Práctica 05: RLS con variables categóricas

1. Regresión Lineal Simple con una variable categórica

Los modelos de regresión simple (RLS) vistos hasta ahora incorporan *variables independientes continuas*, pero se puede expandir a regresiones con *variables independientes categóricas*. De hecho, en general se puede expandir a variables binarias, indicadoras, cualitativas o dummy. Estas variables muestran la presencia de cierto atributo en cada uno de sus niveles.

Por ejemplo, la variable categórica de grupos de edad la podemos ver como:

```
knitr::opts_chunk$set(warning = FALSE, message = FALSE)
# La función kable permite visualizar un data.frame como tabla dentro de un *.RMD
library(knitr)
kable(
  data.frame(
    `Grupos de edad` = c("Bebes 0-3 años",
                         "Niños 3-10 años",
                         "Adolescentes 10-18 años",
                         "Adultos Jóvenes 18-30 años",
                         "Adultos Media Edad 30-45 años",
                         "Adultos Maduros 45-65 años",
                         "Tercera Edad 65+ años")
    )
  )
```

Grupos.de.edad
Bebes 0-3 años
Niños 3-10 años
Adolescentes 10-18 años
Adultos Jóvenes 18-30 años
Adultos Media Edad 30-45 años
Adultos Maduros 45-65 años
Tercera Edad 65+ años

Cada uno de los valores de la variable categórica los conocemos como *niveles*. Si la variable cuenta con n niveles, se puede codificar con $n-1$ variables binarias. Por ejemplo, la variable anterior cuenta con 7 niveles y la regresión se puede codificar con 6 variables binarias o dummy:

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 dummy1 + \hat{\beta}_2 dummy2 + \hat{\beta}_3 dummy3 + \hat{\beta}_4 dummy4 + \hat{\beta}_5 dummy5 + \hat{\beta}_6 dummy6$$

¿Qué valor debe tomar cada variable dummy de acuerdo a cada nivel de la variable categórica?

```
library(knitr)
kable(data.frame(grupos_edad = c("Bebes 0-3 años",
                                 "Niños 3-10 años",
                                 "Adolescentes 10-18 años",
                                 "Adultos Jóvenes 18-30 años",
                                 "Adultos Media Edad 30-45 años",
                                 "Adultos Maduros 45-65 años",
                                 "Tercera Edad 65+ años"),
                 dummy1 = c(0,1,0,0,0,0,0),
                 dummy2 = c(0,0,1,0,0,0,0),
                 dummy3 = c(0,0,0,1,0,0,0),
                 dummy4 = c(0,0,0,0,1,0,0),
                 dummy5 = c(0,0,0,0,0,1,0),
                 dummy6 = c(0,0,0,0,0,0,1),
                 descripción = c("Nivel Control", "","","","","","")))
```

grupos_edad	dummy1	dummy2	dummy3	dummy4	dummy5	dummy6	descripción
Bebes 0-3 años	0	0	0	0	0	0	Nivel Control
Niños 3-10 años	1	0	0	0	0	0	
Adolescentes 10-18 años	0	1	0	0	0	0	
Adultos Jóvenes 18-30 años	0	0	1	0	0	0	
Adultos Media Edad 30-45 años	0	0	0	1	0	0	
Adultos Maduros 45-65 años	0	0	0	0	1	0	
Tercera Edad 65+ años	0	0	0	0	0	1	

De manera que cuando la variable categórica toma el valor de *Adolescentes*, la ecuación nos quedaría como:

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 0 + \hat{\beta}_2 1 + \hat{\beta}_3 0 + \hat{\beta}_4 0 + \hat{\beta}_5 0 + \hat{\beta}_6 0 = \hat{\beta}_0 + \hat{\beta}_2$$

Note que en la tabla un nivel tiene una etiqueta de Nivel Control. Cuando se toma ese nivel (Bebés), entonces todas las βs excepto β_0 se vuelven 0:

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 0 + \hat{\beta}_2 0 + \hat{\beta}_3 0 + \hat{\beta}_4 0 + \hat{\beta}_5 0 + \hat{\beta}_6 0 = \hat{\beta}_0$$

