

# Kickstarter. Análisis de proyectos de crowdfunding

*Adam Kepa*

*08 de junio de 2019*

## Contents

<b>1</b>	<b>Introducción</b>	<b>2</b>
1.1	Descripción del dataset . . . . .	2
1.1.1	Carga de datos . . . . .	2
1.2	Selección de datos . . . . .	3
<b>2</b>	<b>Limpieza de datos</b>	<b>5</b>
2.1	Valores perdidos . . . . .	5
2.2	Valores extremos . . . . .	7
2.2.1	Patrocinadores (backers) . . . . .	7
2.2.2	Cantidad objetivo (usd_goal_real) . . . . .	13
2.2.3	Cantidad recaudada (usd_pledged_real) . . . . .	16
2.2.4	Duración campaña (duration) . . . . .	20
<b>3</b>	<b>Análisis de datos</b>	<b>23</b>
3.1	Análisis de normalidad . . . . .	23
3.2	Reducción de dimensionalidad . . . . .	26
3.3	Análisis descriptivo . . . . .	28
3.4	Análisis inferencial . . . . .	34
3.4.1	Duración . . . . .	34
3.4.2	Dinero recaudado por categoría . . . . .	35
<b>4</b>	<b>Conclusión</b>	<b>38</b>
<b>5</b>	<b>Referencias y bibliografía</b>	<b>38</b>

# 1 Introducción

## 1.1 Descripción del dataset

Para esta práctica se ha elegido el set de datos [Kickstarter Projects](#) de Kaggle, con datos relativos a iniciativas que buscan financiación en la página [Kickstarter.com](#).

Es un set de datos interesante por el gran volumen de observaciones (más de 300K), y que se asemeja al volumen de datos que se puede manejar en proyectos reales de ciencia de datos. Por otro lado, es interesante por ser un set relativamente actual y por las variables disponibles. Éstas son tanto cuantitativas como cualitativas, y alguna de ellas cuenta con un número elevado de categorías.

El listado completo de variables es el siguiente:

1. **ID**. Numérico. Identificador de la iniciativa.
2. **Name**. Categórico. Nombre de la iniciativa.
3. **Main category**. Categórico. Categoría de la iniciativa (nivel 1).
4. **Category**. Categórico. Categoría de la iniciativa (nivel 2).
5. **Currency**. Categórico. Moneda en la que se realiza la recaudación.
6. **Deadline**. Fecha. Fecha en la que acaba la recaudación.
7. **Goal**. Numérico. Cantidad de dinero que se intenta recaudar.
8. **Launched**. Timestamp. Fecha y hora en la que se inició la iniciativa.
9. **Pledged**. Numérico. Dinero recaudado al cumplirse la fecha de fin.
10. **Backers**. Numérico. Número de patrocinadores que han participado en la iniciativa
11. **Country**. Categórico. País de origen de la iniciativa.
12. **USD Pledged**. Numérico. Conversión de la variable *Pledged* a la divisa USD, realizado por Kickstarter.com
13. **USD Pledged Real**. Numérico. Conversión de la variable *Pledged* a la divisa USD, realizado por el autor del dataset.
14. **USD Goal Real**. Numérico. Conversión de la variable *Goal* a la divisa USD, realizado por el autor del dataset.

El resultado de una iniciativa se indica en el campo *state*:

1. **State**. Categórico. Resultado de la iniciativa al finalizar el plazo.

El objetivo del análisis será tratar de entender qué variables tienen un impacto significativo en el resultado final o en la cantidad de dinero que consigue reunir un proyecto.

### 1.1.1 Carga de datos

Realizamos la carga de datos. En este caso, y puesto que tenemos variables de tipo fecha, vamos a cargar las variables categóricas como strings para realizar la conversión al tipo esperado a posteriori.

```
path <- "../csv/ks-projects-201801.csv"  
original <- read.csv(path, header=T, sep=",", encoding = "UTF-8", stringsAsFactors=FALSE)  
  
glimpse(original)
```

```
## Observations: 378,661  
## Variables: 15  
## $ ID              <int> 1000002330, 1000003930, 1000004038, 100000754...  
## $ name            <chr> "The Songs of Adelaide & Abullah", "Greeting ...  
## $ category        <chr> "Poetry", "Narrative Film", "Narrative Film",...  
## $ main_category   <chr> "Publishing", "Film & Video", "Film & Video",...  
## $ currency         <chr> "GBP", "USD", "USD", "USD", "USD", "USD", "US...  
## $ deadline         <chr> "2015-10-09", "2017-11-01", "2013-02-26", "20...  
## $ goal             <dbl> 1000, 30000, 45000, 5000, 19500, 50000, 1000,...
```

```

## $ launched      <chr> "2015-08-11 12:12:28", "2017-09-02 04:43:57",...
## $ pledged       <dbl> 0.00, 2421.00, 220.00, 1.00, 1283.00, 52375.0...
## $ state         <chr> "failed", "failed", "failed", "failed", "canc...
## $ backers        <int> 0, 15, 3, 1, 14, 224, 16, 40, 58, 43, 0, 100, ...
## $ country        <chr> "GB", "US", "US", "US", "US", "US", "US...
## $ usd.pledged    <dbl> 0.00, 100.00, 220.00, 1.00, 1283.00, 52375.00...
## $ usd_pledged_real <dbl> 0.00, 2421.00, 220.00, 1.00, 1283.00, 52375.0...
## $ usd_goal_real   <dbl> 1533.95, 30000.00, 45000.00, 5000.00, 19500.0...

```

Realizamos la conversión, y observamos que el tipo de dato es el esperado para cada una de las variables:

```

original$launched <- as.Date(original$launched)
original$deadline <- as.Date(original$deadline)
original$category <- as.factor(original$category)
original$main_category <- as.factor(original$main_category)
original$currency <- as.factor(original$currency)
original$state <- as.factor(original$state)
original$country <- as.factor(original$country)

glimpse(original)

## Observations: 378,661
## Variables: 15
## $ ID              <int> 1000002330, 1000003930, 1000004038, 100000754...
## $ name            <chr> "The Songs of Adelaide & Abullah", "Greeting ...
## $ category        <fct> Poetry, Narrative Film, Narrative Film, Music...
## $ main_category   <fct> Publishing, Film & Video, Film & Video, Music...
## $ currency        <fct> GBP, USD, USD, USD, USD, USD, USD, USD, ...
## $ deadline        <date> 2015-10-09, 2017-11-01, 2013-02-26, 2012-04-...
## $ goal             <dbl> 1000, 30000, 45000, 5000, 19500, 50000, 1000, ...
## $ launched         <date> 2015-08-11, 2017-09-02, 2013-01-12, 2012-03-...
## $ pledged          <dbl> 0.00, 2421.00, 220.00, 1.00, 1283.00, 52375.0...
## $ state            <fct> failed, failed, failed, failed, canceled, suc...
## $ backers           <int> 0, 15, 3, 1, 14, 224, 16, 40, 58, 43, 0, 100, ...
## $ country           <fct> GB, US, US, US, US, US, US, US, US, CA, U...
## $ usd.pledged       <dbl> 0.00, 100.00, 220.00, 1.00, 1283.00, 52375.00...
## $ usd_pledged_real  <dbl> 0.00, 2421.00, 220.00, 1.00, 1283.00, 52375.0...
## $ usd_goal_real     <dbl> 1533.95, 30000.00, 45000.00, 5000.00, 19500.0...

```

## 1.2 Selección de datos

En el listado de variables de la sección anterior vemos que el set de datos incluye información duplicada. La cantidad objetivo está disponible en la moneda original de la promoción (*Goal*), así como su conversión a USD (*USD Goal Real*). Por otro lado, la cantidad final recaudada está disponible en la divisa original (*Pledged*), la conversión a USD realizada por la plataforma (*USD Pledged*), y la conversión a USD realizada por el autor del set de datos (*USD Pledged Real*). Puesto que la información de estas variables es redundante, se va a mantener únicamente la versión estandarizada por el autor.

Por otro lado, para realizar la analítica de datos no son necesarios los campos que identifican las observaciones (*ID* y *Name*).

Finalmente, y puesto que no se va a realizar un análisis de series temporales, se puede descartar las variables de tipo fecha (*Launched* y *Deadline*). Sin embargo, a partir de estas variables es posible derivar un campo nuevo que podría tener influencia en el resultado de la recaudación de fondos: la duración de la campaña.

También puede ser interesante comprobar si una iniciativa ha conseguido recaudar el dinero que se proponía, por lo que también se derivará esta variable a partir de *Goal* y *Pledged*.

A continuación se deriva estos datos y se elimina las variables innecesarias. Estas modificaciones se realizarán sobre una copia del set original, por si fuese necesario realizar alguna comprobación más adelante sobre los datos originales.

```
# Copia del set de datos
mydata <- original
# Derivación de la duración
mydata$duration_tmp <- mydata$deadline - mydata$launched
mydata$duration <- as.numeric(mydata$duration_tmp, units="days")
original$duration <- as.numeric(mydata$duration_tmp, units="days")
# Derivación de goal reached
mydata$goal_reached <- mydata$usd_pledged_real >= mydata$usd_goal_real
# Borrado de las variables innecesarias
mydata <- dplyr::select(mydata, -ID, -name, -goal, -pledged, -usd.pledged,
                       -launched, -deadline, -duration_tmp)
```

Como se puede observar en la tabla siguiente, se ha reducido el set a nueve inputs y la etiqueta de clase:

```
# Resumen de los datos seleccionados
glimpse(mydata)
```

```
## Observations: 378,661
## Variables: 10
## $ category      <fct> Poetry, Narrative Film, Narrative Film, Music...
## $ main_category <fct> Publishing, Film & Video, Film & Video, Music...
## $ currency       <fct> GBP, USD, USD, USD, USD, USD, USD, ...
## $ state          <fct> failed, failed, failed, failed, canceled, suc...
## $ backers         <int> 0, 15, 3, 1, 14, 224, 16, 40, 58, 43, 0, 100, ...
## $ country         <fct> GB, US, US, US, US, US, US, US, US, CA, U...
## $ usd_pledged_real <dbl> 0.00, 2421.00, 220.00, 1.00, 1283.00, 52375.0...
## $ usd_goal_real   <dbl> 1533.95, 30000.00, 45000.00, 5000.00, 19500.0...
## $ duration        <dbl> 59, 60, 45, 30, 56, 35, 20, 45, 35, 30, 30, 3...
## $ goal_reached    <lgl> FALSE, FALSE, FALSE, FALSE, TRUE, TRUE...
```

## 2 Limpieza de datos

### 2.1 Valores perdidos

En primer lugar, vamos a comprobar si existen observaciones con el valor NA en alguna variable:

```
colSums(is.na(mydata))
```

```
##          category      main_category      currency      state
##            0                  0                  0                  0
##      backers      country usd_pledged_real usd_goal_real
##            0                  0                  0                  0
##      duration      goal_reached
##            0                  0
```

Vemos que no existen valores sin imputar. Vamos a comprobar también cuales son los posibles valores de las variables de tipo “Factor” para comprobar si los valores nulos se han reemplazado por alguna etiqueta:

```
unique(mydata$category)
```

```
## [1] Poetry           Narrative Film   Music
## [4] Film & Video    Restaurants     Food
## [7] Drinks           Product Design Documentary
## [10] Nonfiction     Indie Rock      Crafts
## [13] Games           Tabletop Games  Design
## [16] Comic Books    Art Books      Fashion
## [19] Childrenswear  Theater        Comics
## [22] DIY             Webseries     Animation
## [25] Food Trucks    Public Art     Illustration
## [28] Photography    Pop           People
## [31] Art             Family        Fiction
## [34] Accessories    Rock          Hardware
## [37] Software       Weaving       Gadgets
## [40] Web             Jazz          Ready-to-wear
## [43] Festivals      Video Games   Anthologies
## [46] Publishing     Shorts        Electronic Music
## [49] Radio & Podcasts Apps          Cookbooks
## [52] Apparel         Metal         Comedy
## [55] Hip-Hop        Periodicals   Dance
## [58] Technology     Painting      World Music
## [61] Photobooks     Drama         Architecture
## [64] Young Adult    Latin         Mobile Games
## [67] Flight          Fine Art      Action
## [70] Playing Cards  Makerspaces  Punk
## [73] Thrillers      Children's Books Audio
## [76] Performance Art Ceramics     Vegan
## [79] Graphic Novels Fabrication Tools Performances
## [82] Sculpture      Sound         Stationery
## [85] Print           Farmer's Markets Events
## [88] Classical Music Graphic Design  Spaces
## [91] Country & Folk Wearables     Mixed Media
## [94] Journalism      Movie Theaters  Animals
## [97] Digital Art    Horror        Knitting
## [100] Small Batch   Installations Community Gardens
## [103] DIY Electronics Embroidery   Camera Equipment
## [106] Jewelry        Farms        Conceptual Art
```

```

## [109] Fantasy           Webcomics          Experimental
## [112] Science Fiction   Puzzles            R&B
## [115] Music Videos      Calendars          Video
## [118] Plays              Blues              Bacon
## [121] Faith              Live Games         Woodworking
## [124] Places             Footwear           3D Printing
## [127] Academic           Zines              Musical
## [130] Workshops          Photo              Immersive
## [133] Letterpress        Gaming Hardware    Candles
## [136] Television          Space Exploration Couture
## [139] Nature              Robots             Typography
## [142] Crochet             Translations       Textiles
## [145] Pottery             Interactive Design Video Art
## [148] Quilts              Glass              Pet Fashion
## [151] Printing            Romance            Civic Design
## [154] Kids                Literary Journals Taxidermy
## [157] Literary Spaces     Chiptune          Residencies
## 159 Levels: 3D Printing Academic Accessories Action Animals ... Zines
unique(mydata$main_category)

```

```

## [1] Publishing  Film & Video Music      Food      Design
## [6] Crafts      Games      Comics        Fashion   Theater
## [11] Art         Photography Technology Dance     Journalism
## 15 Levels: Art Comics Crafts Dance Design Fashion Film & Video ... Theater
unique(mydata$currency)

```

```

## [1] GBP USD CAD AUD NOK EUR MXN SEK NZD CHF DKK HKD SGD JPY
## Levels: AUD CAD CHF DKK EUR GBP HKD JPY MXN NOK NZD SEK SGD USD
unique(mydata$state)

```

```

## [1] failed      canceled    successful live      undefined suspended
## Levels: canceled failed live successful suspended undefined
unique(mydata$country)

