Doble Grado en
Matemáticas y Estadística

TRABAJO FIN DE GRADO

Conociendo el streaming de video bajo demanda: Matemáticas, Estadística e IA para el análisis macro y la construcción de modelos micro

María Lobato Ripoll

Tutor: Luis Valencia Cabrera

Sevilla, Junio de 2021

Índice general

Resumen

El éxito de las plataformas de streaming de vídeo queda patente en el hecho de que hoy día se haya convertido en uno de los principales medios de entretenimiento. Las compañías propietarias de estas plataformas han manifestado en varias ocasiones el uso de técnicas de machine learning, estadística e inteligencia artificial tanto para la construcción de sus algoritmos como para predecir su éxito a largo plazo.

El principal objetivo de este estudio es realizar una descripción, desde el punto de vista analítico, del panorama actual de las plataformas de streaming. Para ello, se analizan exhaustivamente los datos accesibles de las mismas y, haciendo uso intensivo de técnicas estadísticas e informáticas que van desde la importación y tratamiento, hasta el análisis exploratorio y el análisis de la evolución de sus variables, se da a conocer su estado tanto a nivel nacional como global, indicando cómo cambian los rasgos de las plataformas dependiendo del país en el que nos encontremos.

De manera análoga, se estudia la popularidad de estas plataformas, a través de las características del contenido que producen, realizando previamente un análisis descriptivo de las producciones, sirviéndonos tanto de técnicas de machine learning como de estadística, acompañando los distintos modelos de clasificación y regresión con el estudio cuidadoso de la bondad de los mismos.

Abstract

The success of video streaming platforms is evident, being one of the main ways of digital entertainment nowdays. The companies that own these platforms have stated on several occasions the use of machine learning, statistics and artificial intelligence techniques both for the construction of their algorithms and for predicting their long-term success.

The main objective of this study is to make a description, from an analytical point of view, of the current state of nature of streaming platforms. To this end, the accessible data of these platforms are exhaustively analyzed and, making intensive use of statistical and computer techniques ranging from import and processing to exploratory analysis and analysis of the evolution of their variables, their status is revealed both nationally and globally, indicating how the features of the platforms change depending on the country of origin.

Similarly, the popularity of these platforms is studied, through the characteristics of the content they produce, previously performing a descriptive analysis of the productions, using both machine learning and statistical techniques, accompanying the different classification and regression models with a careful study of their goodness of fit.

Índice de figuras

Índice de tablas

Parte I

Preliminares

Este primer bloque está orientado a exponer la descripción del estudio realizado, así como a la realización de una sintésis de las técnicas empleadas en el desarrollo del mismo.

Capítulo 1

Introducción

El mundo del cine ha encontrado en las dos últimas décadas lo que podríamos denominar como su mejor (o peor) aliado: las plataformas de streaming de video. El nacimiento de estas plataformas marcó un antes y un después en la industria: conocidas por tener una inmensa biblioteca de contenido y producir algunas de las series más exitosas, se han adentrado poco a poco en grandes galardones como los Óscar.

La popularidad de estas plataformas es evidente, lo cuál hace que los estudios que se efectúan de las mismas sean de lo más diverso, tanto por su autoría como por su contenido: desde individuos analizando una serie o película en blogs particulares hasta revistas de renombre que especulan cual será la plataforma líder en los siguientes años. Por consiguiente, son también numerosas las ramas desde las que se realizan estudios sobre este asunto, encontrándonos desde análisis económicos hasta críticas subjetivas.

El objetivo de este trabajo es investigar este tema abordando algunos de sus aspectos más relevantes desde la perspectiva analítica mediante el uso de las matemáticas, la estadística y la inteligencia artificial. Con este propósito se atenderán principalmente dos vías: la primera de ellas orientada a dar una visión general acerca del estado de estas plataformas en la actualidad, y la segunda enfocada a la construcción de modelos para predecir el éxito de las producciones originales de las mismas.

Para hacer frente a estas dos vías se ha dividido el estudio en cuatro bloques, subdividiendo cada uno de ellos en los capítulos pertinentes. El primer bloque contiene la introducción y un breve resumen de las técnicas estadísticas y los paquetes de R usados en el desarrollo del trabajo. El segundo trata la primera vía de las dos mencionadas, investigando en dos capítulos el estado de estas plataformas en la actualidad: el primero resume el panorama actual de las mismas a nivel global y nacional, mientras que el segundo recoge la distribución del contenido de las plataformas en los diferentes países, comparando su contenido, precio y calidad.

El tercer bloque se ocupa de la segunda vía, abordando una de las actividades más populares de estas plataformas: su propia producción de contenido. En él se expone cuál es el objetivo principal de las compañías con esta práctica: alcanzar el éxito de sus producciones. Así, haciendo uso de un conjunto de datos que engloba las principales características de diferentes películas, se realiza un estudio predictivo de este factor, midiéndolo a través de la puntuación de la película y del número de premios que posee. Para ello, se divide el bloque en tres capítulos: el primero de ellos orientado al pretratamiento y análisis de

los datos, el segundo dedicado a la construcción de modelos de regresión y el último a la elaboración de modelos de clasificación.

Por último, se encuentra el cuarto bloque, que contiene las conclusiones finales del trabajo y esboza algunas posibles líneas de trabajo futuro.

Capítulo 2

Preliminares

En este capítulo introductorio se detallan las técnicas estadísticas empleadas en el desarrollo del estudio, así como los paquetes de R utilizados en el mismo, dando una visión general de cada una de sus funcionalidades.

2.1. Técnicas estadísticas

2.1.1. Regresión lineal multiple

El objetivo general de la regresión lineal es, dada una variable aleatoria Y y varias variables aleatorias $X_1, \ldots, X_p$, encontrar una relación lineal de la forma $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p + \epsilon$, donde ϵ es el error aleatorio. Para ello, dada una muestra de datos $(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_n, y_n)$ donde cada $\boldsymbol{x}_i$ es un vector de $\mathbb{R}^p$, buscaremos estimadores $\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_p$, de forma que dado $\boldsymbol{x} = (x_1, \ldots, x_p) \in \mathbb{R}^p$ predecimos su valor correspondiente de y, $\hat{y}$, como $\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \ldots + \hat{\beta}_p x_p$. Definiendo $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$, escogeremos $\boldsymbol{\beta}$ tal que minimice la suma de cuadrados de los residuos, RSS, definida en este caso como:

$$\mathrm{RSS}(\boldsymbol{\beta}) = \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij})^2$$

Sea $\boldsymbol{y} = (y_1, \ldots, y_n)'$ y $\mathbf{X}$, la matriz de diseño dada por

$$\mathbf{X} = \begin{pmatrix} 1 & \boldsymbol{x}_1' \\ 1 & \boldsymbol{x}_2' \\ \vdots & \vdots \\ 1 & \boldsymbol{x}_n' \end{pmatrix}$$

podemos expresar el modelo como $\boldsymbol{Y} = \mathbf{X}\boldsymbol{\beta} + \epsilon$, siendo $\epsilon = (\epsilon_1, \ldots, \epsilon_n)$ el error aleatorio. Esto nos permite reescribir el $\mathrm{RSS}(\boldsymbol{\beta})$ con lenguaje matricial mediante la siguiente expresión:

$$\mathrm{RSS}(\boldsymbol{\beta}) = (\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta})'(\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta})'$$

Al ser la función RSS convexa, el mínimo se alcanzará cuando $\nabla \text{RSS}(\boldsymbol{\beta}) = 0$. Resolviendo, se tiene que el estimador de $\boldsymbol{\beta}$ será $\hat{\boldsymbol{\beta}} = (\mathbf{X'X})^{-1}\mathbf{X'}\boldsymbol{y}$, donde la matriz $\mathbf{X'X}$ tiene determinante no nulo. De todos los estimadores insesgados de $\boldsymbol{\beta}$, el $\hat{\boldsymbol{\beta}}$ obtenido es el de mínima varianza.

2.1.1.1. Diagnóstico del modelo

Para poder calcular las distintas propiedades del estimador y realizar inferencia sobre el modelo, se deben admitir una serie de hipótesis sobre los residuos del mismo. A continuación se expone cada una de ellas, indicando cómo comprobarlas:

- **Media cero**

Se debe verificar que $\text{E}(\epsilon_i) = 0$, $\forall i$.

- **Normalidad**

Los residuos deben seguir una distribución normal de media cero, esto es, $\epsilon_i \sim \text{N}(0, \sigma^2)$, $\forall i$.

La forma habitual de detectar la no normalidad es mediante los gráficos qq de normalidad. En estos se representa en el eje de abscisas los cuantiles teóricos de la distribución normal y en el de ordenadas los muestrales. Si la nube de puntos obtenida se sitúa alrededor de la recta $y = x$, indica que la distribución muestral coincide con la normal, y por tanto se aceptaría la hipótesis de normalidad.

Otra forma de identificar la normalidad, es mediante algún test de bondad de ajuste. Estos test determinan a que familia de distribuciones pertenecen las variables bajo estudio. Existen diversos test para comprobar la normalidad, como el de Shapiro-Wilk (específico para normalidad), el de Kolmogorov-Smirnov o el de Anderson-Darling.

- **Homocedasticidad**

La hipótesis de homocedasticidad establece que los residuos tienen igual varianza y que esta es constante, es decir, $\text{Var}(\epsilon_i) = \sigma^2$, $\forall i$.

Se puede comprobar gráficamente representando los residuos frente a los valores ajustados. Si la distribución de los puntos es aleatoria, diremos que se cumple la hipótesis de homocedasticidad. Por el contrario, si se observa cierta tendencia creciente o decreciente en los residuos, esto indicaría que la varianza de los mismos no es fija, contrariando la hipótesis de homocedasticidad.

Asimismo, existen diversos test para verificar dicha hipótesis como el de Barlett, el de Levene o el de Breusch-Pagan.

- **Incorrelación**

Las diferentes observaciones de la variable respuesta $y_1, \ldots, y_n$ deben estar incorreladas entre sí, esto es, $\text{Cor}(\epsilon_i, \epsilon_j) = 0$, $\forall i, j$.

Si esta hipótesis no es cierta, cabe esperar que un gráfico secuencial de los residuos manifieste alguna tendencia. Sin embargo, hay muchas formas en que los errores pueden estar correlados, por lo que lo mejor es realizar algún test estadístico para comprobarlo, como el test de Durbin-Watson.

2.1.1.2. Multicolinealidad

Las variables implicadas en el modelo deben ser independientes. A pesar de que esta hipótesis no sea de obligatorio cumplimiento como las anteriores, sí es cierto que la ausencia de ella puede ocasionar problemas en el modelo, pues si las variables están relacionadas podría ocurrir que la matriz $\mathbf{X'X}$ fuese singular, por lo que no se podrían calcular los estimadores $\hat{\beta}_i$.

Uno de los efectos principales de la multicolinealidad es la inflación de la varianza en las estimaciones, por tanto, una forma de detectarla es mediante el cálculo de los denominados factores de inflación de la varianza, VIF, para cada una de las variables involucradas en el modelo. Estos vienen dados por la siguiente expresión

$$\text{VIF}_i = \frac{1}{1 - R_i^2}$$

donde R_i^2 es el coeficiente de determinación para la regresión lineal múltiple de la variable X_i sobre las $p - 1$ variables restantes. Valores entre dos y cinco indican cierta multicolinealidad, mientras que valores superiores a diez son señal de una multicolinealidad muy alta.

