PRAC2 Tipologia & Ciclo de Vida de Datos

Jon Ortiz Abalia, Gabriel Peso Bañuelos

11 de junio de 2019

Contents

---

1. Descripción del dataset

¿Por qué es importante y qué pregunta/problema pretende responder?

Nuestra primera práctica sobre *web scraping* la desarrollamos contra la web de venta de entaradas on-line "Atrapalo.com" con la idea de obtener un listado de todos los espectáculos a nivel nacional tanto en cartera actualmente como históricamente desde el momento que se empezaran a captar los datos.

Nos hubiera gustado utilizar el *dataset* obtenido en dicha práctica para continuar con esta segunda y haber podido plantear un ejercicio de regresión o previsión pero nos resultó dificil plantearlo por no disponer de los datos recuadatorios asociados a cada espectaculo ya que obviamente esa información no existía en la web scrappeada.

Por este motivo hemos elegido otro conjunto de datos público que está muy relacionado y sí dispone de esta información y que, además, es objeto actualmente de un concurso en la plataforma **kaggle**, https://www.kaggle.com. El dataset de estudio consiste en un listado de películas en la que se citan diferentes atributos de las mismas junto a su recaudación en taquilla.

El **objetivo** del concurso es **plantear un modelo de regresión lineal que permita preveer la recaudación de una película futura en base al conocimiento de algunos de estos atributos**.

Concretamente la base de datos origen del concurso, se encuentre en https://www.kaggle.com/c/tmdb-box-office-prediction y se compone de 2 datasets iniciales, uno de training y otro de test.En nuestro caso utilizaremos sólo el dataset de training, denominado 'train.csv', para inferir el modelo de regresión lineal y que será el objeto de este ejercicio.

2. Selección de los datos de interés a analizar.

En primer lugar inspecionamos el formato del fichero train.csv visualmente y comprobamos que se encuentra en formato anglosajón: Decimales separados por punto, campos separados por comas, etc por lo que podemos cargarlo con el comando: read.csv().

Una primera revisión nos indica que se trata de un dataset con las siguientes dimensiones.

```
+ Filas: 3000 registros
+ Columnas: 23 variables
```

Cuyas variables son las siguientes:

1. **X.U.FEFF.id.**: Id asociado a la película.
2. **belongs_to_collection**: Indica si la variable pertenece o no a una saga.
3. **budget**: Presupuesto invertido en la película (en dólares).
4. **genres**: Género(s) de la película. (Nota. Una película puede estar catalogada como más de un género)
5. **homepage**: Página web de la película

6. **imdb_id**: Id de la película en la página web de IMDB (https://www.imdb.com/)
7. **original_language**: Idioma original de la película.

8. **original_title**: Título original de la película.

9. **overview**: Resumen de la película

10. **popularity**: Popularidad de la película en base a UN algoritmo interno creado por la página web de TMDB https://www.themoviedb.org/?language=en-US.
11. **poster_path**: Página donde podemos visualizar el cartel oficial de la película: https://www.themoviedb.org/?language=en-US.
12. **production_companies**: Listado de las productoras que han particpado en la grabación de la película
13. **production_countries**: Paises que participan en la producción de una película
13. **release_date**: Variable donde se indica la fecha de estreno de la película.
14. **runtime**: Duración de la película (en minutos).
15. **spoken_languages**: Idiomas a los que se ha traducido la película

16. **status**: Situación en la que se encuentra una pelicula

17. **tagline**: ¿?
18. **title**: Título de la película

19. **Keywords**: Palabras asociadas a la película
20. **cast**: Variable en formato json donde se listan las personas del reparto.
21. **crew**: Variable en formato json donde se listan las personas del equipo de l película.
22. **revenue**: Beneficio obtenido con la película (en dólares).

Recordemos que el objetivo será plantear un modelo de regresión para la variable depeniente, **revenue**, en función de otras independientes y que tendremos que determinar.

Observamos que uno de los atributos de las películas es **status** y que los valores posibles de esta variable son: Released, Rumored. Es decir aquellas ya estrenadas: `Released` y aquellas todavía en previsión: `Rumored`.

Entre estas últimas vemos que hay 4 películas que no tendremos en cuenta y que descartaremos del listado inicial puesto que los datos se basan obviamente en valores estimados y no reales:

```
##                                            title  status budget
## 610  The Swan Princess: Escape from Castle Mountain Rumored      0
## 1008                         Billy Gardell: Halftime Rumored      0
## 1217                                     Extremities Rumored      0
## 1619                          A Place Called Chiapas Rumored 891000
```

Eliminamos pues los 4 registros y nos quedamos con **2996** registros del listado original.

Una vez realizado este primer análisis de las variables existentes y de la estimación del posible impacto que cada una de ellas podría tener sobre el potencial beneficio de la película (**revenue**) procedemos a descartar, por sentido común, algunas de ellas que consideramos más irrelevantes como **homepage**, **original_title** etc etc.

Nos quedaremos para el análisis del ejercicio únicamente con aquellas que consideramos, a priori, más correlacionadas con la variable buscada y que serían las siguientes:

- "X.U.FEFF.id"

- "belongs_to_collection"
- "budget"
- "genres"
- "original_language"

- “popularity”
- “production_companies”
- “release_date”
- “runtime”
- “title”
- “cast”
- “crew”
- “revenue”

`Nota. Los atributos 'X.U.FEFF.id' y 'title' no se consideran relavantes para el ejercicio de regresión posterior pero nos servirán para identificar unívocamente a cada película dentro del dataset`

De esta forma nos quedaremos con un dataset de análisis con las siguientes dimensiones:

```
+ Filas: 2996 registros
+ Columnas: 13 variables
```

Una vez elegidas mostramos un primer resumen del contenido de las variables de análisis

```
## Observations: 3,000
## Variables: 13
## $ X.U.FEFF.id           <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 1...
## $ belongs_to_collection <chr> "[{'id': 313576, 'name': 'Hot Tub Time M...
## $ budget                <int> 14000000, 40000000, 3300000, 1200000, 0,...
## $ genres                <chr> "[{'id': 35, 'name': 'Comedy'}]", "[{'id...
## $ original_language     <chr> "en", "en", "en", "hi", "ko", "en", "en"...
## $ popularity            <dbl> 6.575393, 8.248895, 64.299990, 3.174936,...
## $ production_companies  <chr> "[{'name': 'Paramount Pictures', 'id': 4...
## $ release_date          <chr> "2/20/15", "8/6/04", "10/10/14", "3/9/12...
## $ runtime               <int> 93, 113, 105, 122, 118, 83, 92, 84, 100,...
## $ title                 <chr> "Hot Tub Time Machine 2", "The Princess ...
## $ crew                  <chr> "[{'credit_id': '59ac067c92514107af02c8c...
## $ cast                  <chr> "[{'cast_id': 4, 'character': 'Lou', 'cr...
## $ revenue               <int> 12314651, 95149435, 13092000, 16000000, ...
```

Observación:

Las variables:

- **belongs_to_collection**
- **genres**
- **production_companies**
- **cast**
- **crew**

se encuentran en formato JSON y no resultan muy intuitivas visualmente.
Para dar una idea de la estructura de estas variables vamos a mostrar el valor de cada una de ellas referentes al primer registro del dataset.

belongs_to_collection

Registro.num.9
[{’id’: 256377, ’name’: ’The Muppet Collection’, ’poster_path’: ’/8Ew8EIdFFurMMYjSbWPu1Hl4vLX.jpg’, ’backdrop_path’: ’/1AWd3MM90G47mxtD112gRDxSXY9.jpg’}]

En esta variable se recoge información de la saga ('collection') a la que pertenece cada película.

genres

Registro.num.9
[{'id': 28, 'name': 'Action'}, {'id': 35, 'name': 'Comedy'}, {'id': 10402, 'name': 'Music'}, {'id': 10751, 'name': 'Family'}, {'id': 12, 'name': 'Adventure'}]

Se aprecia que a la película se le asocian varios géneros: `Action`, `Comedy`,`Music`,`Family` y `Adventure`.

`Nota: Obviamente cada pelicula puede tener ó no una clasificación de género asociada y ésta pueder ser múltiple ó no`

cast

Registro.num.9
[{'cast_id': 1, 'character': 'Long John Silver', 'credit_id': '52fe43c89251416c7501dea1', 'gender': 2, 'id': 13472, 'name': 'Tim Curry', 'order': 0, 'profile_path': '/eo8AHZqSKuPconj1ueXHHBS37pM.jpg'}, {'cast_id': 2, 'character': 'Jim Hawkins', 'credit_id': '52fe43c89251416c7501dea5', 'gender': 2, 'id': 19996, 'name': 'Kevin Bishop', 'order': 1, 'profile_path': '/uiRRSgBK05xrjzGecbp2Oz8SYN8.jpg'}, {'cast_id': 3, 'character': 'Mrs. Bluveridge', 'credit_id': '52fe43c89251416c7501dea9', 'gender': 1, 'id': 12094, 'name': 'Jennifer Saunders', 'order': 2, 'profile_path': '/c1qRVTYqUhWd7y2ws426zdyqhln.jpg'}, {'cast_id': 4, 'character': 'Billy Bones', 'credit_id': '52fe43c89251416c7501dead', 'gender': 0, 'id': 9188, 'name': 'Billy Connolly', 'order': 3, 'profile_path': '/7e1rVdJah2r0DaMpovrhbR2dHPS.jpg'}, {'cast_id': 13, 'character': '', 'credit_id': '533145589251410b48000fdd', 'gender': 2, 'id': 64181, 'name': 'Dave Goelz', 'order': 4, 'profile_path': '/hVfv7gsUPDRDFxUk7fyCktIL7Ar.jpg'}, {'cast_id': 15, 'character': 'Miss Piggy / Fozzie Bear / Sam the Eagle / Animal (voice)', 'credit_id': '5920e2e99251414ab104bc9f', 'gender': 2, 'id': 7908, 'name': 'Frank Oz', 'order': 5, 'profile_path': '/aLH5bYwMIlVxCe4rIDaEsVJqDKn.jpg'}, {'cast_id': 16, 'character': 'Kermit the Frog / Rizzo the Rat / Beaker (voice)', 'credit_id': '5920e2fe925141485e0513d1', 'gender': 2, 'id': 64180, 'name': 'Steve Whitmire', 'order': 6, 'profile_path': '/47ovNnEHh2fMYjTPH7A3MEAuKdW.jpg'}, {'cast_id': 17, 'character': 'Captain Flint', 'credit_id': '5974ef70925141580601aac4', 'gender': 2, 'id': 133876, 'name': 'David Nicholls', 'order': 7, 'profile_path': '/i6SfR3vndOofwycA5RzZ9RBQG3w.jpg'}, {'cast_id': 18, 'character': 'Calico Jerry', 'credit_id': '5974ef7dc3a3685e0101aade', 'gender': 0, 'id': 1705493, 'name': 'Frederick Warder', 'order': 8, 'profile_path': None}, {'cast_id': 19, 'character': 'Easy Pete', 'credit_id': '5974ef88925141580601aae1', 'gender': 0, 'id': 121112, 'name': 'Harry Jones', 'order': 9, 'profile_path': None}, {'cast_id': 20, 'character': 'Black Eyed Pea', 'credit_id': '5974ef96c3a3685da401a998', 'gender': 0, 'id': 1220350, 'name': 'Peter Geeves', 'order': 10, 'profile_path': None}, {'cast_id': 21, 'character': "Big-Fat-Ugly-Bug-Faced-Baby-Eating O'Brien", 'credit_id': '5974efb592514106a8012c91', 'gender': 0, 'id': 1314585, 'name': 'Jessica Hamilton', 'order': 11, 'profile_path': None}]

Se aprecia que esta variable tiene un formato complejo puesto que sintetiza el reparto completo de cada película indicando datos del personaje, actor/actriz que lo interpreta etc etc.

crew

Registro.num.9
[{'credit_id': '52fe43c89251416c7501deb3', 'department': 'Directing', 'gender': 2, 'id': 65298, 'job': 'Director', 'name': 'Brian Henson', 'profile_path': '/m2Bczi1gvhnIYCGp8Fhg2QCPuNf.jpg'}, {'credit_id': '52fe43c89251416c7501deb9', 'department': 'Production', 'gender': 2, 'id': 7908, 'job': 'Producer', 'name': 'Frank Oz', 'profile_path': '/aLH5bYwMIlVxCe4rIDaEsVJqDKn.jpg'}, {'credit_id': '52fe43c89251416c7501debf', 'department': 'Production', 'gender': 2, 'id': 65298, 'job': 'Producer', 'name': 'Brian Henson', 'profile_path': '/m2Bczi1gvhnIYCGp8Fhg2QCPuNf.jpg'}, {'credit_id': '52fe43c89251416c7501dec5', 'department': 'Writing', 'gender': 2, 'id': 64184, 'job': 'Screenplay', 'name': 'Jerry Juhl', 'profile_path': '/cgNumNNGSb5MeN0WIXkkhY0iXGV.jpg'}, {'credit_id': '52fe43c89251416c7501decb', 'department': 'Writing', 'gender': 2, 'id': 29533, 'job': 'Novel', 'name': 'Robert Louis Stevenson', 'profile_path': '/fGEGp5kpR2mhX89XqAJoJQFGeuG.jpg'}, {'credit_id': '52fe43c89251416c7501ded1', 'department': 'Sound', 'gender': 2, 'id': 947, 'job': 'Original Music Composer', 'name': 'Hans Zimmer', 'profile_path': '/7IjJpvGtCfY0DsritmfCh2iX9I4.jpg'}, {'credit_id': '52fe43c89251416c7501ded7', 'department': 'Editing', 'gender': 2, 'id': 12940, 'job': 'Editor', 'name': 'Michael Jablow', 'profile_path': None}, {'credit_id': '546892b422136e68d50007c5', 'department': 'Camera', 'gender': 0, 'id': 1385880, 'job': 'Director of Photography', 'name': 'John Fenner', 'profile_path': None}]

Variable similar a la anterior pero referente al equipo de producción y así desglosa por cada película, los diferentes técnicos implicados junto a su role en la misma: `Director`, `Productor`, `Compositor` etc etc.

`Nota. El desglose de roles asociados (jobs) es exhaustiva por película`

Conclusión: Se observa que se trata de variables relevantes para el análisis de regresión (es de suponer que el reparto de una película puede influir en su éxito comercial ó no) , pero que teniendo en cuenta su formato original no son muy explotables originalmente por lo que tendremos que plantear algún tipo de transformación que se detallará más adelante.

Respecto al resto de variables numéricas y para hacernos una idea de la estructura y formato de las mismas mostramos las 2 primeras filas del dataset .

`Nota. De este listado se han omitido las anteriores 'json' pra facilitar la visualización`

```
##   X.U.FEFF.id   budget original_language popularity release_date runtime
## 1           1 14000000                en   6.575393      2/20/15      93
## 2           2 40000000                en   8.248895       8/6/04     113
##                                    title  revenue
## 1                   Hot Tub Time Machine 2 12314651
## 2 The Princess Diaries 2: Royal Engagement 95149435
```

Pasamos a continuación a analizar más en detalle cada una de estas variables independientes para ver si necesitan o no limpieza de datos y analizar más en detalle su potencial real para el modelo de regresión lineal sobre la variable 'revenue'.

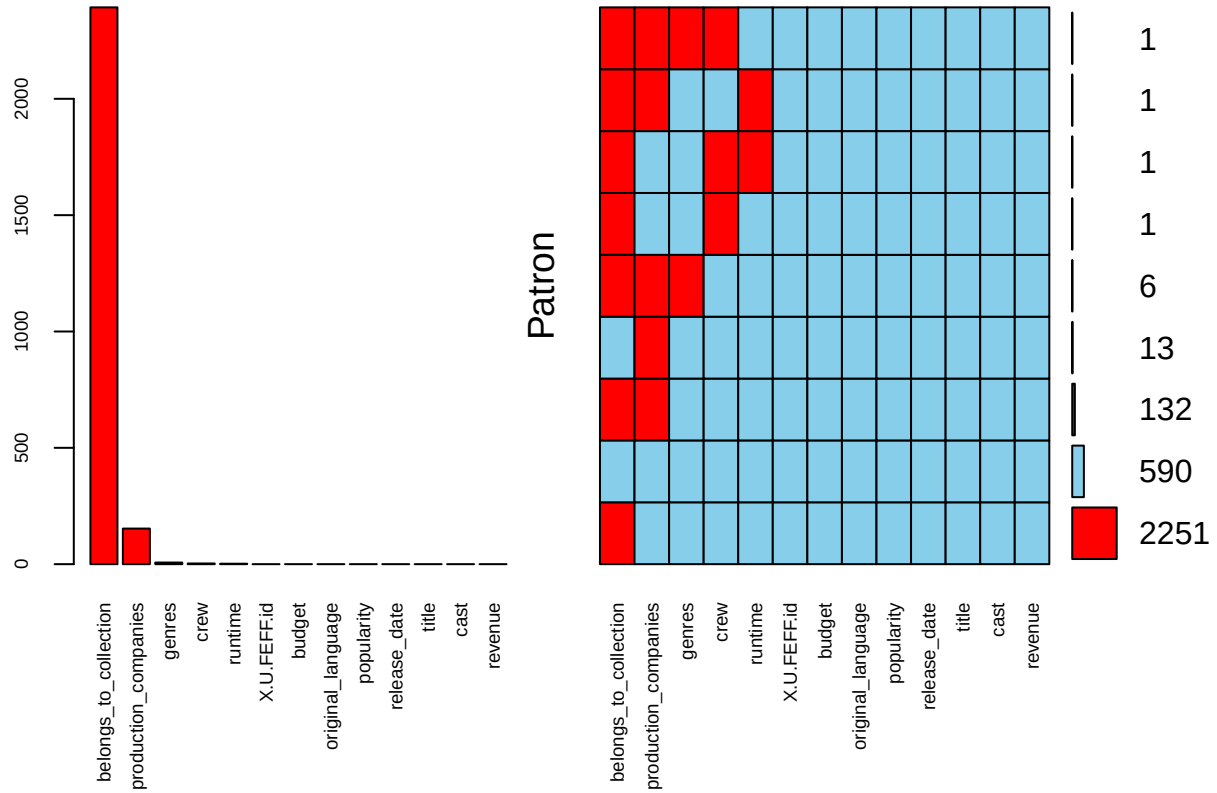
3. Limpieza de los datos

3.1. Valores Cero - Nulos

Valores Nulos

Identificamos qué variables de las seleccionadas presentan valores vacíos ó nulos para ver la calidad de los datos existentes.

```
##           X.U.FEFF.id belongs_to_collection                budget
##                     0                  2393                     0
##                genres     original_language            popularity
##                     7                     0                     0
##  production_companies          release_date               runtime
##                   153                     0                     2
##                 title                  cast                  crew
##                     0                     0                     3
##               revenue
##                     0
```




```
##
##  Variables sorted by number of missings:
##                Variable Count
##  belongs_to_collection  2393
##   production_companies   153
##                 genres     7
##                   crew     3
##                runtime     2
##             X.U.FEFF.id     0
##                 budget     0
##      original_language     0
##             popularity     0
##           release_date     0
##                  title     0
##                   cast     0
```

```
##                revenue     0
```

Observamos que la variable con más valores vacíos es **belongs_to_collection** con **2393** casos, seguida de **production_companies** con **153** casos.

En el caso de **belongs_to_collection** es normal encontrar valores vacíos ya que se trataría de todas aquellas películas que no pertenecen a ninguna colección (o saga), es decir que serán valores nulos.

Mientras que en el caso de **production_companies**, **genres**, **crew** y **runtime** se tratará de valores perdidos ó "NA".

Valores Cero

Veamos a continuación la existencia de los valores "0" en las variables numéricas para ver si éstos son coherentes con su naturaleza ó no.

Si su existencia estuviera dentro de lo posible, no haríamos cambios mientras que si no fuera posible, habría que transformar los valores a "NA".

```
##      budget             popularity         runtime         revenue
##  Min.   :        0   Min.   :  0.000   Min.   :  0.0   Min.   :1.000e+00
##  1st Qu.:        0   1st Qu.:  4.029   1st Qu.: 94.0   1st Qu.:2.402e+06
##  Median :  8000000   Median :  7.389   Median :104.0   Median :1.685e+07
##  Mean   : 22561119   Mean   :  8.469   Mean   :107.9   Mean   :6.681e+07
##  3rd Qu.: 29000000   3rd Qu.: 10.894   3rd Qu.:118.0   3rd Qu.:6.896e+07
##  Max.   :380000000   Max.   :294.337   Max.   :338.0   Max.   :1.520e+09
##                                        NA's   :2
```

Observamos que existen valores "0" en las variables **budget** (concretamente **809** casos) y **runtime** (con **12**) para los cuales no es un valor muy coherente (*).

En el caso de **popularity**, llama la atención que el valor mínimo no es "0" sino 10^{-6}. Existen concretamente **10** casos en los que el valor es menor de 0.01, no obstante, por tratarse de un valor numérico legítimo no le aplicaremos ninguna corrección.

```
(*) Nota: Según lo explicado en: <https://www.themoviedb.org/talk/5ba87d119251412f0103e87b>
si budget="0" querrá decir que los datos aún no están registrados, con lo cual habría que
modificar los valores a "NA". Idem para la variable 'runtime'.
```

3.2. Tratamiento de valores extremos I (*outliers*)

Este estudio de valores extremos se hará en dos fases diferentes.

1. Una primera, en este punto, en el que se identificarán inicialmente los valores extremos considerados incorrectos.
2. Otra segunda, más adelante en el apartado de Análisis de Datos (`Apartado 4.2.3`), cuando se identifiquen valores extremos en variables numéricas normalizadas mediante el método de $mean \pm 3SD$.

Se observa la presencia de valores anormalmente bajos en las variables **budget** y **revenue**.
Cotejando con el foro tanto de la competición de Kaggle, (https://www.kaggle.com/c/tmdb-box-office-prediction/discussion), como de TMDB, (https://www.themoviedb.org/talk?language=ca-ES), descubrimos que existen los siguientes problemas con ciertos valores de dichas variables:

- Valores con unidades dispares en **budget** y **revenue**. Algunos valores muy bajos han de multiplicarse por 1 millón

- Valores inconsistentes en **revenue**. Hay ciertos valores que recogen la recaudación sólo de "US & Canada" mientras que la gran mayoría recoge la recaudación internacional ("Worldwide").

