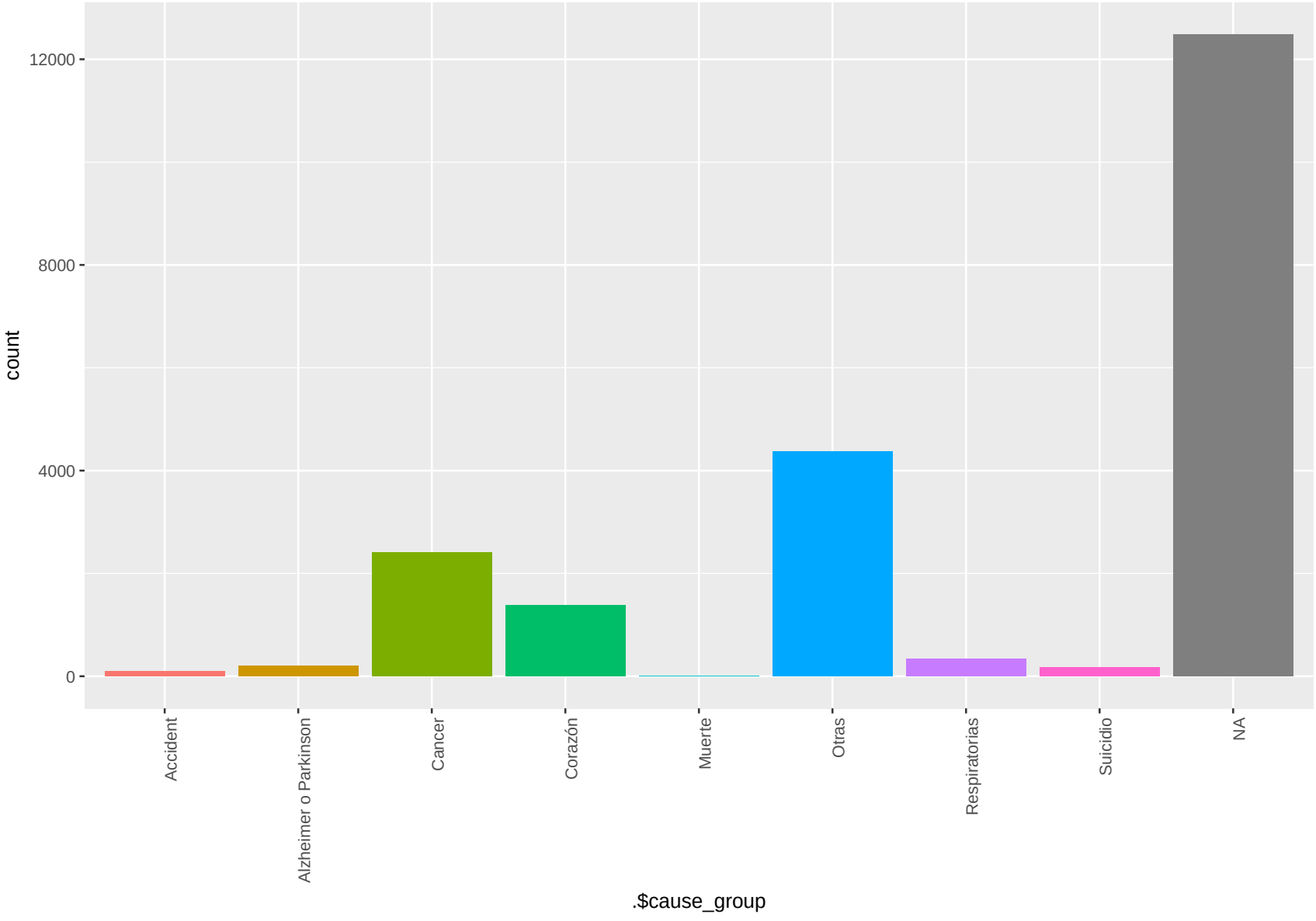
```
## # A tibble: 10 x 2
##                                                                .
##                                                           <chr>
##  1 " \"Father of San Jose International Airport"
##  2 " (Dream Baby) for Roy Orbison et al"
##  3 " [176]"
##  4 " [177]"
##  5 " [188]"
##  6 " [227]"
##  7 " [93]"
##  8 " 1936 Olympic bronze medallist"
##                                                                n
##                                                            <int>
##                                                                1
##                                                                1
##                                                                1
##                                                                1
##                                                                1
##                                                                1
##                                                                1
##                                                                1
```

Por lo anterior se le designa dentro de unos parámetros a que categoria pertenece, lo que da como resultado

```
celebrity_deaths_4$cause_group%>%
  table()%>%
  cbind()
```

```
##                                .
## Accident                      97
## Alzheimer o Parkinson        197
## Cancer                      2413
## Corazón                     1382
## Muerte                         7
## Otras                       4366
## Respiratorias                335
## Suicidio                     177
```

Una forma de verlo es la siguiente



Forma de ver estos resultados

- En forma de tabla de conteo se usa la función table

```
celebrity_deaths_4$cause_group%>%
  table()%>%
  cbind
```

```
##                              .
## Accident                    97
## Alzheimer o Parkinson      197
## Cancer                    2413
## Corazón                   1382
## Muerte                       7
## Otras                     4366
## Respiratorias              335
## Suicidio                   177
```

- Otra forma es con la función count

```
attach(celebrity_deaths_4)
count(celebrity_deaths_4, cause_group)
```

```
## # A tibble: 9 x 2
##   cause_group                n
##   <chr>                  <int>
## 1 Accident                  97
## 2 Alzheimer o Parkinson    197
## 3 Cancer                  2413
## 4 Corazón                 1382
## 5 Muerte                     7
## 6 Otras                   4366
## 7 Respiratorias            335
## 8 Suicidio                 177
## 9 <NA>                   12484
```