```

```

## [1] GB   US   CA   AU   NO   IT   DE   IE   MX   ES   N,O" SE   FR   NL
## [15] NZ   CH   AT   DK   BE   HK   LU   SG   JP
## 23 Levels: AT AU BE CA CH DE DK ES FR GB HK IE IT JP LU MX N,O" NL ... US

```

Vemos que la etiqueta de clase (variable *state*) tiene seis posibles valores entre los que se encuentran “undefined” y “live”. El primero se corresponde a una etiqueta que se ha dado a los valores perdidos, y el segundo a campañas que estaban en activo cuando se hizo la recopilación de datos. Las observaciones de este segundo caso no aportan valor para predecir el resultado de una campaña al tratarse de observaciones de campañas no finalizadas, y por tanto habrá que eliminarlas. En cuanto al primer caso, sería posible imputar valores o eliminar también las observaciones asociadas. Para decidir entre una u otra, vamos a comprobar el porcentaje de observaciones de esta clase en una tabla de frecuencias:

```
tabyl(mydata$state)
```

```

## mydata$state      n      percent
## canceled  38779 0.102410864
## failed    197719 0.522153060
## live      2799 0.007391836
## successful 133956 0.353762336
## suspended  1846 0.004875073

```

```
##      undefined 3562 0.009406831
```

Las observaciones con estado “*undefined*” suponen menos del 1% del total. Por ello, y por el hecho de que el número total de observaciones es muy amplio, vamos a eliminar las observaciones asociadas:

```
mydata <- subset(mydata, state != "undefined" & state != "live")
# Creamos un factor nuevo para eliminar las categorías descartadas
mydata$state <- as.factor(as.character(mydata$state))
# Tabla de frecuencias después del ajuste
tabyl(mydata$state)
```

```
##   mydata$state      n    percent
##     canceled 38779 0.104160623
##     failed   197719 0.531074402
##     successful 133956 0.359806608
##     suspended    1846 0.004958367
```

Por otro lado, vemos que existe un valor anómalo (“N,0”) en la variable *country*. Este valor probablemente se deba a un error a la hora de hacer el scrapping de los datos. Comprobamos el número de observaciones afectadas:

```
nrow(subset(mydata, (country == "N,0\")))
```

```
## [1] 234
```

Y procedemos de forma análoga a la anterior: el número de observaciones afectadas es despreciable comparado con el total, y por tanto se pueden descartar sin problemas:

```
mydata <- subset(mydata, (country != "N,0\""))
# Creamos un factor nuevo para eliminar la categoría incorrecta
mydata$country <- as.factor(as.character(mydata$country))
```

Para el resto de variables cualitativas no se observa ningún valor que pueda representar un valor perdido o erróneo.

## 2.2 Valores extremos

Vamos a analizar las variables cuantitativas en búsqueda de valores extremos.

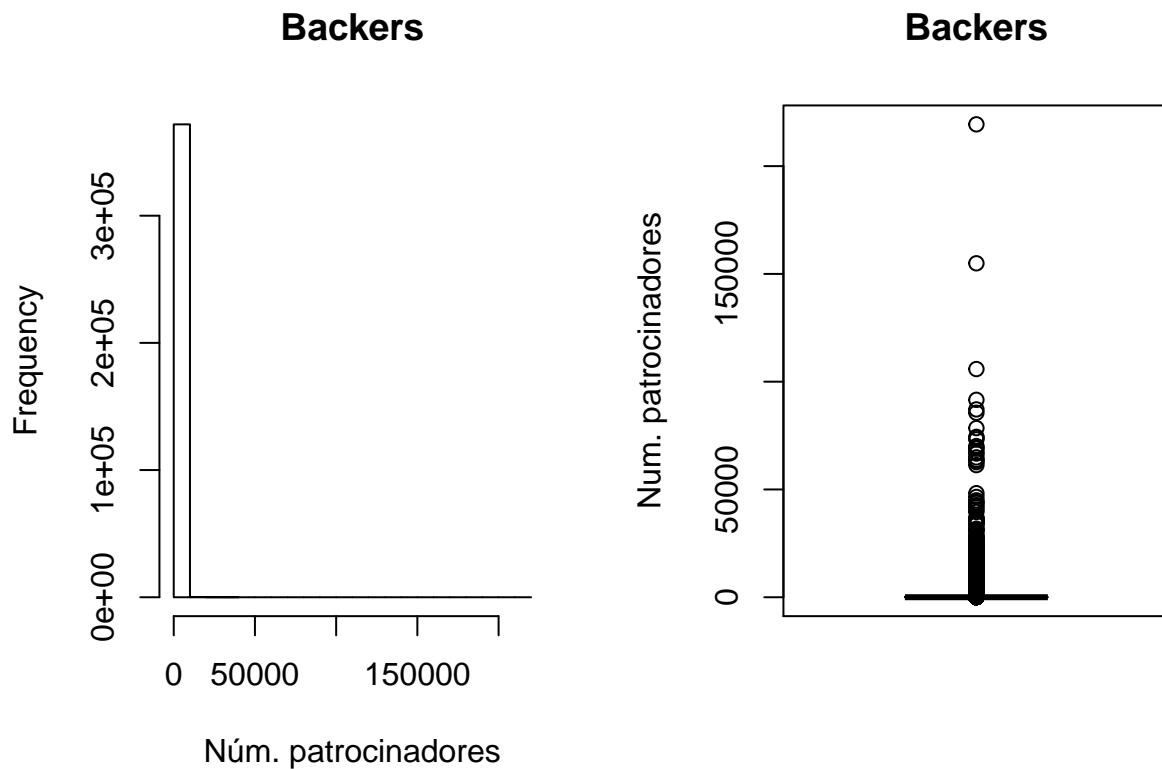
### 2.2.1 Patrocinadores (backers)

Esta variable representa al número de patrocinadores que han aportado dinero a la iniciativa. Vamos a comprobar la distribución de los valores:

```
summary(mydata$backers)

##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
##          0       2      12     107      57 219382

par(mfrow=c(1,2))
hist(mydata$backers,
     main="Backers",
     xlab="Núm. patrocinadores")
boxplot(mydata$backers,
        main = "Backers",
        ylab = "Num. patrocinadores")
```



Vemos que existen valores muy extremos. El 75% de los datos (3er cuartil) está por debajo de 57 patrocinadores. Sin embargo, existen valores superiores a 200000. Vamos a comprobar a cuantas iniciativas han contribuído más de 100000 personas:

```
nrow(subset(mydata, backers > 100000))
```

```
## [1] 3
```

Comprobamos estas observaciones en el set original:

```
subset(original, backers > 100000)
```

```
##          ID                               name
## 75901 1386523707      Fidget Cube: A Vinyl Desk Toy
## 187653 1955357092      Exploding Kittens
## 292245 557230947 Bring Reading Rainbow Back for Every Child, Everywhere!
##           category main_category currency   deadline   goal launched
## 75901 Product Design       Design      USD 2016-10-20 15000 2016-08-30
## 187653 Tabletop Games     Games      USD 2015-02-20 10000 2015-01-20
## 292245        Web Technology    Technology USD 2014-07-02 1000000 2014-05-28
##      pledged      state backers country usd.pledged usd_pledged_real
## 75901 6465690 successful 154926     US      13770      6465690
## 187653 8782572 successful 219382     US      8782572      8782572
## 292245 5408917 successful 105857     US      5408917      5408917
##      usd_goal_real duration
## 75901         15000      51
## 187653         10000      31
## 292245        1000000     35
```

Acudiendo a Kickstarter, comprobamos que estos valores son correctos, y que se corresponden a las iniciativas que históricamente han tenido más éxito en esta plataforma:

- Fidget Cube: A Vinyl Desk Toy
- Exploding Kittens
- Bring Reading Rainbow Back for Every Child, Everywhere!

Por tanto, son observaciones legítimas que no se deben descartar. Sin embargo, y por intuición, las iniciativas con un número alto de patrocinadores salen adelante. Vamos a identificar la observación con mayor número de patrocinadores que NO ha salido adelante:

```
aux <- mydata[mydata$state != "successful",]
aux <- aux[order(-aux$backers),]
head(aux, n = 1)

##           category main_category currency      state backers country
## 264292 Technology     Technology      USD suspended  20632      US
##          usd_pledged_real usd_goal_real duration goal_reached
## 264292          4005111        160000       28         TRUE
```

Vamos a comprobar cuántas observaciones existen con un mayor número de patrocinadores (y que han salido adelante):

```
nrow(mydata[mydata$backers > 20632,])
```

```
## [1] 76
```

Únicamente 76. Esto quiere decir que imputando el valor 20632 a todas estas observaciones obtendríamos el mismo resultado y eliminaríamos valores extremos. Sin embargo, para valores inferiores, seguramente la frecuencia de las iniciativas que han tenido éxito es mucho superior al resto de resultados:

```
tabyl(mydata[mydata$backers > 100,]$state)
```

```
##  mydata[mydata$backers > 100, ]$state      n      percent
##                      canceled  2086  0.034946056
##                      failed   5535  0.092725993
##                      successful 51855  0.868709375
##                      suspended  216  0.003618575
```

```
tabyl(mydata[mydata$backers > 250,]$state)
```

```
##  mydata[mydata$backers > 250, ]$state      n      percent
##                      canceled   674  0.026731181
##                      failed    1333  0.052867455
##                      successful 23094  0.915919727
##                      suspended   113  0.004481637
```

```
tabyl(mydata[mydata$backers > 500,]$state)
```

```
##  mydata[mydata$backers > 500, ]$state      n      percent
##                      canceled   234  0.018541997
##                      failed    379  0.030031696
##                      successful 11946  0.946592710
##                      suspended   61  0.004833597
```

```
tabyl(mydata[mydata$backers > 1000,]$state)
```

```
##  mydata[mydata$backers > 1000, ]$state      n      percent
##                      canceled   76  0.012432521
##                      failed    95  0.015540651
```

```

##                               successful 5915 0.967610011
##                               suspended   27 0.004416817
tabyl(mydata[mydata$backers > 2000,]$state)

##  mydata[mydata$backers > 2000, ]$state      n      percent
##                                         canceled   25 0.009633911
##                                         failed    34 0.013102119
##                                         successful 2519 0.970712909
##                                         suspended  17 0.006551060

```

Vemos que a partir de 1000 patrocinadores, la frecuencia de éxito es de prácticamente un 95%. Por tanto, si imputamos el 1000 como valor máximo al número de patrocinadores conseguiremos eliminar valores extremos sin apenas pérdida de información.

```

nrow(mydata[mydata$backers > 1000,])

## [1] 6113
mydata <- within(mydata, backers[backers > 1000] <- 1000)

```

Volvemos a comprobar la distribución de los valores:

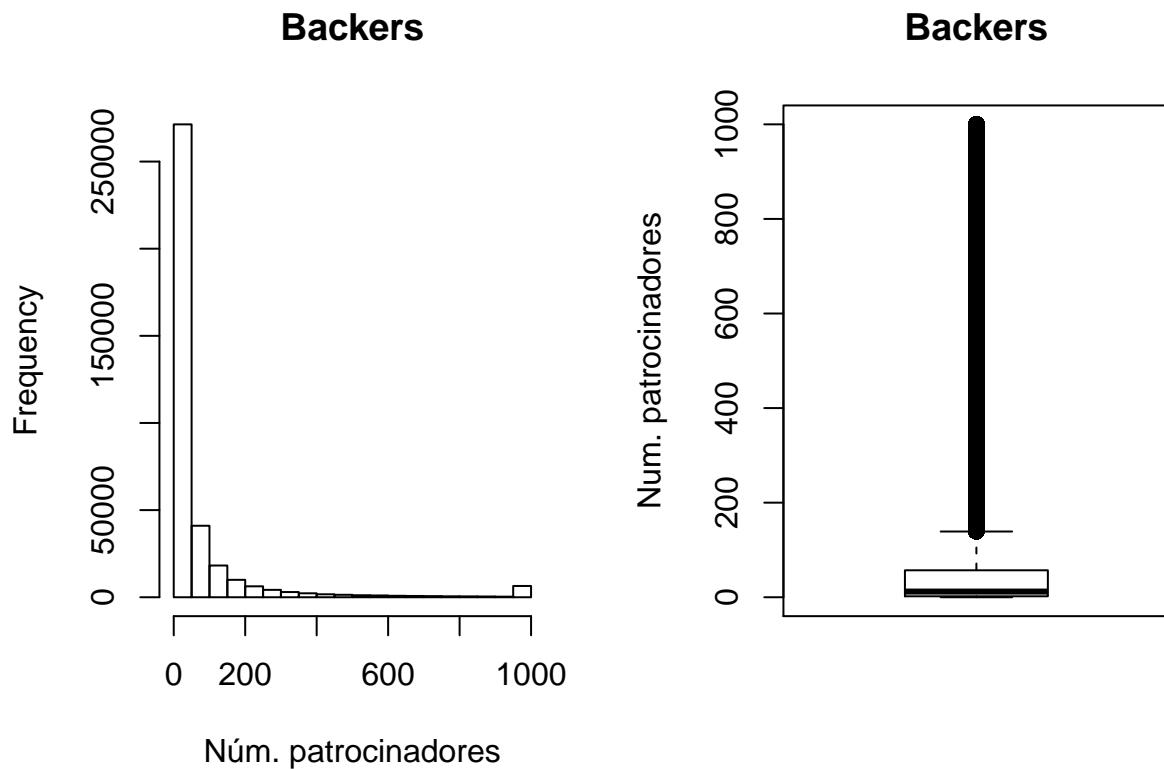
```

summary(mydata$backers)

##      Min.   1st Qu.   Median     Mean  3rd Qu.   Max.
##      0.00    2.00   12.00   70.44   57.00 1000.00

par(mfrow=c(1,2))
hist(mydata$backers,
     main="Backers",
     xlab="Núm. patrocinadores")
boxplot(mydata$backers,
        main = "Backers",
        ylab = "Num. patrocinadores")