2. Ejercicio clase Titanic:

2.1 Introducción:

El hundimiento del Titanic es uno de los accidentes marítimos con mayor fatalidad en la historia. En Abril de 1912 durante su primer viaje, el considerado imposible de hundir RMS Titanic, se hundió despues de colisionar con un iceberg. Desafortunadamente, no había suficientes botes salvavidas para todos en el barco, resultando en la muerte de 1502 de los 2224 pasajeros y tripulación. Aunque hubo factores de suerte involucrados para sobrevivir, parece ser que ciertos grupos de personas tenían mayores posibilidades de sobrevivir que otras. En el siguiente ejercicio, se busca evaluar como afectan cuatro variables categóricas en aumentar la posibilidad de sobrevivencia (Grupo de Edad, Género, Clase del boleto, Título de la persona)

Para este ejercicio, solo se toman 891 registros del total de pasajeros. (Nota: La variable a predecir es una variable binaria, por lo que lo técnicamente correcto sería aplicar una regresión logística, pero para fines del ejercicio, supondremos continua la variable.)

El dataset lo pueden encontrar en el website de *kaggle* y a continuación se presenta su diccionario de datos:

```
kable(data.frame(
  variable = c("survival", "pclass", "sex", "age", "sibsp",
               "parch", "ticket", "fare", "cabin", "embarked"),
  descripción = c("Sobrevivió", "Ticket class", "Sex", "Age in years", "# of siblings",
                  "# of parents", "Ticket number", "Passenger Fare", "Cabin Number",
                  "Port of Embarkation"),
  valores = c("0 = No, 1 = Yes", "1 = 1st, 2 = 2nd, 3 = 3rd", "", "", "", "",
              "", "", "", "C = Cherbourg, Q = Queenstown, S = Southampton")
))
```

variable	descripción	valores
survival	Sobrevivió	0 = No, 1 = Yes
pclass	Ticket class	1 = 1st, 2 = 2nd, 3 = 3rd
sex	Sex	
age	Age in years	
sibsp	# of siblings	
parch	# of parents	
ticket	Ticket number	
fare	Passenger Fare	
cabin	Cabin Number	
embarked	Port of Embarkation	C = Cherbourg, Q = Queenstown, S = Southampton

2.2 Preprocesamiento de datos y funciones para graficar:

Lectura de datos:

```
library(tidymodels)
library(readr)
library(gridExtra)

datos <-
  read_csv(
    "https://raw.githubusercontent.com/savrgg/class_ITAM_metodos/main/notas_r/train.csv"
    )
```

La variable Age viene en años, por lo que se agrupan en clases con la función *case_when*. Adicional, se extrae el título de la persona de su nombre:

```
datos_o <-
  datos %>%
  mutate(
    Survived_prev = Survived,
    Survived = factor(Survived, labels = c("No Sobrevivió", "Sobrevivió")),
    Age = case_when(
      Age < 3 ~ "[0-03) \n Bebés",
      Age < 10 ~ "[03-10) \n Niños/Niñas",
      Age < 18 ~ "[10-18) \n Adolescentes",
      Age < 30 ~ "[18-30) \n Adultos Jóvenes",
      Age < 45 ~ "[30-45) \n Adultos Media",
      Age < 65 ~ "[45-65) \n Adultos Maduros",
      Age < 120 ~ "[65+) \n Tercera Edad",
      TRUE ~ "No proporcionado"
      ),
    Title = gsub('(.*, )|(\\..*)', '', Name)
    )
```

Ahora, se procede a generar dos funciones auxiliares para graficar los datos (no es necesario que las examinen a detalle):

```
plot_general <- function(var_p){
  datos_o %>%
  ggplot(aes(x = factor(!!sym(var_p)), fill = Survived))+
  geom_bar()+
  geom_text(
    aes(label = after_stat(count)),
    stat = "count", position = position_stack(), vjust = 1, color = "gray50", size = 3)+
  scale_fill_brewer(palette = "Greens")+
  theme_minimal()+
  theme(legend.position = "bottom", text = element_text(size = 8))+
  labs(
    x = "",
    y = "Número de sobrevivientes",
    title = "Número de sobrevivientes en Titanic",
    subtitle = "891 pasajeros, 38.4% Sobreviven",
    caption = "Base extraída de: kaggle.com/competitions/titanic",
```