La solución que planteamos ante este problema es fijar unos umbrales por encima de los cuales podemos considerar los valores correctos. Tras revisar las discusiones al respecto en las páginas web mencionadas anteriormente, decidimos fijar el umbral de **budget** en 1000 USD y el de **revenue** en 75.000 USD.

Contabilizamos el número de registros con valores anormalmente bajos y obtenemos los siguientes resultados:

Observamos que hay **18** registros con valores inferiores al umbral de 1000 USD de **budget** mientras que hay **218** registros con valores inferiores a 75.000 USD de **revenue**. De dichos registros, **13** películas, presentan valores inferiores a los umbrales en ambas variables.

3.3. Primeras conclusiones tras el análisis inicial

Después de este análisis preliminar llegamos a las siguientes conclusiones sobre las acciones necesarias de limpieza y transformación que tendremos que realizar sobre cada una de las variables independientes elegidas:

1. **belongs_to_collection**. Extraeremos la información de si pertenece o no a una saga y crearemos una nueva variable dicotómica con dicha información a la que llamaremos **collection**. (`Necesita Transformación`)
2. **budget**. Transformaremos a "NA" tanto los **809** valores "0" como aquellos **18** valores por debajo del umbral de 1000 USD. (`(Necesita gestión de nulos`)
3. **genres**. Nos interesa extraer los géneros asociados a cada película. (`Necesita Transformación`)
4. **original_language**. Nos interesa crear una nueva variable dicotómica a la que llamaremos **english_speaking** en función de si el idioma original es "english" ("yes") u otro ("no") (`Necesita Transformación`)
5. **runtime**. Imputaremos los **12** valores "0" y los **2** nulos. (`Necesita gestión de nulos`)
6. **release_date**. Vamos a generar 2 nuevas variables referentes al mes (**mes**) y año (**year**) de lanzamiento de cada película para intentar establecer alguna correlación entre éstos y el posible revenue de la misma. (`Necesita Transformación`)
7. **crew**. Se analiza en detalle más adelante para asociar a esta variable `json` un valor numérico . (`Necesita Transformación`)
8. **cast**. Idem al anterior. (`Necesita Transformación`)
9. **production_companies**. Idem al anterior. (`Necesita Transformación`)
10. **revenue**. Eliminaremos los registros correspondientes a los **218** casos con valores por debajo del umbral de 75.000 USD. (`Necesita gestión de nulos`)
11. **popularity**. Valor numérico correcto (`NO Necesita tratamiento`)
12. **title**. Atributo literal que funciona como ID de la película y que no se utilizará en la regresión (`NO Necesita tratamiento`)

3.4. Transformaciones y Tratamiento de las variables iniciales

- ***belongs_to_collection***

Como se ha mencionado anteriormente crearemos una nueva variable dicotómica llamada **collection** que recoja los valores "Yes"/"No" dependiendo de que el registro pertenezca o no a una saga, respectivamente.

Observamos que sólo en aquellas películas que pertenecen a una colección (o saga) tienen información presente en la variable **belongs_to_collection** mientras que el resto de películas tienen valores vacíos.

Por lo tanto creamos una nueva variable **collection** y enseñamos los primeros registros a modo de ejemplo:

title	belongs_to_collection	collection
Hot Tub Time Machine 2	[{'id': 313576, 'name': 'Hot Tub Time Machine Collection', 'poster_path': '/iEhb00TGPucF0b4joM1ieyY026U.jpg', 'backdrop_path': '/noeTVcgpBiD48fDjFVic1Vz7ope.jpg'}]	Yes
The Princess Diaries 2: Royal Engagement	[{'id': 107674, 'name': 'The Princess Diaries Collection', 'poster_path': '/wt5AMbxPTS4Kfjx7Fgm149qPfZl.jpg', 'backdrop_path': '/zSEtYD77pKRJlUPx34BJgUG9v1c.jpg'}]	Yes
Whiplash	NA	No
Kahaani	NA	No
Marine Boy	NA	No

- ***genres***

Observamos que una misma película puede estar clasificada como diferentes géneros.

Vamos a proceder a determinar cuántos posibles géneros de película hay y cuántas películas hay asociadas a cada uno de ellos.

Observamos que hay **20** géneros diferentes y que el género más frecuente entre las películas es el de "Drama" (**1530** películas) seguido de comedia (**1027** películas).

`Nota. Concretamente los géneros son: "Drama", "Comedy", "Thriller", "Action", "Romance", "Crime", "Adventure", "Horror", "Science_Fiction", "Family", "Fantasy", "Mystery", "Animation", "History", "Music", "War", "Documentary", "Western", "Foreign", "Tv_movie"`

Al haber registros que cuentan con múltiples valores de género, nos convendrá crear 20 variables nuevas con los distintos géneros y binarizar así la información.

Veamos cómo quedaría la transformación una vez realizado este proceso y mostramos los primeros registros:

title	genre	Drama	Comedy	Thriller	Action	Romance	Crime	Adventure	Horror	Science_Fiction
Hot Tub Time Machine 2	[{'id': 35, 'name': 'Comedy'}]	0	1	0	0	0	0	0	0	0
The Princess Diaries 2: Royal Engagement	[{'id': 35, 'name': 'Comedy'}, {'id': 18, 'name': 'Drama'}, {'id': 10751, 'name': 'Family'}, {'id': 10749, 'name': 'Romance'}]	1	1	0	0	1	0	0	0	0

Family	Fantasy	Mystery	Animation	History	Music	War	Documentary	Western	Foreign	Tv_movie
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0

Podemos observar que la película *Hot Tub Machine 2* está asociada exclusivamente al género "Comedy" en la variable **genres** y confirmamos, efectivamente, que la única columna donde hay un valor "1" es efectivamente la de "Comedy".

En cuanto a la siguiente película, *The Princess diaries 2: Royal Engagement*, observamos que está asociada a múltiples géneros: "Comedy", "Drama", "Family" y "Romance". Confirmamos que la binarización se ha realizado correctamente ya que las únicas columnas con valor "1" corresponden a dichos 4 géneros.

- ***original_language***

Analizamos los distintos idiomas originales (con sus frecuencias) presentes en la variable **original_language** y ordenamos de mayor a menor.

```
##
##   en   fr   ru   es   hi   ja   it   cn   ko   zh   de   ta   sv   nl   pt
## 2571   78   47   43   42   37   24   20   20   19   18   16    8    6    6
##   da   fa   ro   hu   tr   fi   ml   no   pl   te   ar   bn   cs   el   he
##    5    5    4    3    3    2    2    2    2    2    1    1    1    1    1
##   id   mr   nb   sr   ur   vi
##    1    1    1    1    1    1
```

Podemos observar que la gran mayoría de películas están rodadas en inglés. Crearemos pues una nueva variable dicotómica llamada **english_speaking** que recoja la información de si el idioma original de una película es el inglés ("Yes") u otro ("No").

title	original_language	english_speaking
Hot Tub Time Machine 2	en	Yes
The Princess Diaries 2: Royal Engagement	en	Yes
Whiplash	en	Yes
Kahaani	hi	No
Marine Boy	ko	No

- ***budget***

Transformamos los **809** valores "0" en "NA" y además transformamos los **18** valores por debajo del umbral de 1000 USD a "NA". De esta forma finalmente obtenemos 827 valores nulos.

Habría múltiples posibilidades de gestionar estos valores nulos como:

- Eliminar la variable 'budget' de cara al modelo de regresión pero sospechamos que ésta será una de las variables que más peso tenga en el mismo.
- Eliminar estos 827 registros con valor nulo del dataset de entenamiento del modelo
- Intentar reasignar estos valores nulos eligiendo aleatorimente otros valores de 'budget' existentes en el dataset.
- Intentar inferir estos valores con algún algoritmo como el de los k-vecinos más cercanos, que suele tener buenos resultados incluso cuando el porcentaje de valores ausentes es alto respecto al número de registros totales.

`Nota. Previo a decantarnos por una opción se consultó con la profesora, quien nos asesoró elegir esta última opción`

Finalmente procedemos, entonces, a imputar los **827** nulos usando la técnica de los k-vecinos más próximos (**KNN-imputation**) mediante la función `kNN()` de la `library(VIM)`.

X.U.FEFF.id	title	budget	budget_imput
5	Marine Boy	NA	8.0e+07
8	Control Room	NA	3.5e+05
9	Muppet Treasure Island	NA	2.6e+07
12	Revenge of the Nerds II: Nerds in Paradise	NA	1.7e+07
18	The Invisible Woman	NA	1.0e+07
23	V/H/S	NA	1.3e+07

runtime

Al igual que en el caso anterior imputamos los 12 valores "0" y los nulos mediante los k-vecinos más próximos (**KNN-imputation**) por medio de la función kNN() de la library(VIM).

X.U.FEFF.id	runtime	runtime_imput
391	NA	100
592	NA	102
925	NA	90
978	NA	106
1256	NA	117
1336	NA	116
1542	NA	105
1875	NA	91
2151	NA	95
2303	NA	90
2499	NA	93
2646	NA	115
2786	NA	96
2866	NA	100

revenue

Eliminamos los registros correspondientes a los **218** casos con valores por debajo del umbral de 75.000 USD.

Nos quedamos con 2778 registros después de haber eliminado 222 filas.

release_date

Tal y como se ha comentado anteriormente vamos a generar 2 nuevas variables a partir de la la fecha de estreno: **mes** y **year** (*) y que utilizaremos para ver si son significativas para la regresión del beneficio.

`(*) Nota. Estas 2 nuevas variables las hemos generado mediante sendas funciones que se encuentran definidas en el código`

Mostramos como quedan los primeros resgistros del *dataset* una vez aplicadas:

title	Release	mes	year
Hot Tub Time Machine 2	2/20/15	Febrero	2015
The Princess Diaries 2: Royal Engagement	8/6/04	Agosto	2004
Whiplash	10/10/14	Octubre	2014
Kahaani	3/9/12	Marzo	2012
Marine Boy	2/5/09	Febrero	2009

production_companies

Como ya se ha visto nos interesa sacar información numérica o categórica de la variables descriptiva **production_companies** para poder analizar si los nombres de las productoras son significativas para el posible 'revenue' de la película.

Para ello intentaremos transformar esta variable descriptiva en una variable numérica cuyo peso ó nivel de significación se establecerá en función de la importancia de cada productora en el ranking total de productoras según su número de producciones. Para explicarlo vamos a utilizar un ejemplo.

Previamente sacaremos el listado de todas las productoras que se referencian en el conjunto de datos junto al número de producciones de cada una de ellas lo que nos permitirá generar un ranking de las mismas.

Nota: Para realizar este cálculo se ha definido una función específica sacar_attr() que extrae los valores de los campos json.

Así obtenemos un listado de **3524** productoras en total. A continuación mostramos la tabla de productoras junto con el número de producciones de cada una de ellas.

```
##
##    1    2    3    4    5    6    7    8    9   10   11   21   12   13   14
## 2517  492  180  106   58   37   28   17   15    7    7    5    4    4    4
##   16   15   18   24   30   19   23   28   17   25   27   29   31   36   40
##    4    3    3    3    3    2    2    2    1    1    1    1    1    1    1
##   44   48   53   61   62   63   75   84   91  138  156  161  188  202
##    1    1    1    1    1    1    1    1    1    1    1    1    1    1
```

Un total de 2517 productoras sólo han producido una película y una de ellas ha producido 202. Mostramos el ranking de las 10 primeras.

```
##                                  nombre Freq
## 1                          Warner Bros.  202
## 2                     Universal Pictures  188
## 3                     Paramount Pictures  161
## 4                                 VACIO  156
## 5  Twentieth Century Fox Film Corporation  138
## 6                       Columbia Pictures   91
## 7              Metro-Goldwyn-Mayer (MGM)   84
## 8                         New Line Cinema   75
## 9                    Touchstone Pictures   63
## 10                  Walt Disney Pictures   62
```

Es decir la Warner ha producido 202 películas en total, Universal Pictures 188 etc.

Nota: Como era de esperar cuantas más películas producidas mas conocida es la productora para el gran público y potencialmente más taquillera la película

Notar que la 4º productora corresponde a VACIO que corresponde a aquellas películas que como se ha visto no tenían informado el campo. Para estas películas, sencillamente, la productora no será ningún valor añadido y tendrá un peso nulo

Y por último mostramos un resumen estadístico del número de producciones por compañía.

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   1.000   1.000   2.255   2.000 202.000
```

A la vista de los resultados se observa que sólo 15 productoras superan las 40 producciones en total contituyendo lo que denominaremos, por entendernos, el grupo de las 'superproductoras' (Aquellas que pueden tener cierto peso en el revenue esperado por su renombre)

A este valor, 40, le vamos a denominar 'umbral' y nos servirá como criterio para obtener diferentes métricas de las productoras asociadas a cada película(*) y a las que denominaremos prod1, prod2, pro3 y prod4. La idea es calcular inicialmente todas ellas para analizar posteriormente cual es la que presenta más correlación con el revenue de la película y así poder integrarla finalmente en el modelo de regresión final y descartando el resto.

`Nota. Cada película puede estar producida por una o varias productoras`

(*) Para asignar un posible peso de las productoras a la película se ha creado una función específica `puntuacion_plus()` que permite extraer del json los siguientes indicadores asociados a cada película'

1. **prod1**: Valor binario (1/0) en el supuesto de que la película haya participado al menos una superproductora

2. **prod2**: Valor entero correspondiente a la suma de superproductoras que participan en cada película.

3. **prod3**: Valor entero que corresponde a la suma de películas totales producidos por todas las superproductoras

4. **prod4**: Valor entero que corresponde a la suma de todas las películas producidas por cada productora participante (sin tener en cuenta si es superproductora o no) y con más de 1 película producida.(1 productora con una sola película es totalmene desconocida por lo que, apriori, no supondría un valor añadido para el éxito de la película).

Para evidenciar el valor asignado a cada una de las peliculas, vamos a analizar el caso de la primera película: **Hot Tub Time Machine 2** cuyas productoras asignadas vienen indicadas en la variable:

production_companies: [{'name': 'Paramount Pictures', 'id': 4}, {'name': 'United Artists', 'id': 60}, {'name': 'Metro-Goldwyn-Mayer (MGM)', 'id': 8411}]

El número de producciones asociado a cada una de ellas, según el listado que hemos obtenido anteriormente, es:

```
##                        nombre Freq
## 3         Paramount Pictures  161
## 7  Metro-Goldwyn-Mayer (MGM)   84
## 14            United Artists   44
```

`Nota. Se observa que en este caso las 3 productoras superan el valor umbral indicado, 40, y que permite considerarlas como superproductoras`

En base a todo lo anterior tenemos que los indicadores prod1..prod4 asociados a la película **Hot Tub Time Machine 2** serían:

1. $prod1 = 1$ (Al menos una de las productoras es una 'superproductora')

2. $prod2 = 3$ (Número de superproductoras implicadas en la película)

3. $prod3 = 289(161 + 84 + 44)$ (Suma de producciones asociadas a las superproductoras)
4. $prod4 = 289(161 + 84 + 44)$. Suma de producciones asociadas a todas las productoras implicadas (sean o no sean superproductoras) con más de 1 película producida. En este caso coincide con el valor anterior porque todas las productoras participantes son top.

Realizamos el mismo cálculo para todas las películas del dataset y lo comprobamos mostrando los valores *prodN* asociados a las primeras películas:

```
##                                     title prod1 prod2 prod3 prod4
## 1                   Hot Tub Time Machine 2     1     3   289   289
## 2 The Princess Diaries 2: Royal Engagement     1     1    62    62
## 3                                 Whiplash     0     0     0    26
## 4                                  Kahaani     0     0     0     0
## 5                                Marine Boy     0     0     0     0
## 6   Pinocchio and the Emperor of the Night     0     0     0     0
```

Una vez obtenidas estas nuevas variables nos interesa ver cual presenta mayor de coeficiente de correlación con la variable dependiente **revenue**.

Más adelante se demostrará que no se puede asumir la hipótesis de normalidad de la variables 'revenue', por lo que no podremos realizar un test de correlación paramétrico y tendremos que recurrir a una prueba no-paramétrica como el **test de Kendall**.

En la siguiente tabla se recoge el resultado de realizar dicho test de correlación entre la variable 'revenue' y cada uno de los indicadores anteriores. Se observa que el correspondiente al indicador *Prod*4 es el que mejor valor arroja y por tanto será el que elegiremos como variable independiente de cara al modelo de regresión final.

Variable	Tau	pvalor
Prod1	0.3518593	0
Prod2	0.3550216	0
Prod3	0.3183697	0
Prod4	0.3687573	0

Cast (Reparto)

Análogamente a lo visto anteriormente nos interesa sacar información de las variable descriptiva **cast** para poder analizar si los nombres de actores y actrices del reparto, respectivamente, son significativas para el posible 'revenue' de la película.

El planteamiento será totalmente similar al explicado anteriormente para las productoras

Para ello, de nuevo, intentaremos transformar estas variable decriptiva en una variable numérica cuyo peso ó nivel de significación se establecerá en función de la importancia de actor-actriz en el ranking total según el número de películas interpretadas.

Para ello, de nuevo, previamente vamos a sacar el listado de actores-actrices junto al número de películas interprestadas.

`Nota. Utilizaremos la  función ya mencionada sacar_attr() que extrae los valores de los campos json.`

Así obtenemos un listado de 36840 artistas de los cuales mostramos la tabla de distribución

```
##
##     1     2     3     4     5     6     7     8     9    10    11    12
## 27800  4747  1747   840   504   333   223   139   139    82    48    50
##    13    14    15    16    17    18    19    20    21    22    23    24
##    44    34    29    19    14     9    10     9     6     3     2     2
##    25    27    30
##     4     1     2
```

Sólo 2 actores han interpretado 30 películas y 27800, totalmente desconocidos, sólo 1.

Mostramos los 10 actores-actrices que más películas han interpretado según los datos disponibles

```
##                nombre Freq
## 1     Robert De Niro   30
## 2  Samuel L. Jackson   30
## 3     Morgan Freeman   27
## 4       Bruce Willis   25
## 5       J.K. Simmons   25
## 6        Liam Neeson   25
## 7     Susan Sarandon   25
## 8       Bruce McGill   24
## 9      John Turturro   24
## 10   Forest Whitaker   23
```

A continuación mostramos un resumen estadístico del número de películas interpretadas por cada artista.

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    1.00    1.00    1.61    1.00   30.00
```

A la vista de los resultados se observa que sólo 20 artistas superan las 20 películas en total contituyendo el grupo de 'superestrellas'.

A este valor, 20, le vamos a denominar 'umbral' y nos servirá para obtener métricas similares a las planteadas para las productoras pero asociadas en esta caso a cada artista. (*).

```
Nota. Obviamente cada película está interprestada por más de un artista
(*) Utilizaremos la misma función mencionada anteriormente, puntuacion_plus() que permite
extraer del json el nombre de los actores y actrices implicadas
```

1. **rep1**: Valor binario (1/0) en el supuesto de que en el reparto haya al menos una superestrella

2. **rep2**: Valor entero correspondiente al total de superestrellas del reparto.

3. **rep3**: Valor entero que corresponde al total de todas las películas interpretadas por las superestrellas del reparto.