2.1.1.3. Inferencia sobre los coeficientes del modelo

Bajo la hipótesis de normalidad tenemos que el estimador de β, $\hat{\beta}$, se distribuye según una $\text{N}(\beta, \sigma^2(\mathbf{X'X})^{-1})$, siendo por tanto la distribución de cada uno de los $\hat{\beta}_i$ la siguiente

$$\hat{\beta}_i \sim N(\beta_i, \sigma^2 C_{ii}^X)$$

donde C_{ii}^X es la diagonal de la matriz $(\mathbf{X'X})^{-1}$. Al conocer la distribución de los $\hat{\beta}_i$ podemos construir intervalos de confianza o realizar contrastes de hipótesis. Así pues, para saber si la i-ésima variable involucrada en el modelo es significativa, nos planteamos el siguiente contraste:

$$\begin{cases} H_0: & \beta_i = 0 \\ H_1: & \beta_i \neq 0 \end{cases}$$

Teniendo en cuenta que

$$\frac{\hat{\beta}_i - \beta_i}{\sqrt{s^2 C_{ii}^X}} \sim t_{n-p}$$

donde s^2 es el estimador de σ^2, rechazamos H_0 a un nivel 1 - α si $|\hat{\beta}_i| > t_{(n-p,1-\alpha/2)}\sqrt{s^2 C_{ii}^X}$.

2.1.2. Regresión logística

El objetivo general de la regresión logística es encontrar la relación entre una variable binaria aleatoria Y y varias variables aleatorias $X_1, \ldots, X_p$. Este modelo se basa en las probabilidades por lo que, dada una observación se calculará la probabilidad de que esta pertenezca a una clase u otra, y se asignará a aquella a la que sea más probable que pertenezca.

El principal problema de este método es que intentando ajustar un modelo de regresión lineal como el descrito en la sección 2.1.1, puede ocurrir que los valores que obtengamos estén fuera del intervalo [0,1], por lo que no se trataría de probabilidades. Para solucionar este hecho, se transforma el valor devuelto por la regresión lineal aplicando la función sigmoide, cuyo valor devuelto está siempre entre 0 y 1:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Sustituyendo la x de la ecuación por la función lineal obtenemos:

$$P(Y = 1|\underline{X} = \underline{x}) = \frac{1}{1 + e^{-(\beta_0+\beta_1 X_1+\ldots+\beta_p X_p)}} = \ldots = \frac{e^{\beta_0+\beta_1 X_1+\ldots+\beta_p X_p}}{1 + e^{\beta_0+\beta_1 X_1+\ldots+\beta_p X_p}}$$

donde $P(Y = 1|\underline{X} = \underline{x})$ se interpreta como la probabilidad de que la variable cualitativa adquiera el valor 1. Por tanto, $P(Y = 0|\underline{X} = \underline{x}) = 1 - P(Y = 1|\underline{X} = \underline{x})$.

Los coeficientes del modelo, se estiman mediante el método de máxima verosimilitud que no se detalla en este estudio. Una vez estimados, dada una observación es posible conocer la probabilidad de que esta pertenezca al nivel de referencia. Así pues, se clasificará en dicho nivel si la probabilidad es superior a 0.5.

2.1.3. Random Forest

Los modelos Random Forest están integrados por un grupo de árboles de decisión individuales, cada uno entrenado con una muestra de datos levemente diferente del conjunto de entrenamiento generada mediante bootstrapping.

Los **árboles de decisión** son modelos usados para predecir, generalmente formados por reglas binarias aunque existen otras variantes. En ellos, se reparten las observaciones en función de las variables que los definen, para predecir así el valor de la variable respuesta. Esta técnica se usa tanto para problemas de regresión como de clasificación.

Los árboles de regresión se utilizan cuando la variable respuesta es continua. Las observaciones se distribuyen por diferentes nodos hasta alcanzar un nodo terminal. Así, cuando se quiere predecir una nueva observación se recorre el árbol en función del valor de sus variables hasta alcanzar un nodo terminal. La predicción de esa observación es la media de las observaciones del conjunto de entrenamiento que están en ese nodo terminal.

Por el contrario, los árboles de clasificación se utilizan cuando la variable respuesta es categórica. De igual modo, las observaciones se distribuyen por los diferentes nodos hasta alcanzar un nodo terminal. Cuando se quiere predecir la clase de una nueva observación, al igual que antes se recorre el árbol en función de sus variables hasta alcanzar un nodo

terminal. La clase asignada a esa observación será la clase que más se repita en las observaciones del conjunto de entrenamiento que están en ese nodo terminal, es decir, la moda de la variable respuesta.

El modelo de Random Forest surge, como otros, para intentar solventar el problema de equilibrio entre sesgo y varianza que tienen todos los modelos de aprendizaje estadístico. El problema del sesgo hace referencia a cuánto se alejan las predicciones de un modelo respecto a los valores reales, mientras que el de la varianza hace referencia a cuanto cambia el modelo dependiendo de los datos que se utilicen. El modelo ideal es aquel que consigue un equilibrio entre ambos.

Una de las soluciones a este problema, son los métodos de *ensemble*. Estos métodos tratan de solucionar el problema de equilibrio entre sesgo y varianza, combinando varios modelos en uno nuevo. Dos de los tipos de *ensemble* más conocidos son *bagging* y *boosting*. El primero de ellos ajusta varios modelos, usando para cada uno un subconjunto distinto de los datos de entrenamiento, mientras que el segundo ajusta secuencialmente múltiples modelos sencillos, de forma que cada uno aprenda de los errores del anterior. El modelo Random Forest se basa en la primera de estas estrategias, el *bagging*.

La idea principal del método *bagging* es reducir la varianza y aumentar la precisión obteniendo diversas muestras de la población, ajustando un modelo distinto para cada una de ellas y haciendo la media (o moda si la variable objetivo es categórica) de las predicciones resultantes. Como no se suele tener acceso a múltiples muestras, el método simula el proceso mediante bootstrapping, generando así pseudo-muestras para ajustar los diferentes modelos.

Random Forest no es más que una modificación del proceso de *bagging*. Este proceso se ha creado con objeto de reducir la varianza, pero esto es cierto siempre y cuando los diferentes modelos no estén correlacionados, en cuyo caso la disminución de la varianza sería mínima lo cual suele ocurrir cuando una de las variables predictoras es muy influyente, pues todos los árboles creados con el proceso de *bagging* estarán condicionados por el mismo predictor y serán muy parecidos entre ellos. Random Forest soluciona este problema seleccionando aleatoriamente m variables explicativas que serán empleadas para la construcción de cada árbol del bosque, consiguiendo disminuir la correlación entre los árboles, y, por tanto, disminuir la varianza.

Dada una nueva observación, esta se distribuye en cada uno de los árboles por los diferentes nodos hasta llegar a un nodo terminal. Para cada árbol, se calcula el promedio del valor de la variable objetivo en las observaciones que constituyen el nodo terminal (o la moda en el caso de clasificación), obteniendo por tanto un valor de la predicción en cada uno de ellos. La predicción final será la media de las predicciones de todos los árboles (en el caso de regresión) o la moda (en el caso de clasificación).

2.1.4. Análisis de componentes principales

El objetivo principal del análisis de componentes principales es reducir el tamaño de un conjunto de datos descrito por un número elevado de variables relacionadas entre sí.

Mediante esta técnica, se pretende describir la estructura de varianzas y covarianzas del conjunto de variables originales a través de un nuevo conjunto de variables, más pequeño, que están definidas como combinación lineal de las variables iniciales.

Dado un vector aleatorio $\underline{X}$, p dimensional, las componentes principales son, por tanto, combinaciones lineales de las p variables. La i-ésima componente es la variable Y_i tal que

$$Var(Y_i) = \sup_{\{\underline{t}\in\mathbb{R}^p:\underline{t}'\underline{t}=1, Cov(Y_i,Y_j)=0, j=1\ldots i-1\}} Var(\underline{t}'\underline{X})$$

Se cumple que, dado $\underline{X}$ un vector aleatorio p dimensional con matriz de varianzas y covarianzas Σ, si $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_p \geq 0$ son los autovalores de Σ con autovectores unitarios asociados $e_1, \ldots, e_p$, entonces la i-ésima componente principal de $\underline{X}$ viene dada por $Y_i = e_i'\underline{X}$, verificando entonces que $Var(Y_i) = \lambda_i$.

La correlación entre las variables originales y las componentes principales viene dada por

$$\rho(Y_i, X_j) = e_{ji}\sqrt{\lambda_i}/\sqrt{\sigma_{jj}}$$

y permite explicar el significado de las nuevas variables en función de las anteriores.

Así pues, la proporción de la variabilidad total que explican las k primeras componentes principales es

$$\frac{\sum_{i=1}^{k}\lambda_i}{\sum_{s=1}^{p}\lambda_s}$$

y la variabilidad explicada por la i-ésima componente principal será

$$\frac{\lambda_i}{\sum_{s=1}^{p}\lambda_s}$$

Se retendrán las k primeras componentes principales que acumulen un determinado porcentaje de la variabilidad explicada, tomando k menor que p.

En la práctica, la matriz de varianzas y covarianzas es desconocida, por lo que se trabajará con la matriz de varianzas y covarianzas muestrales $\hat{\Sigma}$ y con sus autovalores $\hat{\lambda}_1, \ldots, \hat{\lambda}_p$.

2.1.5. Medidas de bondad del ajuste en modelos de regresión

2.1.5.1. Coeficientes de determinación

La suma de cuadrados de los residuos se define como

$$\text{RSS} : \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

La suma total de cuadrados, TSS, y la suma de cuadrados explicada, ESS, se define como

$$\text{TSS} : \sum_{i=1}^{n}(y_i - \bar{y})^2$$

$$\text{ESS} : \sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2$$

donde TSS = RSS + ESS.

Definiremos el **coeficiente de determinación**, R^2 como

$$R^2 = \frac{\text{ESS}}{\text{TSS}} = 1 - \frac{\text{RSS}}{\text{TSS}}$$

Se tiene que $0 \leq R^2 \leq 1$, y que $R^2 = 1 \Leftrightarrow \text{RSS} = 0$, por tanto, el ajuste es mejor cuanto más cerca esté el valor de R^2 del uno. Además, se define el **coeficiente de determinación ajustado**, R^2_{adj}, como

$$R^2_{adj} = 1 - \frac{\frac{1}{n-(p+1)}\sum_{i=1}^{n}(y_i - \hat{y})^2}{\frac{1}{n-1}\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

Este coeficiente es más fiable que el anterior, ya que al añadir más variables el R^2 siempre mejorará pero no sabremos si hace el modelo menos estable, mientras que al añadir variables inadecuadas el R^2_{adj} puede disminuir, indicándonos que incorporarlas no ha sido una buena idea.

2.1.5.2. Error absoluto medio

El **error absoluto medio**, MAE, mide la magnitud del promedio de los errores en un conjunto de datos, sin tener en cuenta su dirección. Viene dado por:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

Toma valores entre cero e infinito. Cuanto más bajo sea su valor, mejor es la estimación. Su interpretación es muy sencilla, pues indica la media de las diferencias, en valor absoluto, entre las predicciones y los valores reales.

2.1.5.3. Error cuadrático medio

El **error cuadrático medio**, RMSE, mide el promedio de los errores al cuadrado. Viene dado por:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Toma valores entre cero e infinito. Cuanto más bajo sea su valor, mejor es la estimación. Al elevar los errores al cuadrado antes de realizar el promedio, esta medida otorga un peso superior a los errores grandes.

2.1.5.4. Criterios de información

El **criterio de información de Akaike**, AIC, mide la calidad relativa de un modelo estadístico, permitiendo comparar unos modelos con otros. Viene dado por:

$$\text{AIC} = 2k - 2\ln(L)$$

donde k es el número de parámetros del modelo y L es el máximo valor de la función de verosimilitud del modelo considerado.

Dado un conjunto de modelos, el mejor de ellos será el que proporcione un menor AIC. Además, al tener en cuenta el número de parámetros del modelo, esta medida otorga una penalización, considerando peores aquellos modelos con más parámetros.