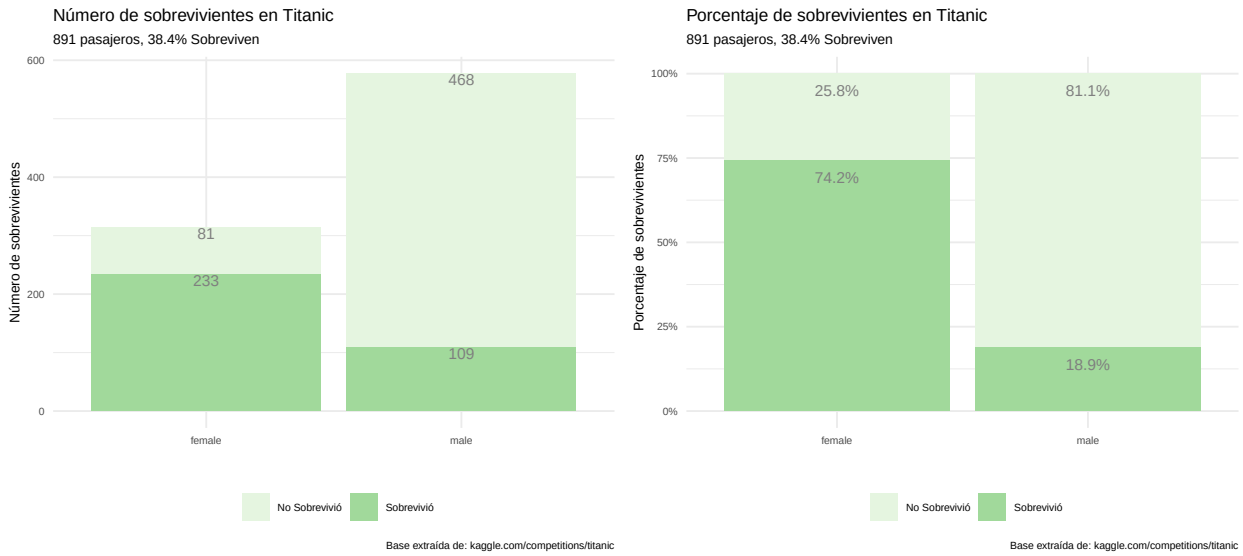
```
    fill = ""
  )
}
```

```r
plot_general_norm <- function(var_p){
  datos_o %>%
  ggplot(aes(x = factor(!!sym(var_p)), fill = Survived))+
  geom_bar(position = "fill", aes(fill = Survived))+
  geom_text(
    aes(label = scales::percent(after_stat(count/tapply(count, x, sum)[x])), group = Survived),
    stat = "count", position = position_fill(), vjust = 2, color = "gray50", size = 3)+
  scale_fill_brewer(palette = "Greens")+
  theme_minimal()+
  scale_y_continuous(labels = percent_format())+
  theme(legend.position = "bottom", text = element_text(size = 8))+
  labs(
    x = "",
    y = "Porcentaje de sobrevivientes",
    title = "Porcentaje de sobrevivientes en Titanic",
    subtitle = "891 pasajeros, 38.4% Sobreviven",
    caption = "Base extraída de: kaggle.com/competitions/titanic",
    fill = ""
  )
}
```

2.3 Regresión lineal Survived~Sex

Primero se realiza un EDA:

```
var_p <- "Sex"
# la función grid.arrange nos permite colocar dos o más gráficas en una
grid.arrange(
  plot_general(var_p),
  plot_general_norm(var_p), ncol = 2
)
```



Ahora se realiza la regresión lineal

```
lm_model <-
  linear_reg() %>%
  fit(Survived_prev ~ Sex, data = datos_o)

tidy(lm_model)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic   p.value
##   <chr>          <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)    0.742    0.0231      32.2 3.35e-151
## 2 Sexmale       -0.553    0.0287     -19.3 1.41e- 69
```

```
glance(lm_model)
```