4. **rep4**: Valor entero que corresponde al total de todas las películas interpretadas por todo el reparto(sin tener en cuenta si son superestrellas o no)

```
Nota. Al igual que en el caso anterior plantearemos  diferentes indicadores y elegiremos
el que mejor margen de correlación arroje con la variable dependiente revenue
```

Para evidenciar el valor asignado a cada una de las peliculas, en base al reparto vamos a analizar el caso de una de las películas en las que participa uno de los actores con más films en su haber como **Robert De Niro** en la película: **The Deer Hunter**.

```
Se observa que como cabeza de cartel aparece Rober de Niro interpretando al personaje
masculino Michael Vronsky
```

Los datos se encuentran disponibles en la variable:
cast=[{'cast_id': 11, 'character': 'Michael Vronsky', 'credit_id': '52fe44849251416c750378c7', 'gender': 2, 'id': 380, 'name': 'Robert De Niro', 'order': 0, 'profile_path': '/lvTSwUcvJRLAJ2FB5qFaukel516.jpg'}, {'cast_id': 15, 'character': 'Stan', 'credit_id': '52fe44849251416c750378d7', 'gender': 2, 'id': 3096, 'name': 'John Cazale', 'order': 1, 'profile_path': '/d8lIfeeAbX8qALMYpVBc3y5Lkp6.jpg'}, {'cast_id': 13, 'character': 'Steven Pushkov', 'credit_id': '52fe44849251416c750378cf', 'gender': 2, 'id': 47879, 'name': 'John Savage', 'order': 2, 'profile_path': '/4cNKdr2E2gAvAjjydcwJGscoBsD.jpg'}, {'cast_id': 12, 'character': 'Nikanor "Nick" Chevotarevich', 'credit_id': '52fe44849251416c750378cb', 'gender': 2, 'id': 4690,

‘name’: ‘Christopher Walken’, ‘order’: 3, ‘profile_path’: ‘/ysO1GwRzLT9OVAB9Y2SKHxomqDr.jpg’}, {‘cast_id’: 14, ‘character’: ‘Linda’, ‘credit_id’: ‘52fe44849251416c750378d3’, ‘gender’: 1, ‘id’: 5064, ‘name’: ‘Meryl Streep’, ‘order’: 4, ‘profile_path’: ‘/oTJj6bLpbmseLww03MOn0eDqYuh.jpg’}, {‘cast_id’: 16, ‘character’: ‘John’, ‘credit_id’: ‘52fe44849251416c750378db’, ‘gender’: 2, ‘id’: 10477, ‘name’: ‘George Dzundza’, ‘order’: 5, ‘profile_path’: ‘/jyjr4P3uCpfrbwk8ySXNaDpBMbc.jpg’}, {‘cast_id’: 19, ‘character’: ‘Angela’, ‘credit_id’: ‘52fe44849251416c750378e7’, ‘gender’: 1, ‘id’: 80135, ‘name’: ‘Rutanya Alda’, ‘order’: 6, ‘profile_path’: ‘/gVOyv8nNYwXmQU7kixKxILUMM50.jpg’}, {‘cast_id’: 20, ‘character’: ‘Julien’, ‘credit_id’: ‘52fe44849251416c750378eb’, ‘gender’: 0, ‘id’: 133859, ‘name’: ‘Pierre Segui’, ‘order’: 7, ‘profile_path’: ‘/XYGQKCDdGtT3XNbUqCy83hoNhZ.jpg’}, {‘cast_id’: 22, ‘character’: ‘Bridesmaid’, ‘credit_id’: ‘52fe44849251416c750378f3’, ‘gender’: 1, ‘id’: 87007, ‘name’: ‘Amy Wright’, ‘order’: 8, ‘profile_path’: ‘/mEV42JRtrTTZg92GjyjRkQSXpsn.jpg’}, {‘cast_id’: 64, ‘character’: “Linda’s Father”, ‘credit_id’: ‘54f5d1d29251412ba90023a1’, ‘gender’: 0, ‘id’: 1222806, ‘name’: ‘Richard Kuss’, ‘order’: 9, ‘profile_path’: ‘/6MtqcyZklzUK03uVGV8ARZwSaPp.jpg’}, {‘cast_id’: 25, ‘character’: ‘Bandleader’, ‘credit_id’: ‘52fe44849251416c750378ff’, ‘gender’: 2, ‘id’: 4887, ‘name’: ‘Joe Grifasi’, ‘order’: 10, ‘profile_path’: ‘/cesXbWZaKHeOhsSdfbF9mB9750w.jpg’}, {‘cast_id’: 70, ‘character’: ‘Cab Driver’, ‘credit_id’: ‘54f5f32d925141068b000195’, ‘gender’: 0, ‘id’: 1258752, ‘name’: ‘Dennis Watlington’, ‘order’: 11, ‘profile_path’: ‘/mB23y8qO2I3lxvVV7n4OYwmMHFn.jpg’}, {‘cast_id’: 18, ‘character’: “Steven’s Mother”, ‘credit_id’: ‘52fe44849251416c750378e3’, ‘gender’: 1, ‘id’: 67513, ‘name’: ‘Shirley Stoler’, ‘order’: 12, ‘profile_path’: ‘/cRMsHh1giZv4ha5tSeXmHBu0LDQ.jpg’}, {‘cast_id’: 17, ‘character’: ‘Axel’, ‘credit_id’: ‘52fe44849251416c750378df’, ‘gender’: 0, ‘id’: 123056, ‘name’: ‘Chuck Aspegren’, ‘order’: 13, ‘profile_path’: ‘/tWVzZsHizswaLalLkGjHisacNST.jpg’}, {‘cast_id’: 23, ‘character’: “Stan’s Girl”, ‘credit_id’: ‘52fe44849251416c750378f7’, ‘gender’: 0, ‘id’: 133861, ‘name’: ‘Mary Ann Haenel’, ‘order’: 14, ‘profile_path’: None}, {‘cast_id’: 21, ‘character’: “Axel’s Girl”, ‘credit_id’: ‘52fe44849251416c750378ef’, ‘gender’: 1, ‘id’: 133860, ‘name’: ‘Mady Kaplan’, ‘order’: 15, ‘profile_path’: None}, {‘cast_id’: 27, ‘character’: ‘Sergeant’, ‘credit_id’: ‘53c41f4c0e0a26157c00cdeb’, ‘gender’: 0, ‘id’: 134148, ‘name’: “Paul D’Amato”, ‘order’: 16, ‘profile_path’: None}, {‘cast_id’: 65, ‘character’: ‘Wedding Man’, ‘credit_id’: ‘54f5d4609251412ba90023c0’, ‘gender’: 0, ‘id’: 1434628, ‘name’: ‘Christopher Colombi Jr.’, ‘order’: 17, ‘profile_path’: None}, {‘cast_id’: 66, ‘character’: ‘Sad Looking Girl’, ‘credit_id’: ‘54f5d4c29251412ba20025df’, ‘gender’: 0, ‘id’: 1434629, ‘name’: ‘Victoria Karnafel’, ‘order’: 18, ‘profile_path’: None}, {‘cast_id’: 67, ‘character’: ‘Cold Old Man’, ‘credit_id’: ‘54f5d5879251412bb60024eb’, ‘gender’: 0, ‘id’: 1434630, ‘name’: ‘Jack Scardino’, ‘order’: 19, ‘profile_path’: None}, {‘cast_id’: 68, ‘character’: ‘Bingo Caller’, ‘credit_id’: ‘54f5d60cc3a36834f30024a8’, ‘gender’: 0, ‘id’: 1434631, ‘name’: ‘Joe Strnad’, ‘order’: 20, ‘profile_path’: None}, {‘cast_id’: 69, ‘character’: ‘Helen’, ‘credit_id’: ‘54f5d66f9251412ba90023eb’, ‘gender’: 0, ‘id’: 1434632, ‘name’: ‘Helen Tomko’, ‘order’: 21, ‘profile_path’: None}, {‘cast_id’: 71, ‘character’: ‘Red Head’, ‘credit_id’: ‘54f5f362c3a36815520001d0’, ‘gender’: 0, ‘id’: 1434671, ‘name’: ‘Charlene Darrow’, ‘order’: 22, ‘profile_path’: None}, {‘cast_id’: 72, ‘character’: ‘Girl Checker’, ‘credit_id’: ‘54f5f3aac3a368153e0001b1’, ‘gender’: 0, ‘id’: 1434672, ‘name’: ‘Jane-Colette Disko’, ‘order’: 23, ‘profile_path’: None}, {‘cast_id’: 73, ‘character’: ‘Stock Boy’, ‘credit_id’: ‘54f5f3fc92514106830001fe’, ‘gender’: 0, ‘id’: 1434673, ‘name’: ‘Michael Wollet’, ‘order’: 24, ‘profile_path’: None}, {‘cast_id’: 74, ‘character’: ‘World War Veteran’, ‘credit_id’: ‘54f5f445c3a36815570001bf’, ‘gender’: 0, ‘id’: 1434674, ‘name’: ‘Robert Beard’, ‘order’: 25, ‘profile_path’: None}, {‘cast_id’: 75, ‘character’: ‘World War Veteran’, ‘credit_id’: ‘54f811189251416ee4003382’, ‘gender’: 0, ‘id’: 1435119, ‘name’: ‘Joe Dzizmba’, ‘order’: 26, ‘profile_path’: None}, {‘cast_id’: 76, ‘character’: ‘Priest’, ‘credit_id’: ‘54f8116b92514118ba000c99’, ‘gender’: 0, ‘id’: 1435120, ‘name’: ‘Stephen Kopestonsky’, ‘order’: 27, ‘profile_path’: None}, {‘cast_id’: 77, ‘character’: ‘Bar Patron’, ‘credit_id’: ‘54f8120b92514151c6003fe5’, ‘gender’: 0, ‘id’: 1435121, ‘name’: ‘John F. Buchmelter III’, ‘order’: 28, ‘profile_path’: None}, {‘cast_id’: 78, ‘character’: ‘Barman’, ‘credit_id’: ‘54f81255c3a368131c000b93’, ‘gender’: 0, ‘id’: 1435124, ‘name’: ‘Frank Devore’, ‘order’: 29, ‘profile_path’: None}, {‘cast_id’: 79, ‘character’: ‘Doctor’, ‘credit_id’: ‘54f812f9c3a36833bb003ba6’, ‘gender’: 0, ‘id’: 1435133, ‘name’: ‘Tom Becker’, ‘order’: 30, ‘profile_path’: None}, {‘cast_id’: 80, ‘character’: ‘Nurse’, ‘credit_id’: ‘54f81341c3a368131c000bac’, ‘gender’: 0, ‘id’: 1435135, ‘name’: ‘Lynn Kongkham’, ‘order’: 31, ‘profile_path’: None}, {‘cast_id’: 81, ‘character’: ‘Bar Girl’, ‘credit_id’: ‘54f81381c3a36833dc00343c’, ‘gender’: 0, ‘id’: 1435136, ‘name’: ‘Nongnuj Timruang’, ‘order’: 32, ‘profile_path’: None}, {‘cast_id’: 82, ‘character’: ‘Chinese Referee’, ‘credit_id’: ‘54f813f692514118ba000cd4’, ‘gender’: 0, ‘id’: 1435138, ‘name’: ‘Po Pao Pee’, ‘order’: 33, ‘profile_path’: None}, {‘cast_id’: 83,

‘character’: ‘Embassy Guard’, ‘credit_id’: ‘54f8143692514118ba000cdd’, ‘gender’: 0, ‘id’: 1435141, ‘name’: ‘Dale Burroughs’, ‘order’: 34, ‘profile_path’: None}, {‘cast_id’: 84, ‘character’: ‘Sergeant’, ‘credit_id’: ‘54f814879251416f370036c7’, ‘gender’: 0, ‘id’: 1435144, ‘name’: ‘Parris Hicks’, ‘order’: 35, ‘profile_path’: None}, {‘cast_id’: 85, ‘character’: ‘Chinese Bodyguard’, ‘credit_id’: ‘54f8150cc3a36834a500388f’, ‘gender’: 0, ‘id’: 1435145, ‘name’: ‘Samui Muang-Intata’, ‘order’: 36, ‘profile_path’: None}, {‘cast_id’: 86, ‘character’: ‘Chinese Man’, ‘credit_id’: ‘54f8154592514151c6004047’, ‘gender’: 0, ‘id’: 1435146, ‘name’: ‘Sapox Colisium’, ‘order’: 37, ‘profile_path’: None}, {‘cast_id’: 87, ‘character’: ‘NVA Officer’, ‘credit_id’: ‘54f81599c3a36833dc003477’, ‘gender’: 0, ‘id’: 1435147, ‘name’: ‘Vitoon Winwitoon’, ‘order’: 38, ‘profile_path’: None}, {‘cast_id’: 88, ‘character’: ‘V.C. Referee’, ‘credit_id’: ‘54f815eb9251411812000cf2’, ‘gender’: 0, ‘id’: 1435148, ‘name’: ‘Somsak Sengvilai’, ‘order’: 39, ‘profile_path’: None}, {‘cast_id’: 89, ‘character’: ‘Chinese Boss’, ‘credit_id’: ‘54f8162c9251416ee4003428’, ‘gender’: 0, ‘id’: 1435149, ‘name’: ‘Charan Nusvanon’, ‘order’: 40, ‘profile_path’: None}, {‘cast_id’: 90, ‘character’: ‘Chinese Man At Door’, ‘credit_id’: ‘54f81667c3a368351d003782’, ‘gender’: 0, ‘id’: 1435150, ‘name’: ‘Jiam Gongtongsmoot’, ‘order’: 41, ‘profile_path’: None}, {‘cast_id’: 91, ‘character’: ‘South Vietnamese Prisoner’, ‘credit_id’: ‘54f816aac3a36834a50038c8’, ‘gender’: 0, ‘id’: 1435153, ‘name’: ‘Chai Peyawan’, ‘order’: 42, ‘profile_path’: None}, {‘cast_id’: 92, ‘character’: ‘South Vietnamese Prisoner’, ‘credit_id’: ‘54f816fb92514118ba000d55’, ‘gender’: 0, ‘id’: 1435154, ‘name’: ‘Mana Hansa’, ‘order’: 43, ‘profile_path’: None}, {‘cast_id’: 93, ‘character’: ‘South Vietnamese Prisoner’, ‘credit_id’: ‘54f81754c3a368126c000e2e’, ‘gender’: 0, ‘id’: 1435155, ‘name’: ‘Sombot Jumpanoi’, ‘order’: 44, ‘profile_path’: None}, {‘cast_id’: 94, ‘character’: ‘Woman In Village’, ‘credit_id’: ‘54f8178d92514124110040ae’, ‘gender’: 0, ‘id’: 1435156, ‘name’: ‘Phip Manee’, ‘order’: 45, ‘profile_path’: None}, {‘cast_id’: 95, ‘character’: ‘V.C. Guard’, ‘credit_id’: ‘54f817e79251416f6e0037ea’, ‘gender’: 0, ‘id’: 1435159, ‘name’: ‘Ding Santos’, ‘order’: 46, ‘profile_path’: None}, {‘cast_id’: 96, ‘character’: ‘V.C. Guard’, ‘credit_id’: ‘54f8184bc3a36833bb003c54’, ‘gender’: 0, ‘id’: 1435160, ‘name’: ‘Krieng Chaiyapuk’, ‘order’: 47, ‘profile_path’: None}, {‘cast_id’: 97, ‘character’: ‘V.C. Guard’, ‘credit_id’: ‘54f8188792514118ba000d84’, ‘gender’: 0, ‘id’: 1435161, ‘name’: ‘Ot Palapoo’, ‘order’: 48, ‘profile_path’: None}, {‘cast_id’: 98, ‘character’: ‘V.C. Guard’, ‘credit_id’: ‘54f818c492514151c60040ae’, ‘gender’: 0, ‘id’: 1435162, ‘name’: ‘Chok Chai Mahasoke’, ‘order’: 49, ‘profile_path’: None}]

De la cual hemos extraído el siguiente reparto asignado junto al número de películas de cada uno de los artistas:

```
##                        nombre Freq
## 1                Robert De Niro   30
## 33          Christopher Walken   19
## 76                Meryl Streep   16
## 1357            George Dzundza    5
## 1747                Amy Wright    4
## 2396              Rutanya Alda    4
## 3318               Joe Grifasi    3
## 3348               John Savage    3
## 6499               John Cazale    2
## 13248             Chai Peyawan    1
## 13284          Charan Nusvanon    1
## 13294          Charlene Darrow    1
## 13616       Chok Chai Mahasoke    1
## 13911 Christopher Colombi Jr.    1
## 14017           Chuck Aspegren    1
## 14631           Dale Burroughs    1
## 15752        Dennis Watlington    1
## 15972              Ding Santos    1
## 17741             Frank Devore    1
## 19265              Helen Tomko    1
## 20029            Jack Scardino    1
## 20534       Jane-Colette Disko    1
## 21426       Jiam Gongtongsmoot    1
```

```
## 21742             Joe Dzizmba    1
## 21811              Joe Strnad    1
## 22029  John F. Buchmelter III    1
## 24163        Krieng Chaiyapuk    1
## 25670           Lynn Kongkham    1
## 25754             Mady Kaplan    1
## 25838              Mana Hansa    1
## 26517         Mary Ann Haenel    1
## 27563          Michael Wollet    1
## 28960        Nongnuj Timruang    1
## 29217              Ot Palapoo    1
## 29350            Parris Hicks    1
## 30145              Phip Manee    1
## 30189            Pierre Segui    1
## 30232              Po Pao Pee    1
## 30959            Richard Kuss    1
## 31250            Robert Beard    1
## 32298      Samui Muang-Intata    1
## 32385          Sapox Colisium    1
## 33141          Shirley Stoler    1
## 33339         Sombot Jumpanoi    1
## 33341        Somsak Sengvilai    1
## 33679     Stephen Kopestonsky    1
## 34936              Tom Becker    1
## 35791       Victoria Karnafel    1
## 35940        Vitoon Winwitoon    1
```

En base a todo lo anterior tenemos que los indicadores prod1..prod4 asociados a la película **The Last Unicorn** serían:

1. $rep1 = 1$ (Al menos en el reparto hay una superestrella)

2. $rep2 = 1$ (Suma de superestrellas en el reparto. Sólo Robert de Niro)

3. $rep3 = 30$ (Suma de las peliculas interpretadas en total por las superestrellas del reparto. Sólo Robert de Niro)

4. $rep4 = 86(30 + 19 + 16 + 5 + 4 + 4 + 3 + 3 + 2)$ (Suma de las peliculas interpretadas en total por todos los actores-actrices del reparto - sean o no sean superestrellas - y con más de 1 película en su haber)

'Nota. Al igual que en el caso de las productoras, nos contamos los que tienen sólo 1 película en su haber al considerar que son totalmente desconocidos para el gran público y por tanto, a priori, con poco 'reclamo' comercial'‘

Realizamos el mismo cálculo para todas las películas del dataset y mostramos, a modo de ejemplo unas películas y entre ellos la mencionada anteriormente, para evidenciar esta asignación de nuevas variables.

```
##                                      title rep1 rep2 rep3 rep4
## 2049                       The Deer Hunter    1    1   30   86
## 30                                   Caché    0    0    0   13
## 1                  Hot Tub Time Machine 2    0    0    0   96
## 2  The Princess Diaries 2: Royal Engagement    0    0    0   80
```

Al igual que hemos hecho en los casos anteriores procederemos a analizar el coeficiente de correlación de cada uno de estos indicadores con la variable 'revenue' obteniendo la siguiente tabla.

Variable	Tau	pvalor
rep1	0.1504592	0
rep2	0.1491905	0
rep3	0.1450539	0
rep4	0.3550457	0

Elegimos para el modelo de regresión final aquel que presenta mejor valor de Tau y que es `rep4`.

Crew (Equipo Producción)

Al igual que hemos hecho para las otras variables, en este caso vamos a intentar obtener un valor cuantitativo asociado a la variable **crew** que pretende recoger el impacto económico en dividendos que el equipo de producción de una película podría tener de cara a los beneficios de la misma.

Para ello, y al igual que hemos hecho en el caso anterior, previamente vamos a sacar el listado de todos los miembros del equipo de producción junto al número de películas en las que han participado utilizando la misma función `sacar_attr()` que extrae los valores de los campos json.

Así obtenemos un listado de 37125 técnicos de los cuales obtenemos la siguiente tabla de distribución de películas realizadas.

```
##
##     1     2     3     4     5     6     7     8     9    10    11    12
## 24028  6282  2724  1492   825   524   351   243   144   133   107    54
##    13    14    15    16    17    18    19    20    21    22    23    24
##    43    26    28    24     8     6    12     9     9     9     5     4
##    25    26    27    28    29    30    32    33    35    37    38    39
##     6     6     3     4     1     3     1     1     2     3     1     1
##    40    44    50
##     1     1     1
```

Mostramos los 10 técnicos que más han participado en la realización de las películas del dataset sea en el role que sea.

```
##                 nombre Freq
## 1          Avy Kaufman   50
## 2     Robert Rodriguez   44
## 3       Deborah Aquila   40
## 4  James Newton Howard   39
## 5         Mary Vernieu   38
## 6      Jerry Goldsmith   37
## 7           Luc Besson   37
## 8     Steven Spielberg   37
## 9     Francine Maisler   35
## 10         Tricia Wood   35
```

Por último, mostramos un resumen estadístico del número de producciones por técnico.

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    1.00    1.00    1.91    2.00   50.00
```

A la vista de los resultados se observa que sólo 15 personas han trabajado en 30 ó más películas en total contituyendo lo que denominaremos los técnicos 'Top'.

Se observa que en dicho listado aparecen nombres relevantes como el asociado al director "Steven Spielberg".

Este valor, 30, le vamos a denominar 'umbral' y nos servirá para obtener métricas similares a las anteriores asociadas a cada técnico 'top'. (*).

`(*) Utilizaremos la misma función mencionada anteriormente, puntuacion_plus() que permite extraer del json el nombre de los actores y actrices implicadas`

1. **equip1**: Valor binario (1/0) en el supuesto de que en el reparto haya un técnico del 'Top'

2. **equip2**: Valor entero correspondiente al total de técnicos 'Top'.

3. **equip3**: Valor entero que corresponde al total de todas las películas en las que ha participado los técnicos 'Top'.

4. **equip4**: Valor entero que corresponde al total de todas las películas realizadas por cualquier técnico del equipo de producción(sin tener en cuenta si es miembro del denominado equipo 'Top' ó no)

`Nota. Al igual que en el caso anterior plantearemos diferentes indicadores y elegiremos el que mejor margen de correlación arroje`

Para evidenciar el valor asignado a cada una de las peliculas, en base al reparto vamos a analizar el caso de una de las películas en las que participa uno de los técnicos con más films en su haber como **Avy Kaufman** en la película: **Boys Life 2**.

Los valores se extraen de la variable: **crew**=[{'credit_id': '52fe47b9c3a36847f8144e3f', 'department': 'Directing', 'gender': 2, 'id': 33541, 'job': 'Director', 'name': 'Mark Christopher', 'profile_path': None}, {'credit_id': '52fe47b9c3a36847f8144e45', 'department': 'Directing', 'gender': 0, 'id': 56947, 'job': 'Director', 'name': 'Peggy Rajski', 'profile_path': '/wXvcBCMfHWcsTuNI9NbmK4G3g2K.jpg'}, {'credit_id': '52fe47b9c3a36847f8144e4b', 'department': 'Directing', 'gender': 2, 'id': 137392, 'job': 'Director', 'name': 'Nickolas Perry', 'profile_path': None}, {'credit_id': '52fe47b9c3a36847f8144e51', 'department': 'Directing', 'gender': 0, 'id': 1175207, 'job': 'Director', 'name': 'Tom DeCerchio', 'profile_path': None}, {'credit_id': '5555cc549251411e4d0022db', 'department': 'Writing', 'gender': 2, 'id': 137392, 'job': 'Writer', 'name': 'Nickolas Perry', 'profile_path': None}, {'credit_id': '5555cccac3a368777400233e', 'department': 'Production', 'gender': 1, 'id': 2952, 'job': 'Casting', 'name': 'Avy Kaufman', 'profile_path': '/yQPGktsmkKkhkOQAUlmYlxHJOiJ.jpg'}, {'credit_id': '5555cbd69251411e5400215e', 'department': 'Writing', 'gender': 2, 'id': 33541, 'job': 'Writer', 'name': 'Mark Christopher', 'profile_path': None}, {'credit_id': '5555cbec9251411e5f002104', 'department': 'Writing', 'gender': 0, 'id': 1175207, 'job': 'Writer', 'name': 'Tom DeCerchio', 'profile_path': None}, {'credit_id': '5555cc059251412ff9000ae8', 'department': 'Writing', 'gender': 0, 'id': 88758, 'job': 'Writer', 'name': 'James Lecesne', 'profile_path': None}, {'credit_id': '5555cc8ac3a368776a0020a5', 'department': 'Production', 'gender': 1, 'id': 4447, 'job': 'Producer', 'name': 'Ann Ruark', 'profile_path': None}, {'credit_id': '5555cca0c3a368777200242b', 'department': 'Production', 'gender': 0, 'id': 1467299, 'job': 'Producer', 'name': 'Rafi Stephan', 'profile_path': None}, {'credit_id': '5555ccb69251411e5f002120', 'department': 'Production', 'gender': 2, 'id': 21069, 'job': 'Producer', 'name': 'Randy Stone', 'profile_path': None}, {'credit_id': '5555ccdf9251411e51002306', 'department': 'Production', 'gender': 0, 'id': 63459, 'job': 'Casting', 'name': 'Aaron Griffith', 'profile_path': None}, {'credit_id': '5555cc729251411e5b00227c', 'department': 'Production', 'gender': 0, 'id': 56947, 'job': 'Producer', 'name': 'Peggy Rajski', 'profile_path': '/wXvcBCMfHWcsTuNI9NbmK4G3g2K.jpg'}, {'credit_id': '5555cd689251411e620021fa', 'department': 'Production', 'gender': 1, 'id': 2045, 'job': 'Casting', 'name': 'Ferne Cassel', 'profile_path': None}]

Extrayendo los datos de la variable anterior obtenemos que el equipo de producción es el siguiente junto al número de películas de cada uno:

```
##                 nombre Freq
## 1          Avy Kaufman   50
## 1270         Ann Ruark    6
## 3057      Ferne Cassel    4
## 3528  Mark Christopher    4
## 6120      Peggy Rajski    3
## 11244   Nickolas Perry    2
## 11657      Randy Stone    2
## 12752    Tom DeCerchio    2
## 13137   Aaron Griffith    1
## 22749    James Lecesne    1
## 31721     Rafi Stephan    1
```

En base a todo lo anterior tenemos que los indicadores equip1..equip4 asociados a la película **Boys Life 2** serían:

1. $equip1 = 1$ (Al menos en el equipo de producción hay un técnico Top - Avy Kaufman)
2. $equip2 = 1$. (Suma de técnicos top en el equipo de producción - Sólo Avy Kaufman)
3. $equip3 = 50$. (Suma de peliculas realizadas en total por todas las personas del equipo top - Sólo Avy Kaufman) 4. $equip4 = 73 (50+6+4+4+3+2+2+2)$. Suma de peliculas realizadas en total por todos los técnicos del equipo - sean ó no 'sean superequipo'Top', pero con más de 1 película en su haber

Realizamos el mismo cálculo para todas las películas del dataset y mostramos, a modo de ejemplo, algunos films, entre ellos el mencionado anteriormente, para evidenciar esta asignación de nuevas variables.