```



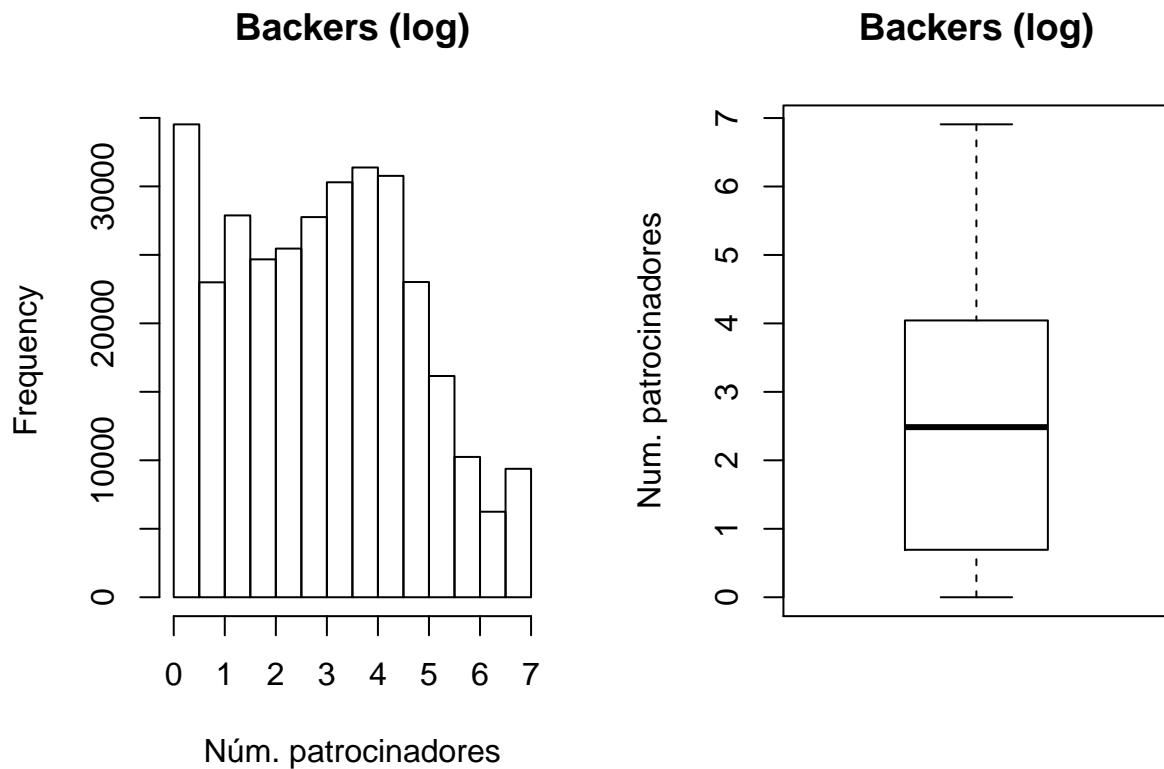
Vemos que de esta forma sigue habiendo valores extremos. Sin embargo, esto probablemente se deba a que los datos no siguen una distribución normal. Aplicando el logaritmo, parece que los datos se distribuyen de una forma más homogénea, y se puede comprobar que no existen datos fuera del rango intercuartílico:

```
summary(log(mydata$backers))

##      Min. 1st Qu. Median     Mean 3rd Qu.    Max.
##      -Inf  0.6931  2.4849     -Inf  4.0431  6.9078

par(mfrow=c(1,2))

hist(log(mydata$backers),
     main="Backers (log)",
     xlab="Núm. patrocinadores")
boxplot(log(mydata$backers),
        main = "Backers (log)",
        ylab = "Num. patrocinadores")
```



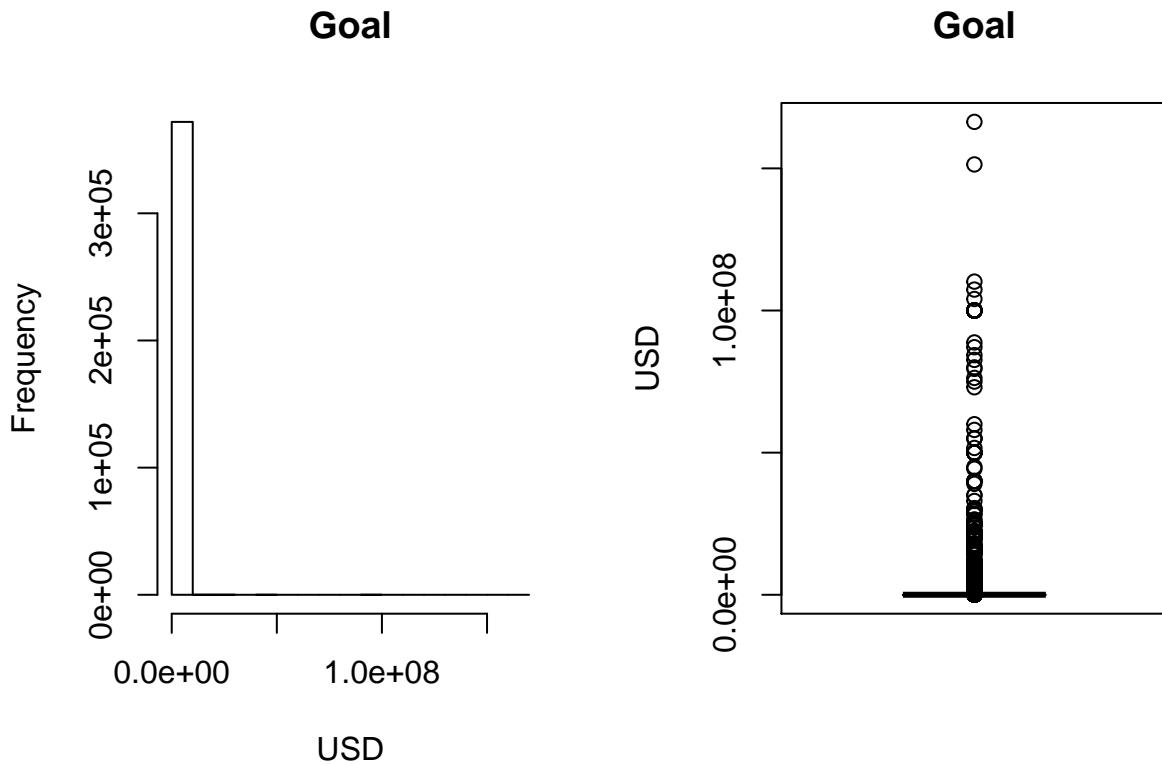
### 2.2.2 Cantidad objetivo (usd\_goal\_real)

Esta variable representa la cantidad que tienen como objetivo recaudar los proyectos. Vamos a observar la distribución de los valores:

```
summary(mydata$usd_goal_real)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max. 
##      0       2000     5500    45738   16000 166361391

par(mfrow=c(1,2))
hist(mydata$usd_goal_real,
     main="Goal",
     xlab="USD")
boxplot(mydata$usd_goal_real,
        main = "Goal",
        ylab = "USD")
```



Vemos que en este caso también existen valores extremos. Comprobamos las observaciones asociadas:

```
aux <- original[order(-original$usd_goal_real),]
head(aux, n = 3)
```

```
##           ID
## 47804 1243678698
## 196532 2000749004
## 367929 944541075
##                                     name
## 47804          FUCK Potato Salad. Paleo Potato Brownies!
```

```

## 196532                               A Celtic Lovestory
## 367929 Hydroponic's Skyscraper(un gratte-ciel hydroponique)e-solar
##           category main_category currency   deadline   goal  launched
## 47804        Food          Food      GBP 2014-08-08 9.9e+07 2014-07-09
## 196532     Drama  Film & Video      GBP 2015-11-30 1.0e+08 2015-11-17
## 367929 Technology  Technology      EUR 2015-10-24 1.0e+08 2015-08-25
##           pledged state backers country usd.pledged usd_pledged_real
## 47804        0 failed       0    GB      0.00          0.0
## 196532        0 failed       0    GB      0.00          0.0
## 367929        2 failed       2    FR      2.29          2.2
##           usd_goal_real duration
## 47804      166361391       30
## 196532      151395870       13
## 367929      110169772       60

```

De forma análoga a la variable de *backers*, acudimos a Kickstarter para verificar que, efectivamente, son iniciativas que intentaron recaudar esa cantidad de dinero:

- [FUCK Potato Salad. Paleo Potato Brownies!](#)
- [A Celtic Lovestory](#)
- [Hydroponic's Skyscraper](#)

Acudiendo a estos enlaces podemos ver que, aparentemente, son proyectos que se crearon como broma. Por ello, vamos a comprobar si es posible eliminar valores extremos de la misma forma que para la variable *Backers*. Vamos a comprobar la frecuencia con de los resultados, filtrando por la cantidad objetivo.

```
tabyl(mydata[mydata$usd_goal_real > 100000,]$state)
```

```

##  mydata[mydata$usd_goal_real > 1e+05, ]$state    n    percent
##                                         canceled 2722 0.21644402
##                                         failed 9019 0.71715967
##                                         successful 734 0.05836514
##                                         suspended 101 0.00803117

```

```
tabyl(mydata[mydata$usd_goal_real > 1000000,]$state)
```

```

##  mydata[mydata$usd_goal_real > 1e+06, ]$state    n    percent
##                                         canceled 225 0.20930233
##                                         failed 827 0.76930233
##                                         successful 11 0.01023256
##                                         suspended 12 0.01116279

```

```
tabyl(mydata[mydata$usd_goal_real > 10000000,]$state)
```

```

##  mydata[mydata$usd_goal_real > 1e+07, ]$state    n    percent
##                                         canceled 32 0.21768707
##                                         failed 111 0.75510204
##                                         successful 0 0.00000000
##                                         suspended 4 0.02721088

```

```
tabyl(mydata[mydata$usd_goal_real > 50000000,]$state)
```

```

##  mydata[mydata$usd_goal_real > 5e+07, ]$state    n    percent
##                                         canceled 6 0.13636364
##                                         failed 36 0.81818182
##                                         successful 0 0.00000000
##                                         suspended 2 0.04545455

```

En este caso no observamos un patrón tan significativo como en la variable anterior, por lo que no se realizará

la imputación de un valor máximo para evitar la pérdida de información.

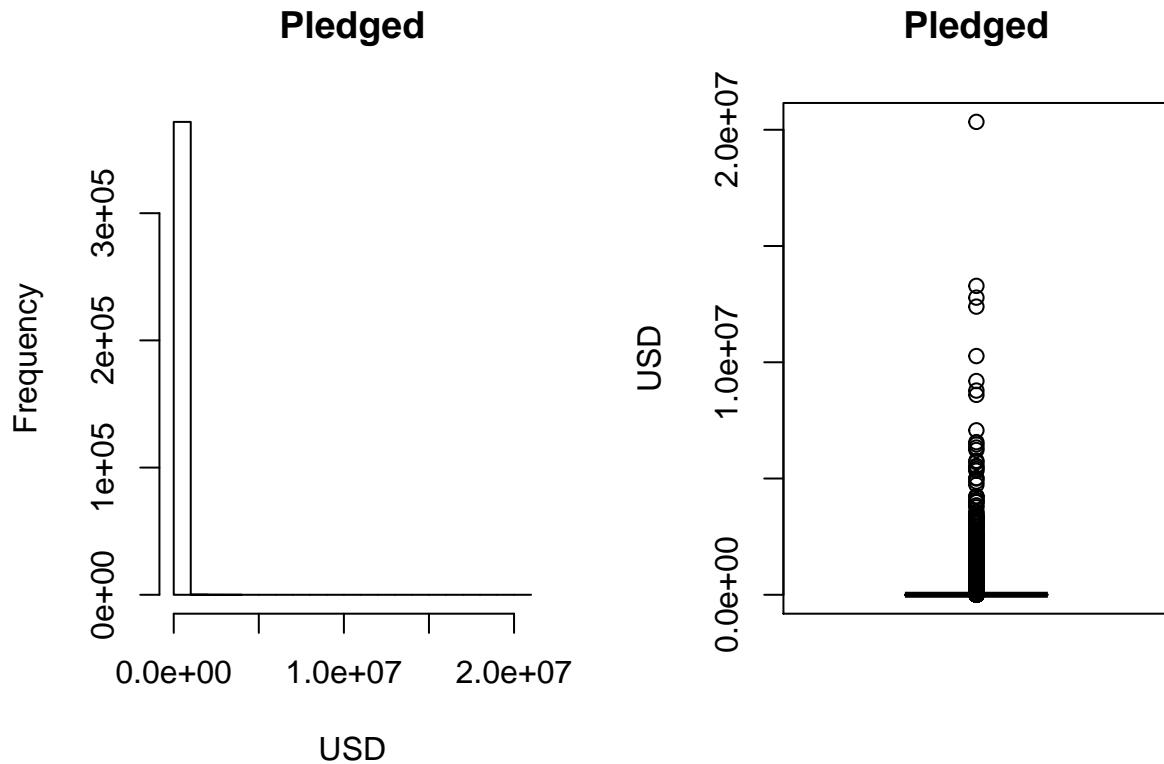
### 2.2.3 Cantidad recaudada (usd\_pledged\_real)

Esta variable es la cantidad de dinero recaudado para poder realizar el proyecto. Vamos a comprobar la distribución de los valores:

```
summary(mydata$usd_pledged_real)

##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
##          0       31      628     9145    4066 20338986

par(mfrow=c(1,2))
hist(mydata$usd_pledged_real,
     main="Pledged",
     xlab="USD")
boxplot(mydata$usd_pledged_real,
        main = "Pledged",
        ylab = "USD")
```



Al igual que en el caso anterior, también existen valores extremos (basta con observar la gran diferencia entre la media, la mediana y el valor máximo). Vamos a extraer las observaciones con los valores más altos de esta variable:

```
aux <- original[order(-original$usd_pledged_real),]
head(aux, n = 2)
```

```
##           ID
## 157271 1799979574
## 250255 342886736
##                                     name
```

```

## 157271 Pebble Time - Awesome Smartwatch, No Compromises
## 250255 COOLEST COOLER: 21st Century Cooler that's Actually Cooler
##           category main_category currency   deadline   goal   launched
## 157271 Product Design      Design      USD 2015-03-28 5e+05 2015-02-24
## 250255 Product Design      Design      USD 2014-08-30 5e+04 2014-07-08
##       pledged     state backers country usd.pledged usd_pledged_real
## 157271 20338986 successful    78471     US 20338986          20338986
## 250255 13285226 successful    62642     US 13285226          13285226
##       usd_goal_real duration
## 157271          5e+05      32
## 250255          5e+04      53

```

Acudiendo a Kickstarter vemos que estos valores extremos son correctos y se corresponden también a proyectos con mucho éxito.