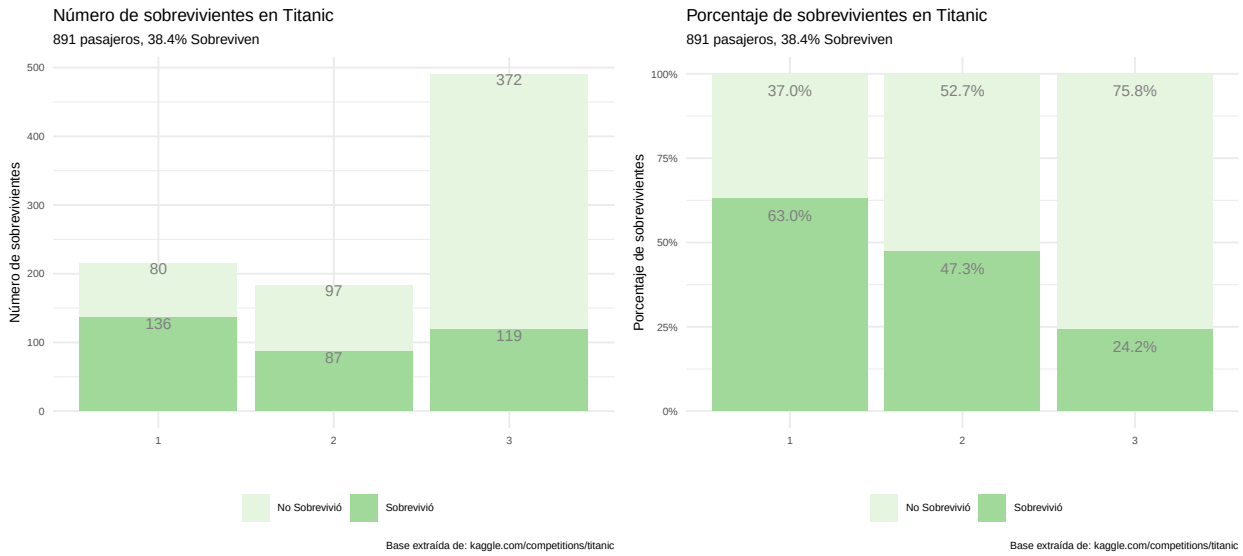
```
## # A tibble: 1 x 12
##   r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC
##       <dbl>         <dbl> <dbl>     <dbl>    <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1     0.295         0.294 0.409      372. 1.41e-69     1  -466.  938.  953.
## # ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

```
# como comprobación podemos agrupar por Sex y sacar el promedio de sobrevivientes:
# datos_o %>%
#   group_by(Sex) %>%
#   summarise(porc_survived = sum(Survived_prev)/n())
```

2.4 Regresión lineal Survived~Pclass

Primero se realiza un EDA:

```
var_p <- "Pclass"
grid.arrange(
  plot_general(var_p),
  plot_general_norm(var_p), ncol = 2
)
```



Ahora se realiza la regresión lineal:

```
lm_model <-
  linear_reg() %>%
  fit(Survived_prev  ~ factor(Pclass), data = datos_o)

tidy(lm_model)
```

```
## # A tibble: 3 x 5
##   term            estimate std.error statistic  p.value
##   <chr>              <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)        0.630    0.0312     20.2  6.08e-75
## 2 factor(Pclass)2   -0.157    0.0460     -3.41 6.75e- 4
## 3 factor(Pclass)3   -0.387    0.0374    -10.4  8.60e-24
```