```
##                                     title equip1 equip2 equip3 equip4
## 163                          Boys Life 2      1      1     50     73
## 1                  Hot Tub Time Machine 2      0      0      0    133
## 2   The Princess Diaries 2: Royal Engagement      0      0      0     64
```

Al igual que hemos hecho en los casos anteriores procederemos a analizar el coeficiente de correlación de cada uno de estos indicadores con la variable 'revenue' obteniendo la siguiente tabla.

Variable	Tau	pvalor
Equip1	0.1442500	0
Equip2	0.1436482	0
Equip3	0.1400234	0
Equip4	0.4096003	0

Elegimos para el modelo de regresión final aquel que presenta mejor valor de Tau y que corresponde a 'equip4'.

4. Análisis de los datos

4.1. Planificación de los análisis a aplicar

Procederemos a analizar por un lado las variables cuantitativas, y por otro, las cualitativas.

- **Variables cuantitativas**: *budget, revenue, runtime, popularity, productoras, reparto, produccion*

- Comprobación de normalidad (sólo variables continuas: *budget, revenue, popularity*)
- Análisis visual: histogramas, diagramas de caja, diagramas de dispersión con respecto a **revenue**
- Análisis de correlación con **revenue**
- Modelo de regresión lineal múltiple (junto con variables cualitativas)

- **Variables cualitativas**: *collection, english_speaking, genres(20) mes, year*
  - Comprobación de homogeneidad de varianzas
  - Análisis visual: diagramas de caja
  - Contraste de hipótesis para la media: Student T-test, ANOVA
  - Modelo de regresión lineal múltiple (junto con variables cuantitativas)

4.2. Comprobación de la normalidad y homogeneidad de la varianza.

Interesa comprobar la normalidad de la distribución de las distintas variables y la homogeneidad de sus varianzas para decidir qué tipo de tests utilizamos cuando hagamos contrastes de hipótesis sobre la media. Además, en este proyecto llevaremos a cabo modelos de regresión lineal, con lo cual será importante para tener un buen ajuste de los modelos que los residuos que obtengamos tengan una distribución normal y homogeneidad de varianza (homoscedasticidad).

Para el caso de los contrastes de hipótesis sabemos por el Teorema del Límite Central que para muestras de >30 casos la distribución de las medias muestrales es normal, con lo cual no tendremos la necesidad de comprobar las distribuciones de normalidad de las distintas variables poblacionales en este caso.

Para el caso de los modelos de regresión lineal, si bien es cierto que se asume que con muestras grandes (>200) no sería necesario hacer transformación de los datos (https://www.statisticssolutions.com/normality/) decidimos llevarla a cabo sólo para las variables cuantitativas continuas **budget**, **popularity** y **revenue**, ésta última por ser la variable dependiente y las otras dos por ser las que presentan mayor correlación con aquella. De tal forma que sólo en el caso de encontrarnos con un patrón no aleatorio de los residuos o de encontrarnos con un problema de heteroscedasticidad usaremos dichas variables normalizadas en nuestro modelo.

En relación a la homogeneidad de varianzas, interesará analizarla tanto en el caso de los contrastes de hipótesis de las variables cualitativas (para decidir qué test utilizar como veremos más adelante) como en el caso de los residuos generados por los modelos de regresión lineal.

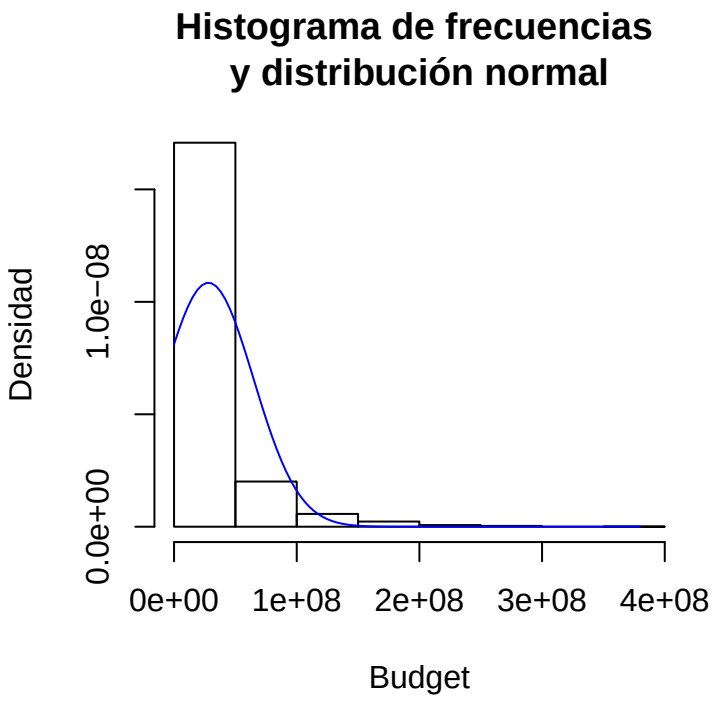
4.2.1. Comprobación de normalidad

Como ya hemos comentado, sólo comprobaremos la distribución de las variables **budget**, **popularity** y **revenue**.

Visualizamos la distribución y usamos el test de *Shapiro-Wilk*, para cada una de las variables según las siguientes hipótesis H_0 y H_1.

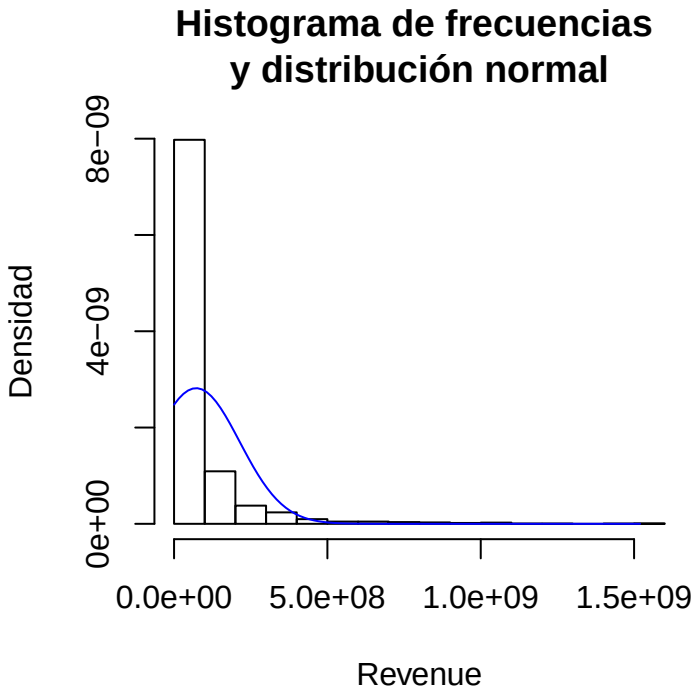
1. Hipótesis nula : H_0: La distribución de la variable es normal.
2. Hipótesis alternativa: H_1: La distribución de la variable no es normal

Variable ***budget***



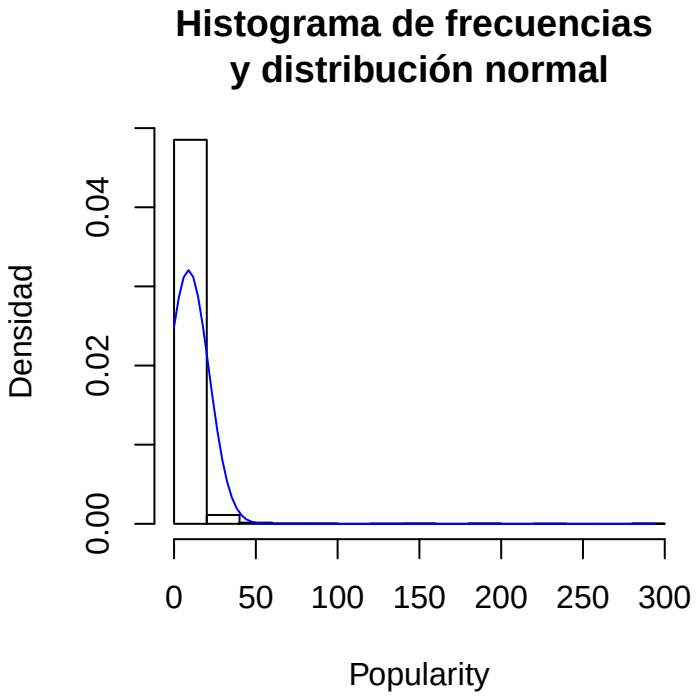

```
##
##  Shapiro-Wilk normality test
##
## data:  clean_train$budget
## W = 0.67697, p-value < 2.2e-16
```

Variable ***revenue***




```
##
##  Shapiro-Wilk normality test
##
## data:  clean_train$revenue
## W = 0.51867, p-value < 2.2e-16
```

Variable ***popularity***




```
##
##  Shapiro-Wilk normality test
##
## data:  clean_train$popularity
## W = 0.3253, p-value < 2.2e-16
```

Comprobamos que ninguna de las variables presenta una distribución normal por lo que procederemos a crear las variables equivalentes transformadas correspondientes: **budget_boxcox**, **revenue_boxcox** y **popularity:boxcox** usando `BoxCox()` y `BoxCoxLambda()` del paquete DescTools.

Variable ***budget***

X.U.FEFF.id	title	budget	budget_boxcox
1	Hot Tub Time Machine 2	1.4e+07	129.33699
2	The Princess Diaries 2: Royal Engagement	4.0e+07	160.72270
3	Whiplash	3.3e+06	95.61733
4	Kahaani	1.2e+06	77.18759
5	Marine Boy	8.0e+07	185.36539
6	Pinocchio and the Emperor of the Night	8.0e+06	115.11244

Variable ***revenue***

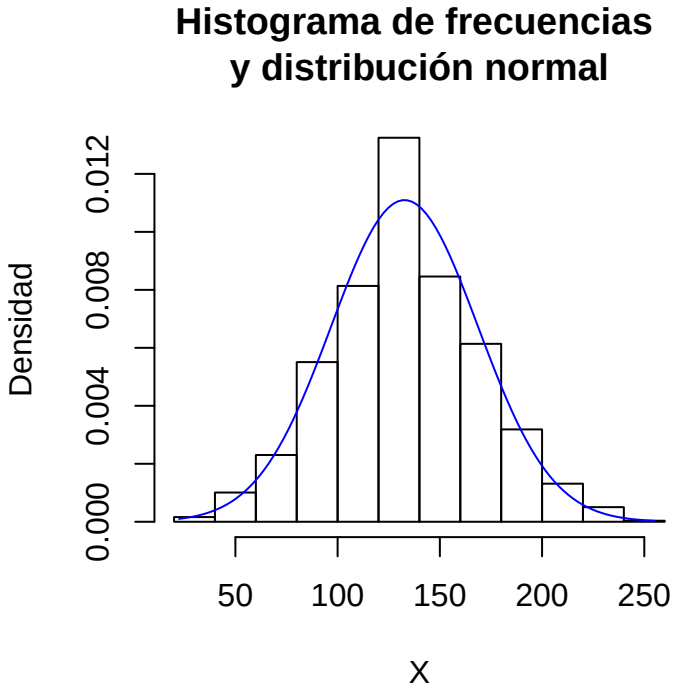
X.U.FEFF.id	title	revenue	revenue_boxcox
1	Hot Tub Time Machine 2	12314651	41.17316
2	The Princess Diaries 2: Royal Engagement	95149435	52.78279
3	Whiplash	13092000	41.48736
4	Kahaani	16000000	42.53056
5	Marine Boy	3923970	35.64283
6	Pinocchio and the Emperor of the Night	3261638	34.80676

Variable ***popularity***

X.U.FEFF.id	title	popularity	popularity_boxcox
1	Hot Tub Time Machine 2	6.575393	2.5314944
2	The Princess Diaries 2: Royal Engagement	8.248895	2.9443221
3	Whiplash	64.299990	8.2903031
4	Kahaani	3.174936	1.3807715
5	Marine Boy	1.148070	0.1409822
6	Pinocchio and the Emperor of the Night	0.743274	-0.2838699

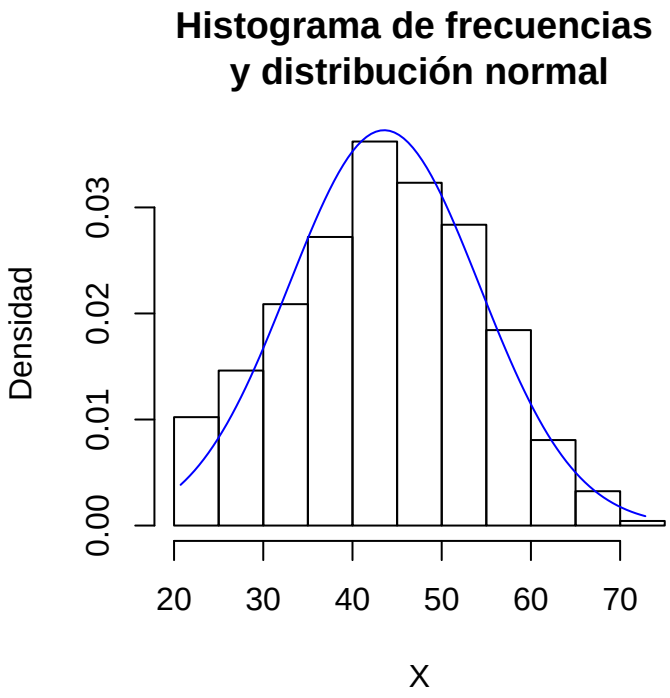
Volvemos a comprobar las distribuciones y el p-valor dado por el test de *Shapiro-Wilk*.

Variable ***budget***



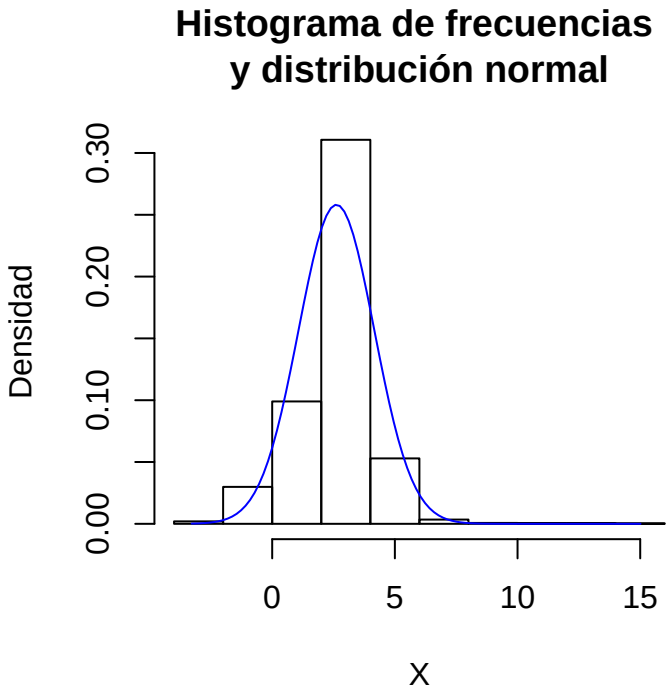

```
##
##  Shapiro-Wilk normality test
##
## data:  clean_train$budget_boxcox
## W = 0.99809, p-value = 0.002041
```

Variable ***revenue***




```
##
##  Shapiro-Wilk normality test
##
## data:  clean_train$revenue_boxcox
## W = 0.99111, p-value = 4.287e-12
```

Variable ***popularity***




```
##
##  Shapiro-Wilk normality test
##
## data:  clean_train$popularity_boxcox
## W = 0.92173, p-value < 2.2e-16
```

Comparamos los p-valores de las variables antes y después de la transformación.

var	p.value.BEFORE_transf	p.value.AFTER_transf
budget	1.85942278168567e-58	0.00204093734448493
revenue	7.03264948613264e-66	4.28726970555669e-12
popularity	1.60561526210663e-72	9.16467829755928e-36

Los p-valores no llegan a ser menores que 0.05 pero han aumentado de forma considerable su valor si comparamos con los valores obtenidos antes de la transformación delas variables. Si bien no hemos conseguido la normalidad, nos hemos aproximado a ella a juzgar por los histogramas y el cambio en los p-valores.

4.2.2. Homogeneidad de varianzas

Nos convendrá saber si existe homoscedasticidad tanto en el caso que hemos comentado anteriormente, cuando estudiemos los residuos de los modelos de regresión lineal, como en el caso de hacer contrastes de hipótesis sobre las media de las distintas variables categóricas, análisis que también haremos más adelante utilizando ANOVA o T de Student, según el caso.

Por lo tanto, estudiaremos si existe homoscedasticidad utilizando el **Test de Levene**, apropiado para variables no normales y que plantea las siguientes hipótesis nula y alternativa:

1. Hipótesis nula:

H_o: Las varianzas de los distintos grupos son iguales: $var_1 = var_2 = ...var_n$ siendo 'n' el número de niveles de la variable.

2. Hipótesis alternativa:

H_1 :: No todas las varianzas son iguales: $var_i \neq var_j$ para algún i, j

variable ***collection***

```
## Levene's Test for Homogeneity of Variance (center = median)
##         Df F value    Pr(>F)
## group    1  295.65 < 2.2e-16 ***
##       2776
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Observamos que el p-valor ($4.7958935 \times 10^{-63}$) es menor que un nivel de significación del 0.05 e incluso del 0.01, con lo cual rechazaríamos la hipótesis nula de que las varianzas serían todas iguales.

variable ***english_speaking***

```
## Levene's Test for Homogeneity of Variance (center = median)
##         Df F value   Pr(>F)
## group    1  49.782 2.16e-12 ***
##       2776
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Observamos que el p-valor ($2.1604657 \times 10^{-12}$) es menor que un nivel de significación del 0.05 e incluso del 0.01, con lo cual rechazaríamos la hipótesis nula de que las varianzas serían todas iguales.

variable ***mes***

```
## Levene's Test for Homogeneity of Variance (center = median)
##         Df F value    Pr(>F)
## group   11  10.738 < 2.2e-16 ***
##       2766
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Observamos que el p-valor ($1.2317833 \times 10^{-19}$) es menor que un nivel de significación del 0.05 e incluso del 0.01, con lo cual rechazaríamos la hipótesis nula de que las varianzas serían todas iguales.

Variables: "Action", "Adventure", "Animation", "Comedy", "Crime", "Documentary", "Drama", "Family", "Fantasy", "Foreign", "History", "Horror", "Music", "Mystery", "Romance", "Science_Fiction", "Thriller", "Tv_movie", "War", "Western"

vars	pvalue	signif	Freq
Action	4.51487116131127e-19	**	741
Adventure	1.05285071446517e-63	**	439
Animation	2.79597588996473e-11	**	140
Comedy	0.0247178430584744	*	1027
Crime	0.0494621285138095	*	469
Documentary	0.00089489739161472	**	86
Drama	8.72850404352556e-16	**	1530
Family	4.4499590005443e-13	**	259
Fantasy	1.97935429078281e-16	**	231
Foreign	0.0484963138241036	*	31
History	0.00904108713246332	**	132
Horror	0.0334616490369043	*	301
Music	0.0769564129745728		100
Mystery	0.193654989981778		225
Romance	0.00755945137029328	**	570
Science_Fiction	4.58467938099327e-11	**	290
Thriller	0.757032342069677		788
Tv_movie	NA	NA	1
War	0.159988292038516		100
Western	0.572924322222386		43

Note:
** pvalor<0.01, * pvalor<0.05

Vemos que el p-valor es mayor que el nivel de significación del 0.5 sólo en 5 variables (**Music**, **Mystery**, **Thriller**, **War** y **Western**), con lo cual sólo existe homogeneidad de varianzas en dichos casos.

`Nota: en el caso de 'Tv_movie' al haber sólo 1 caso con este tipo de género no se ha podido hacer el contraste`.

4.2.3. Tratamiento de valores extremos II (*outliers*)

Se consideran valores extremos todos aquellos por encima (y por debajo) de 3 desviaciones estándar de la media (mean ± 3*SD).Pueden ser causados por errores (en la entrada de datos, en las mediciones), por falsas asunciones debidas a distribuciones no normales, o puede que sean incluso valores correctos.

Identificamos si existen valores extremos en las variables **budget_boxcox popularity_boxcox** y **revenue_boxcox**.