- Pebble Time - Awesome Smartwatch
- COOLEST COOLER

Vemos si es posible imputar un valor máximo:

```
tabyl(mydata[mydata$usd_pledged_real > 10000,]$state)
```

```

## mydata[mydata$usd_pledged_real > 10000, ]$state      n      percent
##                                         canceled 1963 0.038438944
##                                         failed  5218 0.102177489
##                                         successful 43676 0.855251821
##                                         suspended 211 0.004131746

```

```
tabyl(mydata[mydata$usd_pledged_real > 100000,]$state)
```

```

## mydata[mydata$usd_pledged_real > 1e+05, ]$state      n      percent
##                                         canceled  78 0.015869786
##                                         failed   122 0.024821974
##                                         successful 4684 0.953001017
##                                         suspended  31 0.006307223

```

A partir de 100000 USD recaudados, el 95% de los proyectos acaban con éxito, por lo que podemos imputar este valor máximo para reducir los valores extremos sin perder apenas información en los datos:

```
nrow(mydata[mydata$usd_pledged_real > 100000,])
```

```

## [1] 4915
mydata <- within(mydata, usd_pledged_real[usd_pledged_real > 100000] <- 100000)

```

Volvemos a visualizar los datos, y vemos que no parecen ajustarse a una distribución normal:

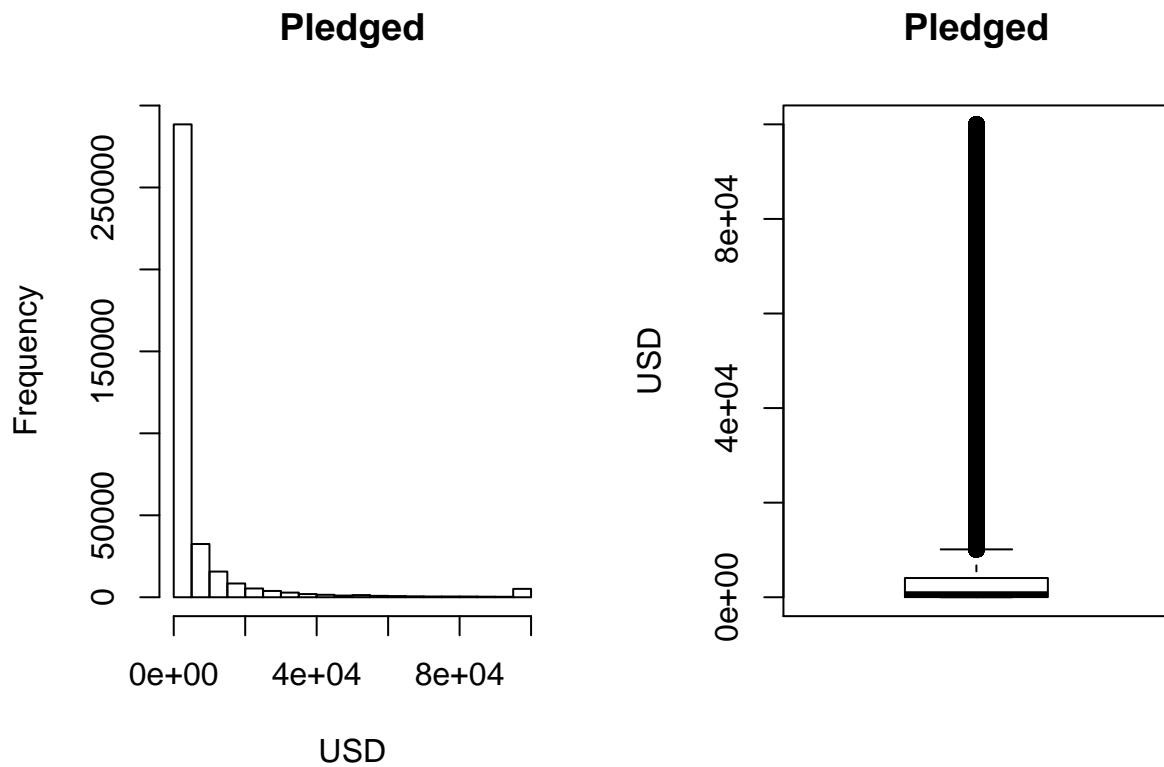
```
summary(mydata$usd_pledged_real)
```

```

##      Min.   1st Qu.   Median   Mean   3rd Qu.   Max.
##      0.00    31.24   627.59  5805.95  4066.00 100000.00

par(mfrow=c(1,2))
hist(mydata$usd_pledged_real,
     main="Pledged",
     xlab="USD")
boxplot(mydata$usd_pledged_real,
        main = "Pledged",
        ylab = "USD")

```

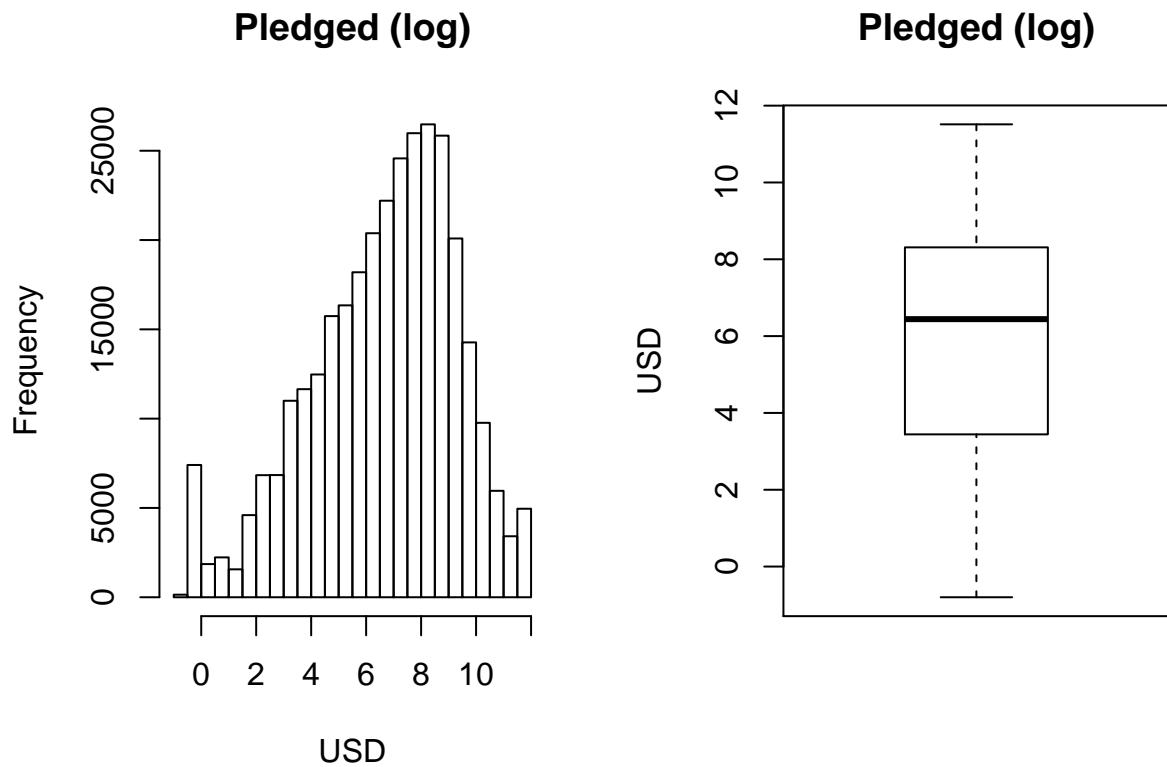


Sin embargo, en el caso de aplicar el logaritmo, la distribución se asemeja a lo que podría ser una normal, con un aumento de frecuencia en el valor 0.

```
summary(log(mydata$usd_pledged_real))

##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
## -Inf     3.442   6.442     -Inf    8.310   11.513

par(mfrow=c(1,2))
hist(log(mydata$usd_pledged_real),
  main="Pledged (log)",
  xlab="USD")
boxplot(log(mydata$usd_pledged_real),
  main = "Pledged (log)",
  ylab = "USD")
```



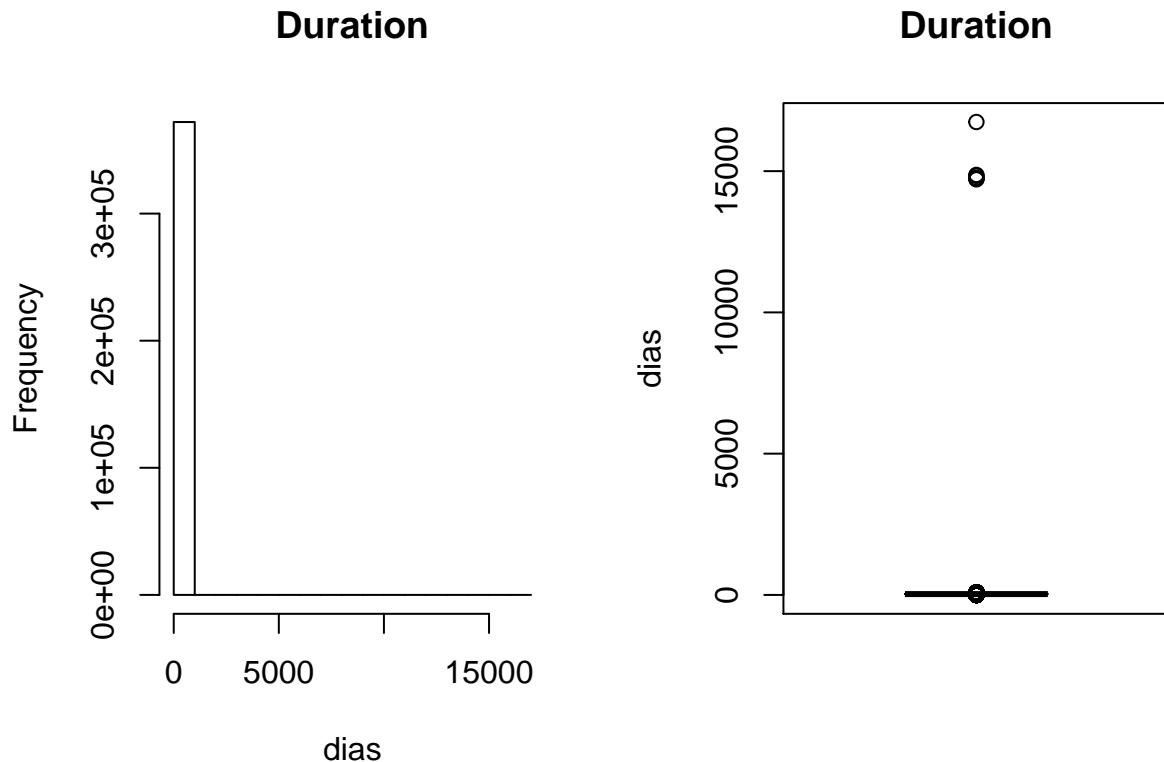
## 2.2.4 Duración campaña (duration)

Vamos a comprobar la distribución de valores de la variable derivada al inicio:

```
summary(mydata$duration)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max. 
##      1.00    30.00    30.00   34.46    37.00 16739.00 

par(mfrow=c(1,2))
hist(mydata$duration,
     main="Duration",
     xlab="días")
boxplot(mydata$duration,
        main = "Duration",
        ylab = "días")
```



Vemos que en este caso también existen valores muy extremos. El valor máximo son 16739 días (45 años), por lo que ha de tratarse de un valor erróneo. Vamos a comprobar cuantas iniciativas hay con más 100 días de duración:

```
nrow(subset(mydata, duration > 100))
```

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

Únicamente 7, por lo que probablemente se deba a un error en los datos iniciales o en el cálculo de la variable. Para comprobarlo, vamos a examinar las observaciones en el set original:

```
subset(original, duration > 100)
```

```

##          ID
## 2843    1014746686
## 48148   1245461087
## 75398   1384087152
## 94580   1480763647
## 247914  330942060
## 273780  462917959
## 319003  69489148
##                               name
## 2843           Salt of the Earth: A Dead Sea Movie (Canceled)
## 48148          1st Super-Size Painting - Social Network Owned (Canceled)
## 75398          "ICHOR" (Canceled)
## 94580          Support Solo Theater! Help "Ungrateful Daughter" Project Development! (Canceled)
## 247914        Help RIZ Make A Charity Album: 8 Songs, 8 Causes, 1 Song For Each Cause (Canceled)
## 273780          Identity Communications Infographic (Canceled)
## 319003          Student Auditions Music 2015
##      category main_category currency deadline goal launched
## 2843  Film & Video  Film & Video     USD 2010-09-15 5000 1970-01-01
## 48148   Art          Art            USD 2010-08-14 15000 1970-01-01
## 75398  Film & Video  Film & Video     USD 2010-05-21  700 1970-01-01
## 94580   Theater      Theater         USD 2010-06-01  4000 1970-01-01
## 247914   Music        Music          USD 2010-05-04 10000 1970-01-01
## 273780   Design       Design         USD 2010-04-10   500 1970-01-01
## 319003 Publishing    Publishing      CHF 2015-10-31 1900 1970-01-01
##      pledged state backers country usd.pledged usd_pledged_real
## 2843      0 canceled      0   US      0          0
## 48148      0 canceled      0   US      0          0
## 75398      0 canceled      0   US      0          0
## 94580      0 canceled      0   US      0          0
## 247914      0 canceled      0   US      0          0
## 273780      0 canceled      0   US      0          0
## 319003      0 suspended     0   CH      0          0
##      usd_goal_real duration
## 2843      5000.00    14867
## 48148     15000.00   14835
## 75398      700.00    14750
## 94580     4000.00    14761
## 247914    10000.00   14733
## 273780      500.00    14709
## 319003    1905.97    16739

```

Vemos que el error se debe a que la fecha de inicio de algunas iniciativas está mal informada (1970-01-01). Se trata del valor inicial del formato `epoch`, por lo que seguramente se trate de valores perdidos. Eliminamos estas observaciones, y volvemos a analizar la distribución de valores:

```
# Eliminamos observaciones con valores incorrectos
mydata <- subset(mydata, duration < 100)
```

```
summary(mydata$duration)
```