Por otro lado, el **criterio de información bayesiano**, BIC, también mide la calidad relativa del modelo estadístico, permitiendo comparar unos modelos con otros. Su fórmula es la siguiente:

$$\text{BIC} = k\ln(n) - 2\ln(L)$$

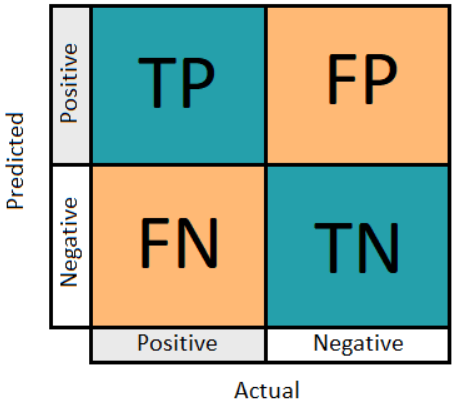
donde k y L representan los mismos valores que antes y n es el número de observaciones del modelo. De igual modo, dado un conjunto de modelos el mejor de ellos será el que proporcione un menor BIC.

La principal diferencia entre ambas medidas está en cómo miden estas la complejidad del modelo. Mientras que el AIC solamente se basa en el número de parámetros, k, el BIC incorpora tanto k como ln(n), apareciendo el número de parámetros como un factor que multiplica al tamaño muestral. Por tanto, el BIC penaliza más la complejidad del modelo que el AIC.

2.1.6. Medidas de bondad del ajuste en modelos de clasificación

La principal herramienta en la que se basan las diferentes medidas de bondad del ajuste en modelos de clasificación es la **matriz de confusión**. Se trata de una matriz M de tamaño $m \times m$, siendo m es el número de categorías de la variable objetivo. Cada fila representa el número de predicciones de cada clase y cada columna los valores reales, aunque también se puede encontrar lo contrario. Así pues, en la posición M_{ij} se hallará el número de instancias de la variable objetivo cuyo valor real era el de la categoría que representa la columna j-ésima y que han sido precedidas con la categoría de la fila i-ésima. La situación óptima es por tanto aquella en la que la matriz M es diagonal.

Si la variable objetivo es una variable binaria, esto es, tan solo tiene dos niveles, la matriz de confusión será por tanto una matriz 2×2, de la siguiente forma:



Se denominan **falsos negativos**, FN, a aquellas observaciones pertenecientes a la categoría codificada con el cero que el modelo ha clasificado como perteneciente a la categoría codificada con el uno. Por su parte, se denominan **falsos positivos**, FP, a aquellas observaciones pertenecientes a la categoría clasificada con el uno que el modelo ha clasificado como perteneciente a la categoría codificada con el cero. Los **verdaderos negativos** son aquellas observaciones pertenecientes a la clase del cero clasificadas como tal y los **verdaderos positivos** aquellas pertenecientes a la clase del uno clasificadas como tal.

Teniendo en mente estas definiciones, se definen las siguientes métricas:

2.1.6.1. Precisión o accuracy

Esta cantidad mide la proporción de observaciones que han sido clasificadas correctamente. Su fórmula viene dada por:

$$\text{ACC (Accuracy)} = \frac{\text{VP+VN}}{\text{VP+VN+FP+FN}}$$

2.1.6.2. Sensibilidad

También conocida como la tasa de verdaderos positivos. Esta cantidad mide la proporción de casos positivos que han sido correctamente clasificados. Su fórmula viene dada por:

$$\text{VPR (Sensitivity)} = \frac{\text{VP}}{\text{VP+FN}}$$

2.1.6.3. Especificidad

También conocida como la tasa de verdaderos negativos. Esta cantidad mide la proporción de casos negativos que han sido correctamente clasificados. Su fórmula viene dada por:

$$\text{SPC (Especificity)} = \frac{\text{VN}}{\text{VN+FP}}$$

2.1.6.4. Curva ROC y área bajo la curva

La curva ROC poblacional representa 1-especificidad frente a la sensibilidad para cada posible valor umbral o punto de corte en la escala de resultados de la prueba de estudio. Se aproxima por la curva ROC muestral, que representa la fracción de falsos positivos en abscisas frente a la fracción de verdaderos positivos en ordenadas. Como se están representando las probabilidades, la curva ROC resultante estará en el espacio [0,1] $\times$ [0,1]. El mejor punto posible se situaría en la esquina superior izquierda, en el punto (0,1), indicando 100 % de sensibilidad y 100 % de especificidad. A este punto se le llama una clasificación perfecta y, por el contrario, una clasificación totalmente aleatoria sería la que proporciona un punto a lo largo de la línea $y = x$.

Para interpretar la curva ROC se suele usar el área bajo la curva, abreviada como AUC. Como la curva ROC se representa en el espacio [0,1] $\times$ [0,1] el valor máximo que puede alcanzar este área es uno y el menor es cero. La clasificación es mejor cuanto mayor sea esta cantidad, considerándose normalmente buena cuando el área es superior a 0.75.

2.1.7. Sobremuestreo

Las técnicas de sobremuestreo se usan para ajustar la distribución de un conjunto de datos, compensando el desequilibrio presente en los datos mediante la generación de más muestras. Este desequilibrio puede ser debido a la falta de representación de una clase en una o más variables predictoras o en la variable objetivo y se soluciona añadiendo ejemplos a la clase minoritaria.

A continuación, se describen las técnicas de sobremuestreo empleadas en el estudio:

2.1.7.1. Sobremuestreo aleatorio

Esta técnica aumenta los datos pertenecientes a la clase minoritaria bien duplicando los datos ya existentes o bien eligiendo algunos de ellos al azar con reemplazo.

2.1.7.2. Algoritmo ROSE

Este algoritmo genera nuevos ejemplos artificiales para las clases, usando *bootstrap*.

Para generar un nuevo ejemplo artificial, el procedimiento ROSE selecciona una observación del conjunto de entrenamiento, la cual puede pertenecer a la clase mayoritaria o minoritaria, con una cierta probabilidad y genera una nueva muestra en su vecino. Combina por tanto técnicas de *downsampling* y *upsampling* generando una muestra de datos superior, incrementándose especialmente la clase minoritaria. El nivel de equilibrio en ambas clases está determinado por la probabilidad con la que selecciona una observación de una u otra clase del conjunto de entrenamiento. Si esta probabilidad es 1/2, tendremos aproximadamente el mismo número de muestras para cada categoría.

2.1.7.3. Algoritmo SMOTE

Este algoritmo sobremuestrea la clase minoritaria, creando muestras sintéticas en lugar de utilizar muestreo con reemplazamiento.

El método selecciona una observación al azar de la clase minoritaria, y determina los k vecinos más cercanos de la misma. La nueva observación será una combinación aleatoria de los predictores de la observación seleccionada al azar y de los k vecinos más cercanos.

2.2. Funciones y paquetes utilizadas

2.2.1. Tidyverse

Tidyverse es una colección de paquetes de R diseñados para Data Science. Algunos de los paquetes integrados en la colección son *ggplot2*, *dplyr*, *tidyr*, *readr*, *purr*, *tibble*, *stringr* o *forcats*. Las funciones usadas en el estudio se detallan a continuación, agrupadas según su funcionalidad.

2.2.1.1. Reorganizar los datos

- **dplyr::arrange(data, ...)**: ordena los datos según las columnas señaladas. Por defecto el orden es ascendente. Para ordenarlas en orden descendente usar *desc()*.
- **tidyr::extract(data, col, into, regex, ...)**: dada una expresión regular, transforma una columna en varias asignando el valor obtenido mediante dicha expresión.
- **tidyr::gather(data, key, value, ...)**: cambia las columnas indicadas a formato largo, esto es, pasa sus etiquetas a la columna *key* y sus valores a la columna *value*.
- **tidytable::pivot_longer(data, cols, names_to = ..., values_to = ..., ...)**: cambia las columnas indicadas en *names_to* a formato largo, incluyendo sus valores en la columna indicada en *values_to*. Es sucesora de la función *gather()*, diseñada para ser más fácil de usar y para poder ser empleada en más ocasiones.
- **tidytable::pivot_wider(data, cols, names_from = ..., values_from = ..., ...)**: incrementa el número de columnas y disminuye el de filas. En *names_from* y *values_from* se encuentran los nombres de las nuevas columnas y sus valores, respectivamente. Es sucesora de la función *spread()*, diseñada para ser más fácil de usar y para poder ser empleada en más ocasiones.
- **dplyr::rename(data, ...)**: renombra las columnas señaladas. Su sintaxis es *nombre_nuevo = nombre_antiguo*.
- **tidyr::spread(data, key, value, ...)**: realiza lo contrario a *gather*. Dadas dos columnas del conjunto de datos, *key* y *value*, crea nuevas columnas nombrandolas con los valores de *key* y usando los valores de *value*.
- **tidyr::separate(data, col, into, sep, ...)**: separa la columna señalada en *col* en varias siguiendo el patrón indicado en *sep*.

- **tidyr::unite(data, col, sep, ...)**: une las columnas señaladas en una sola, cuyo nombre se indica en *col*. La unión se realiza fusionando los valores de las columnas seleccionadas, separándolos mediante el separador señalado en *sep*.

2.2.1.2. Subconjuntos de observaciones

- **dplyr::filter(data, ...)**: extrae las filas que cumplen una determinada condición lógica.
- **dplyr::select(data, ...)**: selecciona las columnas mediante su nombre o una función auxiliar.
- **dplyr::top_n(data, n, ...)**: selecciona y ordena los n mejores valores en función de la variable señalada. Si los datos están agrupados, realiza la selección por grupos.

2.2.1.3. Agrupar datos

- **dplyr::group_by(data, by = cols, ...)**: agrupa los datos en filas con el mismo valor que la variable dada.
- **dplyr::summarise(data, ...)**: se suele usar después de la función *group_by*, aplicando una función (*sd*, *mean*, *count*, ...) a los valores de cada grupo resultante.
- **dplyr::count(data, cols, wt, ...)**: esta función realiza en un solo paso *group_by* y *summarise*, bien contando o sumando según se use o no el parámetro *wt*.

2.2.1.4. Nuevas variables

- **dplyr::mutate(data, ...)**: crea nuevas variables o modifica alguna ya existente a través de una formulación dada.
- **dplyr::transmute(data, ...)**: funciona exactamente igual que *mutate*, exceptuando que se queda únicamente con las columnas creadas, borrando las originales.

2.2.1.5. Combinar datasets

- **tidytable::full_join(a, b, by = ..., ...)**: devuelve todas las filas de ambos conjuntos de datos. La coincidencia se determina a nivel de las columnas coincidentes en nombre o bien por las explícitamente establecidas en el *by*.
- **tidytable::inner_join(a, b, by = ..., ...)**: devuelve todas las filas donde hay coincidencia en ambos conjuntos de datos. La coincidencia se determina a nivel de las columnas coincidentes en nombre o bien por las explícitamente establecidas en el *by*.
- **tidytable::left_join(a, b, by = ..., ...)**: devuelve todas las filas del conjunto de datos a junto a las filas coincidentes del conjunto de datos b. La coincidencia se determina a nivel de las columnas coincidentes en nombre o bien por las explícitamente establecidas en el *by*.

- **tidytable::right_join(a, b, by = ..., ...)**: devuelve todas las filas del conjunto de datos b junto a las filas coincidentes del conjunto de datos a. La coincidencia se determina a nivel de las columnas coincidentes en nombre o bien por las explícitamente establecidas en el *by*.

2.2.1.6. Resumir variables

- **mean()**: calcula la media de las observaciones.
- **sd()**: calcula la desviación típica de las observaciones.
- **var()**: calcula la varianza de las observaciones.

2.2.2. Tidytext

El ecosistema *tidytext* une diferentes paquetes con objeto de agrupar métodos para visualizar y tratar texto, mediante el uso de principios de datos ordenados.