Variable ***budget_boxcox***

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   22.46  108.40  131.20  132.73  156.36  254.97
```

```
## [1] "El número de valores extremos por encima es de 2"
```

X.U.FEFF.id	title	budget	budget_boxcox
2136	Pirates of the Caribbean: On Stranger Tides	3.8e+08	254.9721
2210	Pirates of the Caribbean: At World's End	3.0e+08	242.9672

```
## [1] "El número de valores extremos por debajo es de 2"
```

X.U.FEFF.id	title	budget	budget_boxcox
366	Primer	7000	24.37579
2611	The Tiger: An Old Hunter's Tale	5000	22.46401

Variable ***revenue_boxcox***

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   20.73   35.87   43.72   43.55   51.48   72.83
```

```
## [1] "El número de valores extremos por encima es de 0"
```

```
## [1] "El número de valores extremos por debajo es de 0"
```

Variable ***popularity***

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -3.281   1.929   2.806   2.618   3.533  15.012
```

```
## [1] "El número de valores extremos por encima es de 17"
```

X.U.FEFF.id	title	popularity	popularity_boxcox
3	Whiplash	64.29999	8.290303
519	Wonder Woman	294.33704	15.012279
685	Beauty and the Beast	287.25365	14.878699
935	John Wick: Chapter 2	49.24750	7.396526
1127	The Avengers	89.88765	9.519200
1310	Gone Girl	154.80101	11.795690
1674	The Dark Knight	123.16726	10.792931
1696	Baby Driver	228.03274	13.659974
1716	War for the Planet of the Apes	146.16179	11.537281
1784	Logan	54.58200	7.732741
2017	The Shawshank Redemption	51.64540	7.550660
2019	Guardians of the Galaxy	53.29160	7.653597
2098	Pirates of the Caribbean: The Curse of the Black Pearl	47.32666	7.269221
2127	Guardians of the Galaxy Vol. 2	185.33099	12.635131
2294	The Circle	88.43924	9.456717
2327	Deadpool	187.86049	12.700205
2339	Fight Club	63.86960	8.266907

```
## [1] "El número de valores extremos por debajo es de 10"
```

X.U.FEFF.id	title	popularity	popularity_boxcox
458	Swoon	0.003013	-2.749111
742	And You Thought Your Parents Were Weird!	0.000578	-2.977327
1054	FBI: Frikis buscan incordiar	0.044048	-2.026992
1504	Campus Man	0.000844	-2.934509
1684	The Slugger's Wife	0.000308	-3.038590
1758	Journey from the Fall	0.044561	-2.022446
1926	Monster in a Box	0.009961	-2.497019
2011	Cheetah	0.011574	-2.458503
2427	Overdose	0.021580	-2.278718
2557	Big Time	0.000001	-3.280504

Hemos detectado 4 valores extremos en el caso de **budget** (2 por encima y 2 por debajo) y 27 en el caso de **popularity** (17 encima y 10 por debajo). Lo que haremos será probar a excluir los casos que los contienen cuando lleguemos a la parte de generar los modelos de regresión y comparar la calidad del modelo.

4.4. Análisis visual y estadístico de los datos.

En función de los datos y el objetivo del estudio, aplicar pruebas de contraste de hipótesis, correlaciones, regresiones, etc. Aplicar al menos tres métodos de análisis diferentes.

4.4.1. Análisis de variables cuantitativas

- **Análisis visual**: histogramas, diagramas de caja, diagramas de dispersión con respecto a **revenue**

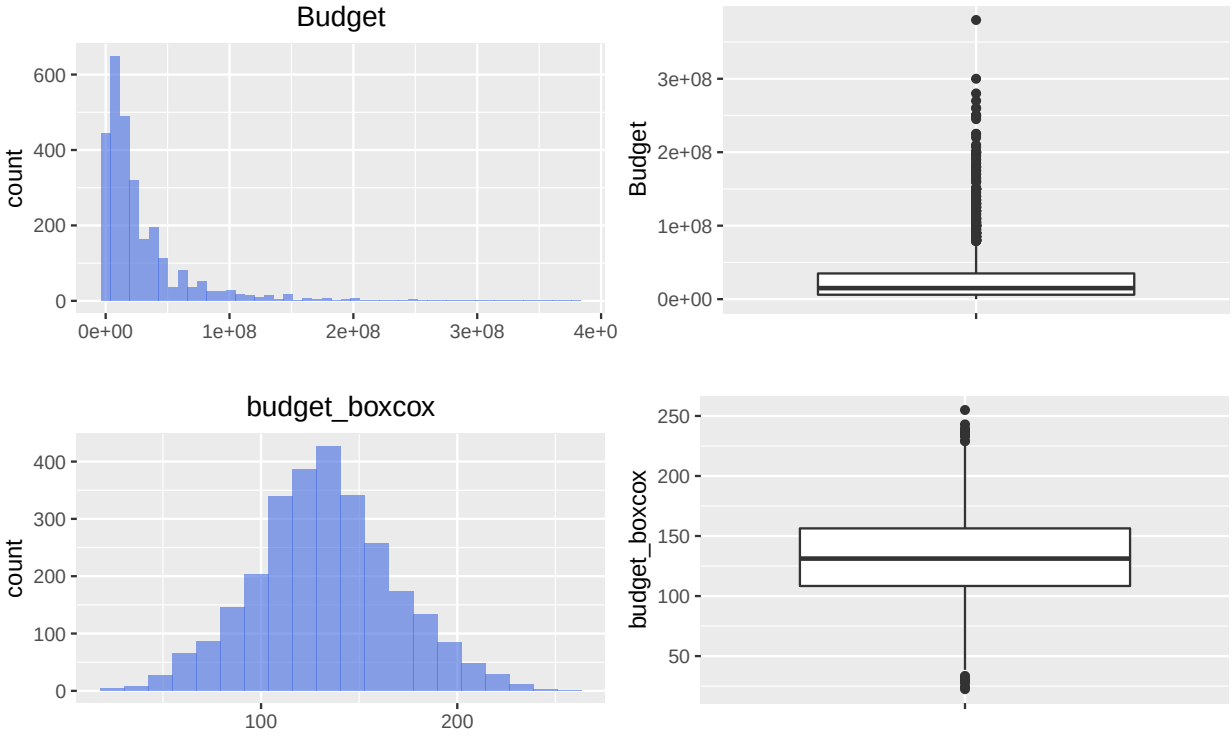
  Variables ***budget*** y ***budget_boxcox***

```
summary(clean_train$budget)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
##      5000   6000000  15000000  28018053  35000000 380000000
```

```
summary(clean_train$budget_boxcox)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   22.46  108.40  131.20  132.73  156.36  254.97
```



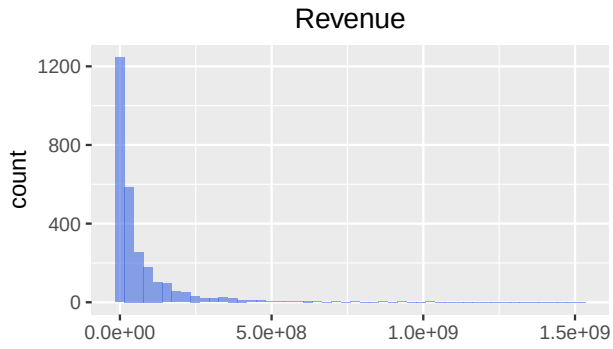
Variables ***revenue*** y ***revenue_boxcox***

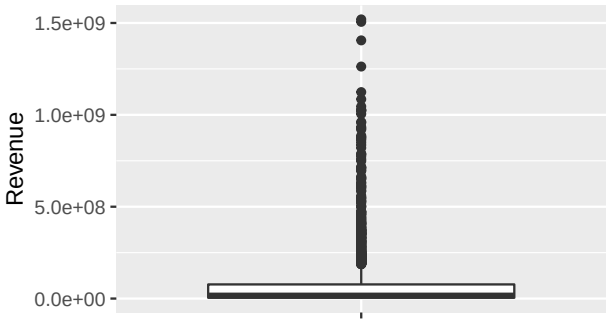
```
summary(clean_train$revenue)
```

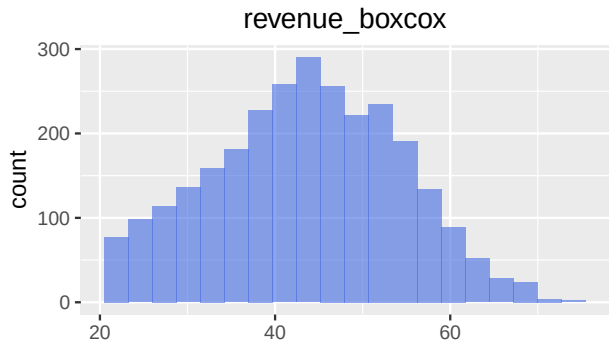
```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## 7.514e+04 4.124e+06 2.002e+07 7.205e+07 7.710e+07 1.520e+09
```

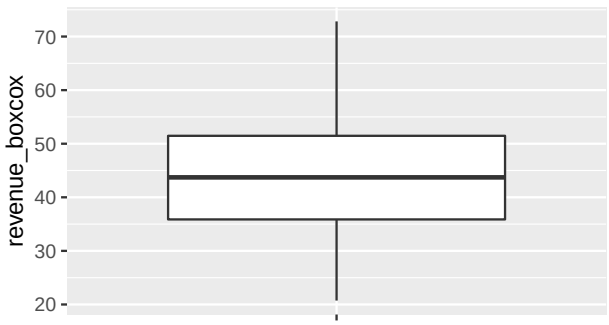
```
summary(clean_train$revenue_boxcox)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   20.73   35.87   43.72   43.55   51.48   72.83
```



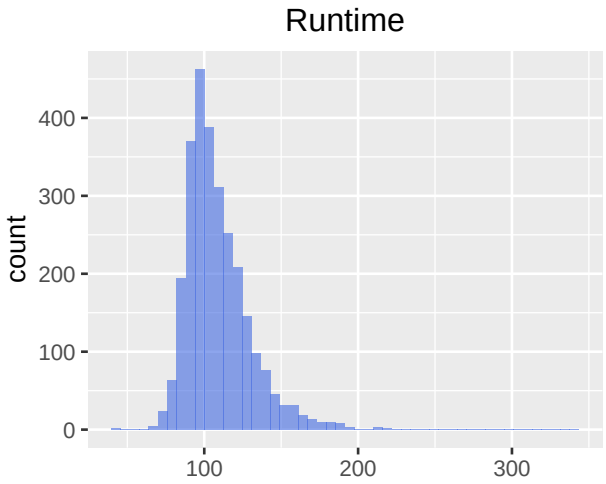


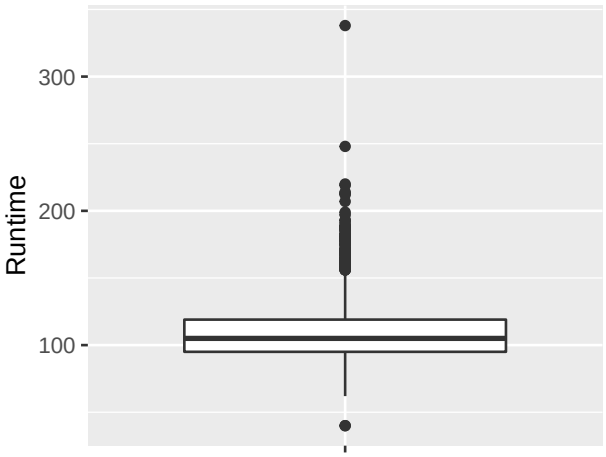




Variable ***runtime***

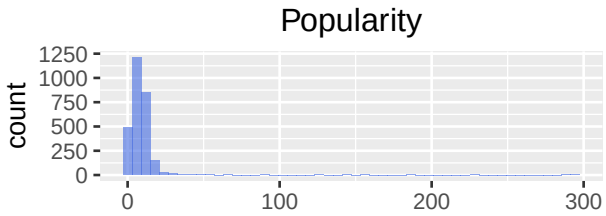
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      40      95     105     109     119     338
```

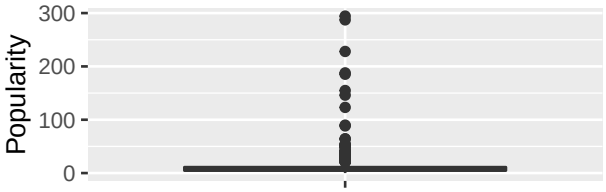


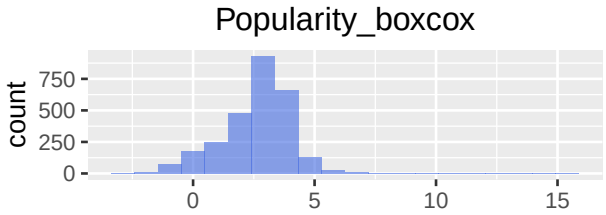


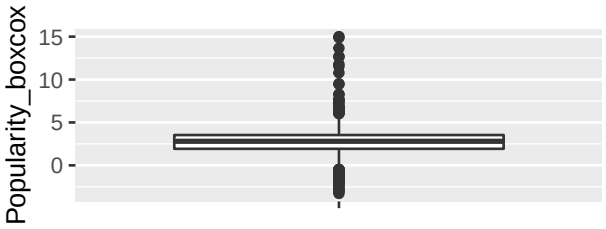
Variable ***popularity***

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   4.582   7.657   8.843  11.120 294.337
```



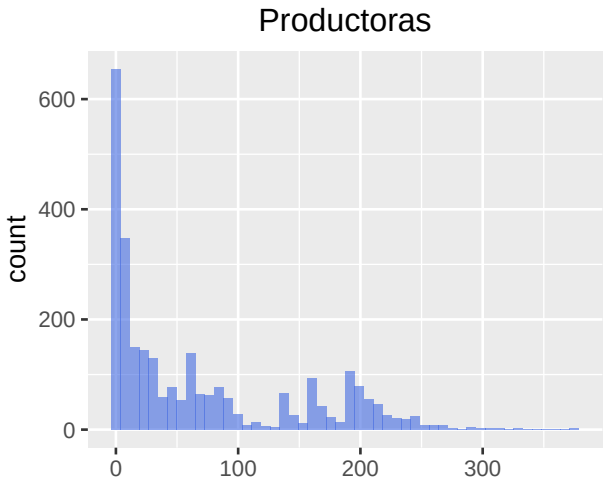


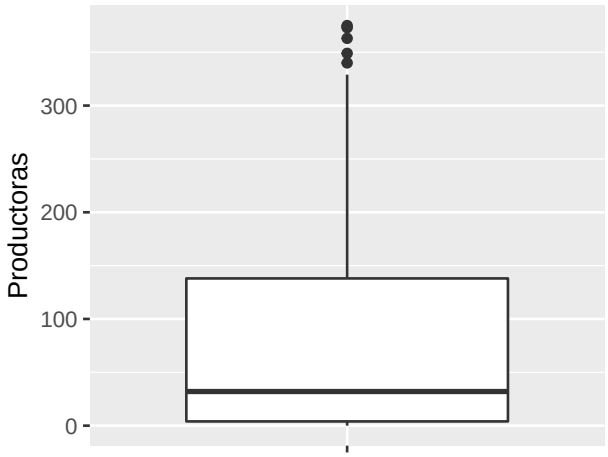




Variable ***productoras***

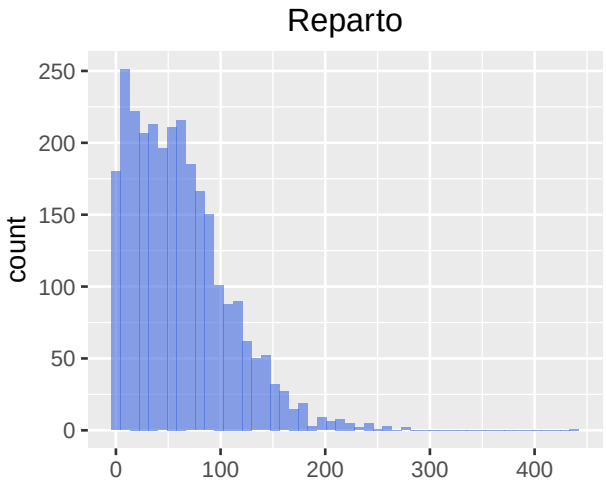
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    4.00   32.00   69.25  138.00  375.00
```

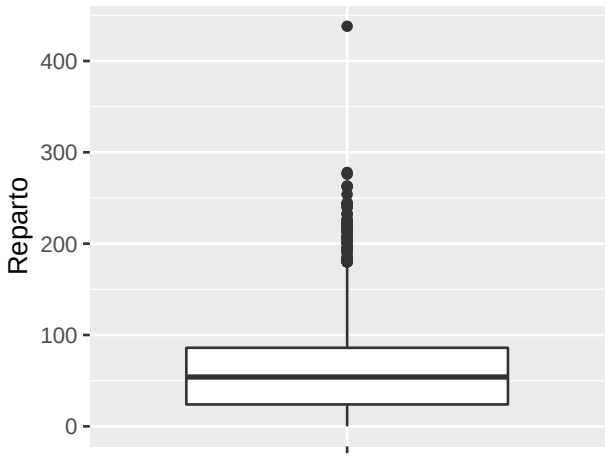




Variable ***reparto***

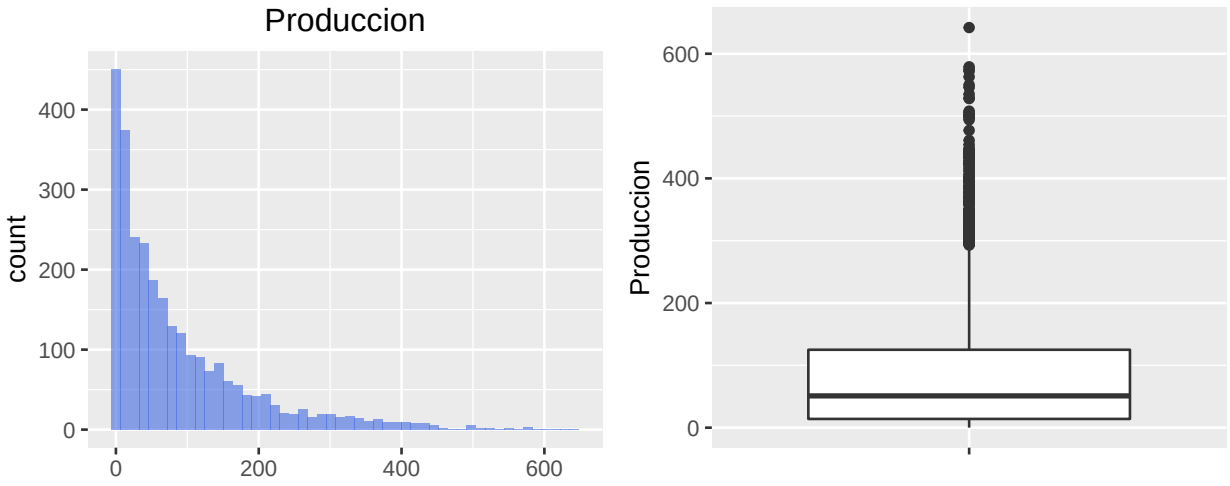
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   24.00   54.00   61.26   86.00  438.00
```





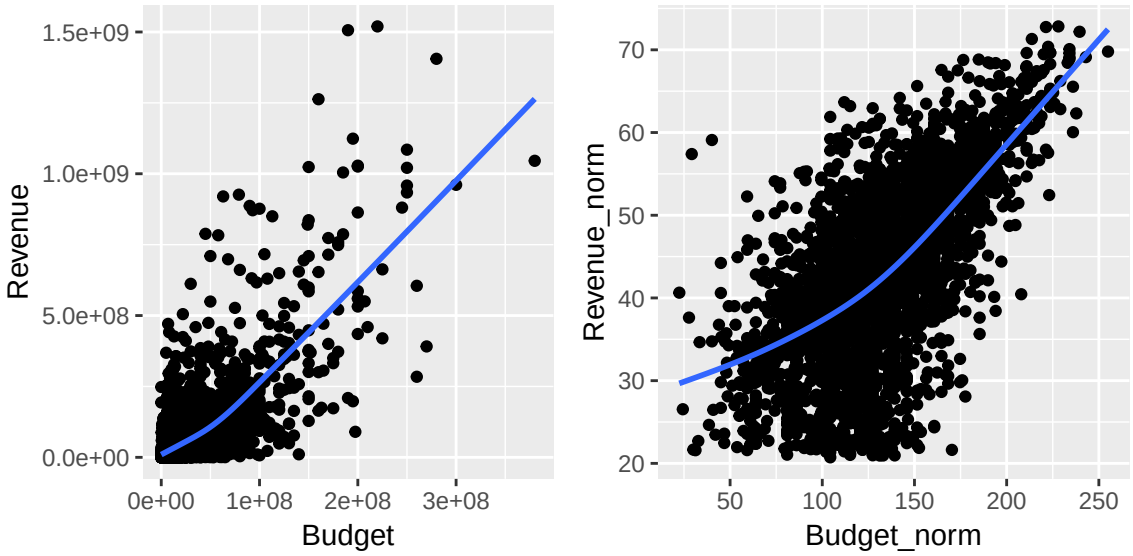
Variable ***produccion***

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   14.00   51.00   87.76  125.00  642.00
```

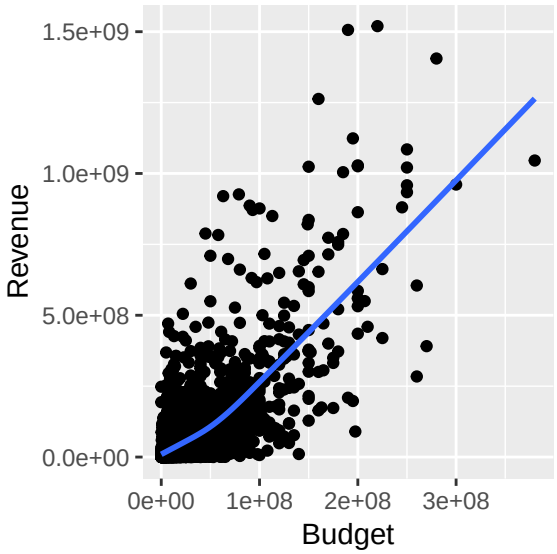


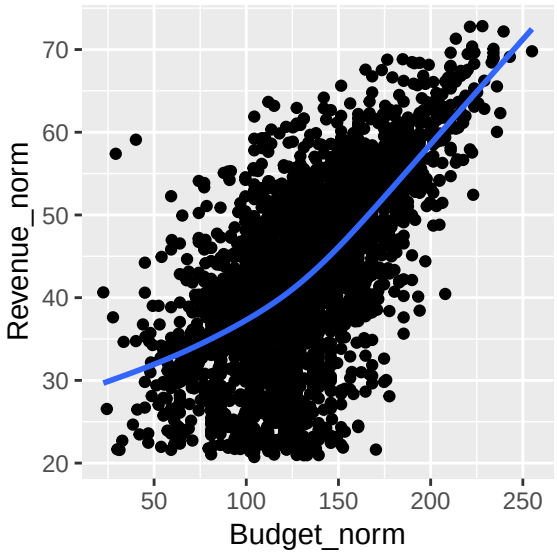
Analizamos el grado de correlación de las variables cuantitativas con la variable **revenue**. Para ello realizamos diagramas de dispersión (*scatterplot*) y test de correlación.