Para verlo de manera de proporción

```
prop.table(table(celebrity_deaths_4$cause_group))%>%
  cbind()
```

```
##                                    .
## Accident                 0.0108090038
## Alzheimer o Parkinson    0.0219523067
## Cancer                   0.2688878984
## Corazón                  0.1540004457
## Muerte                   0.0007800312
## Otras                    0.4865166035
## Respiratorias            0.0373300646
## Suicidio                 0.0197236461
```

Como el resultado da con muchos decimales se usa la función round la cual trunca los decimales a la hora de imprimirlos , más no a la hora de calcularlos

	Porcentaje
Accident	0.01
Alzheimer o Parkinson	0.02
Cancer	0.27
Corazón	0.15
Muerte	0.00
Otras	0.49
Respiratorias	0.04
Suicidio	0.02

Para comprobar que este completo se emplea la siguiente instrucción

```
sum(prop.table(table(celebrity_deaths_4$cause_group)))
```

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

Se desarrolla el procedimiento anterior con el año de la muerte

```
celebrity_deaths_4$death_year <- factor(celebrity_deaths_4$
```

```
##    death_year    n
## 1        2006  440
## 2        2007  797
## 3        2008 1157
## 4        2009 1459
## 5        2010 2023
## 6        2011 2096
## 7        2012 2151
## 8        2013 2496
## 9        2014 2891
## 10       2015 2750
## 11       2016 3198
```

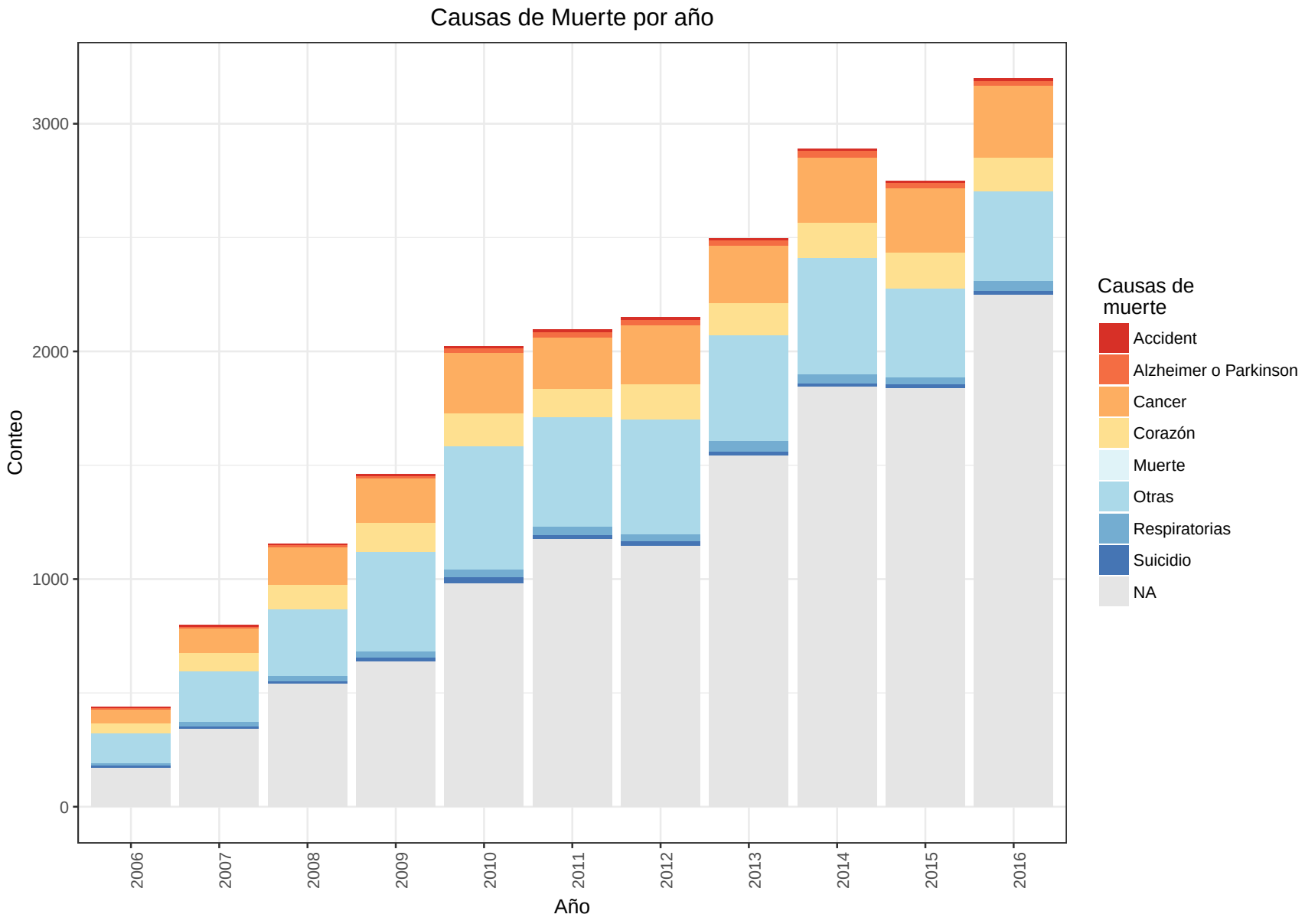
		Proporción
2006	440	2.05
2007	797	3.71
2008	1157	5.39
2009	1459	6.80
2010	2023	9.43
2011	2096	9.77
2012	2151	10.02
2013	2496	11.63
2014	2891	13.47
2015	2750	12.82
2016	3198	14.90

Visualización

La mejor forma de entender el análisis exploratorio viene dado de las buenas practicas del visual Analytics

Observe el siguiente código

```
r<-celebrity_deaths_4%>%
  ggplot(aes(.$death_year))+
  geom_bar(aes(fill=.$cause_group))+
  text_theme+
  color_theme+
  labs(title='Causas de Muerte por año')+
  xlab('Año')+
  ylab('Conteo')+
  theme_bw()+
  theme(plot.title = element_text(hjust = 0.5))
```