```
glance(lm_model)
```

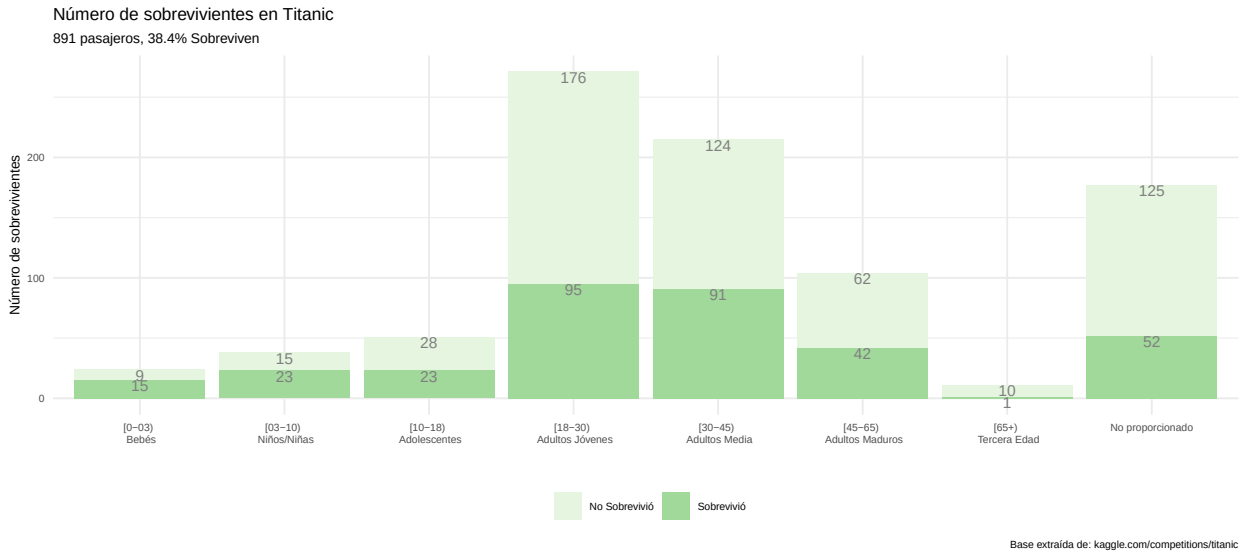
```
## # A tibble: 1 x 12
##   r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC
##       <dbl>         <dbl> <dbl>     <dbl>    <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1     0.115         0.113 0.458      58.0 2.18e-24     2  -567. 1143. 1162.
## # ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

```r
# como comprobación podemos agrupar por Pclass y sacar el promedio de sobrevivientes:
# datos_o %>%
#   group_by(Pclass) %>%
#   summarise(porc_survived = sum(Survived_prev)/n())
```

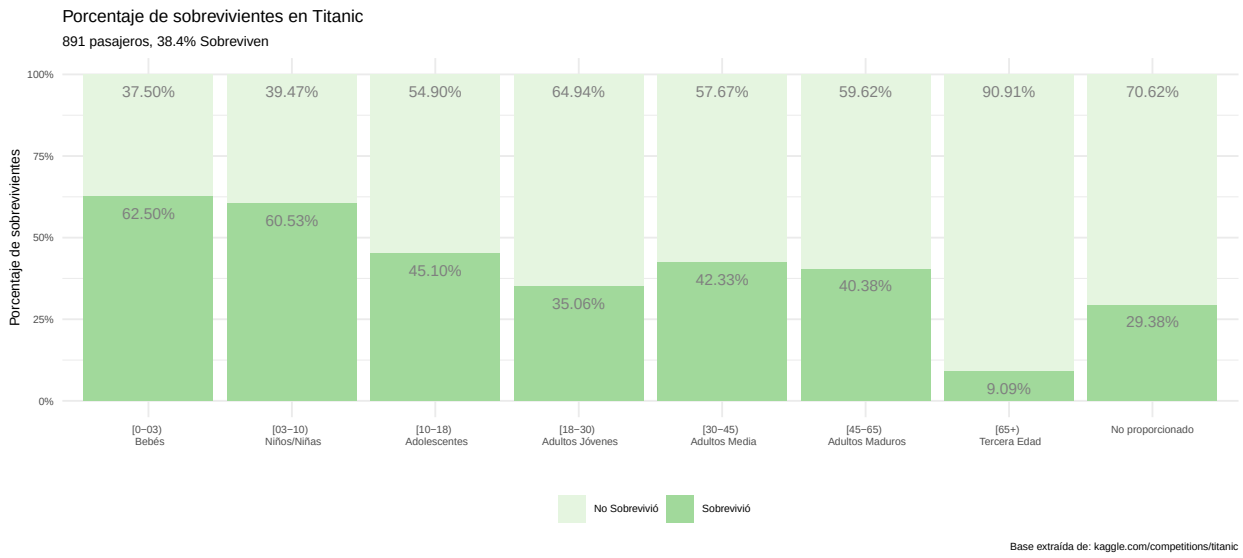
2.5 Regresión lineal Survived~Age

Primero se realiza un EDA:

```
var_p <- "Age"
plot_general(var_p)
```




```
plot_general_norm(var_p)
```