Variable ***budget***

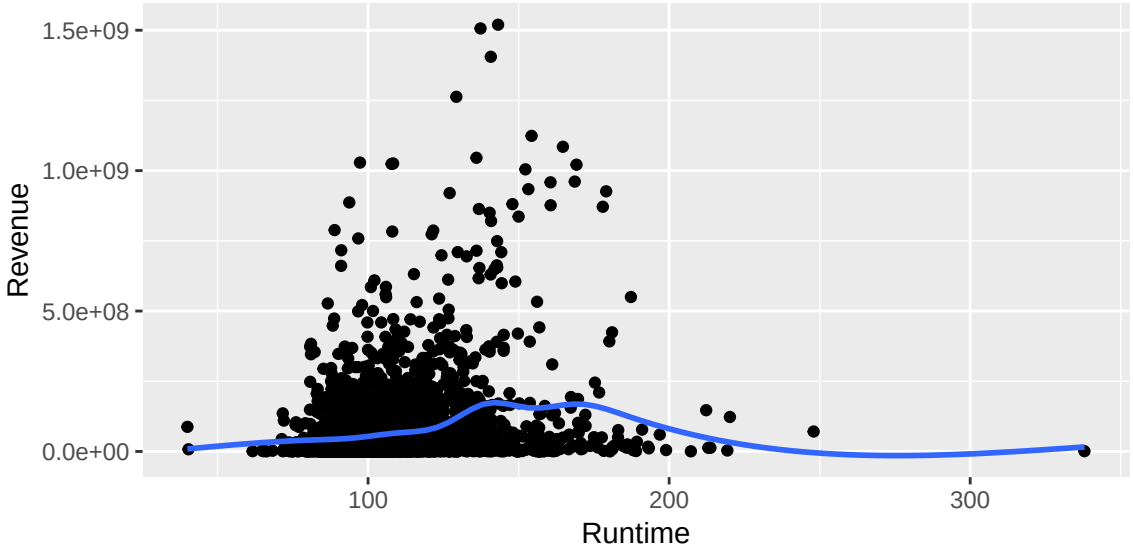


Variable ***popularity***

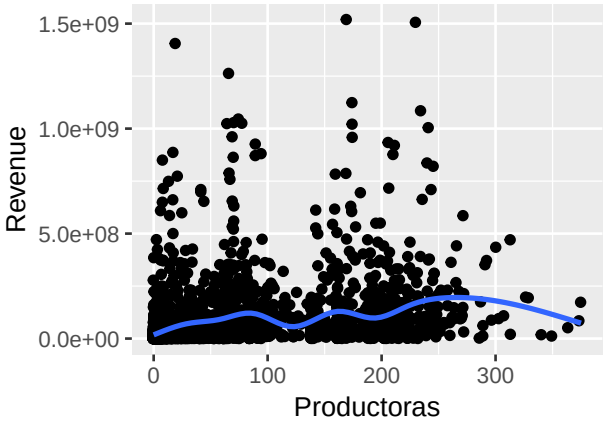


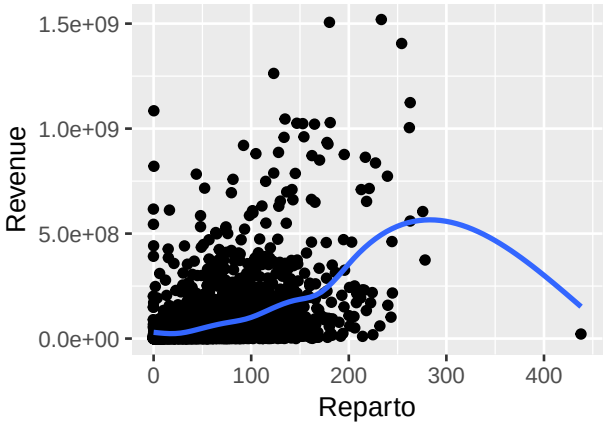


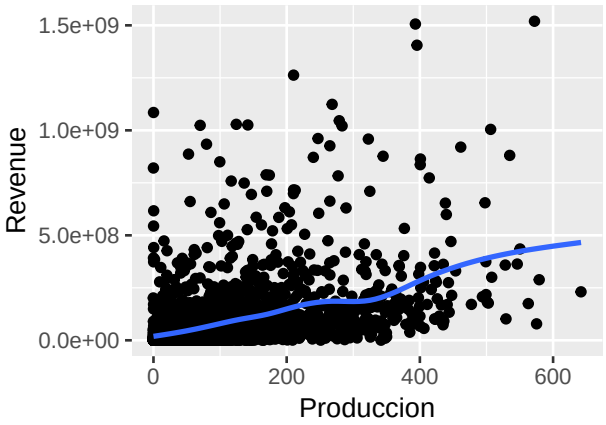
Variable ***runtime***



Variables ***productoras***, ***reparto*** y ***produccion***







- **Análisis de correlación con *revenue***

Realizamos el test de correlación de cada una de las variables cuantitativas con **revenue**. Como hemos visto que ninguna de las variables presentaban una distribución normal aun después de la normalización, decidimos utilizar un test no paramétrico como es **Kendall's test**.

Variable	Tau	pvalor
budget	0.4568930	3.3737030463443e-281
runtime	0.1590958	1.1867693045038e-35
popularity	0.4073581	2.94560894404187e-227
productoras	0.3687573	5.30408054566694e-181
reparto	0.3550457	8.68044387695629e-172
produccion	0.4096003	9.76049831850113e-228

Realizamos la matriz de correlación para las variables cuantitativas en su conjunto.

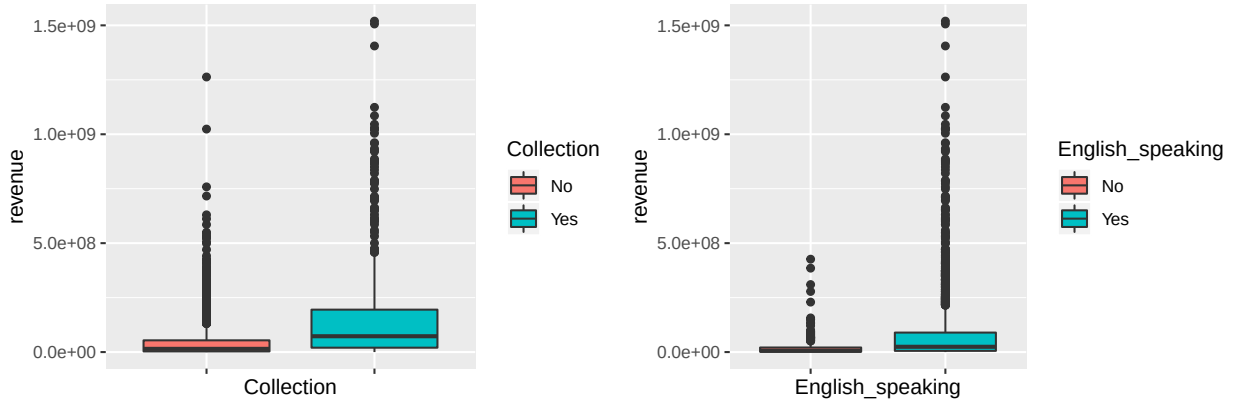
Podemos observar que todas las variables presentan correlación con **revenue** siendo más significativo el caso de **budget** (Tau= 0.46) seguido de **produccion** (Tau= 0.41) y **popularity** (Tau= 0.41).

4.4.2. Análisis de variables categóricas

- **Análisis visual**: diagramas de caja

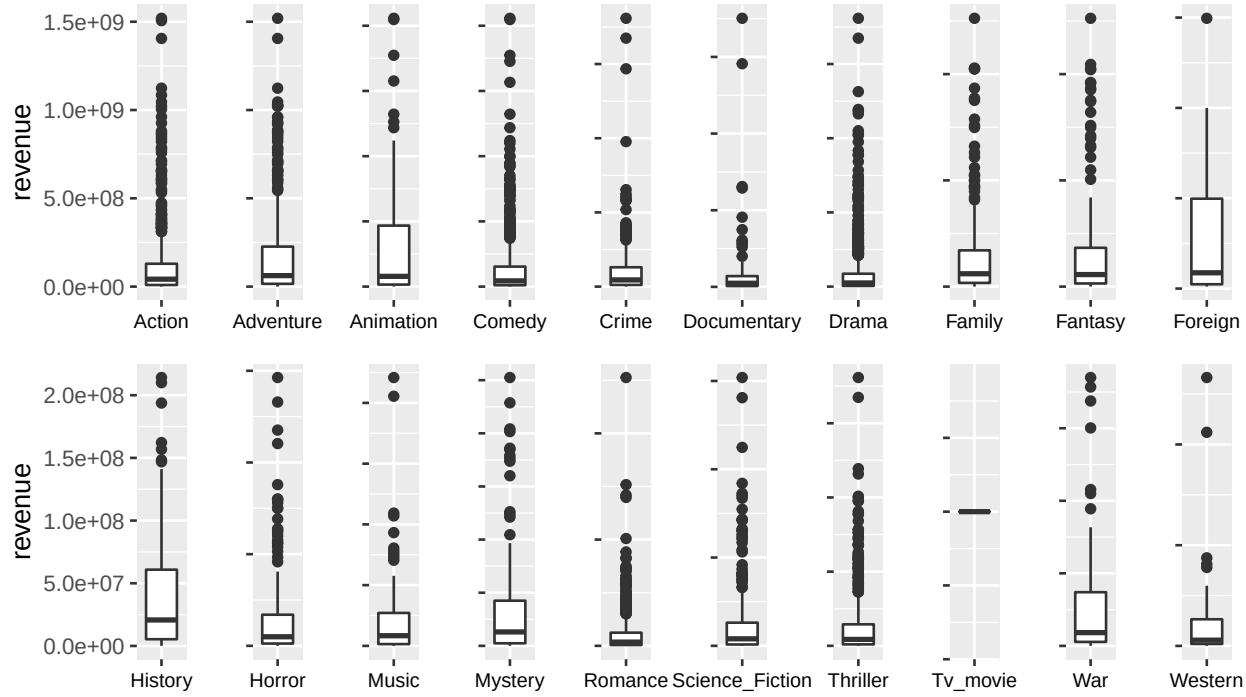
Para el análisis de las variables categóricas utilizaremos diagramas de caja comparando el valor de **revenue** atendiendo a los distintos niveles de cada variable.

Variables ***collection*** y ***english_speaking***



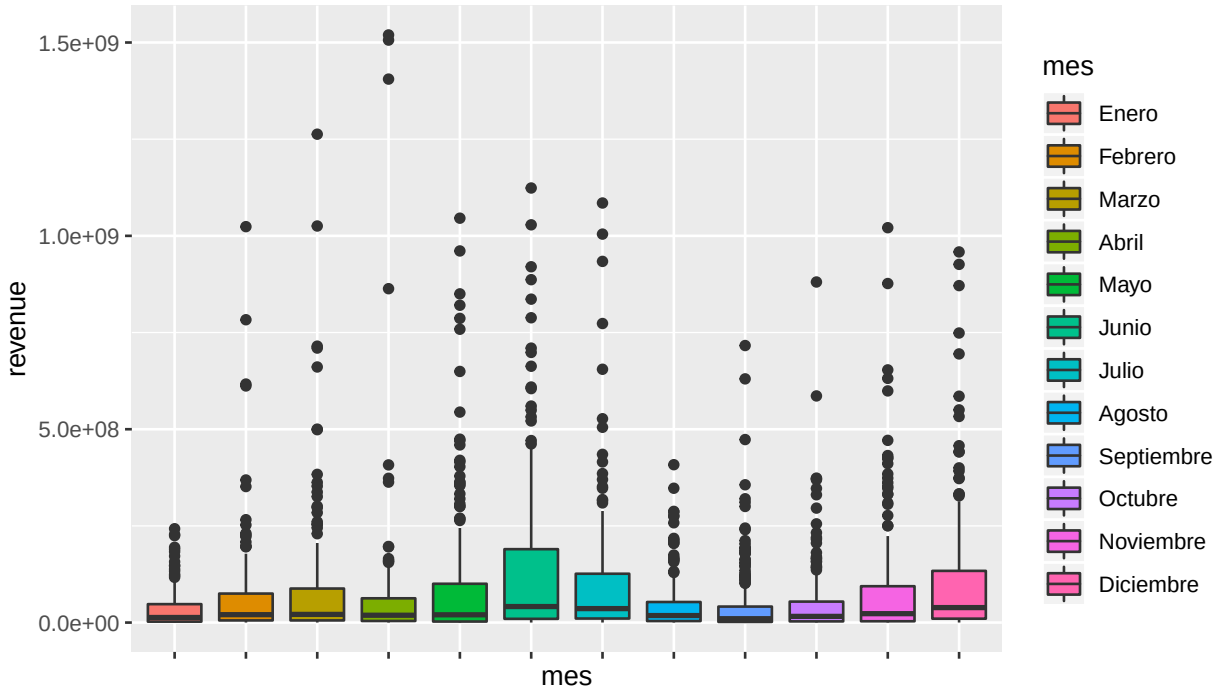
Podemos apreciar que existen diferencias de **revenue** en cuanto a **collection** mientras que no se aprecian muchas diferencias en cuanto a **english_speaking**. Se explorarán mejor estas diferencias en el modelo de regresión lineal.

Variable ***genres***



Podemos apreciar que existen diferencias de **revenue** en cuanto a género se refiere. Se explorarán mejor estas diferencias en el modelo de regresión lineal.

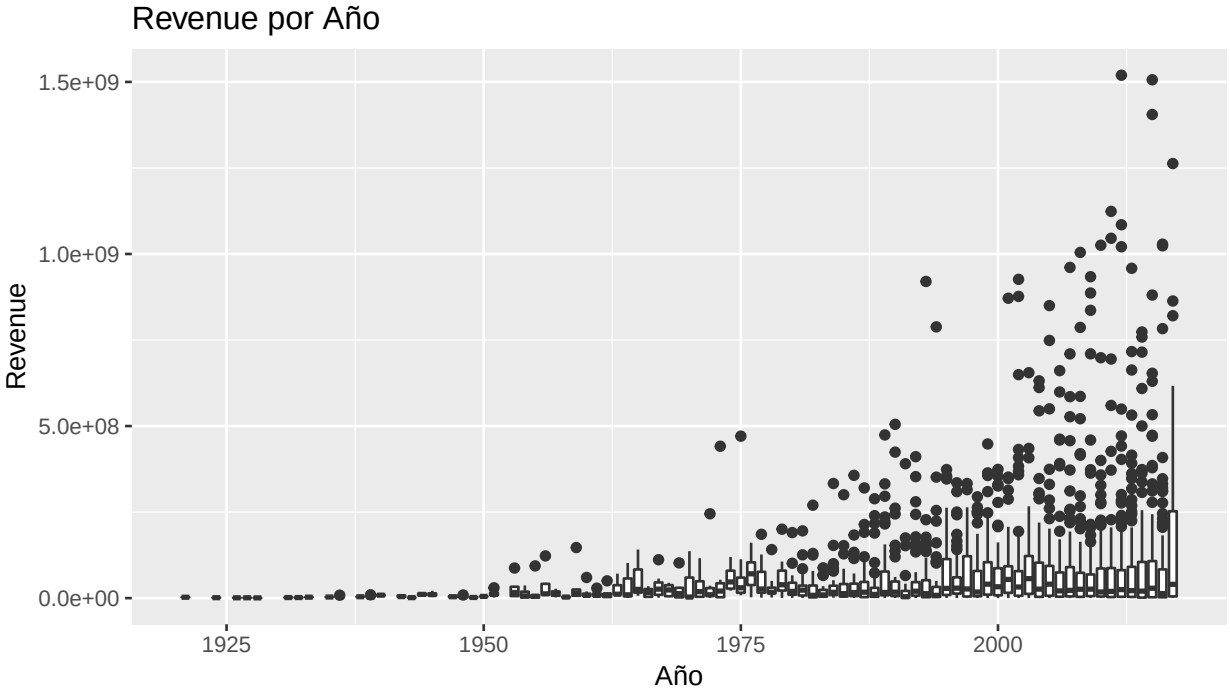
Variable ***Mes***



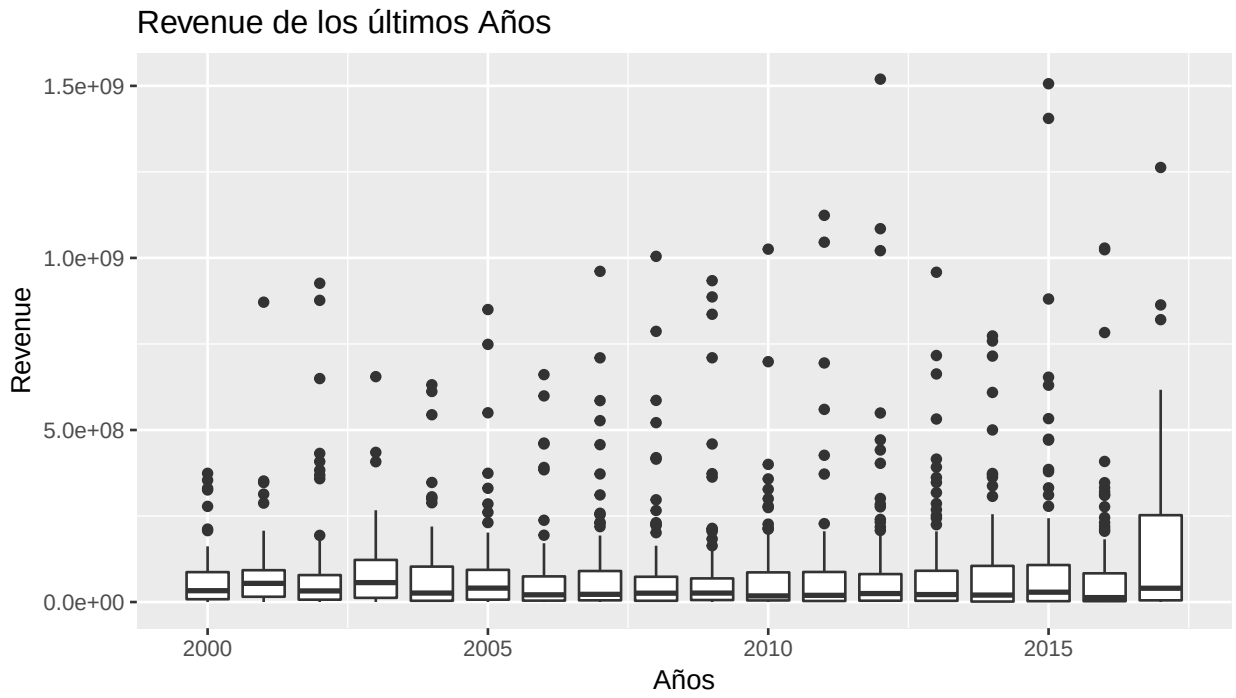
Podemos apreciar que existen ciertas diferencias de **revenue** en cuanto al mes de estreno. En junio, julio y en diciembre parece que sube la recaudación con respecto al resto de meses. Se explorarán mejor estas diferencias en el modelo de regresión lineal.

Variable ***Year***

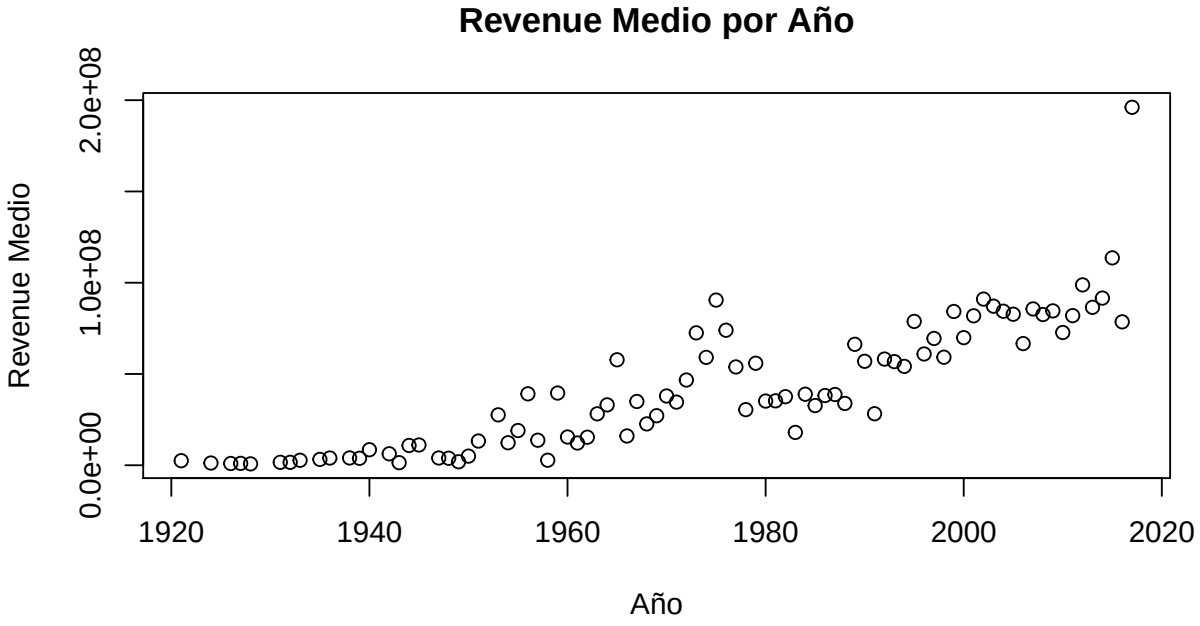
Hacemos un boxplot para el conjunto de todos los años.



No se visualiza correctamente dada la gran cantidad de años existentes así que filtramos los años a partir del 2000 para ver si podemos sacar conclusiones fijándonos en las dos últimas décadas.