Causas de Muerte por año
Conteo
3000
2000
1000
0
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
Año
Causas de muerte
Accident
Alzheimer o Parkinson
Cancer
Corazón
Muerte
Otras
Respiratorias
Suicidio
NA

Creación de una variable nueva

```
celebrity_deaths_4$Grupo_de_fama<-factor(findInterval(celeb
levels(celebrity_deaths_4$Grupo_de_fama)<-c('Localmente Fam
```

Creación de tablas

```
##      [,1]
## 2006  440
## 2007  797
## 2008 1157
## 2009 1459
## 2010 2023
## 2011 2096
## 2012 2151
## 2013 2496
## 2014 2891
## 2015 2750
## 2016 3198
```

```
##      [,1]      [,2]
## 2006  440  2.050517
## 2007  797  3.714232
## 2008 1157  5.391928
## 2009 1459  6.799329
## 2010 2023  9.427719
## 2011 2096  9.767919
## 2012 2151 10.024233
## 2013 2496 11.632025
## 2014 2891 13.472831
## 2015 2750 12.815733
## 2016 3198 14.903532
```

Comprobación de las tablas

```r
sum(table_de_conteo_prp[,2]) ## Suma proporcional
```

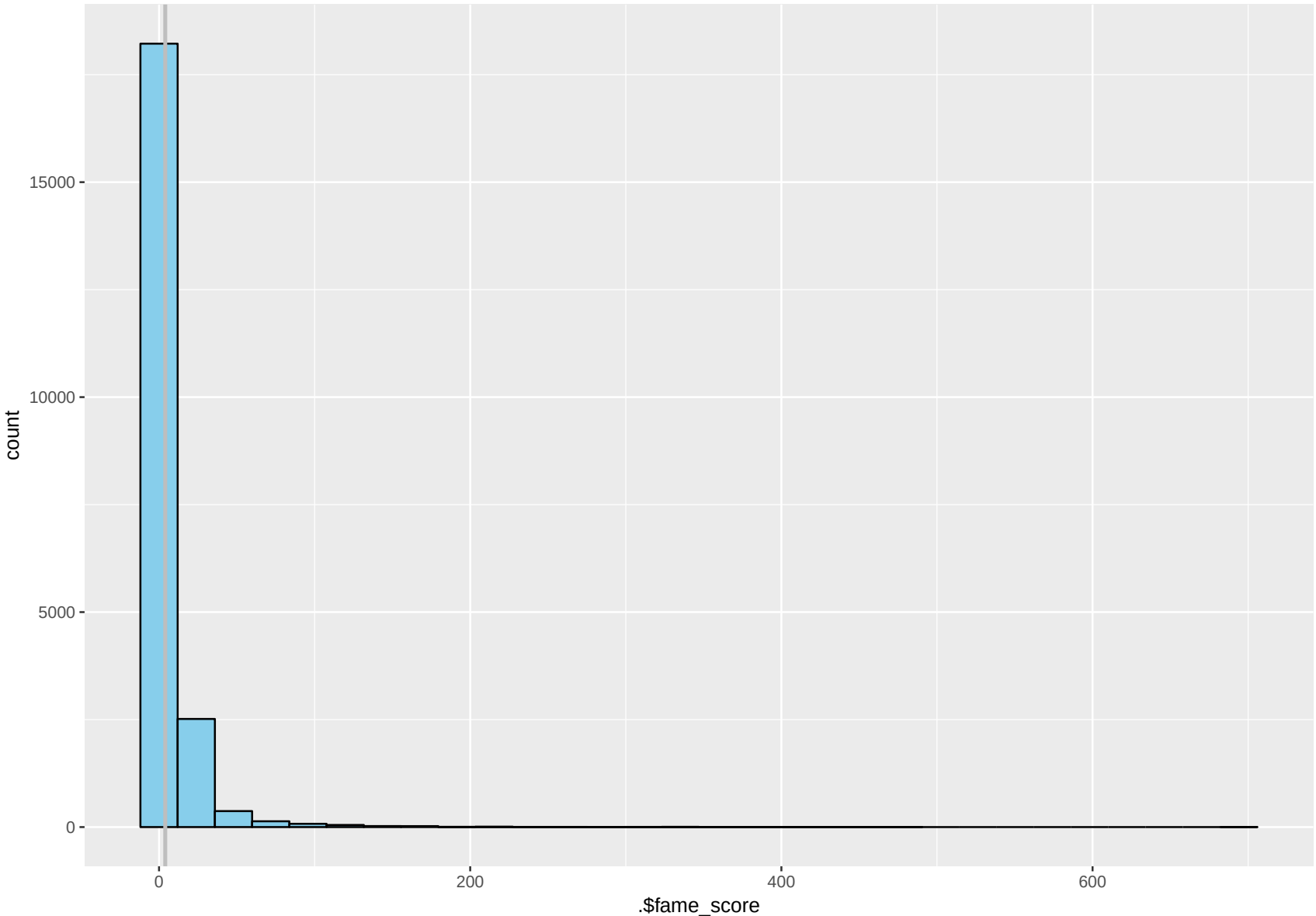
```
## [1] 100
```

```r
sum(table_de_conteo_prp[,1]) ## Suma total
```

```
## [1] 21458
```

Visualización de la fama

```
## `stat_bin()` using `bins = 30`. Pick better value with `
```



Se encuentran tres categorias

- Localmente famoso, debajo de la mediana
- Famoso : Sobre la mediana y superior
- Mundialmente famoso: dos veces la media y superior

```
##   age birth_year  cause_of_death death_month death_year
## 1  85       1921  natural causes     January       2006
## 2  49       1957        murdered     January       2006
##                                          famous_for
## 1  businessman chairman of IBM (1973<89><db><d2>1981)
## 2              musician (House of Freaks Gutterball) Br
##   nationality fame_score cause_group     Grupo_de_fama
## 1    American           1       Otras Localmente Famoso
## 2    American           2      Muerte Localmente Famoso
```