Ahora se realiza la regresión lineal:

```
lm_model <-
  linear_reg() %>%
  fit(Survived_prev  ~ Age, data = datos_o)

tidy(lm_model)
```

```
## # A tibble: 8 x 5
##   term                            estimate std.error statistic  p.value
##   <chr>                              <dbl>     <dbl>     <dbl>    <dbl>
## 1 "(Intercept)"                     0.625     0.0982     6.37  3.09e-10
## 2 "Age[03-10) \n Niños/Niñas"      -0.0197    0.125     -0.157 8.75e- 1
## 3 "Age[10-18) \n Adolescentes"     -0.174     0.119     -1.46  1.44e- 1
## 4 "Age[18-30) \n Adultos Jóvenes"  -0.274     0.102     -2.68  7.50e- 3
## 5 "Age[30-45) \n Adultos Media"    -0.202     0.103     -1.95  5.16e- 2
## 6 "Age[45-65) \n Adultos Maduros"  -0.221     0.109     -2.03  4.26e- 2
## 7 "Age[65+) \n Tercera Edad"       -0.534     0.175     -3.05  2.35e- 3
## 8 "AgeNo proporcionado"            -0.331     0.105     -3.17  1.60e- 3
```

```
glance(lm_model)
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC
##       <dbl>         <dbl> <dbl>     <dbl>    <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1    0.0311        0.0234 0.481      4.04 0.000226     7  -608. 1234. 1277.
## # ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

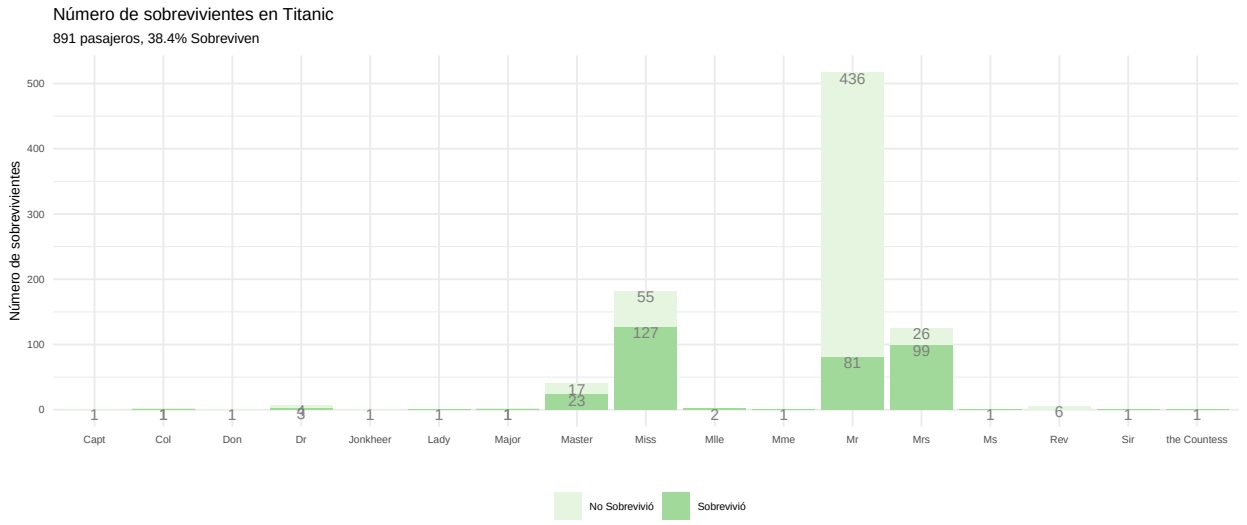
```
# como comprobación podemos agrupar por Age y sacar el promedio de sobrevivientes:
datos_o %>%
  group_by(Age) %>%
  summarise(porc_survived = sum(Survived_prev)/n())
```