La función que se ha usado de este paquete es **unnest_tokens(data, output, input, token, n, ...)**. En el argumento *data* se indica el conjunto de datos, y en input la columna del conjunto a la que se quiere aplicar la función. El argumento *output* indica el nombre de la nueva columna creada conteniendo la salida de la función. En el argumento *token* se indica la unidad a la que se que se quiere *tokenizar*, es decir, la unidad lingüística por la que se quiere separar el texto en secuencias. Esta unidad puede ser palabras, caracteres u oraciones, aunque también se puede emplear una función personalizada. En el estudio se ha usado la función ya integrada *ngrams*, que recibiendo un *n* determinado calcula los n-gramas del texto proporcionado.

2.2.3. Tidymodels

Tidymodels es una colección de paquetes de R diseñados para el Machine Learning. Algunos de los paquetes integrados son *rsample*, *parsnip*, *recipes* o *tune*, cada uno dotado de una funcionalidad específica. A continuación, se detallan los servicios que ofrece cada una de estos paquetes, así como algunas de las funciones que los integran.

2.2.3.1. Paquete rsample

El paquete *rsample* contiene un conjunto de funciones destinadas a aplicar las diferentes técnicas de muestreo estadístico.

Las dos funciones que se han usado de este paquete son *initial_split* y *vfold_cv*, y se detallan a continuación:

- **initial_split(data, strata, prop, ...)**: esta función se usa para particionar los datos en los subconjuntos de entrenamiento y prueba (también llamado conjunto *test*), los cuales se obtienen posteriormente mediante las funciones *training* y *testing*.

En *prop* se indica la proporción de los datos destinada al conjunto de entrenamiento, mientras que en *strata* se precisa la variable por la que se realiza el muestreo estratificado.

- **vfold_cv(data, strata, v, ...)**: esta función se usa para realizar el procedimiento de validación cruzada con v pliegues, el cual divide el conjunto de datos en v grupos de aproximadamente el mismo tamaño. El argumento v indica el número de pliegues, mientras que *strata* señala la variable por la que se realiza el muestreo estratificado.

2.2.3.2. Paquetes recipes y themis

El paquete *recipes* contiene el conjunto de funciones destinadas a realizar las codificaciones y el preprocesamiento de los datos, esto es, *ingeniería de características*.

La función principal de este paquete es *recipe*, que es la que crea la receta. Una receta no es más que la descripción de los procedimientos que se deben aplicar a los datos para tenerlos preparados para el análisis. Su sintaxis es **recipe(formula, data, ...)**, recibiendo la fórmula del modelo y el conjunto de datos al que aplicarla.

Tras definir la receta, se pueden ir aplicando los diferentes pasos para tener los datos preparados. Una función útil es **update_role(recipe, variables, new_role, ...)** que permite alterar el rol de una variable.

Las siguientes funciones más importantes usadas en la receta son las denominadas funciones *step*, pues cada una de ellas sirve para implementar algún paso del preprocesamiento de los datos. A continuación se exponen las que han sido usadas en el estudio, teniendo presente que hay muchas más:

- **step_dummy(recipe, ...)**: esta función permite transformar una variable categórica en las variables dummy correspondientes, es decir, un *one-hot-encoding* posiblemente suprimiendo una de las nuevas columnas correspondientes a una categoría, para evitar colinealidad.
- **step_normalize(recipe, ...)**: normaliza los datos para que tengan desviación típica uno y media cero.
- **step_pca(recipe, ...)**: esta función aplica el análisis de componentes principales a las variables señaladas, y se queda con el número de componentes principales que le indiquemos.

Por su parte, el paquete *themis* complementa el paquete *recipes*, recogiendo funciones *step* para trabajar con datos desbalanceados:

- **step_rose(recipe, ...)**: esta función aplica el algoritmo ROSE de sobremuestreo detallado en la sección 7.3, tomando como variable seleccionada para muestrear los datos la indicada.
- **step_smote(recipe, ...)**: esta función aplica el algoritmo SMOTE de sobremuestreo detallado en la sección 7.4, tomando como variable seleccionada para muestrear los datos la indicada.

- **step_update(recipe, ...)**: esta función aplica el algoritmo de sobremuestreo aleatoria detallado en la sección 7.5, tomando como variable seleccionada para muestrear los datos la indicada.

Cabe destacar también el uso de las funciones *prep()*, *bake()* y *juice()*. La función *prep()* toma el objeto que define la receta y calcula todo para que se puedan ejecutar los pasos de preprocesamiento. La función *bake()* toma una receta preparada y la aplica al conjunto de datos que se indique. Por su parte la función *juice()* realiza la misma labor que la función *bake()* aplicada a los datos de entrenamiento. Estas funciones suelen emplearse automáticamente dentro de las funciones de ajuste como *last_fit*, pero además pueden aplicarse a solas para ver el fruto de su cálculo o aplicación a los conjuntos de entrenamiento y/o prueba.

2.2.3.3. Paquete parsnip

El paquete *parsnip* contiene el conjunto de funciones dedicadas a construir los diferentes modelos estadísticos.

La sintaxis es siempre la misma: se da el **tipo** de modelo, seguido del **modo** (regresión, clasificación, etc) y del **motor** computacional, que es el nombre del paquete de R.

El modo se da mediante la función *set_mode()* y el motor mediante *set_engine()*. Por otro lado, para definir el tipo de modelo nos encontramos varias funciones dependiendo del tipo de modelo que se quiera implementar. Las funciones usadas para los diferentes modelos implementados en el estudio son:

- **linear_reg()**: para regresión lineal.
- **rand_forest()**: para Random Forest.
- **logistic_reg()**: para regresión logística.

Además de permitirnos definir los modelos, mediante a función **fit(data, ...)** podemos ajustarlos utilizando los datos que se indiquen, por ejemplo, los del conjunto de entrenamiento, y mediante la función **predict(data, ...)** de este paquete se pueden realizar las predicciones en los datos deseados, como por ejemplo, los del conjunto de prueba.

2.2.3.4. Paquete workflow

El paquete *workflow* sirve para implementar *workflows* o flujos de trabajo, permitiendo fusionar los modelos creados y las recetas, creando así una interfaz unificada.

2.2.3.5. Paquete tune

El paquete *tune* facilita el ajuste de hiperparámetros. Su función principal es *tune()*, que aplicada en el parámetro sobre el que se desea realizar el contraste facilita el ajuste antes mencionado.

La función **tune_grid(object, preprocessor, resamples, ...)** calcula una serie de medidas de rendimiento para un conjunto predefinido de parámetros de ajuste correspondientes a una receta o modelo dado. A través de estas medidas se seleccionan los parámetros que mejor ajustan los datos.

La función **fit_resamples()** calcula un conjunto de métricas de rendimiento en una o más muestras. En este estudio se ha utilizado para calcular dichas métricas sobre los conjuntos obtenidos mediante validación cruzada. Esta función ha sido usada junto a la función *registerDoParallel()* del paquete *doParallel*, que aprovecha el paralelismo disponible en nuestro equipo y gana eficiencia, haciendo que cada plegado del *fit_resamples* sea independiente y puediéndose así ejecutar en un proceso o hilo independiente.

La función **last_fit()** recibe un *workflow* y el conjunto particionado en entrenamiento/test obtenido mediante *initial_split* y entrena automáticamente el modelo especificado en el *workflow* usando los datos de entrenamiento, y produciendo valoraciones basadas en el conjunto de prueba.

2.2.3.6. Paquete yardstick

El paquete *yardstick* contiene las funciones necesarias para estimar la funcionalidad de los modelos. Algunas de las funciones más importantes son **metrics(data, truth, estimate, ...)** que, dadas las estimaciones y los valores verdaderos, devuelve las medidas de bondad de ajuste que se le indiquen.

Parte II

Estudio global

En este bloque se analiza el estado de las plataformas de streaming en la actualidad, estudiando tanto su situación presente a nivel global y nacional, como su comportamiento en diferentes países, comparando su contenido, precio y calidad, mediante el uso exhaustivo de difrentes técnicas de análisis y ciencia de datos. Todos los análisis y estudios aquí presentados son el fruto de los resultados obtenidos a partir de todas las fuentes de datos que se han podido encontrar tras la búsqueda exhaustiva de otros posibles orígenes de datos.

Capítulo 3

Análisis del panorama actual

3.1. Estudio a nivel mundial

A pesar de que el éxito de las compañías es notorio, la realidad es que la información que corrobora esta prosperidad, como el número de suscriptores que poseen o los ingresos de los que disponen, no siempre es accesible al público: las empresas muestran cierta reticencia a la hora de compartir sus cifras. El caso de Netflix es quizás el más transparente de todos, pues los informes financieros de la empresa, accesibles al público, contienen tanto los ingresos como el número de suscripciones (Netflix [2010-2020a]). Obtener las cifras de otras compañías es, en cambio, menos simple.

En los siguientes apartados estudiaremos la evolución de cuatro de las compañías de streaming de vídeo más notables: Netflix, HBO, Amazon Prime Video y Disney+, analizando el número de suscriptores e ingresos que han obtenido en los últimos años, así como intentando predecir su comportamiento a largo plazo.

3.1.1. Suscripciones

El número de suscripciones que poseen estas plataformas no es fácil de conseguir: algunas empresas consideran esta cifra como su mejor secreto. Nos encontramos con todo tipo de casos, desde empresas que publican abiertamente estos datos, pasando por otras que lo hacen esporádicamente, hasta algunas cuya cifra de suscriptores sigue siendo un misterio. Por ejemplo, Netflix da acceso a estos datos de forma sencilla, tal y como se ha comentado anteriormente, pero en otras empresas como HBO y Amazon es más complicado: HBO no tiene todas sus cifras disponibles para el público, y en Amazon estas son casi imposibles de encontrar. A continuación, se va a analizar esta cifra, en la medida de lo posible, para cada una de estas compañías.

En primer lugar, notemos que el número de suscriptores de Netflix a nivel mundial a lo largo de los años, está, por tanto, disponible (Netflix [2010-2020b], Statista [2011-2020], Statista [2013-2020]). Esta misma cifra para HBO solo ha sido localizada hasta el año 2017. HBO era propiedad de Time Warner, que en sus informes financieros facilitaba el número de suscriptores de la plataforma (Warner [2017]). En el año 2018 Time Warner fue adquirida por AT&T, la compañía de telecomunicaciones más grande del mundo, cambiando su nombre a Warner Bros, y pasando por tanto, HBO, al control de esta

nueva empresa. Desde entonces, los informes financieros no los realiza Warner Bros, sino la propia AT&T, la cual no difunde el número de suscriptores a nivel mundial de la compañía, facilitando solo y eventualmente, los de Estados Unidos. A todo esto hay que añadir el comportamiento especial de HBO, ya que, al contrario que el resto de compañías involucradas en el estudio, en algunos países combina su catálogo en servicio de streaming con la retransmisión en directo.

Posteriormente se muestra un gráfico con la evolución del número de suscriptores en estas dos compañías. Antes bien, se va a hacer hincapié en algunos de los pasos desempeñados en el pretratamiento de los datos, llevado a cabo en vista de poder realizar correctamente la representación.

Primeramente, se debe mencionar que los datos de Netflix han sido obtenidos de dos archivos diferentes. Ambos proporcionan los datos por trimestres. El primero de ellos contiene los datos relativos a los suscriptores de la compañía desde el año 2011 hasta el segundo trimestre del año 2020, mientras que el segundo contiene las mismas cifras, pero comenzando en 2013 y teniendo 2020 completo. Estos dos datasets son muy similares, diferenciándose únicamente en el formato en el que la fecha está guardada. Por ejemplo, si quisiéramos acceder a los datos del segundo cuatrimestre de 2016, en el primer fichero habría que buscar "T2 16" mientras que en el segundo habría que buscar "Q2 2016".