Creación de intervalos de edad

Dado lo anterior para estimar una mejor manera de análisis de datos, se desarrolla un intervalo de edades a través de la función findIterval

```
## # A tibble: 20 x 2
##    Var1      n
##    <chr> <int>
##  1 0         1
##  2 2         1
##  3 3         2
##  4 4         2
##  5 6         4
##  6 7         1
##  7 8         2
##  8 9         2
##  9 10        5
## 10 11        3
## 11 12        3
## 12 13        2
## 13 14        2
```

```
findInterval(celebrity_deaths_4$age, c(20,40,60,80,100))%>%
  head(10)
```

```
##  [1] 4 2 3 4 4 2 1 1 2 3
```

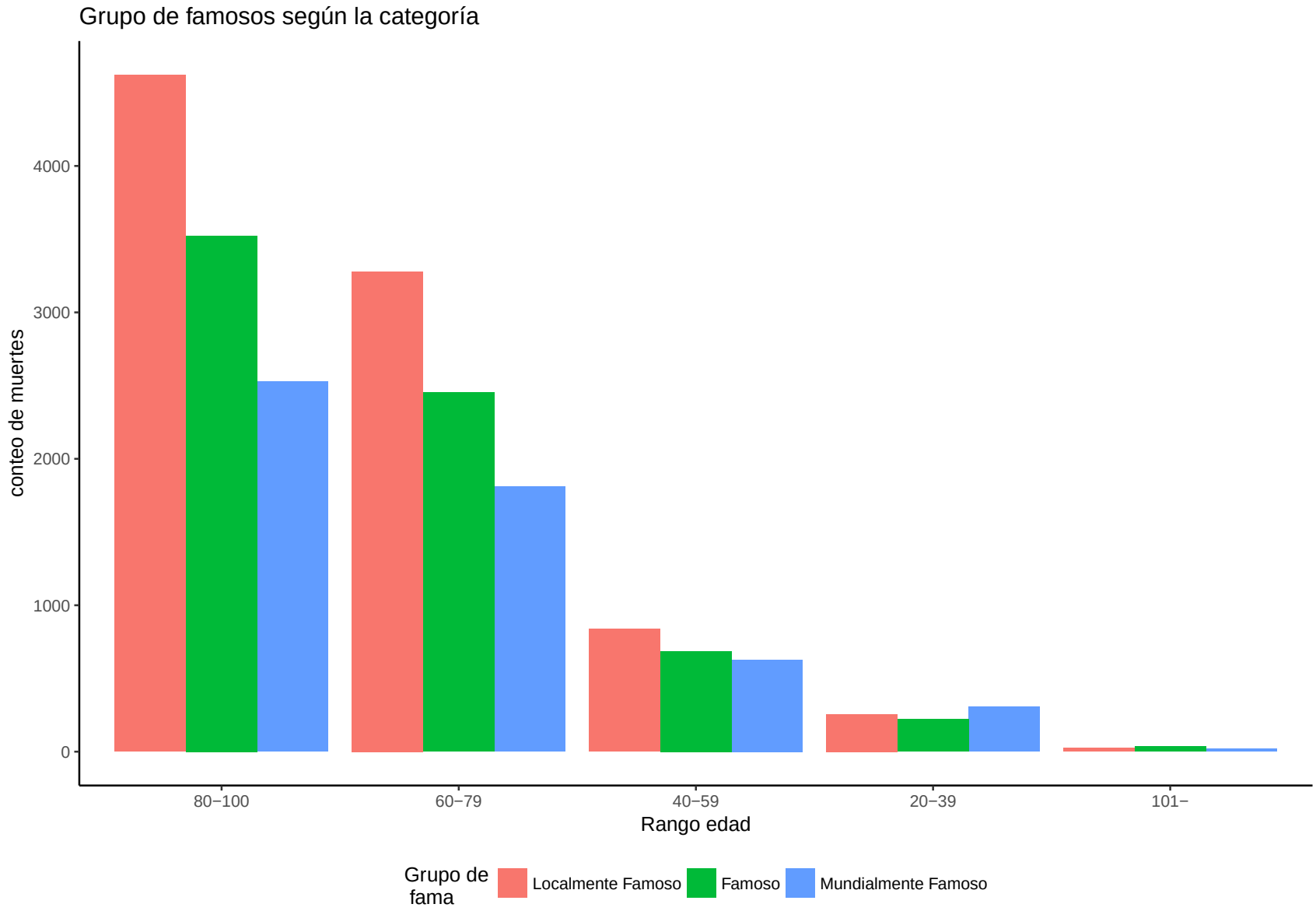
Se eliminan los valores que no aportan valor y se crean niveles

```
celebrity_deaths_4$age[celebrity_deaths_4$age>100 |celebrit
celebrity_deaths_4$grupos_edad<-factor(findInterval(celebr
```