```
## # A tibble: 8 x 2
##   Age                          porc_survived
##   <chr>                                <dbl>
## 1 "[0-03) \n Bebés"                   0.625
## 2 "[03-10) \n Niños/Niñas"            0.605
## 3 "[10-18) \n Adolescentes"           0.451
## 4 "[18-30) \n Adultos Jóvenes"        0.351
## 5 "[30-45) \n Adultos Media"          0.423
## 6 "[45-65) \n Adultos Maduros"        0.404
## 7 "[65+) \n Tercera Edad"             0.0909
## 8 "No proporcionado"                  0.294
```

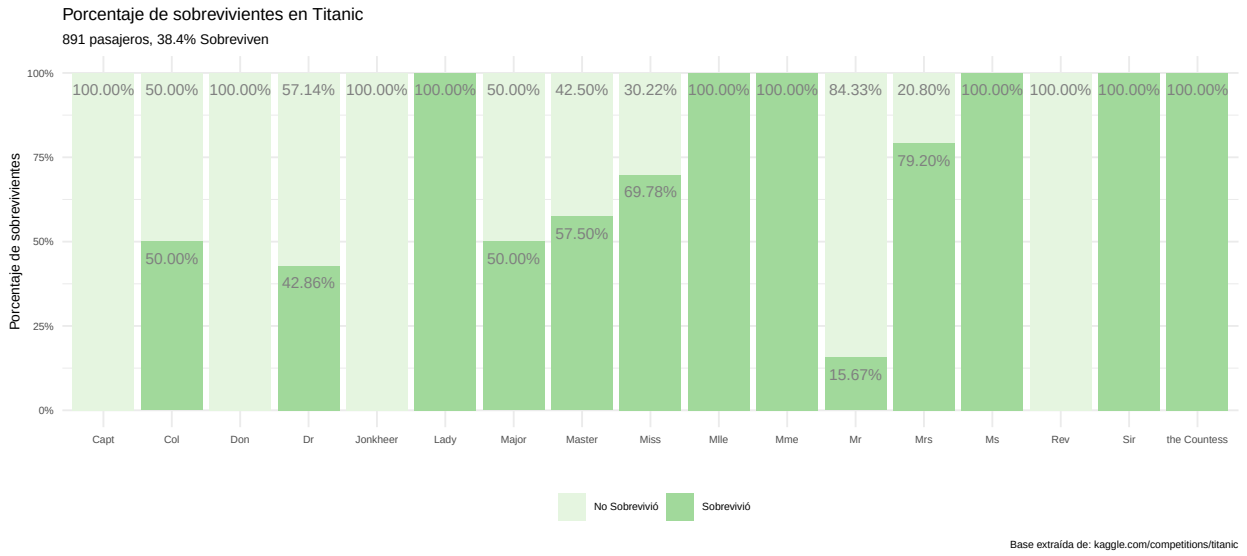
2.6 Regresión lineal Survived~Age

Primero se realiza un EDA:

```
var_p <- "Title"
plot_general(var_p)
```




```
plot_general_norm(var_p)
```



Ahora se realiza la regresión lineal:

```
lm_model <-
  linear_reg() %>%
  fit(Survived_prev  ~ Title, data = datos_o)