Con objeto de fusionar estos dos conjuntos correctamente, se ha cambiado el formato de la fecha en el primer dataset, empleando sobre esta variable la función *mutate* del paquete tidyverse (ver 2.2.1.4) y la función *gsub*, la cual dada una cadena de caracteres, una expresión regular y un patrón, busca coincidencias de la expresión regular en la cadena de caracteres y las cambia por el patrón. En este caso usaremos *mutate* para modificar la variable fecha, y la corregiremos aplicando *gsub*: donde ponga T pondremos Q, y donde haya espacio, es decir, antes de los números, pondremos "20" para así tener los años comenzando por esta cifra. Tras esto, los conjuntos estan listos para ser unidos. Seguidamente, se muestra el código utilizado:

```
netflix_suscriptores_mundial =
  read_xlsx("datos/Netflix-suscriptores-mundial-11-20.xlsx",
            sheet = 2,
            col_names = c("Fecha", "Numero"))[-c(1:3),] %>%
  mutate(Fecha = gsub(Fecha, pattern = "T", replacement = "Q")) %>%
  mutate(Fecha = gsub(Fecha, pattern = " ", replacement = " 20")) %>%
  full_join(read_xlsx("datos/Netflix-suscriptores-mundial-13-20.xlsx",
                      sheet = 2,
                      col_names = c("Fecha", "Numero"))[-c(1:3),])
```

A continuación, se lee el conjunto de datos relativo a HBO, el cual está dado en años, para, por último, combinarlos a fin de poder hacer adecuadamente la representación. Con objeto de poder efectuar la gráfica con el paquete *ggplot*, lo más sencillo es tener un dataframe con tres variables: una indicando la fecha, otra proporcionando el número de suscriptores y una última, variable factor, que indique si dicha cifra pertenece a Netflix o a HBO. Después de esto, tendremos un conjunto listo para ser representado con ggplot, ubicando en el eje de abscisas la variable fecha, en el de ordenadas la cifra de suscriptores, y diferenciando por dos grupos, correspondientes a las dos plataformas.

Inicialmente nos quedaremos con aquellos datos del dataset de Netflix que pertenezcan al cuarto trimestre, para poder fusionar correctamente por años. Para ello, usaremos la función *filter* (ver 2.2.1.2) y la función *substr*: con *substr* substraeremos de cada fecha los dos primeros caracteres, que indican el trimestre: si este coincide con el cuarto, gracias al *filter*, nos quedamos con esa observación. Una vez que tengamos todos los datos de Netflix relativos al cuarto trimestre, en la variable fecha nos quedamos únicamente con el año, usando de nuevo *substr*, para tomar los últimos cuatro dígitos. Tras esto estaremos en condiciones de fusionar ambos datasets, lo que nos dará un dataframe con tres columnas: una columna relativa al año, otra con el número de suscripciones de Netflix y otra con el número de suscripciones de HBO. Aún así, aún no está en el formato deseado, hay que usar la función *gather*: esta función toma varias columnas y las une en pares clave-valor. Se adjunta el código y el resultado obtenido:

```
suscriptores_nivel_mundial =
  netflix_suscriptores_mundial %>%
  filter(substr(Fecha, start = 1, stop = 2) == "Q4") %>%
  mutate(Fecha = substr(Fecha, start = 4, stop = 7)) %>%
  full_join(hbo_suscriptores_mundial, by = c("Fecha" = "Año")) %>%
  rename(`Netflix` = Numero.x, `HBO` = Numero.y, `Año` = Fecha) %>%
  gather(key = "Plataforma", value = "Suscriptores", -Año) %>%
  mutate(Plataforma = factor(Plataforma, levels = c("Netflix", "HBO")))
```

Tabla 3.1: Extracto del número de suscriptores de Netflix y HBO a nivel mundial.

Año	Plataforma	Suscriptores
2011	Netflix	21.60
2011	HBO	93.00
2012	Netflix	30.36
2012	HBO	114.00
2013	Netflix	41.43
2013	HBO	127.00

En último término, mostramos el gráfico obtenido a partir de estos datos:

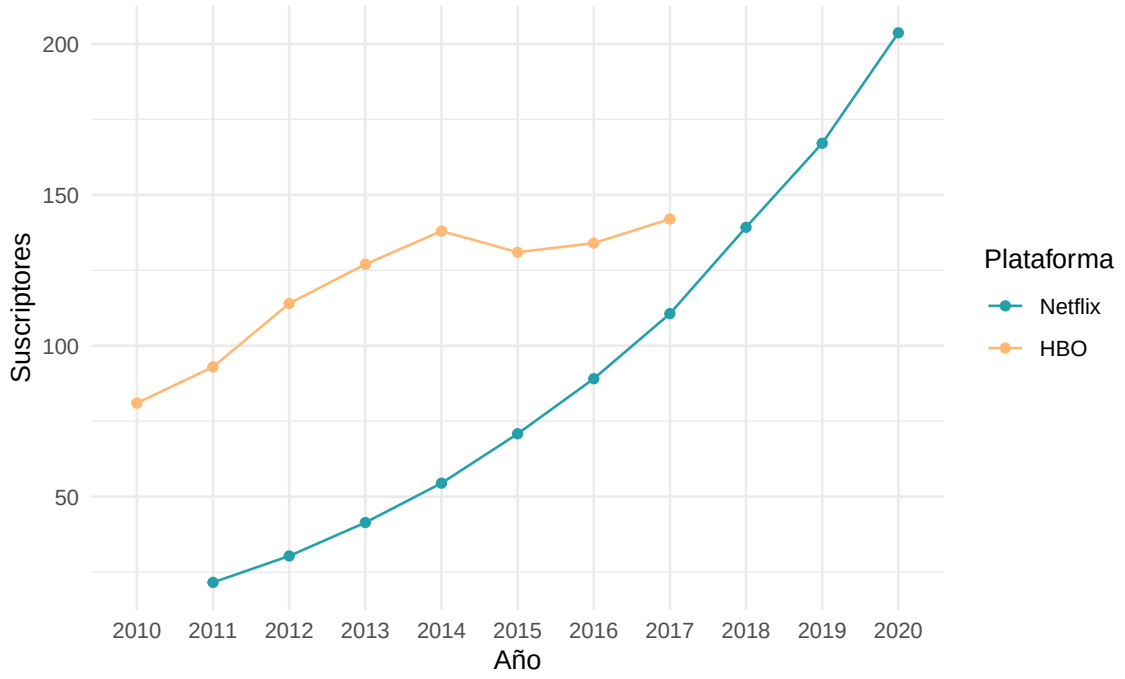


Figura 3.1: Evolución del número de suscriptores de Netflix y HBO a nivel mundial.

Como se puede observar, la evolución del número de suscriptores de Netflix ha sido creciente desde que comenzó como una simple compañía de distribución de DVD en 2007. No obstante, durante sus primeros años en activo, el número de suscriptores de HBO era superior, siendo imposible hacer una comparación en los últimos debido a la escasez de información. Esta última perdió suscriptores en el año 2014, bajada de la que se recuperó en 2017. En 2018 Netflix casi iguala el número de suscriptores que tenía HBO el año anterior, pudiendo, en la actualidad, ir en cabeza. Sin embargo, no disponemos de información suficiente para corroborar este hecho, que en las secciones posteriores intentará deducirse de alguna forma.

El caso de Amazon Prime Video es más especial. La suscripción a esta plataforma de streaming puede hacerse de dos modos: de forma directa, o a través de la suscripción a Amazon Prime. Eso sí, depende del país, no pudiéndose optar en algunos a la suscripción de manera directa. El problema que esto acarrea es que no se sabe verdaderamente el número de personas que usan Amazon Prime Video, ya que es imposible saber cuántos de los suscriptores de Amazon Prime utilizan la plataforma de vídeo. Por si fuera poco, ni siquiera se tiene acceso al número de clientes de Amazon Prime: el CEO de la compañía, Jeff Bezos, reveló por primera vez en 2018 que más de cien millones de personas pagaban esta suscripción en todo el mundo, pero no proporcionó cifras exactas (Spangler [2018], Bezos [2018]). Por tanto, ha sido imposible encontrar el número de suscriptores a nivel mundial para este trabajo.

Finalmente, analicemos la compañía más reciente: Disney+. No es de extrañar que Disney, una empresa de entretenimiento con semejante trayectoria y presencia en todo el mundo, se sumara al mundo de los servicios de streaming bajo demanda, apostando en Noviembre de 2019 por Disney Plus. La empresa ha proporcionado el número de

suscriptores a la plataforma por cuatrimestre a nivel mundial, los cuáles se han recogido de forma manual (Disney). Ahora bien, al ser la plataforma tan reciente, con menos de dos años de vida, no se ha considerado adecuado representarla en el gráfico anterior, realizando su representación por separado:

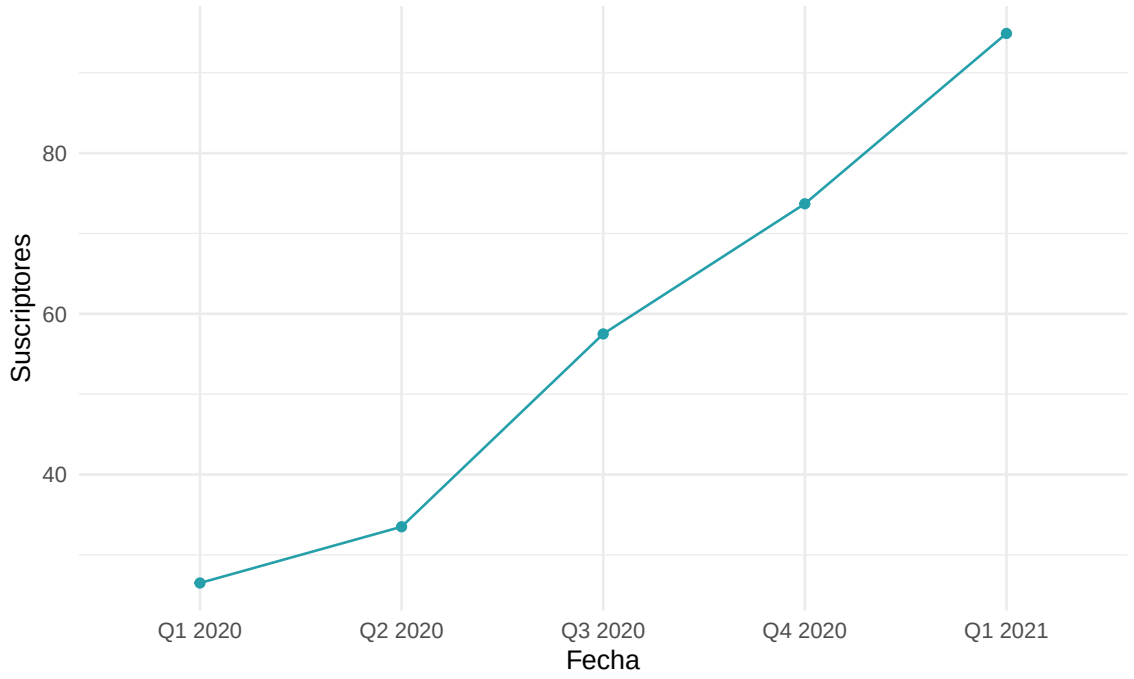


Figura 3.2: Evolución del número de suscriptores de Disney+ a nivel mundial.

Observamos como, apenas después de un año de su creación, Disney+ cuenta con casi noventa millones de suscriptores, que se espera, observando la tendencia trimestral y atendiendo a los contenidos previstos, que no hagan más crecer.