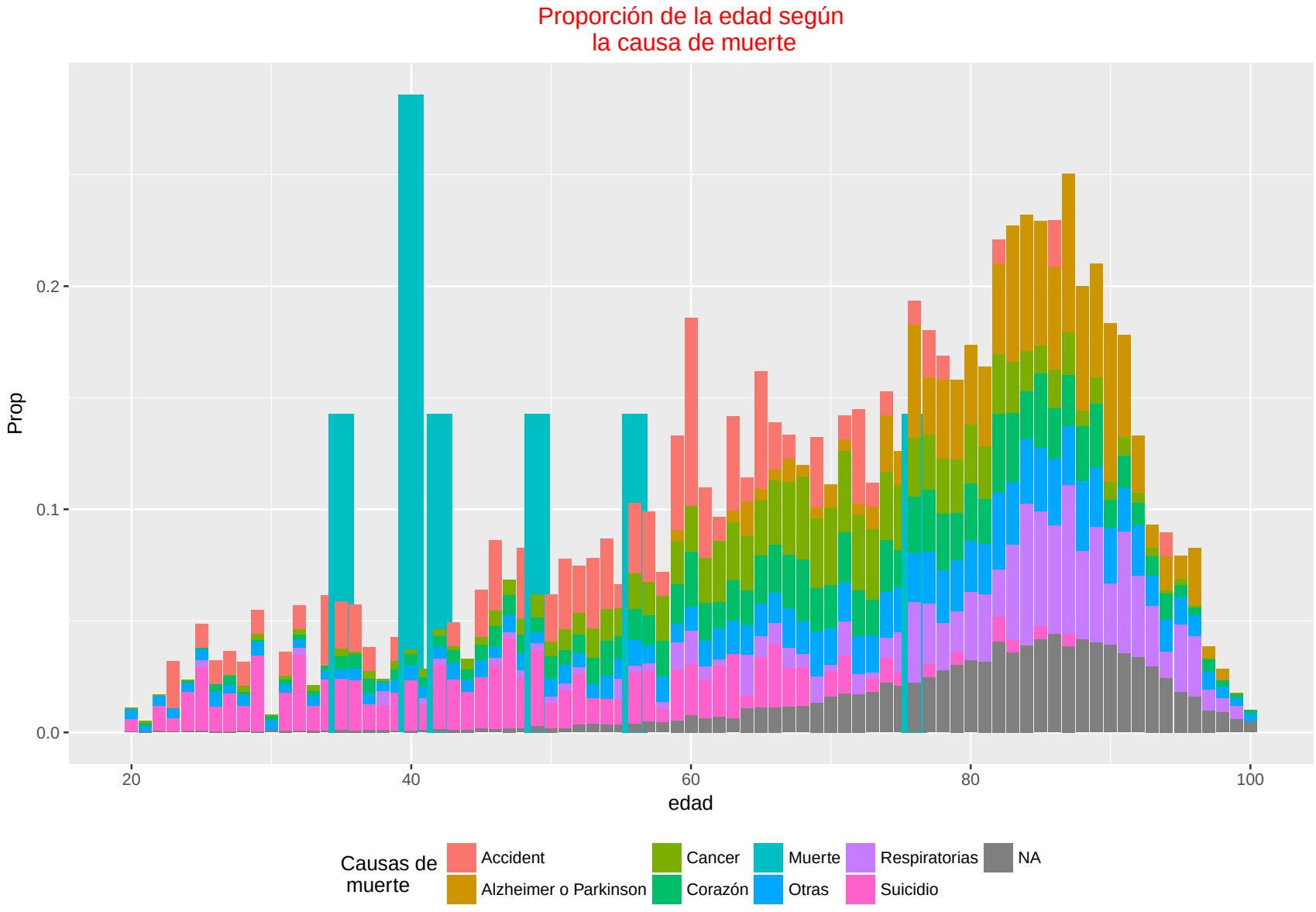
```
## Warning in findInterval(celebrity_deaths_4$age, c(20, 40
## NAs introduced by coercion
```

```
levels(celebrity_deaths_4$grupos_edad)<-c('20-39','40-59',
```