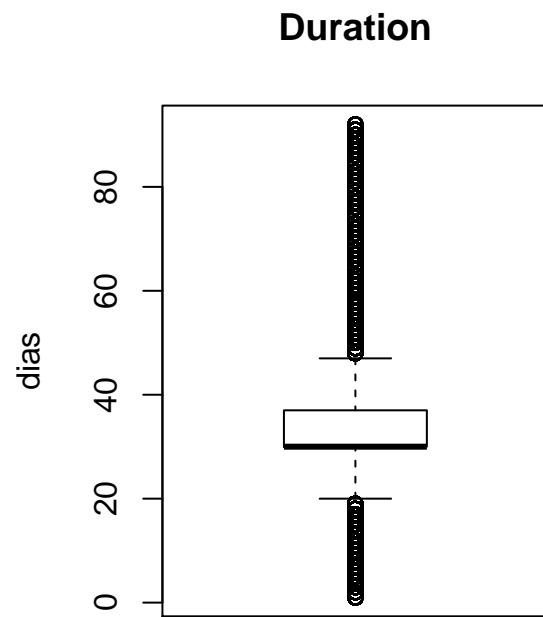
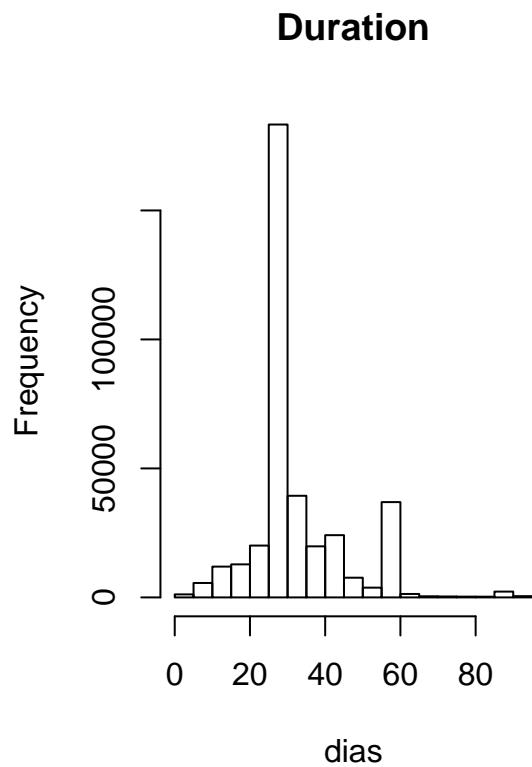
```
##      Min. 1st Qu. Median  Mean 3rd Qu.  Max.
##      1.00  30.00  30.00  34.18  37.00  92.00
```

```
par(mfrow=c(1,2))
hist(mydata$duration,
     main="Duration",
```

```

xlab="dias")
boxplot(mydata$duration,
        main = "Duration",
        ylab = "dias")

```



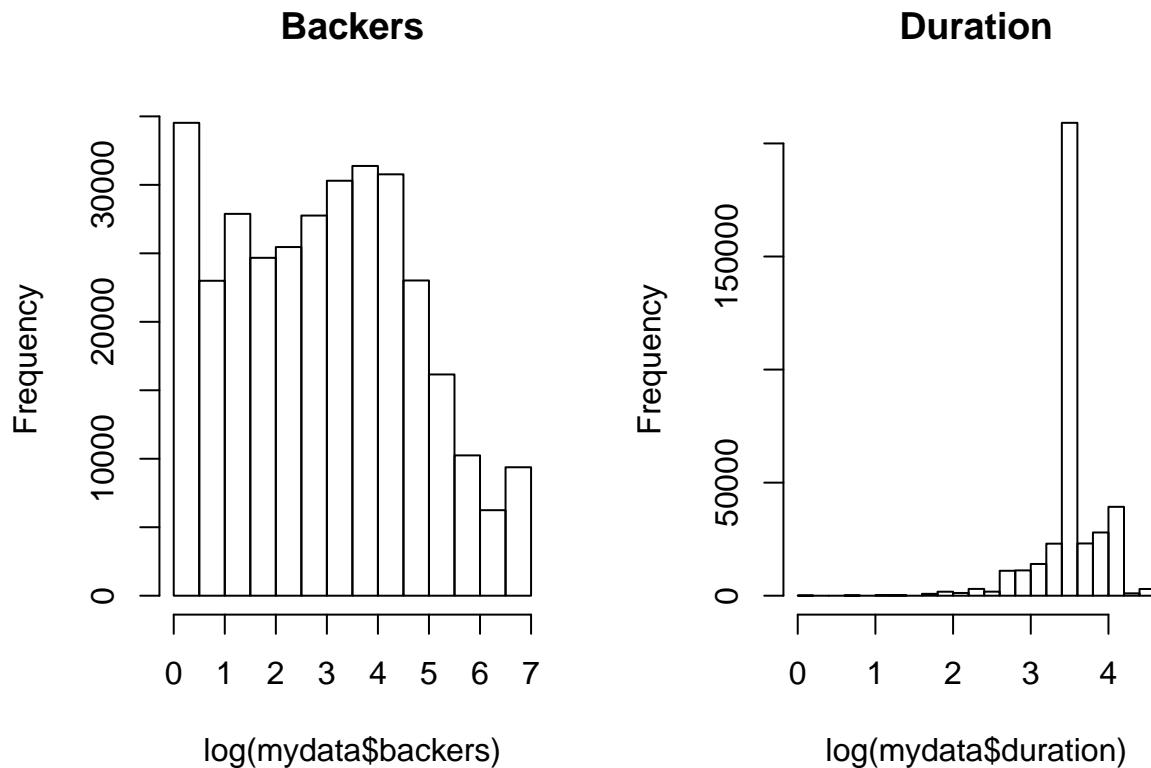
Vemos que ahora existen valores extremos, pero correctos: la campaña más larga se sitúa en torno a los 3 meses, mientras que tanto el valor medio como el mediano están en torno a un mes.

### 3 Análisis de datos

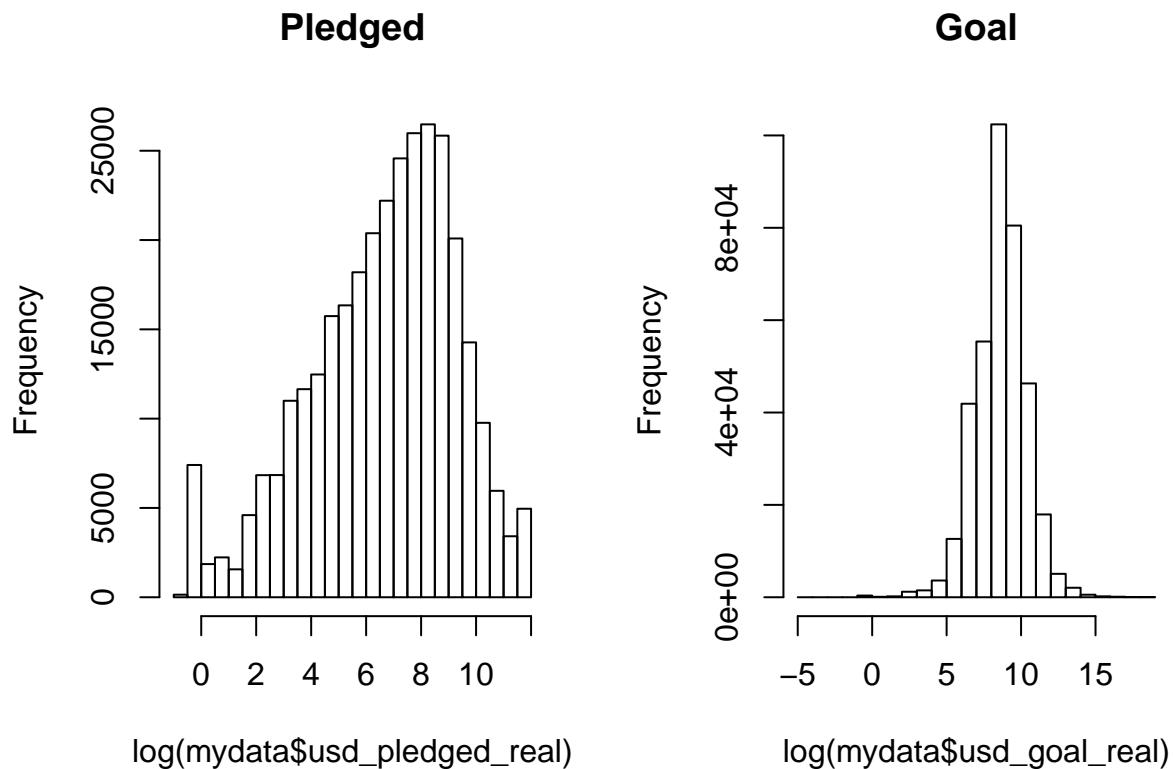
#### 3.1 Análisis de normalidad

En el apartado de valores extremos se ha comprobado a simple vista que las variables no siguen una distribución normal. Sin embargo, dada la forma del histograma es posible que sigan una distribución LogNormal o exponencial. Para ello, visualizamos en primer lugar el histograma del logaritmo de estas variables:

```
par(mfrow=c(1,2))
hist(log(mydata$backers), main = "Backers")
hist(log(mydata$duration), main = "Duration")
```



```
hist(log(mydata$usd_pledged_real), main = "Pledged")
hist(log(mydata$usd_goal_real), main = "Goal")
```



```
summary(log(dplyr::select(mydata, backers, duration, usd_pledged_real, usd_goal_real)))

##      backers          duration      usd_pledged_real      usd_goal_real
##  Min.   : -Inf   Min.   :0.000   Min.   : -Inf   Min.   :-4.605
##  1st Qu.:0.6931  1st Qu.:3.401   1st Qu.: 3.442   1st Qu.: 7.601
##  Median :2.4849   Median :3.401   Median : 6.442   Median : 8.613
##  Mean   : -Inf   Mean   :3.462   Mean   : -Inf   Mean   : 8.639
##  3rd Qu.:4.0431  3rd Qu.:3.611   3rd Qu.: 8.310   3rd Qu.: 9.680
##  Max.   :6.9078   Max.   :4.522   Max.   :11.513   Max.   :18.930
```

Backers y Duration claramente no siguen una distribución LogNormal, por lo que no es necesario aplicar ningún test. En cuanto a las variables Pledged y Goal, vamos a aplicar el test de [Test de Shapiro-Wilk](#) y observar los diagramas QQ comprobar si estas variables pueden que sí se ajusten a normal.

En primer lugar, dado que estas variables contienen el valor 0, antes de aplicar el logaritmo vamos a imputar un valor mínimo, puesto de lo contrario no se podría aplicar el tests al tener valores inválidos. Por otro lado, la implementación en R del test acepta como máximo 5000 observaciones que obtendremos a partir de la muestra original por muestreo aleatorio.

```
# Imputación valor mínimo distinto de 0
mydata <- within(mydata, usd_goal_real[usd_goal_real <= 0] <- 0.00001)
mydata <- within(mydata, usd_pledged_real[usd_pledged_real <= 0] <- 0.00001)
# Muestreo de 5000 observaciones
set.seed(123)
sample <- mydata[sample(1:nrow(mydata), 5000, replace=FALSE),]
```

Comprobamos la variable *Goal*:

```

# Goal
summary(log(sample$usd_goal_real))

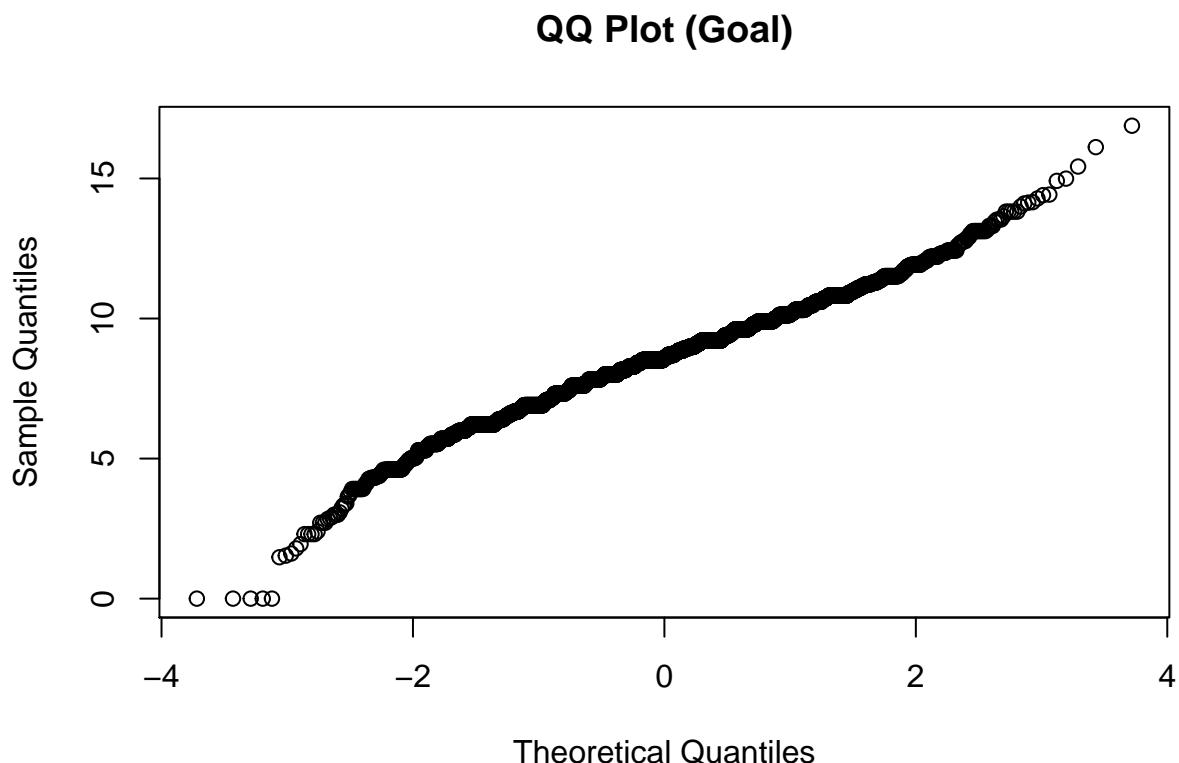
##      Min. 1st Qu. Median    Mean 3rd Qu.    Max.
## 0.000   7.601  8.593  8.614  9.653 16.882

shapiro.test(log(sample$usd_goal_real))

##
## Shapiro-Wilk normality test
##
## data: log(sample$usd_goal_real)
## W = 0.98801, p-value < 2.2e-16

qnorm(log(sample$usd_goal_real), main = "QQ Plot (Goal)")