3.1.2. Ingresos

Si encontrar el número de suscriptores de estas plataformas no es sencillo, encontrar los ingresos no lo iba a ser más. Como se ha manifestado anteriormente, en los informes financieros de Netflix nos encontramos con los ingresos de la compañía a lo largo de los años. El año anterior, en 2020, estos fueron de casi veinticinco millones de dólares estadounidenses.

A diferencia del número de suscriptores, para HBO los ingresos si están disponibles al público hasta el año 2020 (Netflix [2010-2020b]). Antes de 2018, los ingresos quedaban recogidos en los informes de Time Warner (Warner [2017]). A partir de dicho año, debido a la adquisición de Time Warner por AT&T, tal y como se ha indicado con anterioridad, los ingresos se hallan en los informes financieros de esta compañía (ATT).

Comparamos los ingresos de manera gráfica:

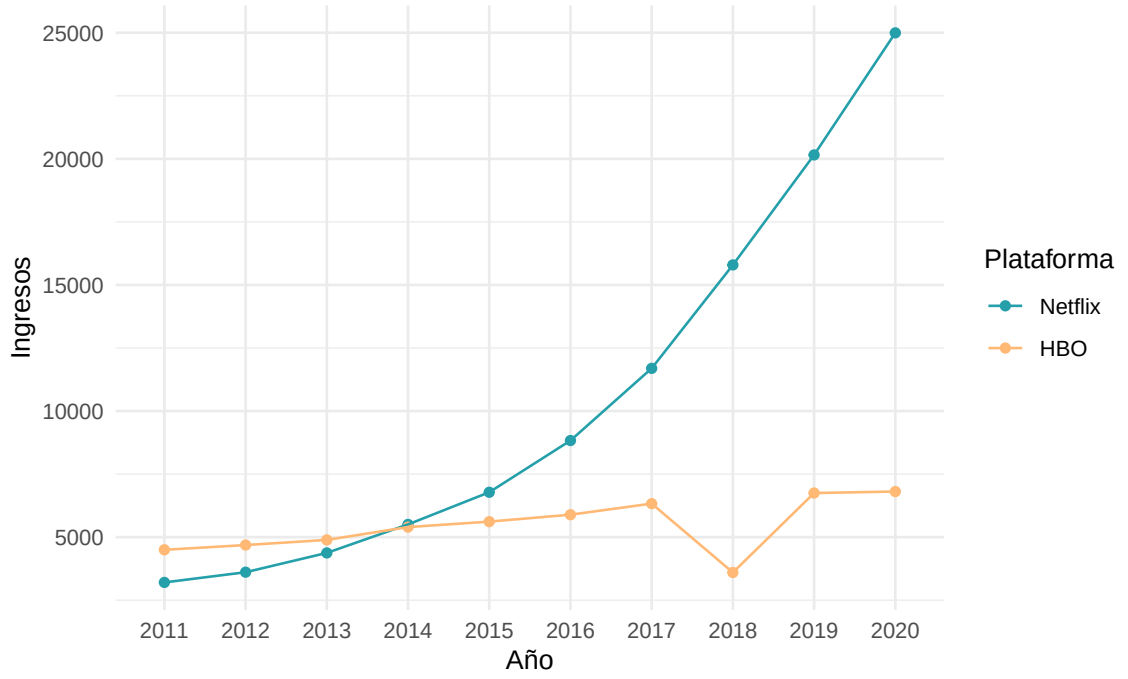


Figura 3.3: Evolución de los ingresos de Netflix y HBO a nivel mundial.

Se aprecia como, hasta el año 2014, los ingresos de HBO superaban ligeramente los de Netflix. Es a partir de ese año cuando los ingresos de este último comienzan a incrementarse considerablemente, distando en la actualidad desmesuradamente de los de HBO y, continuando, a día de hoy, aumentando.

En contraste con la evolución creciente y continua de los ingresos de Netflix, nos encontramos los de HBO: estos aumentan de un modo más suave, a pesar del descenso notable en el año 2018. En este año la compañía tuvo la mitad de los ingresos del año anterior, pasando de más de seis mil millones de dólares estadounidenses a tres mil millones y medio. La explicación mas razonable a este hecho es que ese año la empresa no emitió, por primera vez en siete años, temporada de Juego de Tronos, serie original de la plataforma que se había ganado un gran número de adeptos por todo el mundo y que dio un gran impulso a la compañía.

Con respecto a Amazon Prime Video y Disney+ nos encontramos ante la misma situación: las compañías propietarias de estos dos servicios, Amazon y Disney, en sus informes financieros no desglosan los ingresos, mostrando, y no siempre, únicamente los ingresos globales. Por tanto, en algunos casos podemos acceder a los ingresos globales de las propietarias, pero no podemos determinar que parte de ellos se corresponden a las plataformas de interés.

3.2. Estudio a nivel nacional

Un punto fundamental a tener en cuenta es el papel de España en este marco. Las plataformas de streaming han tenido también un importante impacto en el país, cambiando

la forma en la que consumimos series y películas. Estos servicios también han tenido un efecto significativo en la economía, un ejemplo claro lo encontramos en los videoclubes: a pesar del éxito que experimentaron a principios de siglo, el nacimiento de estas plataformas (entre otros aspectos, ya que no podemos olvidar el impacto de las descargas - tanto lícitas como ilícitas - de contenidos digitales) ha hecho que, hoy en día, estén casi obsoletos.

Para analizar el comportamiento de la población española respecto a estos servicios, se van a usar, entre otros, los resultados de la encuesta realizada por Statista acerca de los hábitos de consumo de la población, denominada *Global Consumer Survey* (Statista [2018-2020]). Statista, portal de estadística en línea alemán, realiza esta encuesta dos veces al año desde el 2018, incluyendo más de cincuenta y cinco países, entre ellos España. En el último año la encuesta fue respondida por más de doce mil quinientos españoles, cifra que, aunque dista de equivaler a la población total, podemos considerar representativa. Se puede acceder a los resultados de la misma desde su página oficial, ofreciendo la posibilidad de desagregar por países y categorías, siendo "Televisión y vídeo bajo demanda" la que nos atañe. Así pues, se incluyen a continuación algunas de las preguntas y respuestas más relevantes.

Desde el 2019 se ha realizado la misma pregunta a los encuestados: si son consumidores o no de alguna plataforma de streaming de vídeo. Acto seguido, mostramos la evolución de las proporciones obtenidas, comparando el primer y último año:

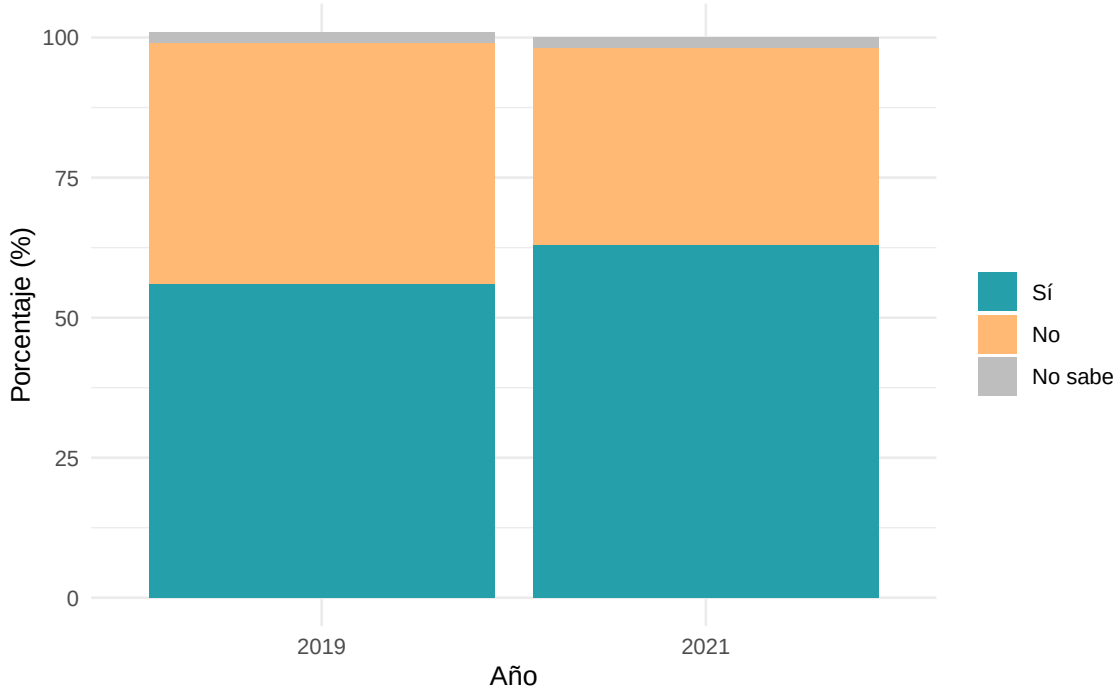


Figura 3.4: Proporción de españoles consumidores de plataformas de streaming.

El porcentaje de consumidores de los servicios de streaming ha aumentado levemente en los dos últimos años, representando ya en el 2019 a más de la mitad de la pobla-

ción. Actualmente, más del 60 % de los españoles son consumidores de alguna de estas plataformas.

Otro aspecto interesante, que queda recogido en la encuesta, es identificar los motivos principales por los que se realizan suscripciones a dichas plataformas. Para ello, el estudio incluye una pregunta de selección múltiple que indica a los encuestados que precisen las razones por las que adquieren estos servicios. Los resultados obtenidos para España fueron los siguientes:

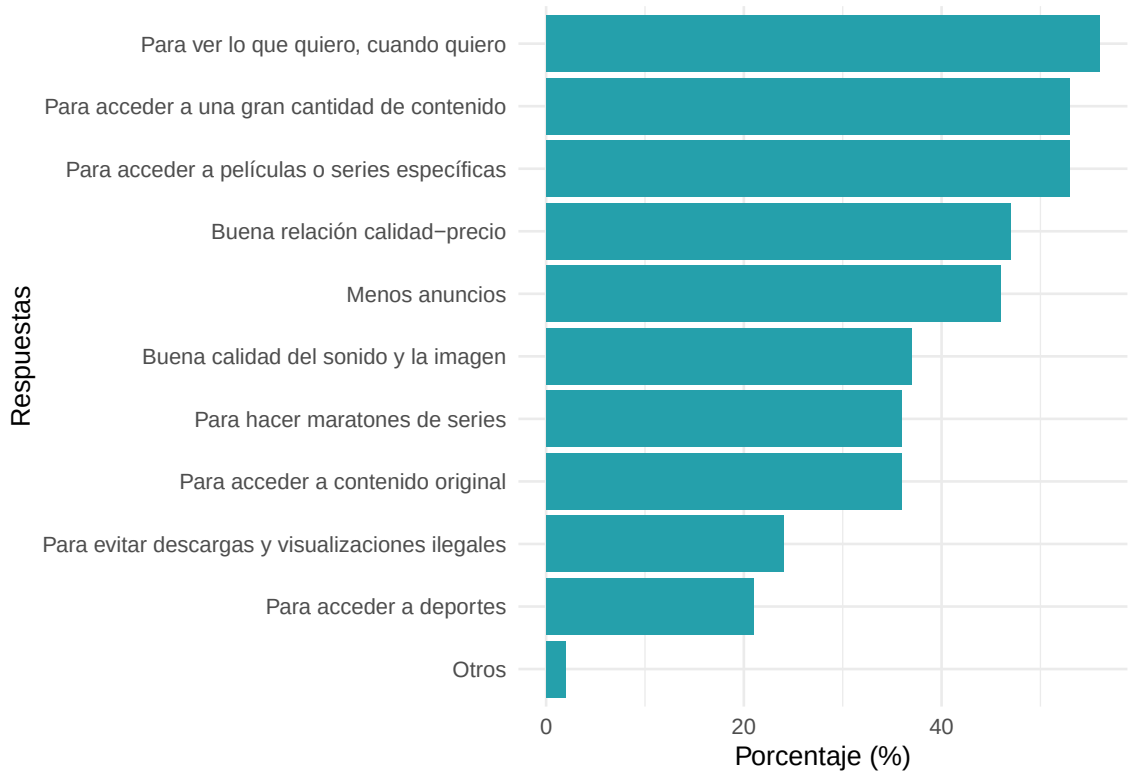


Figura 3.5: Razones principales para adquisición de servicios de streaming en España.