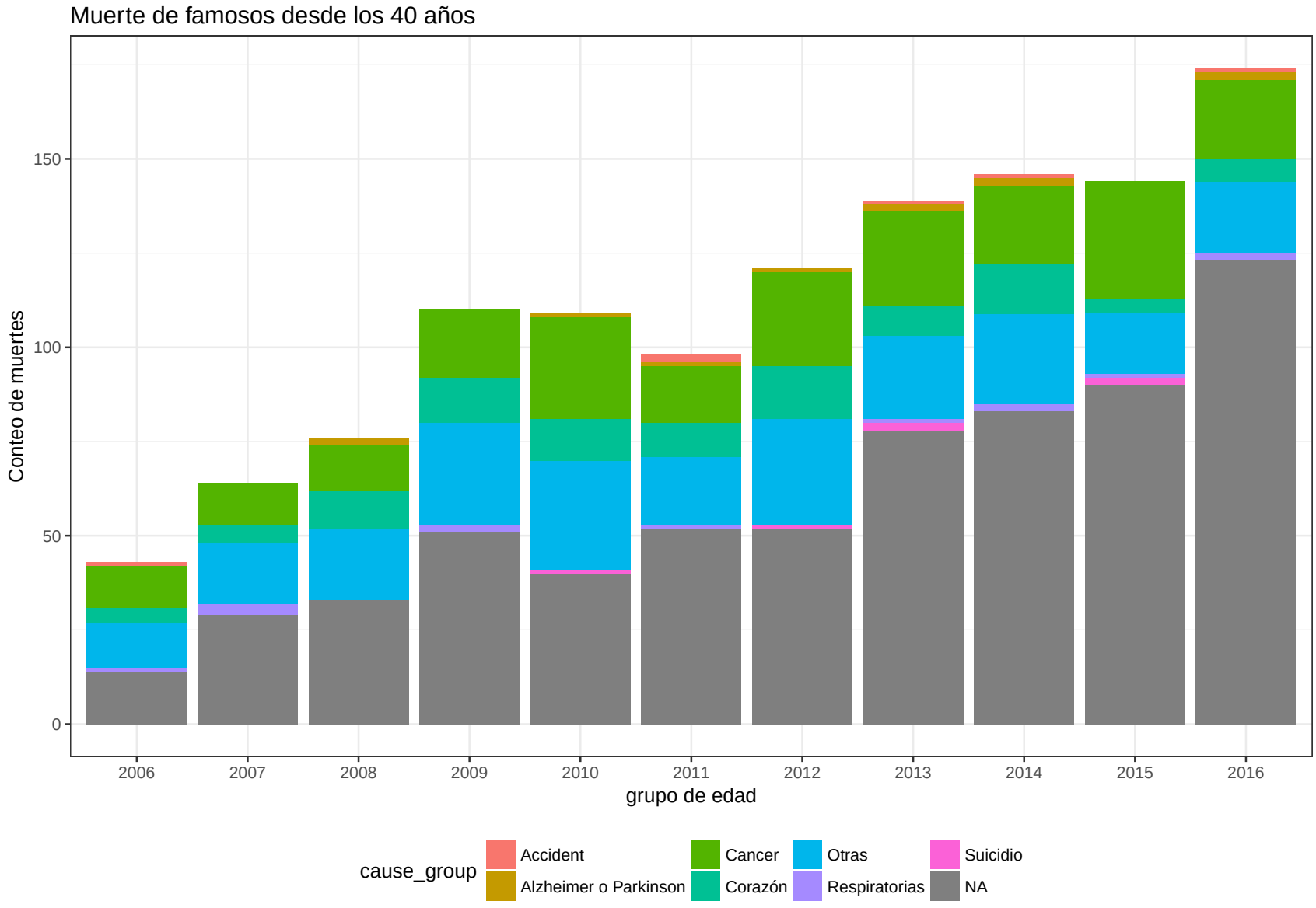
Se limpia la BBDD y se visualiza



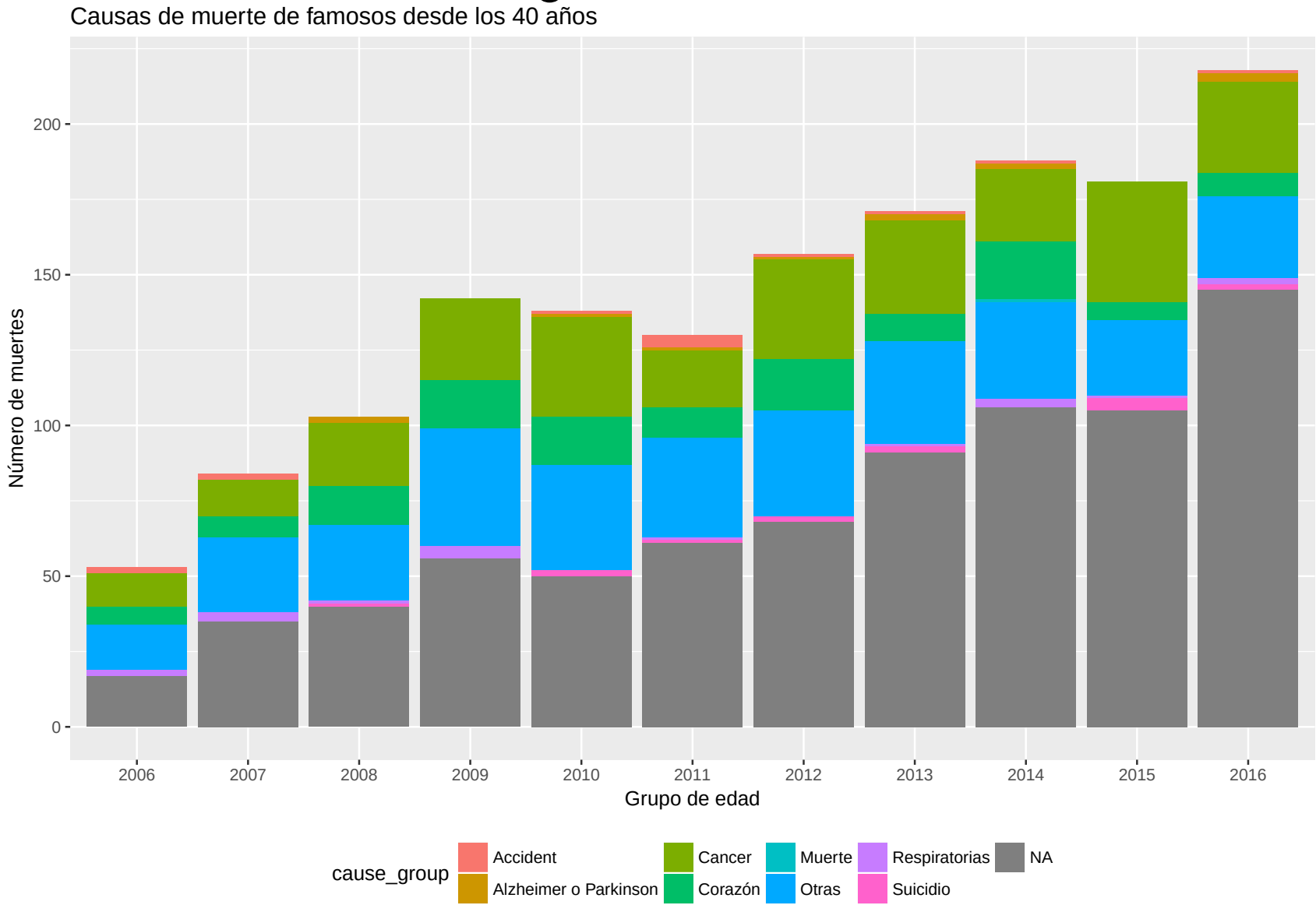
Otro factor a evaluar ‘La causa de la muerte’