Por último y para contrastar el gráfico anterior sacamos las medias de *revenue** por año para ver la evolución de la recaudación de las películas a lo largo del tiempo.



Podemos apreciar que existe una tendencia a aumentar la recaudación a medida que pasan los años. Quizás esto no se deba a un aumento de los espectadores y sólo sea debido al aumento del precio de la entrada. Se explorarán mejor estas diferencias en el modelo de regresión lineal.

- **Contrastes de hipótesis**: contraste de hipótesis para la media: Student T-test, ANOVA

¿Existen diferencias entre las medias de **revenue** atendiendo a los diferentes niveles de las variables?

Hemos visto que ninguna de las variables presenta una distribución normal, pero al contar con una muestra grande (>30), por el Teorema del Límite Central, podemos aproximar la distribución de las medias muestrales a una normal.

Por otro lado hemos comprobado que no existía homogeneidad de varianza para ninguna variable. Esto tiene como consecuencias que en el caso de las variables con 2 niveles debamos utilizar como parámetro `var.equal=FALSE` cuando hagamos el test T de Student. Para el caso de variables con más de 2 niveles, al igual que pasa con la asunción de normalidad, el no cumplimiento de la homoscedasticidad no afectaría de forma sensible el contraste del estadístico F del test paramétrico ANOVA si, por un lado, la muestra es grande (>30) y, por otro, si los grupos tienen aproximadamente el mismo tamaño.

Con lo cual, para la única variable con más de 2 niveles (**mes**) comprobamos si el tamaño de los grupos es similar.

```
##
##      Enero    Febrero      Marzo      Abril       Mayo      Junio
##        188        203        221        229        207        223
##      Julio     Agosto Septiembre    Octubre  Noviembre  Diciembre
##        200        245        327        277        205        253
```

Podemos decir que los tamaños son similares por lo que procedemos a realizar un test ANOVA.

Test ANOVA

Contraste de hipótesis:

1. Hipótesis nula:

H_0: Las medias de los distintos grupos son iguales: $\mu_1 = \mu_2 = ...\mu_n$ para 'n' niveles de la variable.

2. Hipótesis alternativa:

H_1 :: No todas las medias son iguales: $\mu_i \neq \mu_j$ para algún i, j

Variable ***mes***

```
##               Df    Sum Sq   Mean Sq F value Pr(>F)
## mes           11 2.441e+18 2.219e+17   11.53 <2e-16 ***
## Residuals   2766 5.322e+19 1.924e+16
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Observamos que el valor de F es superior a 1. Además, como vemos que la probabilidad asociada al estadístico ($2.5464405 \times 10^{-21}$) es menor que un nivel de significación del 0.05 podemos concluir que hay diferencias entre las medias de los distintos grupos, con lo cual podemos decir que el hecho de que se estrene la película en un determinado mes (**mes**) sí influye en la recaudación (**revenue**).

Test T de Student

Para el resto de variables utilizaremos el `t.test()`.

Variable ***collection***

El contraste de hipótesis para cada una de las variables sería el siguiente:

1. Hipótesis nula:

$H_0 : \mu_1 - \mu_2 = 0$

2. Hipótesis alternativa:

$H_1 : \mu_1 - \mu_2 \neq 0$

Aplicaríamos un contraste de hipótesis de tipo bilateral, es decir que la hipótesis alternativa engloba tanto el caso $\mu_1 > \mu_2$ como $\mu_1 < \mu_2$, contaríamos con $n_1 + n_2 - 2$ grados de libertad.

Utilizaremos un nivel de significación α de 0.05, que sería el error máximo de tipo I (aceptar la hipótesis nula siendo ésta falsa) que estaríamos dispuestos as asumir

```
##
##  Welch Two Sample t-test
##
## data:  clean_train$revenue by clean_train$collection
## t = -11.741, df = 626.6, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -136894287  -97662831
## sample estimates:
##  mean in group No mean in group Yes
##          47312337         164590896
```

Observamos que el p-valor ($6.5482981 \times 10^{-29}$) es menor que un nivel de significación del 0.05 podemos concluir que hay diferencias entre las medias de los distintos grupos, con lo cual podemos decir que el hecho de que la película pertenezca o no a una saga (**collection**) sí influye en la recaudación (**revenue**).

Variable ***english_speaking***

```
##
##  Welch Two Sample t-test
##
## data:  clean_train$revenue by clean_train$english_speaking
## t = -15.121, df = 1830.1, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -65830157 -50714292
## sample estimates:
##  mean in group No mean in group Yes
##          21498505          79770730
```

Observamos que el p-valor ($9.0955134 \times 10^{-49}$) es menor que un nivel de significación del 0.05 podemos concluir que hay diferencias entre las medias de los distintos grupos, con lo cual podemos decir que el hecho de que el idioma original de la la película (**english_speaking**) sea o no el inglés sí influye en la recaudación (**revenue**).

Variables: ***"Action", "Adventure", "Animation", "Comedy", "Crime", "Documentary", "Drama", "Family", "Fantasy", "Foreign", "History", "Horror", "Music", "Mystery", "Romance", "Science_Fiction", "Thriller", "Tv_movie", "War", "Western"***

Habíamos encontrado homogeneidad de varianza sólo en 5 de los 20 géneros. Estos eran: **Music**, **Mystery**, **Thriller**, **War** y **Western**. Ajustaremos el parámetro `var.equal=TRUE` de `t.test()`para estos casos.

vars	pvalue	signif	Freq
Action	1.53316011754993e-13	**	741
Adventure	1.20390363580493e-21	**	439
Animation	1.42595518291965e-05	**	140
Comedy	0.0345000591251033	*	1027
Crime	0.0741777579015846		469
Documentary	9.34932037503644e-58	**	86
Drama	2.02910892894588e-17	**	1530
Family	2.20327132137686e-08	**	259
Fantasy	1.15031448009595e-07	**	231
Foreign	1.34916672633676e-99	**	31
History	2.49796639558126e-09	**	132
Horror	0.00444005919952389	**	301
Music	0.0721193697099548		100
Mystery	0.298439075840218		225
Romance	0.00121345060438395	**	570
Science_Fiction	9.60275107220391e-06	**	290
Thriller	0.893800659514279		788
Tv_movie	NA		1
War	0.186773011783958		100
Western	0.5416960819016		43

Note:
** pvalor<0.01, * pvalor<0.05

Observamos que el p-valor es menor que un nivel de significación del 0.05 en la mayoría de los géneros excepto para 'Crime', 'Music', 'Mystery','Thriller', 'War' y 'Western'. El resto de géneros parece que sí influye en la recaudación (**revenue**).

`Nota: en el caso de 'Tv_movie' al haber sólo 1 caso con este tipo de género no se ha podido hacer el contraste`.

5. Fichero de salida

Después de haber realizado todo el análisis y transformaciones necesarias al dataset original hemos obtenido uno nuevo que será el que utilicemos para plantear y evaluar un posible modelo de regresión y que se basará en las siguientes variables:

1. **X.U.FEFF.id**
2. **title**
3. **collection**
4. **english_speaking**
5. **budget**
6. **budget_boxcox**

7. **popularity**
8. **popularity_boxcox**
9. **runtime** 10.**mes** 11.**year**
10. **productoras**
11. **reparto**
12. **produccion**
13. **revenue**
14. **revenue_boxcox**

15. **genres (las 20 variables binarias)**

Este nuevo dataset transformado se vuelca al fichero **trainOut.csv** adjunto al ejercicio.

6. Modelo de regresión lineal generalizado

El objetivo del proyecto es generar un modelo de predicción de la recaudación de las películas atendiendo a distintas variables. Para ello hemos elegido el modelo de regresión lineal para el cual utilizaremos la función `lm()`.

Seleccionaremos las siguientes variables:

- Variable dependiente: **revenue** (o **revenue_boxcox**)
- Variables independientes:
    - Cuantitativas (6): **runtime**, **budget** (o **budget_boxcox**), **popularity**, **productoras**, **reparto**, **produccion**.
    - Categóricas (24): **collection**, **english_speaking**, **mes**, **year**, **Action**, **Adventure**, **Animation**, **Comedy**, **Crime**, **Documentary**, **Drama**, **Family**, **Fantasy**, **Foreign History**, **Horror**, **Music**, **Mystery**, **Romance**, **Science_Fiction**, **Thriller**, **Tv_movie**, **War**, **Western**.

Generaremos distintos modelos y seleccionaremos aquel que se ajuste mejor a los datos. Para ello nos fijaremos en los residuos que genere el modelo (distribución, homogeneidad de la varianza), en el coeficiente de determinación (R^2), la raíz del error cuadrático medio ($RMSE$) y el p-valor global del modelo.

6.1. Modelo sin las variables normalizadas

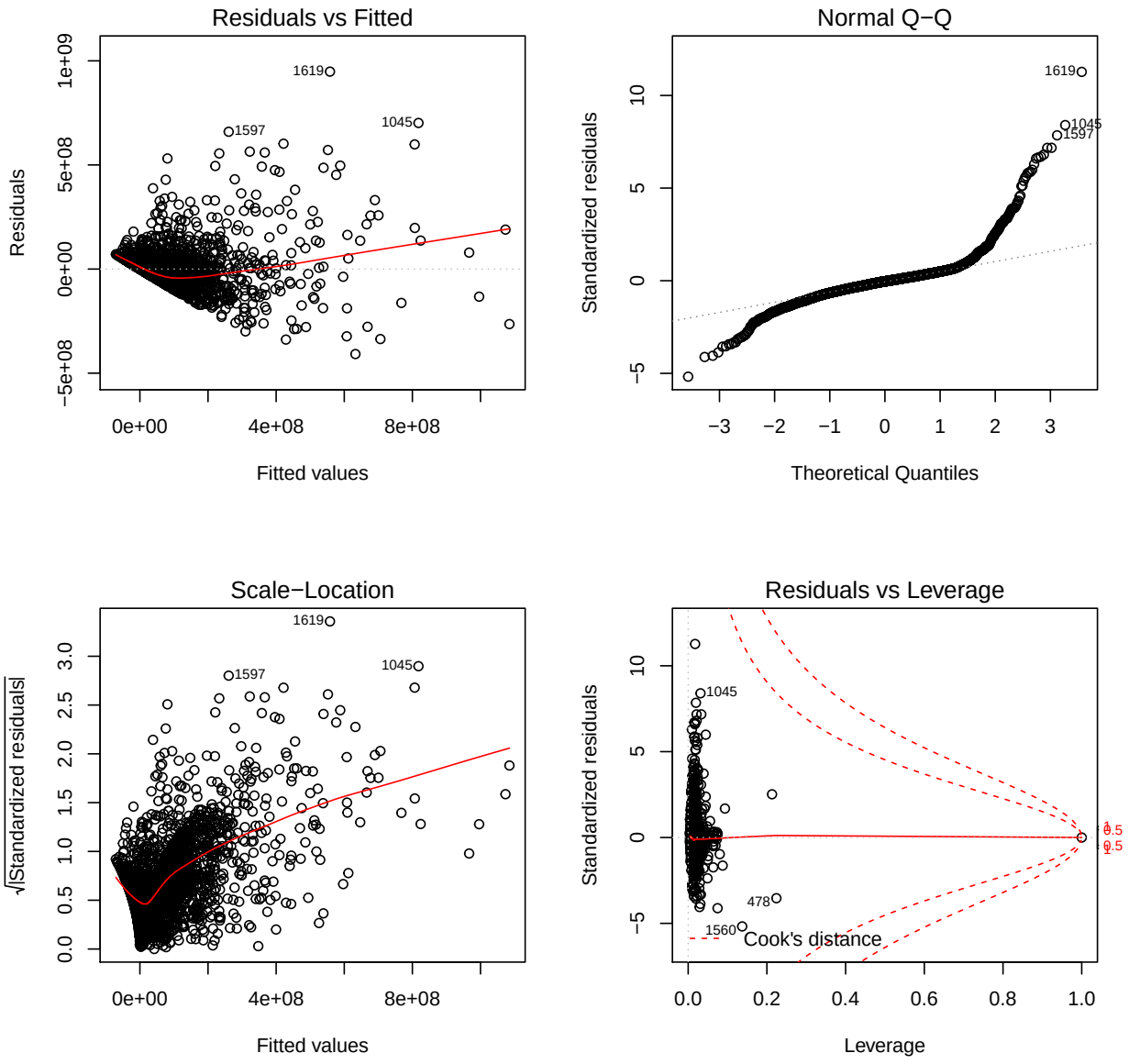
Generamos un primer modelo con las variables **budget**, **popularity** y **revenue** sin normalizar y con los valores imputados de **budget** y **runtime**

```
##
## Call:
## lm(formula = formula, data = clean_train)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -408241596  -36570553   -2921081   25784125  948203551
##
## Coefficients:
```

```
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)           1.817e+08  2.456e+08   0.740 0.459589
## budget                2.113e+00  6.510e-02  32.455  < 2e-16 ***
## popularity            2.499e+06  1.400e+05  17.844  < 2e-16 ***
## runtime               4.639e+05  9.610e+04   4.827 1.46e-06 ***
## collectionYes         5.829e+07  4.435e+06  13.143  < 2e-16 ***
## english_speakingYes -7.946e+06  5.858e+06  -1.356 0.175064
## mesFebrero            7.617e+06  8.674e+06   0.878 0.379969
## mesMarzo              5.333e+06  8.531e+06   0.625 0.531945
## mesAbril              7.578e+06  8.409e+06   0.901 0.367589
## mesMayo               1.609e+07  8.649e+06   1.860 0.062955 .
## mesJunio              2.118e+07  8.601e+06   2.463 0.013833 *
## mesJulio              1.385e+07  8.725e+06   1.587 0.112635
## mesAgosto            -4.399e+06  8.268e+06  -0.532 0.594770
## mesSeptiembre         6.136e+06  7.877e+06   0.779 0.436033
## mesOctubre           -3.875e+06  8.113e+06  -0.478 0.632945
## mesNoviembre          5.816e+06  8.705e+06   0.668 0.504085
## mesDiciembre          1.808e+07  8.407e+06   2.151 0.031563 *
## year                 -1.348e+05  1.217e+05  -1.108 0.268095
## productoras          -2.935e+04  2.375e+04  -1.236 0.216671
## reparto               2.404e+05  4.866e+04   4.941 8.22e-07 ***
## produccion            8.426e+04  2.198e+04   3.833 0.000130 ***
## Action1              -1.270e+07  4.564e+06  -2.782 0.005439 **
## Adventure1            1.971e+07  5.339e+06   3.691 0.000228 ***
## Animation1            2.004e+07  8.927e+06   2.244 0.024894 *
## Comedy1               6.773e+05  4.256e+06   0.159 0.873574
## Crime1               -1.394e+07  4.964e+06  -2.808 0.005017 **
## Documentary1          2.082e+07  1.241e+07   1.678 0.093554 .
## Drama1               -2.468e+06  4.085e+06  -0.604 0.545735
## Family1              -1.121e+06  6.918e+06  -0.162 0.871265
## Fantasy1             -6.772e+06  6.373e+06  -1.063 0.288068
## Foreign1              1.234e+07  2.044e+07   0.604 0.546176
## History1             -2.599e+07  8.495e+06  -3.060 0.002238 **
## Horror1               7.213e+06  6.271e+06   1.150 0.250169
## Music1                1.044e+06  9.204e+06   0.113 0.909656
## Mystery1             -5.682e+06  6.411e+06  -0.886 0.375500
## Romance1              1.316e+07  4.493e+06   2.928 0.003441 **
## Science_Fiction1     -1.249e+07  5.807e+06  -2.150 0.031621 *
## Thriller1            -6.629e+06  4.506e+06  -1.471 0.141393
## Tv_movie1             4.742e+07  8.546e+07   0.555 0.578994
## War1                 -2.810e+07  9.352e+06  -3.005 0.002679 **
## Western1             -7.991e+06  1.422e+07  -0.562 0.574240
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 84880000 on 2737 degrees of freedom
## Multiple R-squared:  0.6457, Adjusted R-squared:  0.6405
## F-statistic: 124.7 on 40 and 2737 DF,  p-value: < 2.2e-16
```

Observamos que el p-valor del modelo, asociado al estadístico F, es inferior a 0.05, lo cual nos llevaría a rechazar la hipótesis nula que nos dice que un modelo sin variables independientes se ajusta a los datos igual que lo hace nuestro modelo.

Extraemos los distintos gráficos explicativos de los residuos para comprobar el patrón de estos.



Por un lado observamos que la distribución no es normal a juzgar por el gráfico **Normal Q-Q** donde vemos que una buena parte de las observaciones no se encuentran sobre la recta. Además observamos heteroscedasticidad en gráfico **Scale-location**, es decir la varianza de los errores parece no ser constante a lo largo de los valores estimados por el modelo ("Fitted values"). Estas dos situaciones podrían llevar a una pérdida de precisión en los coeficientes de regresión y a producir p-valores con valores menores de lo que realmente son.

Para intentar logar un mejor ajuste del modelo procederemos a utilizar las variables **budget**, **popularity** y **revenue** transformadas anteriormente con el método `boxcox`.

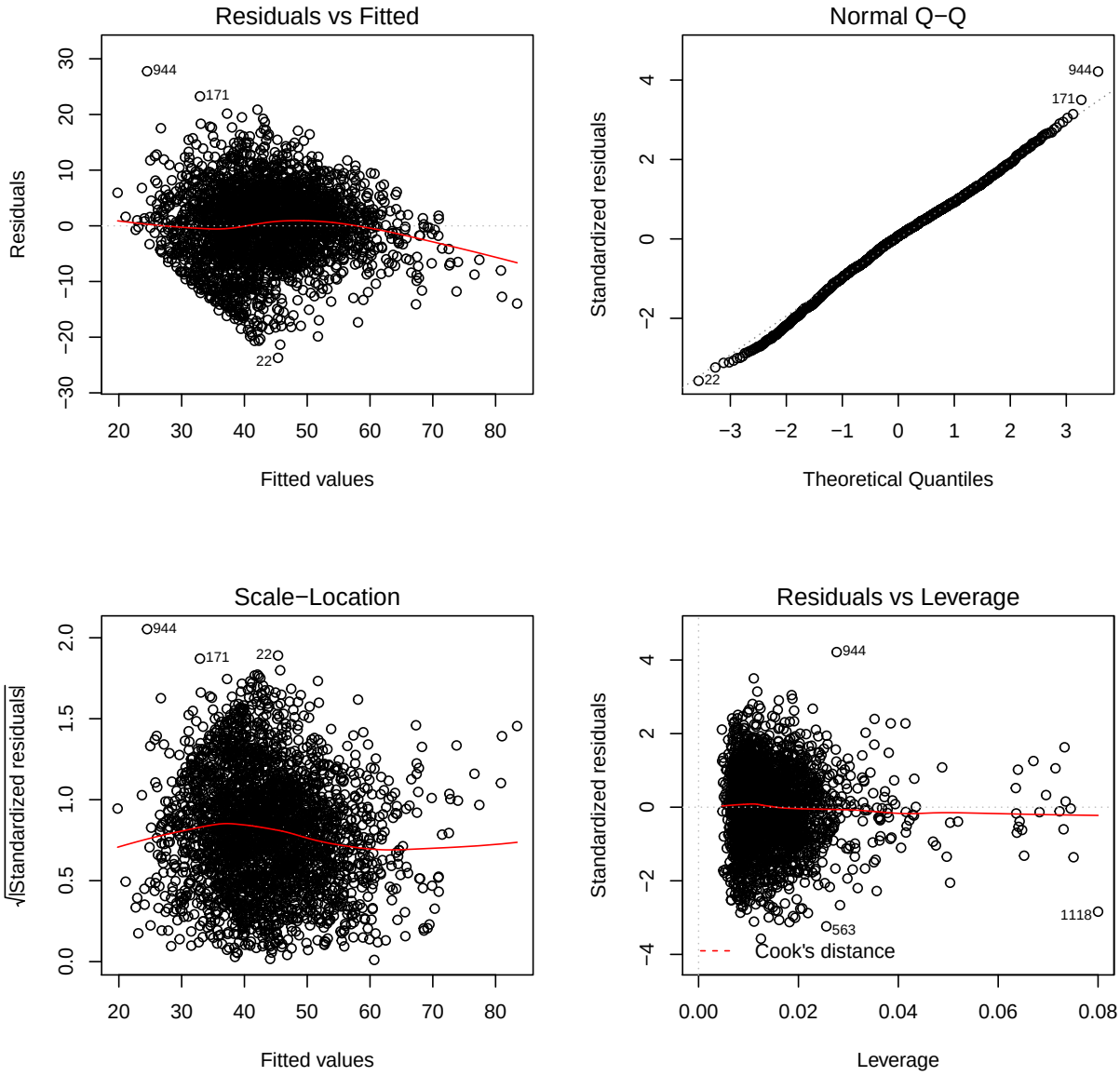
6.2. Modelo con las variables normalizadas

Procedemos pues a utilizar las variables normalizadas **budget_boxcox**, **popularity_boxcox** y **revenue_boxcox** en un segundo modelo.