No es de extrañar que la causa principal por la que se adquieren estos servicios sea ver el contenido que se desee en cualquier momento. Hace apenas unos años, antes de la aparición de estas plataformas, las opciones para ver series o películas, al margen del cine o la compra de las mismas, eran limitadas: o bien a través de su adquisición en un videoclub o bien en algún canal de televisión. El principal problema de la televisión es que los títulos son fijados por la cadena, siendo imposible para el usuario escogerlos, y aún menos, determinar las horas a las que se emiten. Por el contrario, los videoclubes, solucionaban estos dos inconvenientes, permitiendo elegir el título y verlo cuando te apeteciera. Sin embargo, las plataformas de streaming son una opción mucho más asequible, siendo el precio de las suscripciones equivalente al alquiler de un par de películas; y cómoda, ya que anulan la necesidad de desplazarse.

Otros de los motivos principales son la gran cantidad de contenido que ofrecen, siendo a veces posible elegir entre más de mil películas y series diferentes; y la capacidad de poder elegir un título específico, sin depender de la disponibilidad del mismo en el videoclub o de la programación en la televisión.

Además de las cuatro plataformas mencionadas al principio del capítulo, existen muchas otras. Cabe destacar el servicio de televisión prestado por Movistar, Movistar+, que opera desde 2015, siendo la fusión de las ya extinguidas Canal+ y Movistar TV. Esta plataforma es propiedad de Telefónica, empresa consolidada en España y líder de telecomunicaciones en Europa, por lo que no es sorprendente que sea una de las principales plataformas de streaming del país. A continuación, se incluye la evolución del número de suscriptores en los últimos años (CNMCData):

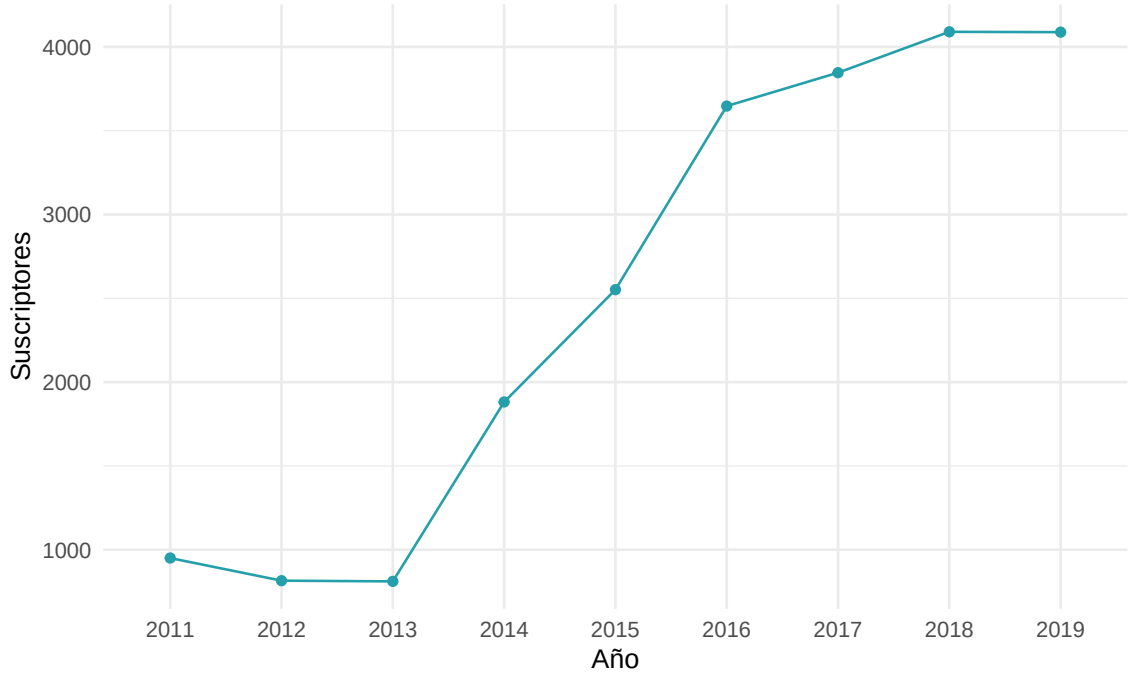


Figura 3.6: Evolución de los suscriptores de Movistar+ en España.

Observamos, teniendo en cuenta que los años anteriores a 2015 corresponden a Movistar TV, como el número de abonados al servicio creció incesantemente desde 2013 hasta 2017. A partir de ese año el número de suscriptores se mantiene más constante, elevándose sólo ligeramente, contando en 2019 con más de cuatro millones de suscriptores. Se plantea entonces la siguiente pregunta: ¿es Movistar+ la plataforma líder de streaming en España o, por el contrario, existe otra que la desbanque?

Con objeto de resolver dicha cuestión, se incluye seguidamente el número de abonados a las diferentes plataformas de streaming en España. La información disponible más reciente es del año 2020, tal y como se muestra a continuación:

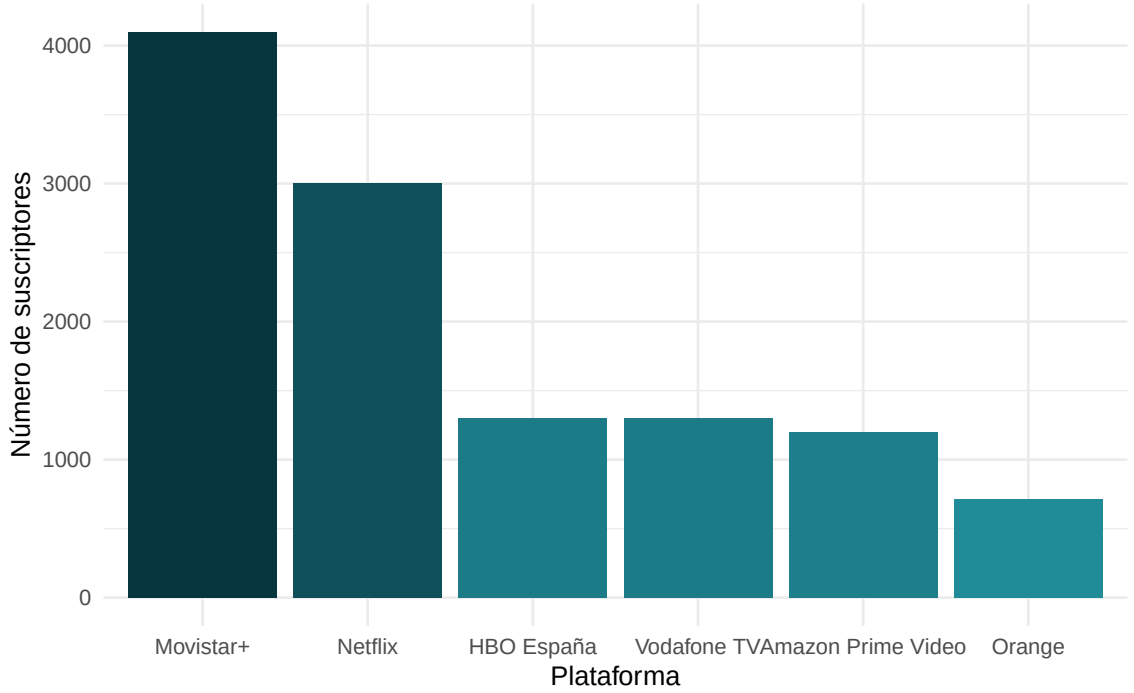


Figura 3.7: Número de suscriptores de cada plataforma en España en el año 2020.

Se advierte como, en efecto, Movistar+ es en 2020, la compañía líder en España, superando a Netflix en un millón de suscripciones. Netflix, introducida en el país en el año 2015, va adquiriendo mayor peso, acercándose a la gran compañía española y pudiendo, si no ha ocurrido ya, superar a Movistar en pocos años.

En el estudio mencionado anteriormente, se indicaba a los encuestados que seleccionaran las plataformas de las que eran usuarios. Por seguir en la misma línea comparativa, exponemos parcialmente estos resultados, disponibles desde el 2019:

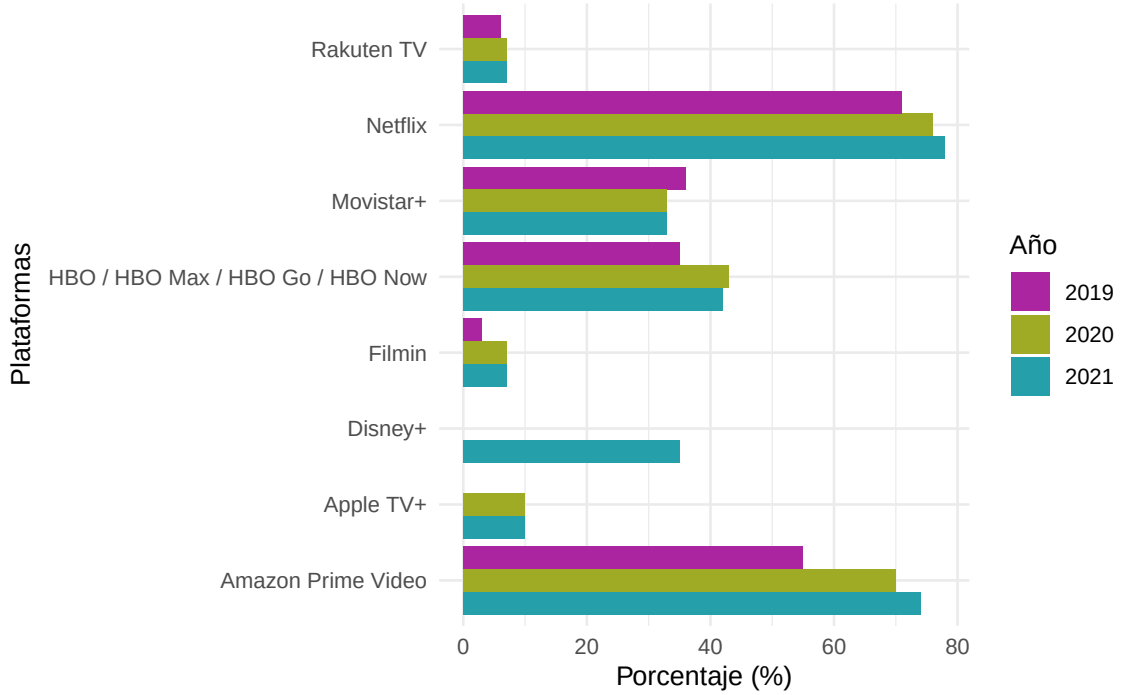


Figura 3.8: Plataformas más usadas en España.

Menos del diez por ciento de los encuestados afirma ser usuario de Filmin, Rakuten TV y/o Apple TV+. Por otra parte, el porcentaje de abonados a Movistar+ ha descendido desde el 2019, al contrario que el de Netflix y Amazon Prime Video. Mientras que la subida en Netflix ha sido moderada, la de Amazon Prime Video es más notable, incrementándose el porcentaje casi un 15 % de 2019 a 2020, a diferencia de las suscripciones a HBO y sus derivados, que han experimentado una leve bajada en el último año. Por último, Disney+ ha sido añadido en la encuesta este año, contando ya con el apoyo de casi un 40 % de los entrevistados.