```



Comprobamos la variable Pledged:

```

# Pledged
summary(log(sample$usd_pledged_real))

##      Min. 1st Qu. Median    Mean 3rd Qu.    Max.
## -11.513   3.401   6.400   4.035   8.290  11.513

shapiro.test(log(sample$usd_pledged_real))

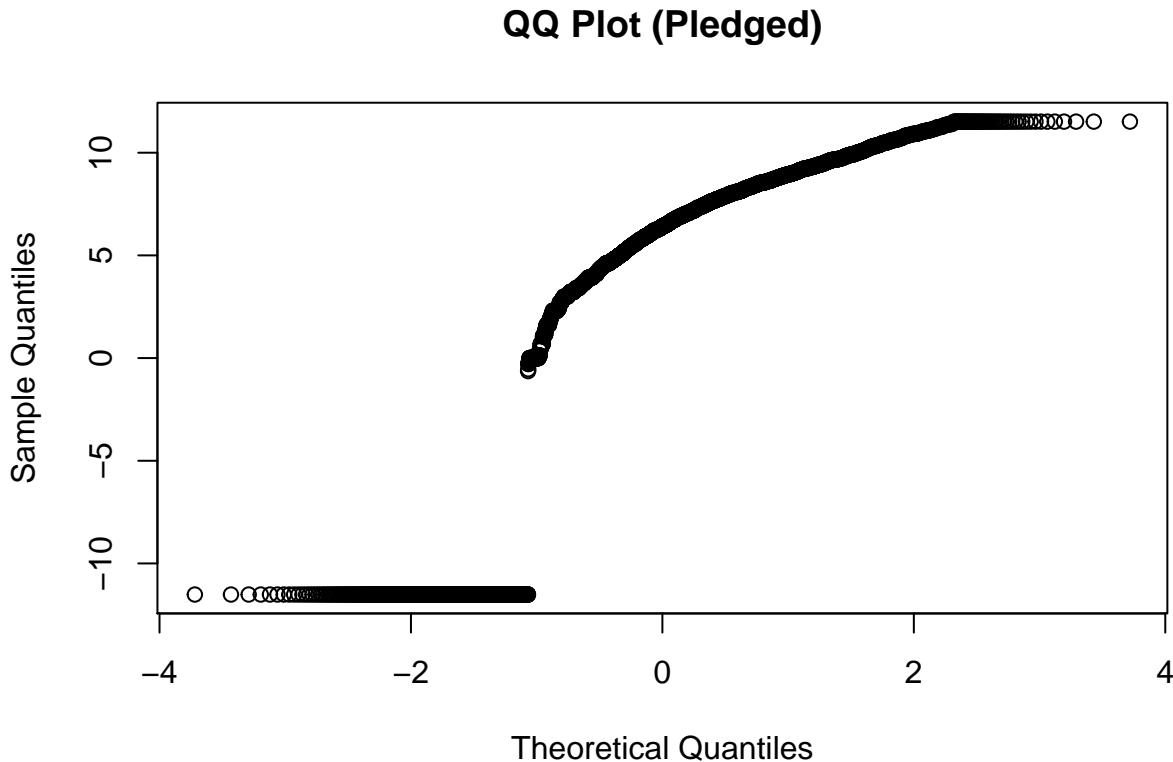
##
## Shapiro-Wilk normality test
##

```

```

## data: log(sample$usd_pledged_real)
## W = 0.73112, p-value < 2.2e-16
qqnorm(log(sample$usd_pledged_real), main = "QQ Plot (Pledged)")

```



En el caso de la variable *Goal* los datos parecen ajustarse a la diagonal salvo por la cola izquierda, mientras que para la variable *Pledged* vemos que estos distan mucho de la diagonal.

Sin embargo, en ambos casos el p-value obtenido por el test es cercano a 0, por lo que se rechaza la hipótesis nula de que las variables siguen una distribución normal, y se acepta la alternativa (es decir, que no siguen una distribución normal, o log-normal en este caso).

Dado este resultado, no se realizará [la transformación de Box-Cox](#) al estar los datos muy alejados de una distribución normal, por lo que a la hora de realizar el análisis de los datos se aplicará algoritmos no paramétricos.

### 3.2 Reducción de dimensionalidad

Para seleccionar los datos sobre los que vamos a realizar el análisis vamos a realizar un análisis previo de correlación para ver si es posible reducir la dimensionalidad descartando algunas de las variables que no son independientes.

Vamos a proceder en primer lugar con las cuatro variables categóricas disponibles.

En primer lugar, las variables *Category* y *Main Category* están correlacionadas por definición (la primera es un desglose de la segunda). Para facilitar la representación de resultados en el análisis descriptivo vamos a quedarnos únicamente con la segunda.

Por otro lado, las variables *Country* y *Currency* probablemente son dependientes. Vamos a realizar un test [chi-square](#) para verificar si se puede obviar una de ellas en la fase de análisis. Planteamos las hipótesis, y

supondremos un nivel de significación del 0.05:

$H_0$  : Las variables son independientes

$H_1$  : Las variables NO son independientes

Calculamos el p-value:

```
t <- table(mydata$currency, mydata$country)
chisq.test(t)

##
##  Pearson's Chi-squared test
##
## data: t
## X-squared = 4836800, df = 273, p-value < 2.2e-16
```

Y vemos que el valor es prácticamente 0, por lo que rechazamos la hipótesis nula y aceptamos la alternativa. Por ello se eliminará la variable *Currency* de los datos a analizar.

Para las variables cuantitativas, vamos a calcular la matrix de correlación usando como método la correlación de Spearman, dado que las variables no se ajustan a una distribución normal:

```
num_vars = dplyr::select(mydata, backers, country, usd_pledged_real, usd_goal_real, duration)
round(cor(x = data.matrix(num_vars), method = "spearman"), 3)

##
backers   country usd_pledged_real usd_goal_real duration
## backers      1.000    0.068        0.960     0.106   -0.001
## country       0.068    1.000        0.072     -0.001    0.029
## usd_pledged_real  0.960    0.072        1.000     0.181    0.018
## usd_goal_real    0.106   -0.001        0.181     1.000    0.214
## duration       -0.001    0.029        0.018     0.214    1.000
```

Vemos que el número de patrocinadores y el dinero recaudado tienen una relación de correlación positiva, lo que significa que éstas no son independientes y por tanto se podría obviar una de ellas a la hora de realizar el análisis de datos.

Eliminamos del set de datos aquellas variables que no tendremos en cuenta para el análisis:

```
mydata <- dplyr::select(mydata, -currency, -category, -backers)
```

Por lo que para el análisis dispondremos de seis variables, y la etiqueta de clase:

```
glimpse(mydata)
```

```
## Observations: 372,059
## Variables: 7
## $ main_category <fct> Publishing, Film & Video, Film & Video, Music...
## $ state          <fct> failed, failed, failed, failed, canceled, suc...
## $ country         <fct> GB, US, US, US, US, US, US, US, US, CA, U...
## $ usd_pledged_real <dbl> 1.00000e-05, 2.42100e+03, 2.20000e+02, 1.0000...
## $ usd_goal_real    <dbl> 1533.95, 30000.00, 45000.00, 5000.00, 19500.0...
## $ duration         <dbl> 59, 60, 45, 30, 56, 35, 20, 45, 35, 30, 30, 3...
## $ goal_reached     <lgl> FALSE, FALSE, FALSE, FALSE, TRUE, TRUE...
```

### 3.3 Análisis descriptivo

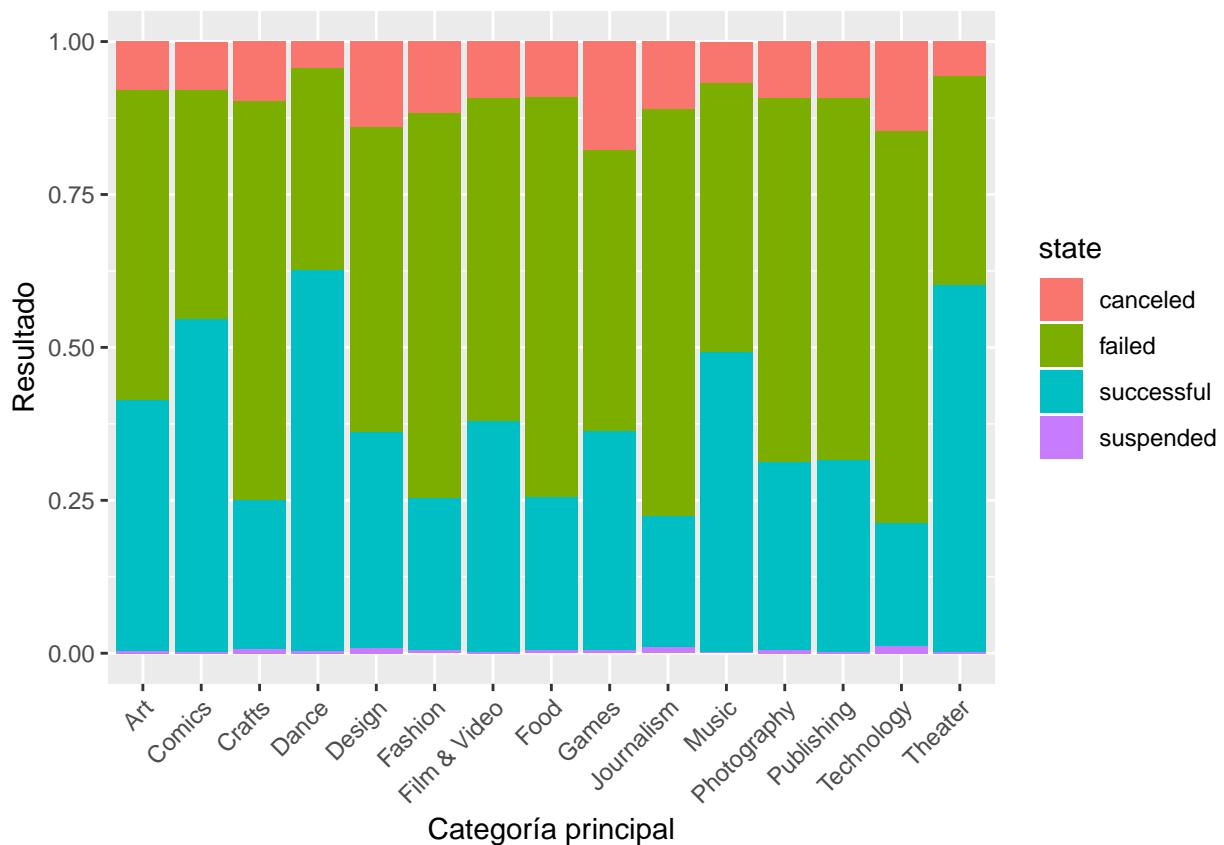
En primer lugar comprobamos las frecuencias del resultado de los proyectos:

```
tabyl(mydata$state)
```

```
##   mydata$state      n    percent
##       canceled 38751 0.104152836
##       failed 197614 0.531136191
##       successful 133851 0.359757458
##       suspended  1843 0.004953515
```

Vemos que, de forma global, únicamente el 40% de los proyectos acaban saliendo adelante. Vamos a comprobar si la frecuencia si algunas de las categorías de las variables cualitativas parecen tener frecuencias superiores a otras.

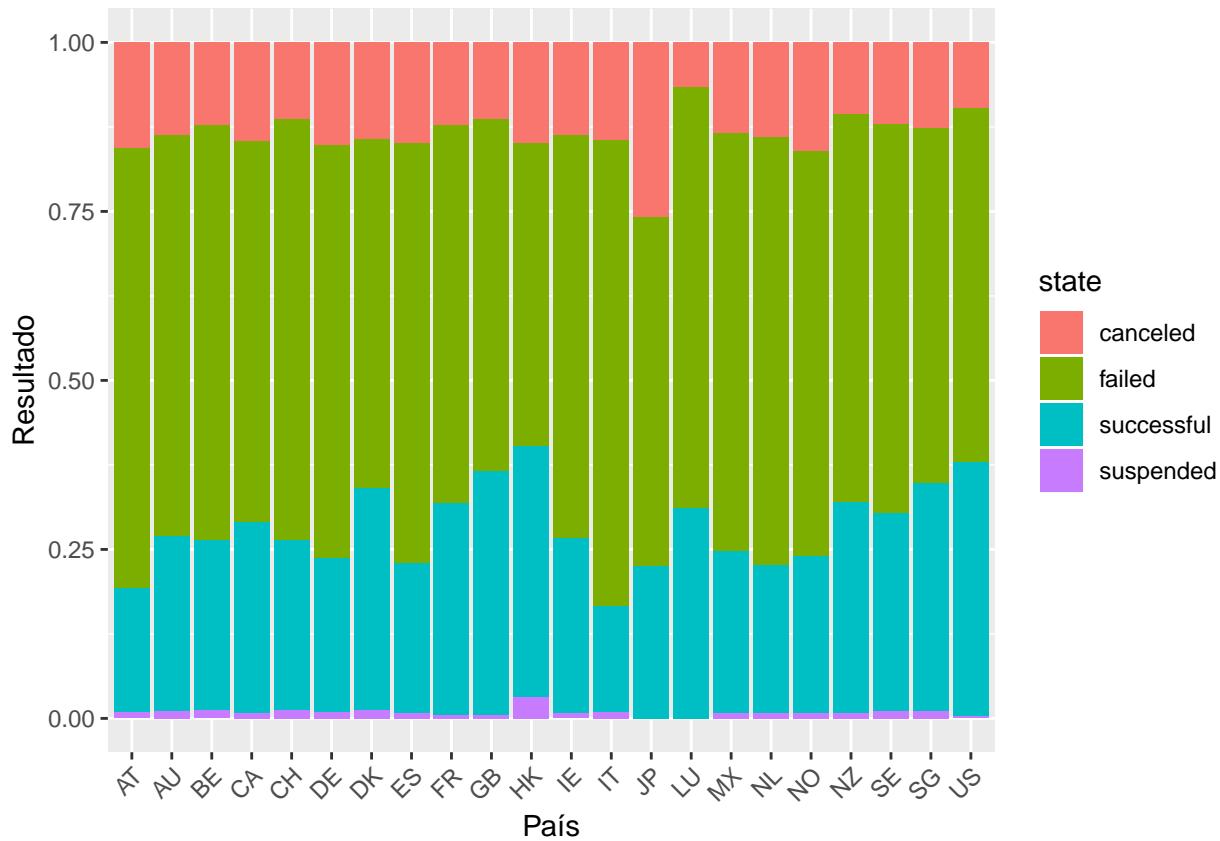
```
ggplot(data=mydata,aes(x=main_category,fill=state))+geom_bar(position="fill") +
  xlab("Categoría principal") + ylab("Resultado") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



En esta gráfica vemos que categorías como *Dance* o *Theatre* tienen una frecuencia de éxito aproximadamente de aproximadamente el doble que otras como “Crafts” o *Journalism*.

También hay diferencias dependiendo del país:

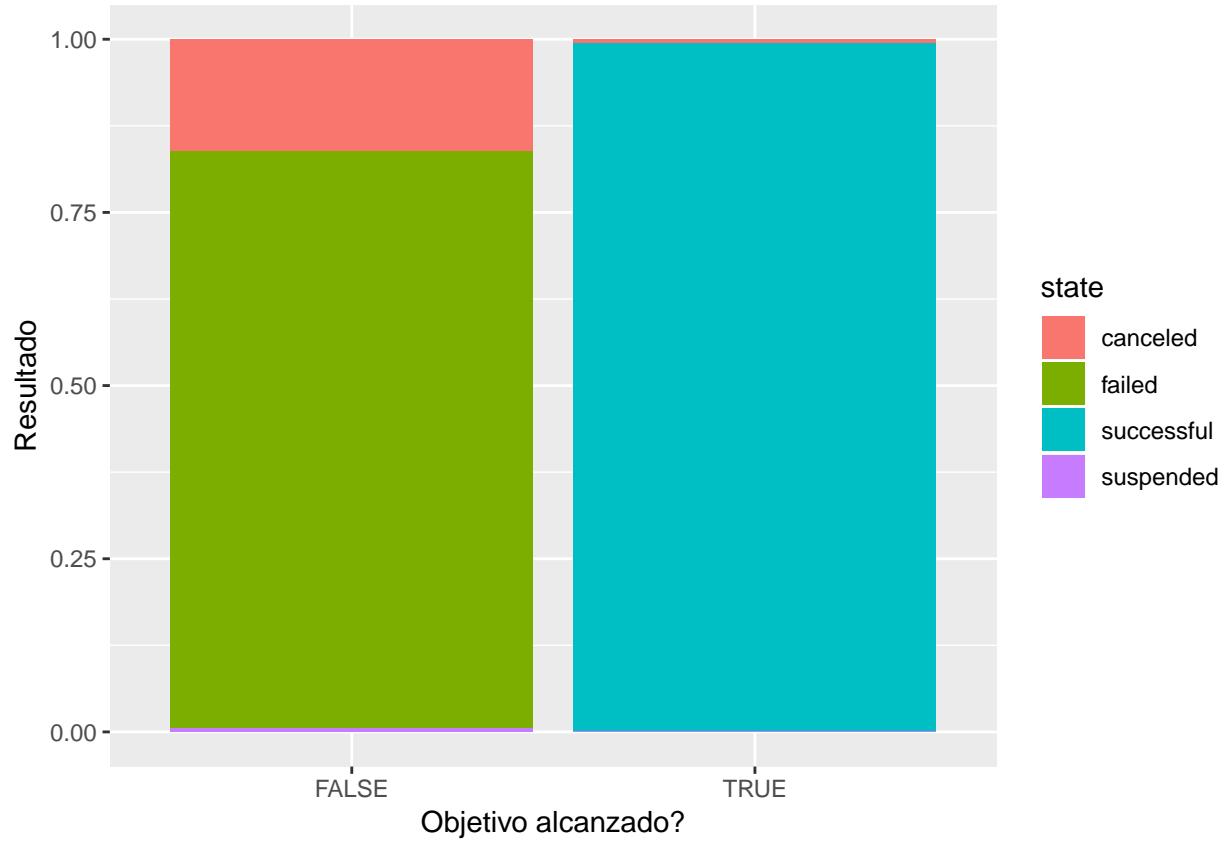
```
ggplot(data=mydata,aes(x=country,fill=state))+geom_bar(position="fill") +
  xlab("País") + ylab("Resultado") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Iniciativas nacidas en Estados Unidos o Hong Kong tienen una frecuencia de éxito superior a la de otros países como, por ejemplo, Italia.