```
##
## Call:
## lm(formula = formula, data = clean_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -23.7174  -4.3218   0.4672   4.3926  27.7641
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         29.129209  19.856902   1.467  0.14250
## budget_boxcox        0.074635   0.005513  13.539  < 2e-16 ***
## popularity_boxcox    1.690151   0.101783  16.605  < 2e-16 ***
## runtime              0.050138   0.007562   6.630 4.03e-11 ***
## collectionYes        4.257784   0.350128  12.161  < 2e-16 ***
## english_speakingYes -0.594786   0.464561  -1.280  0.20054
## mesFebrero           1.318711   0.682463   1.932  0.05343 .
## mesMarzo             0.307320   0.671122   0.458  0.64705
## mesAbril             0.496733   0.661560   0.751  0.45281
## mesMayo              0.656623   0.679821   0.966  0.33419
## mesJunio             1.375938   0.676039   2.035  0.04192 *
## mesJulio             1.811834   0.686513   2.639  0.00836 **
## mesAgosto            0.251466   0.650709   0.386  0.69919
## mesSeptiembre       -0.087534   0.619853  -0.141  0.88771
## mesOctubre          -0.309427   0.638417  -0.485  0.62794
## mesNoviembre         0.652282   0.685203   0.952  0.34120
## mesDiciembre         2.632534   0.661751   3.978 7.13e-05 ***
## year                -0.005451   0.009928  -0.549  0.58299
## productoras          0.019324   0.001911  10.112  < 2e-16 ***
## reparto              0.020196   0.003892   5.189 2.27e-07 ***
## produccion           0.017864   0.001728  10.337  < 2e-16 ***
## Action1              0.582216   0.360446   1.615  0.10637
## Adventure1           0.715662   0.415916   1.721  0.08542 .
## Animation1           0.985933   0.704896   1.399  0.16202
## Comedy1              0.849907   0.334731   2.539  0.01117 *
## Crime1              -0.379842   0.390788  -0.972  0.33114
## Documentary1        -1.413866   0.987148  -1.432  0.15218
## Drama1              -1.359179   0.321220  -4.231 2.40e-05 ***
## Family1              2.654077   0.546798   4.854 1.28e-06 ***
## Fantasy1            -0.008868   0.501853  -0.018  0.98590
## Foreign1            -1.295792   1.618147  -0.801  0.42332
## History1             0.148706   0.668089   0.223  0.82388
## Horror1              0.770185   0.494586   1.557  0.11953
## Music1              -0.018108   0.724577  -0.025  0.98006
## Mystery1             0.494899   0.504649   0.981  0.32684
## Romance1             1.847796   0.353785   5.223 1.89e-07 ***
## Science_Fiction1    -1.235335   0.456543  -2.706  0.00686 **
## Thriller1           -0.350573   0.354439  -0.989  0.32271
## Tv_movie1           13.874808   6.724470   2.063  0.03918 *
## War1                -0.784436   0.737395  -1.064  0.28752
## Western1            -0.809430   1.119795  -0.723  0.46984
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 6.679 on 2737 degrees of freedom
## Multiple R-squared:  0.616,  Adjusted R-squared:  0.6104
## F-statistic: 109.8 on 40 and 2737 DF,  p-value: < 2.2e-16
```

Extraemos los distintos gráficos explicativos de los residuos para comprobar el patrón de estos.



Observamos que la distribución de los residuos se aproxima a la normalidad después de la transformación de las variables. Prácticamente todas las observaciones están encima de la recta en el gráfico **Normal Q-Q**.

En cuanto a la homogeneidad de varianza (gráfico **Scale-Location**), observamos que ha mejorado bastante aunque no hemos conseguido del todo mantener la varianza constante a lo largo de todo el rango de valores.

Existen algunas observaciones extremas marcadas en el gráfico (casos 944, 29 y 2768) que podrían explicar que la varianza sea ligeramente diferente en el rango inferior de valores. Podremos probar a generar un modelo sin estos valores y ver si mejora la calidad.

De ahora en adelante generaremos modelos con las variables transformadas ya que nos darán resultados más fiables y precisos.

6.3. Modelo con variables normalizadas y sin los valores imputados de *budget*

Queremos comprobar la calidad del modelo en el caso de no haber imputado los valores perdidos de **budget**, que recordemos que afectaban a 682 casos y que constituyen el 27% del total.

```
##
## Call:
## lm(formula = formula, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -21.0193  -3.7188   0.3462   3.9419  21.1891
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         65.371016  21.762831   3.004  0.00270 **
## budget_boxcox        0.110477   0.006005  18.397  < 2e-16 ***
## popularity_boxcox    1.525114   0.109466  13.932  < 2e-16 ***
## runtime              0.031977   0.007863   4.067 4.95e-05 ***
## collectionYes        4.392200   0.357634  12.281  < 2e-16 ***
## english_speakingYes -1.224376   0.537364  -2.278  0.02280 *
## mesFebrero           1.153290   0.732122   1.575  0.11535
## mesMarzo             0.188992   0.726387   0.260  0.79475
## mesAbril             0.953267   0.716414   1.331  0.18347
## mesMayo              1.328849   0.733577   1.811  0.07021 .
## mesJunio             1.746534   0.716090   2.439  0.01481 *
## mesJulio             1.929233   0.726904   2.654  0.00801 **
## mesAgosto            0.388318   0.711012   0.546  0.58502
## mesSeptiembre        0.530061   0.671580   0.789  0.43004
## mesOctubre           0.128869   0.682357   0.189  0.85022
## mesNoviembre         0.809985   0.742407   1.091  0.27539
## mesDiciembre         2.417007   0.693288   3.486  0.00050 ***
## year                -0.022969   0.010886  -2.110  0.03497 *
## productoras          0.011812   0.001971   5.992 2.45e-09 ***
## reparto              0.009005   0.003914   2.301  0.02150 *
## produccion           0.013061   0.001690   7.728 1.69e-14 ***
## Action1             -0.294162   0.372029  -0.791  0.42921
## Adventure1           0.433938   0.425239   1.020  0.30763
## Animation1           0.976368   0.756072   1.291  0.19672
## Comedy1              0.716444   0.360024   1.990  0.04672 *
## Crime1              -0.659497   0.403311  -1.635  0.10216
## Documentary1        -0.217681   1.628728  -0.134  0.89369
## Drama1              -1.043183   0.339629  -3.072  0.00216 **
## Family1              1.616242   0.590122   2.739  0.00622 **
## Fantasy1            -0.450825   0.515025  -0.875  0.38149
## Foreign1            -2.778188   2.409393  -1.153  0.24902
## History1            -0.272364   0.698107  -0.390  0.69647
## Horror1              0.644282   0.505095   1.276  0.20225
## Music1               0.383521   0.786125   0.488  0.62570
## Mystery1            -0.056696   0.514464  -0.110  0.91226
```

```
## Romance1             1.581509   0.381636   4.144 3.55e-05 ***
## Science_Fiction1    -1.394099   0.457565  -3.047  0.00234 **
## Thriller1           -0.131900   0.364407  -0.362  0.71742
## Tv_movie1           12.467476   6.195803   2.012  0.04432 *
## War1                -0.839917   0.758221  -1.108  0.26810
## Western1            -1.070280   1.182111  -0.905  0.36536
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.142 on 2055 degrees of freedom
## Multiple R-squared:  0.6373, Adjusted R-squared:  0.6303
## F-statistic: 90.28 on 40 and 2055 DF,  p-value: < 2.2e-16
```

6.4. Modelo con variables normalizadas, sin imputar y eliminado *outliers*

Generamos un modelo eliminando los 3 outliers detectados en los gráficos de los residuos como valores que podrían estar alterando el ajuste del modelo. Corresponden a los casos 39, 944 y 276, que corresponden a las siguientes películas:

	X.U.FEFF.id	title	budget	revenue	runtime	popularity	year
29	31	The Intouchables	1.3e+07	426480871	112	16.086919	2011
276	299	Scenes from a Mall	3.0e+06	9563393	89	1.739182	1991
944	1021	The Living Sea	3.5e+05	87600000	40	1.081517	1995

Generamos el modelo sin esas 3 películas:

```
##
## Call:
## lm(formula = formula, data = df2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -21.0446  -3.7353   0.3507   3.9394  21.1499
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         64.599991  21.722173   2.974 0.002974 **
## budget_boxcox        0.110150   0.005995  18.375  < 2e-16 ***
## popularity_boxcox    1.508709   0.109391  13.792  < 2e-16 ***
## runtime              0.032257   0.007848   4.110 4.11e-05 ***
## collectionYes        4.409193   0.356985  12.351  < 2e-16 ***
## english_speakingYes -1.118197   0.537491  -2.080 0.037612 *
## mesFebrero           1.144190   0.730710   1.566 0.117535
## mesMarzo             0.175446   0.724994   0.242 0.808807
## mesAbril             0.945197   0.715031   1.322 0.186350
## mesMayo              1.317458   0.732165   1.799 0.072102 .
## mesJunio             1.731450   0.714720   2.423 0.015498 *
## mesJulio             1.918668   0.725504   2.645 0.008241 **
## mesAgosto            0.375440   0.709647   0.529 0.596827
## mesSeptiembre        0.519252   0.670288   0.775 0.438624
## mesOctubre           0.114203   0.681052   0.168 0.866846
## mesNoviembre         0.665428   0.742535   0.896 0.370274
```

```
## mesDiciembre         2.406909   0.691953   3.478 0.000515 ***
## year                -0.022600   0.010865  -2.080 0.037643 *
## productoras          0.011837   0.001968   6.016 2.11e-09 ***
## reparto              0.009251   0.003907   2.368 0.017977 *
## produccion           0.013086   0.001687   7.758 1.35e-14 ***
## Action1             -0.284263   0.371323  -0.766 0.444037
## Adventure1           0.429448   0.424417   1.012 0.311727
## Animation1           1.018807   0.754739   1.350 0.177203
## Comedy1              0.670797   0.359649   1.865 0.062303 .
## Crime1              -0.659256   0.402529  -1.638 0.101620
## Documentary1        -0.245088   1.625597  -0.151 0.880174
## Drama1              -1.073223   0.339118  -3.165 0.001575 **
## Family1              1.628955   0.588994   2.766 0.005732 **
## Fantasy1            -0.436789   0.514048  -0.850 0.395588
## Foreign1            -2.759607   2.404731  -1.148 0.251278
## History1            -0.255514   0.696776  -0.367 0.713874
## Horror1              0.639034   0.504119   1.268 0.205076
## Music1               0.382034   0.784602   0.487 0.626371
## Mystery1            -0.049035   0.513473  -0.095 0.923930
## Romance1             1.614569   0.381056   4.237 2.36e-05 ***
## Science_Fiction1    -1.396937   0.456679  -3.059 0.002250 **
## Thriller1           -0.128971   0.363702  -0.355 0.722921
## Tv_movie1           12.479331   6.183796   2.018 0.043714 *
## War1                -0.822779   0.756773  -1.087 0.277067
## Western1            -1.050961   1.179837  -0.891 0.373158
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.13 on 2054 degrees of freedom
## Multiple R-squared:  0.6384, Adjusted R-squared:  0.6314
## F-statistic: 90.67 on 40 and 2054 DF,  p-value: < 2.2e-16
```

6.5. Modelo eliminando variables no significativas

Generamos un modelo eliminando aquellas variables no significativas como son: **english_speaking** y **year**.

```
##
## Call:
## lm(formula = formula, data = df2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -21.4586  -3.7041   0.3338   3.9451  20.4451
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       18.734864   1.088145  17.217  < 2e-16 ***
## budget_boxcox      0.104766   0.005438  19.267  < 2e-16 ***
## popularity_boxcox  1.465642   0.108173  13.549  < 2e-16 ***
## runtime            0.037310   0.007554   4.939 8.48e-07 ***
## collectionYes      4.519547   0.354813  12.738  < 2e-16 ***
## mesFebrero         1.045183   0.730448   1.431 0.152617
## mesMarzo           0.119897   0.724674   0.165 0.868606
```

```
## mesAbril           0.911952   0.715506   1.275 0.202611
## mesMayo            1.309775   0.731572   1.790 0.073544 .
## mesJunio           1.763255   0.713992   2.470 0.013608 *
## mesJulio           1.869941   0.725368   2.578 0.010009 *
## mesAgosto          0.382911   0.709373   0.540 0.589401
## mesSeptiembre      0.389485   0.669044   0.582 0.560528
## mesOctubre         0.066690   0.679143   0.098 0.921786
## mesNoviembre       0.643340   0.741307   0.868 0.385580
## mesDiciembre       2.438483   0.690346   3.532 0.000421 ***
## productoras        0.012520   0.001876   6.672 3.23e-11 ***
## reparto            0.007638   0.003767   2.028 0.042736 *
## produccion         0.012815   0.001685   7.605 4.30e-14 ***
## Action1           -0.195493   0.369980  -0.528 0.597286
## Adventure1         0.454112   0.422709   1.074 0.282819
## Animation1         1.265800   0.745866   1.697 0.089832 .
## Comedy1            0.663615   0.359621   1.845 0.065135 .
## Crime1            -0.670397   0.402763  -1.664 0.096166 .
## Documentary1      -0.676809   1.618779  -0.418 0.675919
## Drama1            -1.080160   0.339473  -3.182 0.001485 **
## Family1            1.594483   0.588306   2.710 0.006778 **
## Fantasy1          -0.308872   0.512214  -0.603 0.546566
## Foreign1          -2.762742   2.402942  -1.150 0.250388
## History1          -0.201217   0.697127  -0.289 0.772888
## Horror1            0.575756   0.503239   1.144 0.252714
## Music1             0.423584   0.779739   0.543 0.587025
## Mystery1           0.016675   0.513265   0.032 0.974086
## Romance1           1.650119   0.380370   4.338 1.51e-05 ***
## Science_Fiction1  -1.381566   0.457027  -3.023 0.002534 **
## Thriller1         -0.143338   0.364013  -0.394 0.693790
## Tv_movie1         12.421576   6.190687   2.006 0.044934 *
## War1              -0.678114   0.754416  -0.899 0.368833
## Western1          -0.823975   1.172776  -0.703 0.482394
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.137 on 2056 degrees of freedom
## Multiple R-squared:  0.6372, Adjusted R-squared:  0.6305
## F-statistic: 95.05 on 38 and 2056 DF,  p-value: < 2.2e-16
```

7. Representación de los resultados a partir de tablas y gráficas

Generamos una tabla para comparar los distintos modelos.

modelos	var.normal	var.imput	outliers	var.excl	adj.r.squared	rmse
model1	No	Sí	Sí	-	0.6405414	8.425305e+07
model2	Sí	Sí	Sí	-	0.6103967	6.629746e+00
model3	Sí	No	Sí	-	0.6302637	6.081520e+00
model4	Sí	No	No	-	0.6313774	6.069704e+00
model5	Sí	No	No	english_speaking, year	0.6305408	6.079546e+00

Al haber transformado variables los parámetros de R^2 y $RMSE$ no pueden utilizarse para comparar la calidad del primer modelo con el resto (https://people.duke.edu/~rnau/rsquared.htm), por lo que habiendo

descartado el primer modelo debido a falta de normalidad y homoscedasticidad en los residuos, pasaremos a comparar la calidad de ajuste de los otros 4 modelos.

En primer lugar, observamos que el p-valor asociado al estadístico F es significativo en todos los modelos generados, con lo cual las variables incluidas tienen un valor explicativo de la variable dependiente en todos ellos.

En segundo lugar, sabemos que cuanto mayor es el valor de R^2 y menor sea el valor de $RMSE$, mayor variabilidad de los datos quedará explicada por el modelo. Observando los valores de todos los modelos generados, podemos decir que la calidad es aceptable, en torno al **60%** de la variabilidad de **revenue** quedaría explicada por las variables independientes incluidas en el modelo.

Podemos decidir quedarnos con aquellos modelos que mayor valor de R^2 y menor valor de $RMSE$ tienen: el 4º o 5º modelo. Observamos que hay muy pocas diferencias cuando eliminamos del modelo las variables **english_speaking** y **year** (modelo 5) por lo que podemos conservarlas y decidir entonces seleccionar el modelo 4.

Recordemos que en el modelo 4 hemos utilizado las variables normalizadas **budget_boxcox** y **revenue_boxcox**, hemos excluido los valores imputados para "budget" y hemos eliminado 3 casos extremos detectados en el análisis de los residuos.

Entre las variables más significativas para explicar la recaudación se encuentran las cuantitativas **budget, popularity, produccion y productoras**, las cuales ya presentaban un grado de correlación moderado con **revenue** mientras que, entre las cualitativas, destaca **collection** que ya vimos también en el análisis visual y estadístico que había diferencias significativas en cuanto a si una película pertenecía a una saga o no.

Aquí mostramos la tabla completa con las variables y sus p-valores ordenados de menor a mayor.

```
##                    var      pvalues
## 2        budget_boxcox 6.147119e-70
## 3    popularity_boxcox 1.888876e-41
## 5        collectionYes 7.399143e-34
## 21          produccion 1.348418e-14
## 19         productoras 2.114305e-09
## 36            Romance1 2.364439e-05
## 4              runtime 4.109784e-05
## 17        mesDiciembre 5.148660e-04
## 28              Drama1 1.574926e-03
## 37    Science_Fiction1 2.250197e-03
## 1          (Intercept) 2.974470e-03
## 29             Family1 5.731639e-03
## 12           mesJulio 8.241013e-03
## 11           mesJunio 1.549771e-02
## 20             reparto 1.797744e-02
## 6  english_speakingYes 3.761228e-02
## 18                year 3.764284e-02
## 39           Tv_movie1 4.371379e-02
## 25             Comedy1 6.230347e-02
## 10             mesMayo 7.210215e-02
## 26              Crime1 1.016195e-01
## 7           mesFebrero 1.175353e-01
## 24          Animation1 1.772033e-01
## 9             mesAbril 1.863499e-01
## 33             Horror1 2.050759e-01
## 31           Foreign1 2.512780e-01
## 40               War1 2.770669e-01
## 23          Adventure1 3.117266e-01
```

```
## 16        mesNoviembre 3.702739e-01
## 41            Western1 3.731583e-01
## 30            Fantasy1 3.955882e-01
## 14       mesSeptiembre 4.386237e-01
## 22             Action1 4.440373e-01
## 13           mesAgosto 5.968267e-01
## 34              Music1 6.263705e-01
## 32            History1 7.138742e-01
## 38           Thriller1 7.229215e-01
## 8             mesMarzo 8.088072e-01
## 15          mesOctubre 8.668465e-01
## 27        Documentary1 8.801736e-01
## 35            Mystery1 9.239298e-01
```

8. Resolución del problema.

> A partir de los resultados obtenidos, ¿cuáles son las conclusiones? ¿Los resultados permiten responder al problema?

Como conclusión final podemos decir que el objetivo del presente proyecto consistía en generar un modelo de regresión lineal para predecir el valor de la recaudación de las películas en base a una serie de variables explicativas, para lo cual se han planteado distintos modelos derivados de diferentes combinaciones y transformaciones de éstas. Finalmente, el modelo elegido que nosotros elegiríamos de los que se han analizado sería el 4º teniendo en cuenta los siguientes criterios:

- El modelo planteado tiene una potencia de previsión media-alta basándonos en el coeficiente de determinación ajustado obtenido, R^2= **0.6313774**, y que consigue explicar aproximadamente el **63%** de la variabilidad total de la recaudación en base a las variables independientes elegidas.
- Además, los residuos del modelo presentan una distribución aproximada a la normal y una varianza bastante homogénea a lo largo de los valores predecidos, lo cual nos da una precisión fiable del modelo.

No obstante esta modelización no tiene porque ser la opción más óptima (*) y sólo representa una prueba de concepto de la capacidad de previsión que tienen los nuevos datos transformados de los originales y que éstos originalmente no tenían. Desde este punto de vista, efectivamente, podemos decir que el ejercicio realizado consigue el objetivo prefijado inicialmente.

`Nota: Posiblemente con más tiempo de análisis y recombinado las nuevas variables (por ejemplo en otras dicotómicas entre generos populares-no populares ó meses optimos-no optimos para estrenos etc etc), eliminando variables poco significativas ó incorporando otras que hayamos podido obviar podríamos conseguir modelos más sencillos y mejores ratios de previsión que los ejemplos mostrados en este ejercicio`

9. Tabla de contribuciones

El ejercicio ha sido realizado por:

Contribuciones	Firma
Investigación previa	Jon Ortiz, Gabriel Peso
Redacción de las respuestas	Jon Ortiz, Gabriel Peso
Desarrollo código	Jon Ortiz, Gabriel Peso

10. Recursos

A continuación se listan los diferentes recursos utilizados para la realización del ejercicio.

Origen del dataset

- https://www.kaggle.com/c/tmdb-box-office-prediction

Conversión de formato JSON

- https://www.kaggle.com/samstiyer/parsing-the-json-columns-in-r-tidy-approach

Expresiones regulares

- https://stringr.tidyverse.org/articles/regular-expressions.html
- https://www.regextester.com/21

Transformación boxcox()

- https://rpubs.com/bskc/288328
- https://www.rdocumentation.org/packages/forecast/versions/8.7/topics/BoxCox.lambda

Test de normalidad

- https://www.youtube.com/watch?v=_YOr_yYPytM
- https://www.youtube.com/watch?v=vo9DssNQA4E
- https://www.statisticssolutions.com/normality/
- https://data.library.virginia.edu/normality-assumption/

Test de homogeneidad de varianzas

- https://www.statisticssolutions.com/homoscedasticity/
- https://biostats.w.uib.no/test-for-homogeneity-of-variances-levenes-test/
- https://www.statisticshowto.datasciencecentral.com/levene-test/
- https://stats.libretexts.org/Bookshelves/Biostatistics/Book%3A_Biological_Statistics_(McDonald)/4.0/4.05%3A_Homoscedasticity_and_Heteroscedasticity

Test de correlación

- https://www.researchgate.net/post/Is_Pearsons_Correlation_coefficient_appropriate_for_non-normal_data
- https://www.statisticssolutions.com/pearson-correlation-assumptions/
- https://www.statisticshowto.datasciencecentral.com/kendalls-tau/
- https://www.youtube.com/watch?v=D56dvoVrBBE
- https://www.statsdirect.com/help/nonparametric_methods/kendall_correlation.htm

Contraste de hipótesis: T-test

- https://www.rdocumentation.org/packages/stats/versions/3.6.0/topics/t.test

- https://www.r-bloggers.com/two-sample-students-t-test-1/

Contraste de hipótesis: ANOVA test

- http://www.sthda.com/english/wiki/two-way-anova-test-in-r
- https://www.r-bloggers.com/two-way-analysis-of-variance-anova/
- https://rcompanion.org/rcompanion/d_08.html
- https://www.researchgate.net/post/One-Way_ANOVA_or_Kruskal_Wallis_which_one_should_I_use
- https://www.statmethods.net/stats/anova.html
- https://arc.lib.montana.edu/book/statistics-with-r-textbook/item/56
- https://www.r-bloggers.com/performing-anova-test-in-r-results-and-interpretation/

Modelos de regresión lineal múltiple (variables cuantitativas y cualitativas):

- Libro:"Regression analysis: An intuitive guide" (Jim Frost, 2019)
- http://analyticspro.org/2016/03/15/r-tutorial-how-to-interpret-f-statistic-in-regression-models/
- https://thestatsgeek.com/2014/01/25/r-squared-and-goodness-of-fit-in-linear-regression/
- https://thestatsgeek.com/2013/08/07/assumptions-for-linear-regression/
- https://www.theanalysisfactor.com/assessing-the-fit-of-regression-models/
- https://www.kaggle.com/samstiyer/parsing-the-json-columns-in-r-tidy-
- https://data.library.virginia.edu/diagnostic-plots/