tidy(lm_model)
```

```
## # A tibble: 17 x 5
##    term               estimate std.error statistic p.value
##    <chr>                 <dbl>     <dbl>     <dbl>   <dbl>
##  1 (Intercept)        1.08e-13     0.400  2.71e-13  1.00
##  2 TitleCol           5.00e- 1     0.490  1.02e+ 0  0.308
##  3 TitleDon          -1.36e-13     0.566 -2.40e-13  1.00
##  4 TitleDr            4.29e- 1     0.428  1.00e+ 0  0.316
##  5 TitleJonkheer     -1.20e-13     0.566 -2.12e-13  1.00
##  6 TitleLady          1.00e+ 0     0.566  1.77e+ 0  0.0774
##  7 TitleMajor         5.00e- 1     0.490  1.02e+ 0  0.308
##  8 TitleMaster        5.75e- 1     0.405  1.42e+ 0  0.156
##  9 TitleMiss          6.98e- 1     0.401  1.74e+ 0  0.0822
## 10 TitleMlle          1.00e+ 0     0.490  2.04e+ 0  0.0415
## 11 TitleMme           1.00e+ 0     0.566  1.77e+ 0  0.0774
## 12 TitleMr            1.57e- 1     0.400  3.91e- 1  0.696
## 13 TitleMrs           7.92e- 1     0.401  1.97e+ 0  0.0489
## 14 TitleMs            1.00e+ 0     0.566  1.77e+ 0  0.0774
## 15 TitleRev          -1.15e-13     0.432 -2.66e-13  1.00
## 16 TitleSir           1.00e+ 0     0.566  1.77e+ 0  0.0774
## 17 Titlethe Countess  1.00e+ 0     0.566  1.77e+ 0  0.0774
```

```
glance(lm_model)
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC
##       <dbl>         <dbl> <dbl>     <dbl>    <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1     0.337         0.325 0.400      27.7 3.92e-67    16  -439.  914. 1000.
## # ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

```
# datos_o %>%
#   group_by(Title) %>%
#  summarise(porc_survived = sum(Survived_prev)/n())
```

3. Ejercicio para clase

Este toy dataset contiene variables que ayudan a predecir el precio de una casa. Factores como metros cuadrados del lote, metros cuadrados de construcción, número de recámaras, número de baños, alberca, etc influyen en el precio final de una casa. En el ejercicio que se realizará hoy se tomarán en cuenta solo las variables categóricas Street, House Style, Central Air, PoolArea:

```
kable(
  data.frame(
    variable = c("Street", "HouseStyle", "CentralAir", "PoolArea"),
    descripción = c("Tipo de acceso", "Número de Pisos",
                    "Cuenta con calefacción central", "Tiene alberca"),
    valores = c("Grvl: Gravel, Pave: Paved", "1Story: One story, 2Story: Two story",
                "N: No, Y: Yes", "")
  )
)
```

variable	descripción	valores
Street	Tipo de acceso	Grvl: Gravel, Pave: Paved
HouseStyle	Número de Pisos	1Story: One story, 2Story: Two story
CentralAir	Cuenta con calefacción central	N: No, Y: Yes
PoolArea	Tiene alberca	

3.1 Carga y preprocesamiento de datos

```
library(tidymodels)
library(readr)
library(gridExtra)

datos <-
  read_csv("https://raw.githubusercontent.com/savrgg/class_ITAM_metodos/main/notas_r/HousePrice.csv")

datos <-
  datos %>%
  select(Street, HouseStyle, CentralAir,
         PoolArea, SalePrice) %>%
  mutate(PoolArea = if_else(PoolArea > 0, "With-Pool", "No-Pool")) %>%
  filter(
    !is.na(Street),
    !is.na(HouseStyle),
    !is.na(CentralAir),
    !is.na(PoolArea),
    !is.na(SalePrice),
    HouseStyle %in% c("1Story", "2Story")
  )
datos %>% head(2)
```

```
## # A tibble: 2 x 5
##   Street HouseStyle CentralAir PoolArea SalePrice
##   <chr>  <chr>      <chr>      <chr>        <dbl>
```

```
## 1 Pave   2Story     Y          No-Pool     208500
## 2 Pave   1Story     Y          No-Pool     181500
```

3.2 SalePrice ~ Street

a) Realiza un análisis exploratorio para la variable independiente

b) Construye la regresión lineal

c) Interpreta los resultados

3.3 SalePrice ~ HouseStyle

a) Realiza un análisis exploratorio para la variable independiente

b) Construye la regresión lineal

c) Interpreta los resultados

3.4 SalePrice ~ CentralAir

a) Realiza un análisis exploratorio para la variable independiente

b) Construye la regresión lineal

c) Interpreta los resultados

3.5 SalePrice ~ PoolArea

a) Realiza un análisis exploratorio para la variable independiente

b) Construye la regresión lineal

c) Interpreta los resultados