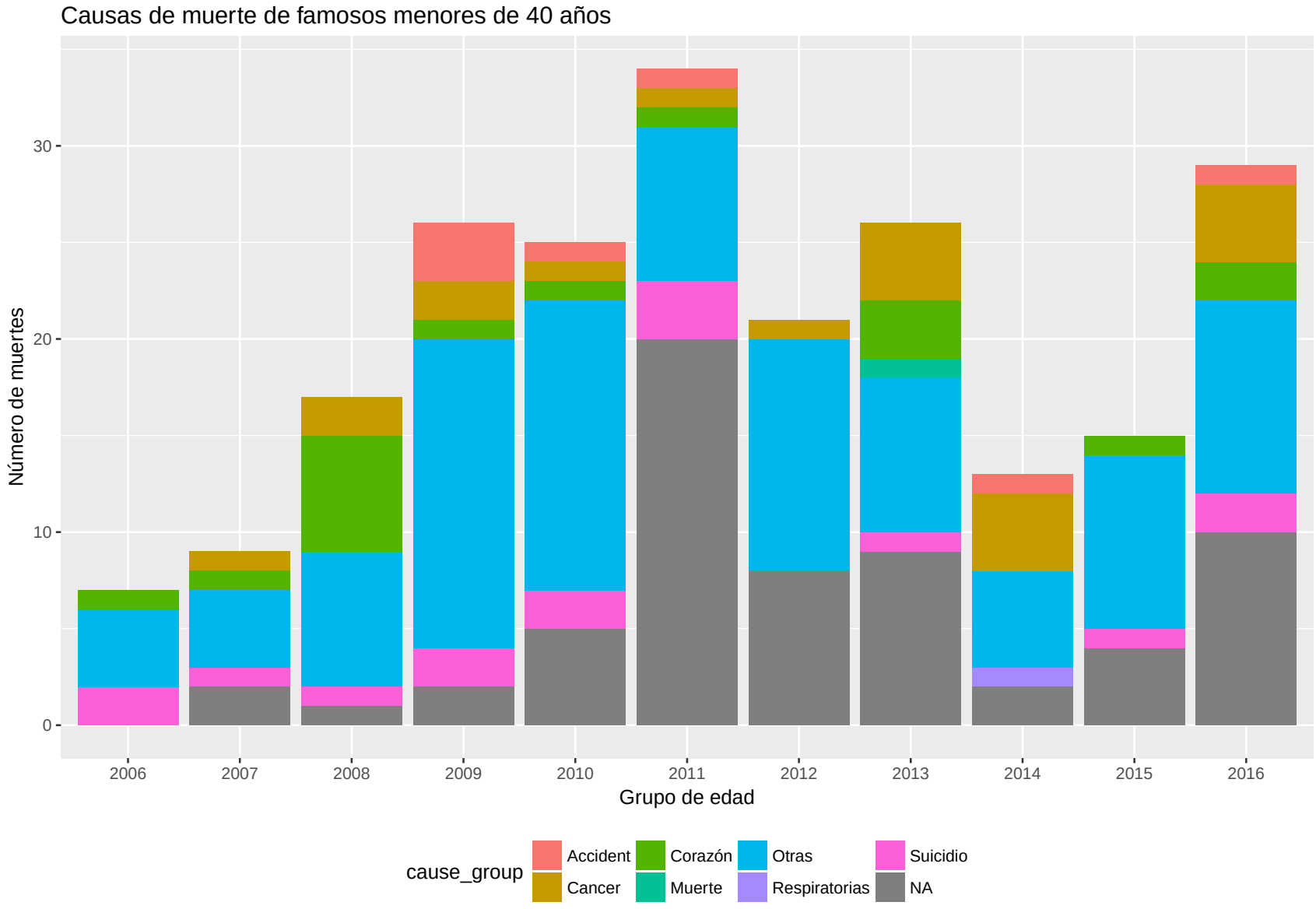
Se hace visualización comparativa

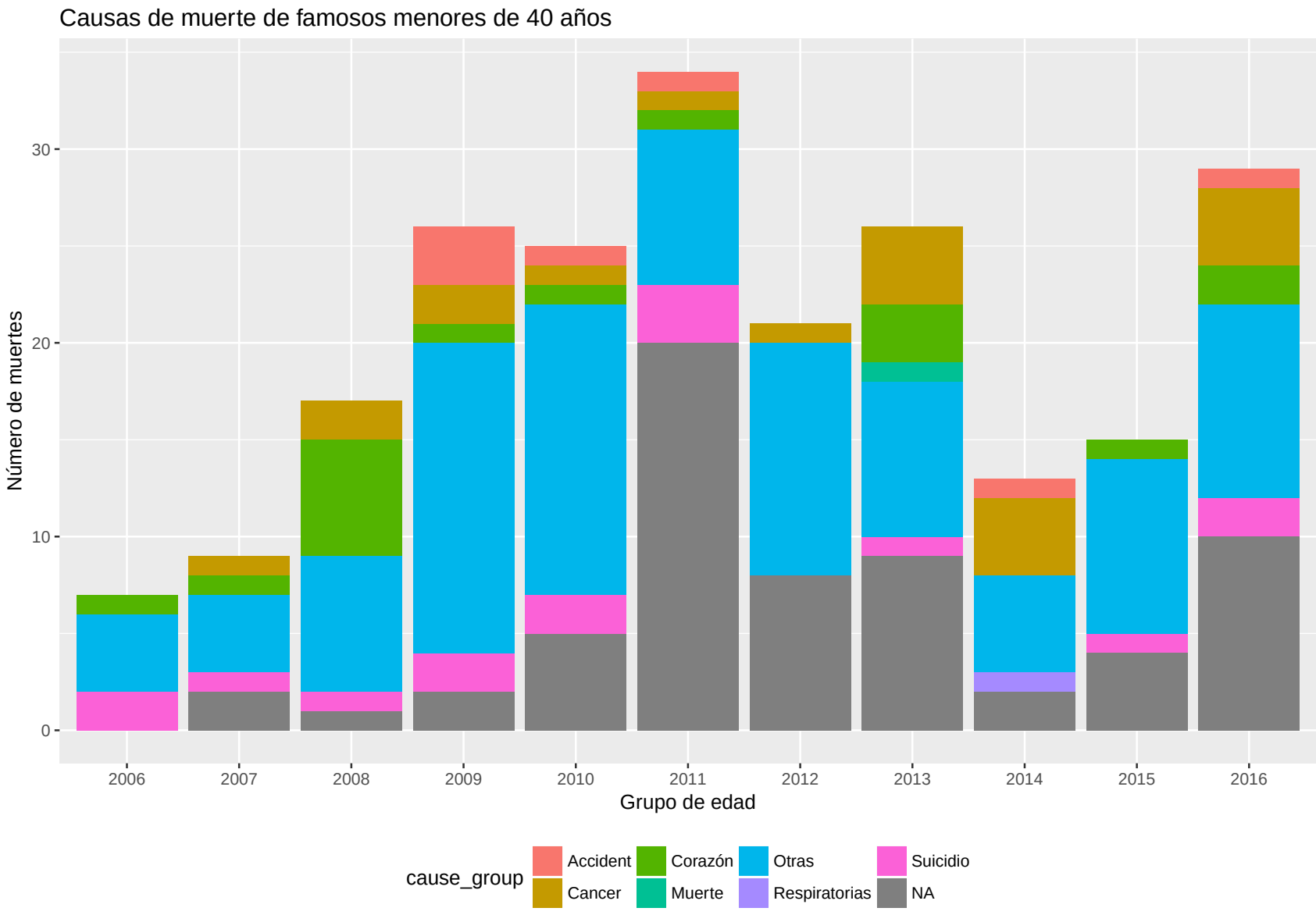


Una forma de verlo más elegante

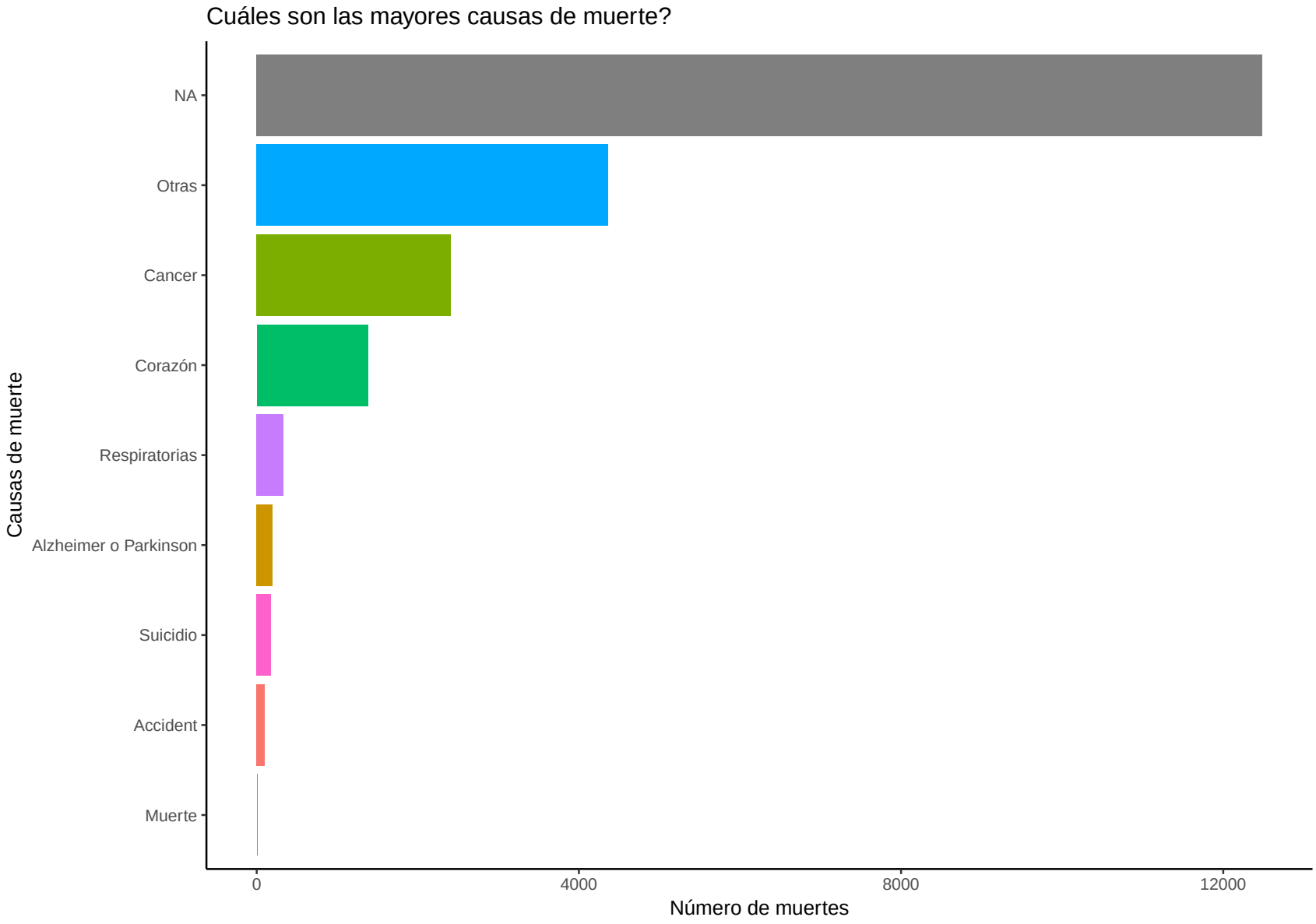


Causas de muertes a menores de 40




Causas de muerte de famosos menores de 40 años
Número de muertes
0
10
20
30
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
Grupo de edad
cause_group
Accident
Cancer
Corazón
Muerte
Otras
Respiratorias
Suicidio
NA

Cúal es la causa mayor de muertes?



age	birth_year	cause_of_death	death_month	death_year	far
85	1921	natural causes	January	2006	bus
49	1957	murdered	January	2006	mu
64	1942	Alzheimer's disease	January	2006	bas
86	1920	Alzheimer's disease	January	2006	po
82	1924	cancer	January	2006	nig

Gracias

Tarea

- Desarrollara una replica del código con la base de datos llamada **Restaurant & Market Health Data**
- Hará una presentación corta con lo que considere relevante
- La presentación se enviara al correo cjimenez187@aol.com