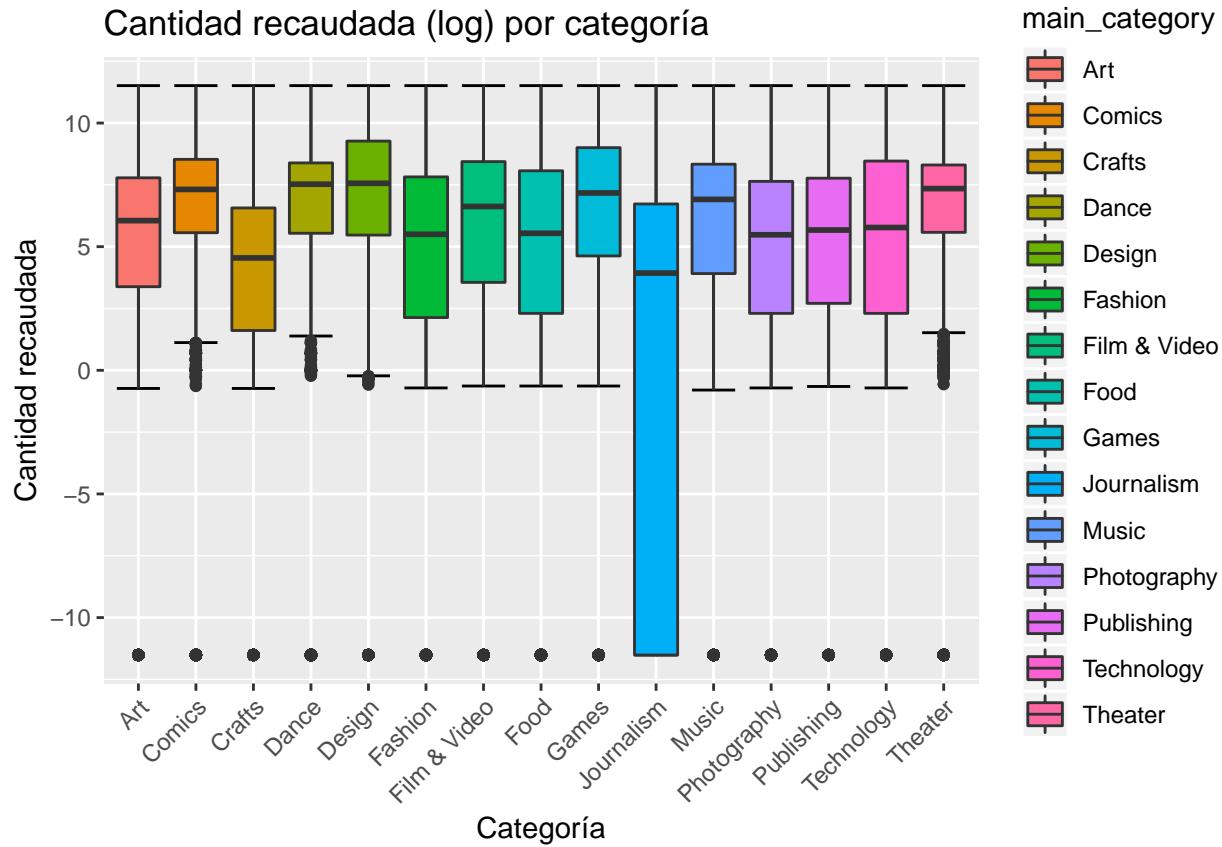
Finalmente, en el siguiente gráfico vemos que prácticamente la totalidad de iniciativas que alcanzan el objetivo tienen éxito. Sin embargo, también vemos que hay una frecuencia muy pequeña de iniciativas que, a pesar de alcanzar el objetivo, acaban cancelándose:

```
ggplot(data=mydata,aes(x=goal_reached,fill=state))+geom_bar(position="fill") +
  xlab("Objetivo alcanzado?") + ylab("Resultado")
```



Vamos a comprobar ahora si existen diferencias significativas en el dinero recaudado dependiendo de la categoría de las iniciativas:

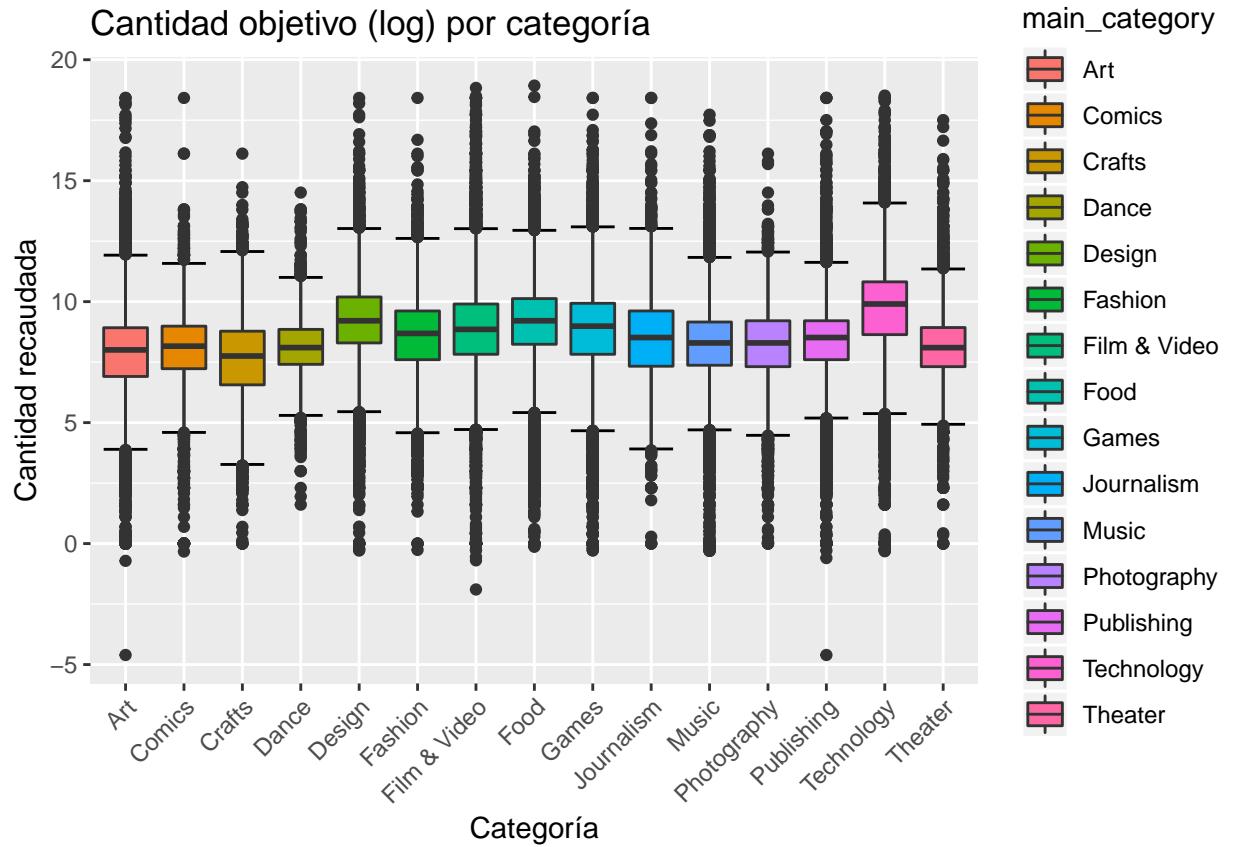
```
ggplot(data = mydata, aes(x=mydata$main_category, y=log(mydata$usd_pledged_real))) +
  stat_boxplot(geom ='errorbar') + geom_boxplot(aes(fill=main_category)) +
  ggtitle("Cantidad recaudada (log) por categoría") +
  xlab("Categoría") + ylab("Cantidad recaudada") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



En algunos casos esta diferencia es clara, como por ejemplo entre “Art” y “Comics”. Sin embargo, en algunos casos esta diferencia no está tan clara, como entre “Dance” y “Design”.

Comprobamos también si, en general, la cantidad objetivo varía dependiendo de la categoría de la iniciativa:

```
ggplot(data = mydata, aes(x=mydata$main_category, y=log(mydata$usd_goal_real))) +
  stat_boxplot(geom ='errorbar') + geom_boxplot(aes(fill=main_category)) +
  ggtitle("Cantidad objetivo (log) por categoría") +
  xlab("Categoría") + ylab("Cantidad recaudada") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

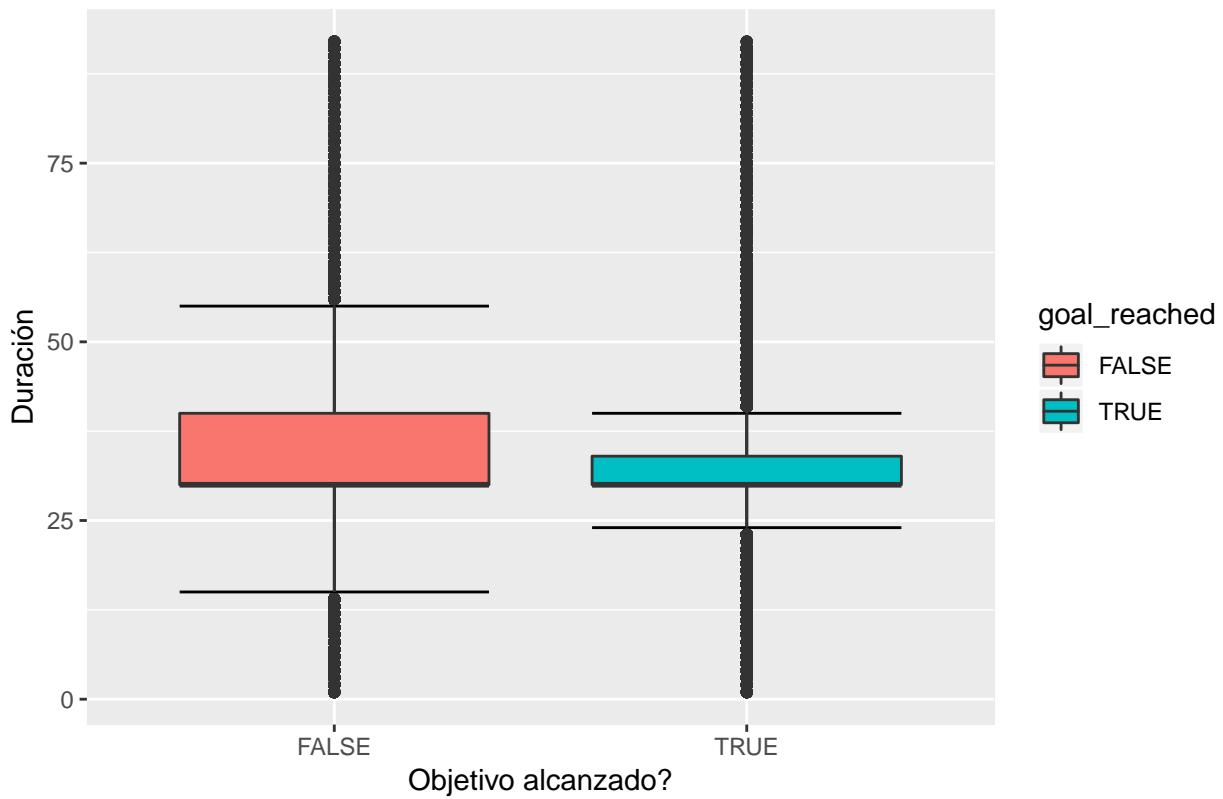


Vemos que, en general, las iniciativas tecnológicas tienen una cantidad a recaudar superior al resto de categorías.

