

# Prueba Markdown

Angel Arcones

17/1/2022

## Intro a RMarkdown

Este script es un ejemplo básico de cómo utilizar **RMarkdown**.  
La idea es mostrar algunas de las funciones que ofrece para:

- Formatear texto para documentar, usando Markdown
- Organizar código de R en celdas
- Mostrar resultados del código de forma limpia y ordenada.

## ¿Para qué usar RMarkdown?

Es muy útil sobretodo para **organizar y documentar** scripts, ya que permite tener el código en bloques separados (*celdas*) que se pueden ejecutar independientemente, y permite además incluir texto con múltiples opciones de formateo.

Un ejemplo de cómo se puede visualizar un bloque de código y sus resultados:

```
for(i in 1:5){  
  print(paste0("Número ", i))  
}
```

```
## [1] "Número 1"  
## [1] "Número 2"  
## [1] "Número 3"  
## [1] "Número 4"  
## [1] "Número 5"
```

## La edición de texto permite además

Usar *cursiva*, ~~tachado~~, esto llamado **verbatim**, poner<sup>índices</sup> y también<sub>subíndices</sub>

Editar el **texto con colores**

Escribir formulas:  $y = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

Utilizar links a webs

## Ejemplos con código

Para comenzar, como se haría en cualquier script, lo primero es cargar las librerías necesarias. En este caso, vamos a usar el conjunto de librerías incluidas en **Tidyverse** para manejo y visualización de datos, y la librería **dslabs** que contiene datasets para usar.

En este caso, en las opciones de celda se selecciona para el output **Show nothing (Run Code)**, para que lo ejecute pero no muestre los mensajes de instalación/carga de librerías en documento

```
data(movielens)
head(movielens)
```

```
##      movieId                title year
## 1         31      Dangerous Minds 1995
## 2       1029             Dumbo 1941
## 3       1061          Sleepers 1996
## 4       1129  Escape from New York 1981
## 5       1172 Cinema Paradiso (Nuovo cinema Paradiso) 1989
## 6       1263      Deer Hunter, The 1978
##              genres userId rating  timestamp
## 1              Drama      1    2.5 1260759144
## 2 Animation|Children|Drama|Musical      1    3.0 1260759179
## 3              Thriller      1    3.0 1260759182
## 4 Action|Adventure|Sci-Fi|Thriller      1    2.0 1260759185
## 5              Drama      1    4.0 1260759205
## 6      Drama|War      1    2.0 1260759151
```

```
nrow(movielens)
```

```
## [1] 100004
```

```
ncol(movielens)
```

```
## [1] 7
```

```
colnames(movielens)
```

```
## [1] "movieId"  "title"    "year"     "genres"   "userId"   "rating"
## [7] "timestamp"
```

```
sort(unique(movielens$year))
```

```
##      [1] 1902 1915 1916 1917 1918 1919 1920 1921 1922 1923 1924 1925 1926 1927 1928
##     [16] 1929 1930 1931 1932 1933 1934 1935 1936 1937 1938 1939 1940 1941 1942 1943
##     [31] 1944 1945 1946 1947 1948 1949 1950 1951 1952 1953 1954 1955 1956 1957 1958
##     [46] 1959 1960 1961 1962 1963 1964 1965 1966 1967 1968 1969 1970 1971 1972 1973
##     [61] 1974 1975 1976 1977 1978 1979 1980 1981 1982 1983 1984 1985 1986 1987 1988
##     [76] 1989 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2003
##     [91] 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016
```

```
df_tabla <- movielens %>% filter(year>=2008) %>%
  group_by(year) %>%
  summarise("Películas por año" = n(), "Nota media"=round(mean(rating),2)) %>% as.data.frame()

df_tabla
```

```
##   year Películas por año Nota media
## 1 2008             1544      3.50
## 2 2009             1375      3.52
## 3 2010             1238      3.54
## 4 2011              926      3.40
## 5 2012              939      3.40
## 6 2013              839      3.36
## 7 2014              798      3.42
## 8 2015              535      3.37
## 9 2016              124      3.22
```

```
kable(df_tabla, caption = "Películas por año y nota media\n(2008-2016)", align="ccc")
```

Table 1: Películas por año y nota media (2008-2016)

year	Películas por año	Nota media
2008	1544	3.50
2009	1375	3.52
2010	1238	3.54
2011	926	3.40
2012	939	3.40
2013	839	3.36
2014	798	3.42
2015	535	3.37
2016	124	3.22

```
regresion <- lm(year~rating, data=movielens)
resultados <- coef(summary(regresion))
kable(resultados, caption="Regresión entre año y valoración", align = "cccc", digits = 2)
```

Table 2: Regresión entre año y valoración

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	1997.88	0.17	12066.16	0
rating	-1.71	0.04	-38.17	0

Se pueden encontrar mas opciones a utilizar con kable aquí.

```
df_plot <- movielens %>% filter(year>=2000) %>%
  group_by(year) %>%
  summarise("Películas" = n(),
            "Nota_media"=round(mean(rating),2),
```

```

      "Nota_maxima"=max(rating),
      "Nota_minima"=min(rating)) %>% as.data.frame()
df_plot

```

##	year	Peliculas	Nota_media	Nota_maxima	Nota_minima
## 1	2000	4054	3.36	5	0.5
## 2	2001	3442	3.46	5	0.5
## 3	2002	3090	3.46	5	0.5
## 4	2003	2508	3.44	5	0.5
## 5	2004	2879	3.49	5	0.5
## 6	2005	1852	3.45	5	0.5
## 7	2006	1786	3.51	5	0.5
## 8	2007	1606	3.55	5	0.5
## 9	2008	1544	3.50	5	0.5
## 10	2009	1375	3.52	5	0.5
## 11	2010	1238	3.54	5	0.5
## 12	2011	926	3.40	5	0.5
## 13	2012	939	3.40	5	0.5
## 14	2013	839	3.36	5	0.5
## 15	2014	798	3.42	5	0.5
## 16	2015	535	3.37	5	0.5
## 17	2016	124	3.22	5	0.5