Por último, comprobaremos si la duración influye en la cantidad recaudada o en el resultado final de los proyectos:

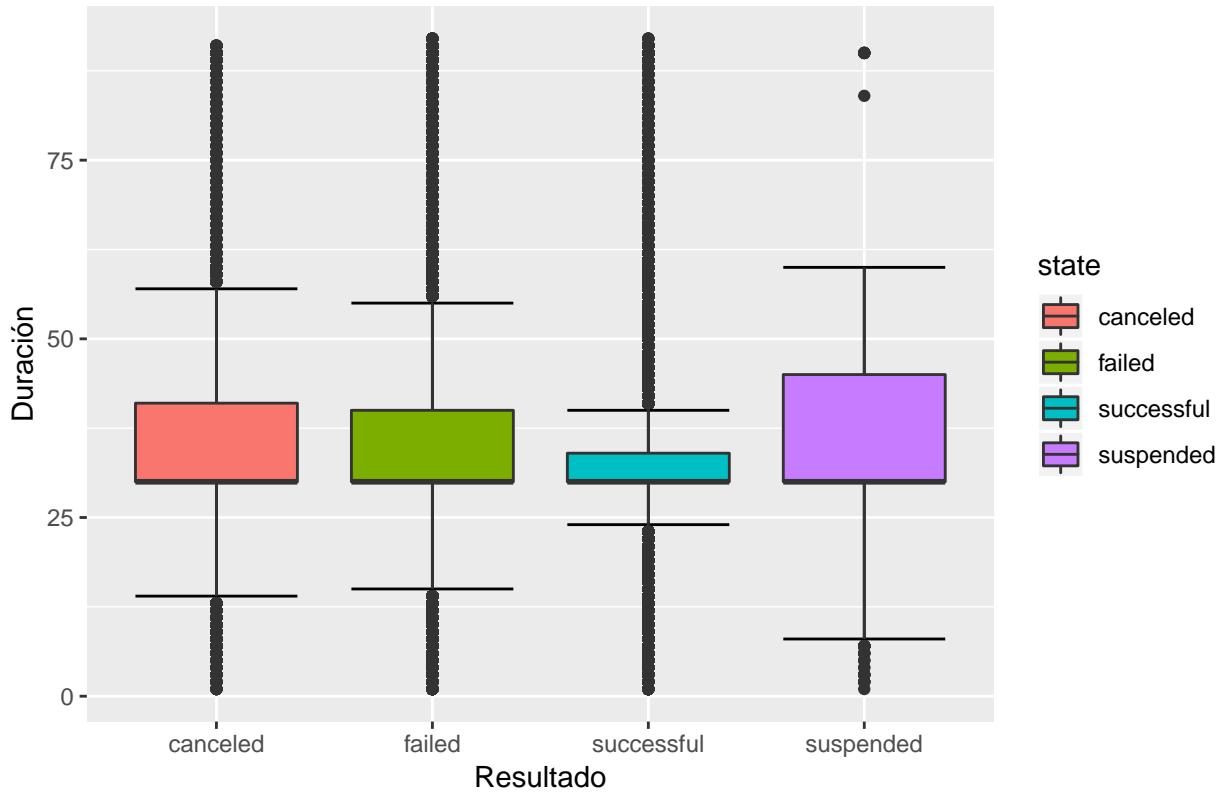
```
ggplot(data = mydata, aes(x=mydata$goal_reached, y=mydata$duration)) +
  stat_boxplot(geom ='errorbar') + geom_boxplot(aes(fill=goal_reached)) +
  ggtitle("Duración dependiendo del objetivo alcanzado") + xlab("Objetivo alcanzado?") + ylab("Duración")
```

## Duración dependiendo del objetivo alcanzado



```
ggplot(data = mydata, aes(x=mydata$state, y=mydata$duration)) +  
  stat_boxplot(geom ='errorbar') + geom_boxplot(aes(fill=state)) +  
  ggtitle("Duración por resultado") + xlab("Resultado") + ylab("Duración")
```

## Duración por resultado



Vemos que, a simple vista, parece que la duración no tiene una influencia significativa.

En base a este análisis descriptivo, podemos plantear las siguientes pruebas:

- Verificar que no hay una diferencia significativa de la duración entre aquellos proyectos que salen adelante, y aquellos que no.
- Dado que aparentemente existen diferencias entre las cantidades que recaudan los proyectos en función de su categoría, verificar qué categorías tienen cantidades recaudadas distintas a las demás.

## 3.4 Análisis inferencial

### 3.4.1 Duración

Vamos a comprobar si existe una diferencia significativa en la duración de los proyectos, entre aquellos proyectos que llegan a recaudar la cantidad objetivo, y aquellos que no. Como se ha podido observar en el boxplot de apartado de análisis descriptivo, las tendencias centrales son similares:

```

reached <- mydata[mydata$goal_reached == TRUE,]$duration
not_reached <- mydata[mydata$goal_reached == FALSE,]$duration

# Medias
mean(reached)

## [1] 32.17455
mean(not_reached)

## [1] 35.31228

```

```
# Medianas
median(reached)

## [1] 30

median(not_reached)

## [1] 30
```

Dado que la variable duration no sigue una distribución normal, no es posible el empleo del test T. Por ello, vamos a usar el [test de Wilcoxon](#), que es la alternativa no paramétrica al anterior. Suponemos un nivel de significación del 0.05 y las hipótesis siguientes:

$$H_0 : \mu_{Reached} = \mu_{Not Reached}$$

$$H_1 : \mu_{Reached} \neq \mu_{Not Reached}$$

```
wilcox.test(reached, not_reached, paired = FALSE, alternative = "two.sided")
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data: reached and not_reached
## W = 1.4165e+10, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

El p-value obtenido es cercano a 0, por lo que rechazaremos la hipótesis nula y aceptaremos la alternativa. Esto es, que la duración si que es distinta dependiendo de si un proyecto logra alcanzar la cantidad objetivo.

### 3.4.2 Dinero recaudado por categoría

Vamos a probar si hay diferencias significativas en las cantidades medias recaudadas dependiendo del tipo de proyecto. Dado que hemos visto que los datos no siguen una distribución normal no podemos aplicar un análisis de varianza unifactorial (ANOVA), por lo que aplicaremos el test no paramétrico [Kruskal-Wallis](#) para comprobar si las medias de todas las categorías son iguales. Planteamos la hipótesis nula y alternativa, y suponemos un nivel de significación del 0.05:

$$H_0 : \text{Todas las medias son iguales}$$

$$H_1 : \text{No todas las medias son iguales}$$

```
mydata$usd_pledged_real_log <- log(mydata$usd_pledged_real)
kruskal.test(usd_pledged_real_log~main_category, data=mydata)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data: usd_pledged_real_log by main_category
## Kruskal-Wallis chi-squared = 16724, df = 14, p-value < 2.2e-16
```

Obtenemos un p-value cercano a cero, por lo que rechazamos la hipótesis nula: existe al menos alguna media significativamente distinta de las demás.

Para comprobar qué medias difieren significativamente de las demás vamos a aplicar el [test post hoc de Dunn](#). Este test no paramétrico permite realizar comparaciones múltiples, y es adecuado para muestras en las que los distintos grupos tienen un número distinto de observaciones.

```
dunn_res <- dunnTest(usd_pledged_real_log~main_category,data=mydata,method="by")
print(dunn_res,dunn.test.results=TRUE)
```

```
##  Kruskal-Wallis rank sum test
##
##  data: x and g
##  Kruskal-Wallis chi-squared = 16724.1465, df = 14, p-value = 0
##
##
##                                     Comparison of x by g
##                                     (Benjamini-Yekutieli)
##   Col Mean-|
##   Row Mean |      Art     Comics    Crafts     Dance    Design   Fashion
##   -----
##   Comics | -38.11357
##           | 0.0000*
##           |
##   Crafts | 27.97786  53.83176
##           | 0.0000*  0.0000*
##           |
##   Dance  | -22.65730  2.033148 -37.74774
##           | 0.0000*  0.2288  0.0000*
##           |
##   Design | -64.55574 -9.331822 -72.36186 -8.286099
##           | 0.0000*  0.0000*  0.0000*  0.0000*
##           |
##   Fashion | 7.718880  42.80117 -21.73279  26.26030  68.73478
##           | 0.0000*  0.0000*  0.0000*  0.0000*  0.0000*
##           |
##   Film & V | -23.35236  25.32654 -44.72866  13.43936  52.45982 -30.52589
##           | 0.0000*  0.0000*  0.0000*  0.0000*  0.0000*  0.0000*
##           |
##   Food   | 2.883527  39.55018 -25.48130  23.90535  65.19400 -4.745284
##           | 0.0218*  0.0000*  0.0000*  0.0000*  0.0000*  0.0000*
##           |
##   Games  | -48.84524  3.688341 -61.42939  0.123535  18.47381 -53.98000
##           | 0.0000*  0.0013*  0.0000*  1.0000  0.0000*  0.0000*
##           |
##   Journali | 23.93701  46.34978  1.860573  35.23148  58.36993  19.21678
##           | 0.0000*  0.0000*  0.3385  0.0000*  0.0000*  0.0000*
##           |
##   Music   | -25.32732  22.81854 -45.87310  12.06533  47.40002 -32.18444
##           | 0.0000*  0.0000*  0.0000*  0.0000*  0.0000*  0.0000*
##           |
##   Photogra | 9.242088  39.38860 -16.51524  26.30343  57.06945  3.059376
##           | 0.0000*  0.0000*  0.0000*  0.0000*  0.0000*  0.0124*
##           |
##   Publishi | 7.937185  45.42168 -23.72369  26.68473  77.96754 -0.825958
##           | 0.0000*  0.0000*  0.0000*  0.0000*  0.0000*  1.0000
##           |
##   Technolo | -7.677581  33.19470 -33.62790  19.19809  59.05408 -15.18462
```

```

##          | 0.0000* 0.0000* 0.0000* 0.0000* 0.0000* 0.0000*
##          |
## Theater | -32.07185 5.154283 -49.09373 1.665989 15.62902 -36.96577
##          | 0.0000* 0.0000* 0.0000* 0.5109 0.0000* 0.0000*
## Col Mean-|
## Row Mean | Film & V      Food      Games   Journali    Music   Photogra
## -----
## Food   | 25.60962
##          | 0.0000*
##          |
## Games  | -33.50089 -50.01663
##          | 0.0000* 0.0000*
##          |
## Journali | 36.08930 22.10105 49.57195
##          | 0.0000* 0.0000* 0.0000*
##          |
## Music   | -3.579651 -27.45797 28.92051 -37.17232
##          | 0.0020* 0.0000* 0.0000* 0.0000*
##          |
## Photogra | 26.12402 6.880917 45.02117 -15.55496 27.65339
##          | 0.0000* 0.0000* 0.0000* 0.0000* 0.0000*
##          |
## Publishi | 35.68826 4.525247 61.67795 -20.40912 37.17277 -3.925439
##          | 0.0000* 0.0000* 0.0000* 0.0000* 0.0000* 0.0005*
##          |
## Technolo | 15.34372 -10.37194 42.60665 -28.19567 17.70285 -15.04581
##          | 0.0000* 0.0000* 0.0000* 0.0000* 0.0000* 0.0000*
##          |
## Theater  | -18.70728 -33.63027 2.679803 -42.40915 -16.31186 -34.35137
##          | 0.0000* 0.0000* 0.0405* 0.0000* 0.0000* 0.0000*
## Col Mean-|
## Row Mean | Publishi Technolo
## -----
## Technolo | -16.59840
##          | 0.0000*
##          |
## Theater  | -39.16471 -27.02017
##          | 0.0000* 0.0000*
##          |
## alpha = 0.05
## Reject Ho if p <= alpha

```

En la tabla anterior vemos el p-value asociado a la comparación entre todos los pares de grupos. En esta comparación vemos que todas las medias son significativamente distintas salvo las de los siguientes pares de grupos:

- Dance - Comics
- Dance - Games
- Journalism - Crafts
- Publishing - Fashion
- Theater - Dance
- Theater - Games

## 4 Conclusión

A lo largo de esta práctica se ha podido analizado las características que tienen los poyectos de crowdfunding.

En primer lugar, durante la búsqueda de valores extremos se ha podido observar que existen proyectos con muchísimo más éxito que los demás (en cuanto a patrocinadores y dinero recaudado), y cuyos valores se han tenido que suavizar para que evitar que éstos tuviesen demasiada incluencia y produjesen sesgo a la hora de realizar la analítica de datos.

Por otro lado se ha podido comprobar que ninguna de las variables cuantitativas se ajusta a una distribución normal o log-normal, a pesar de que la distribución de valores sigue una curva similar a la gausiana aplicando el logaritmo sobre los valores reales.

El hecho de que ninguna de las variables siguiese una distribución normal ha obligado al uso de tests no paramétricos (y de menor poder estadístico) en la fase de analítica.

Los resultados obtenidos se pueden consultar en los apartados de análisis descriptivo y análisis inferencial. Sin embargo, la fase analítica de la práctica se ha visto impactada por el hecho de que los datos no siguiesen distribuciones conocidas, y por la falta de base teórica sólida sobre cómo tratar con este tipo de datos.

La versión del set de datos tras el preprocesado está disponible en el directorio /csv del repositorio.

```
write.csv(mydata, file = "../csv/kickstarter_data_clean.csv", row.names=TRUE)
```

## 5 Referencias y bibliografía

1. Mouillé, M. [Kickstarter projects](#). Kaggle.com
2. Squire, M. Clean Data. Packt Publishing, 2015.
3. Magnifico, S.S. [Kruskal-Wallis Test](#). An R Companion for the Handbook of Biological Statistics. Consultado el 02.06.2019
4. Calvo M., Subirats L., Pérez D. (2019). Introducción a la limpieza y análisis de los datos. Editorial UOC.