Para dar por concluida esta sección, analizamos en última instancia los precios de algunas de estas plataformas en España (Perez and Mendez [2019]):

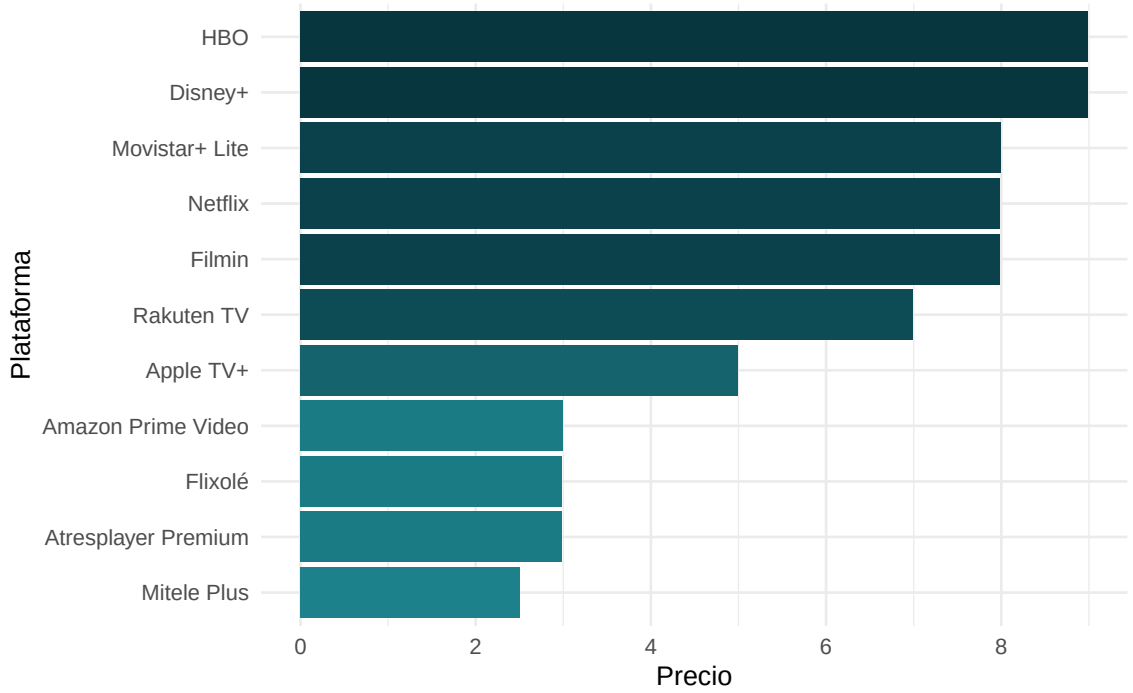


Figura 3.9: Precios por plataforma en España.

Actualmente, las plataformas más caras son HBO y Disney+. Esta última ha incrementado su precio el 23 de Febrero de este mismo año, pasando de costar 6.99 euros al mes a 8.99, subida que no ha pasado desapercibida para los suscriptores. Sin embargo, son muchos los que aún pueden disfrutar de la antigua tarifa: Disney+ ofrece una suscripción anual, por lo que los usuarios adscritos a esta antes del 23 de Febrero seguirán pagando la antigua tasa durante el año que dure su membresía.

Por otra parte, Netflix, Filmin y Movistar Lite tienen el mismo precio. Hay que tener cuidado de no confundir este último con Movistar+, cuyo precio es más elevado al poseer contenido exclusivo, como retransmisiones en directo de fútbol. Señalar que Netflix presenta diferentes tarifas en función del paquete que seleccione el usuario. En los resultados aquí presentes se ha usado el precio del plan básico (una única pantalla).

Los precios más bajos corresponden a Amazon Prime Video, Fixolé, Atresplayer Premium y Mitele Plus. Estos tres últimos no fueron incluidos en la encuesta realizada por Statista, al tener un carácter menos internacional, aunque cuentan con una gran presencia en el país contando, entre Atresplayer Premium y Mitele Plus, con más de medio millón de suscriptores en Diciembre de 2020.

Capítulo 4

Análisis del comportamiento por países

4.1. Estudio del contenido disponible

Una de las razones por las que el número de suscriptores de las plataformas de streaming se ha incrementado a lo largo de los años es el aumento del número de países que pueden acceder a ellas. Netflix, en particular, estuvo hasta 2010 únicamente disponible en Estados Unidos, año en el que lanza su servicio a Canadá. Desde entonces, el número de países en los que se puede acceder a la plataforma no ha hecho más que crecer, llegando a estar actualmente disponible en más de ciento noventa territorios.

A pesar de que en la actualidad el número de países que puede acceder a las plataformas es mucho mayor, lo cierto es que las condiciones no son las mismas en todos ellos: las diferencias van desde el precio hasta el contenido disponible. El hecho de que el precio sea diferente no parece tan sorprendente, ya que el nivel económico es diferente en cada país, y es coherente pensar que al igual que algunos productos son más caros en unos países que en otros, también lo sean las suscripciones a las plataformas de streaming. Puede parecer más insólito que el contenido cambie dependiendo del país. Las razones que hacen que esto suceda son bastante diversas: desde adaptar el servicio a las preferencias de cada territorio, hasta que los derechos de una película o serie no estén disponibles en alguna región. En cualquier caso, la realidad es que el tamaño del catálogo es significativamente diferente entre algunos países, y es algo que merece la pena analizar.

La dificultad para comparar los tamaños de estos catálogos radica en el hecho de que no es sencillo saber la cifra exacta de contenido que ofrece cada una de estas plataformas: esta cambia constantemente debido a que las compañías añaden contenido diferente cada mes e incluso cada día. Por tanto, es complicado encontrar un registro del contenido disponible en cada plataforma cada año para analizar su evolución y la tendencia que se pueda observar en la misma.

Localizar dónde está disponible una serie o película actualmente es, sin embargo, más simple. Un problema habitual entre las personas que son consumidoras de varias de estas plataformas es no saber dónde se encuentra una película o serie en concreto. Aprovechando esta cuestión, han surgido diversas páginas web que actúan como buscadores: se introduce el título de la película o serie de interés y devuelve la plataforma en la que se encuentra.

Una de estas guías de streaming es JustWatch (JustWatch). Fundada en 2014 y ubicada en diversas capitales europeas, esta página es usada cada día por una gran variedad de personas para ver dónde se encuentra el contenido que desean.

JustWatch permite ver el contenido disponible en cuarenta y seis países diferentes, repartidos por todo el mundo. Esta página no solo actúa como buscador, al seleccionar un país y una plataforma específica nos muestra una lista de todas las películas y series que contiene, así como el número exacto de cada una de ellas. Como se ha mencionado antes, esta cifra no permanece constante sino que cambia cada día. En consecuencia, utilizando esta información, se ha podido crear de forma artesanal un dataset que agrupe cada uno de estos cuarenta y seis países junto al número de contenidos disponibles en cuatro plataformas seleccionadas: Netflix, HBO, Amazon Prime Video y Disney+. El dataset hace referencia a la lista de contenidos de cada una de estas plataformas el día 11 de Abril de 2021.

Representando cada uno de estos cuarenta y seis países junto al número total de contenido disponible en cada uno de ellos, calculado mediante la suma del número de títulos en las cuatro plataformas seleccionadas, se obtiene el siguiente gráfico:

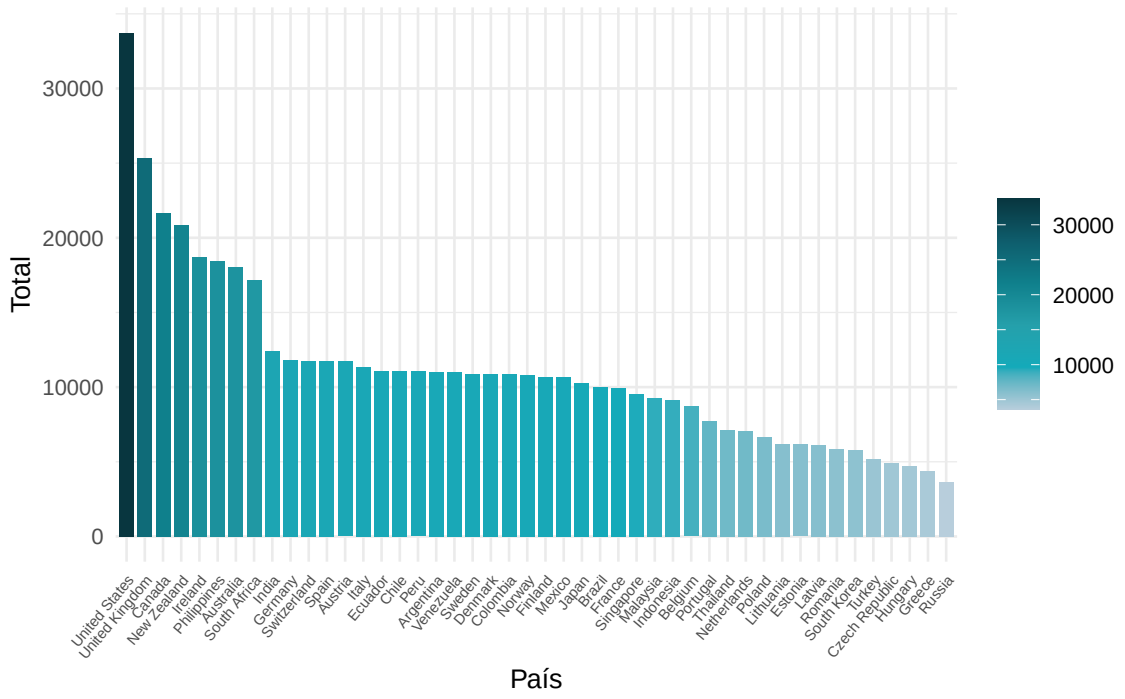


Figura 4.1: Contenido audiovisual disponible en cada país.

En él se pueden distinguir aproximadamente cuatro grupos: el primero estaría formado únicamente por Estados Unidos, cuyo catálogo es muy superior al del resto de países, disponiendo de más de treinta mil títulos; el segundo, encabezado por Reino Unido, engloba los países cuyo volumen de catálogo fluctúa entre quince mil y veinticinco mil; el tercero, donde se encuentran la mayoría de territorios, es un grupo más homogéneo que el anterior, oscilando el número de títulos en este conjunto en torno a ocho mil y doce mil, y, por último, el cuarto grupo, que estaría constituido por aquellos países que tienen

una cantidad inferior de contenido, reuniendo entre las cuatro plataformas menos de siete mil medios de entretenimiento digital.

La mayor parte de estos cuarenta y seis países se localiza por tanto en el tercer grupo. El segundo estaría formado únicamente por siete: Reino Unido, Canadá, Nueva Zelanda, Irlanda, Filipinas, Australia y Sudáfrica, en orden de extensión de catálogo. La diferencia entre este segundo grupo y el último es abismal: mientras que Reino Unido cuenta con más de veinticinco mil series y películas, Rusia entre las cuatro plataformas no reúne ni cinco mil. Podemos observar más detalladamente estas diferencias en la siguiente gráfica comparativa, mostrada junto al código que la genera:

```
datos %>%
  top_n(5, Total) %>%
  full_join(datos %>% top_n(5, -Total)) %>%
  ggplot(aes(x = reorder(Country, -Total), y = Total, fill = Total)) +
  scale_y_continuous(limits = c(0, 35000)) +
  geom_bar(width = 0.8, stat = 'identity') +
  geom_text(aes(label = Total), vjust = -.8) +
  scale_fill_gradientn(colours = c("#B8CEDC",
                                   "#14AABA",
                                   "#25A0AB",
                                   "#0F808C",
                                   "#0E5D6C",
                                   "#08363F")) +
  labs(x = "País", y = "Total") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 50, hjust = 1, vjust = 1),
        legend.position ="none")
```

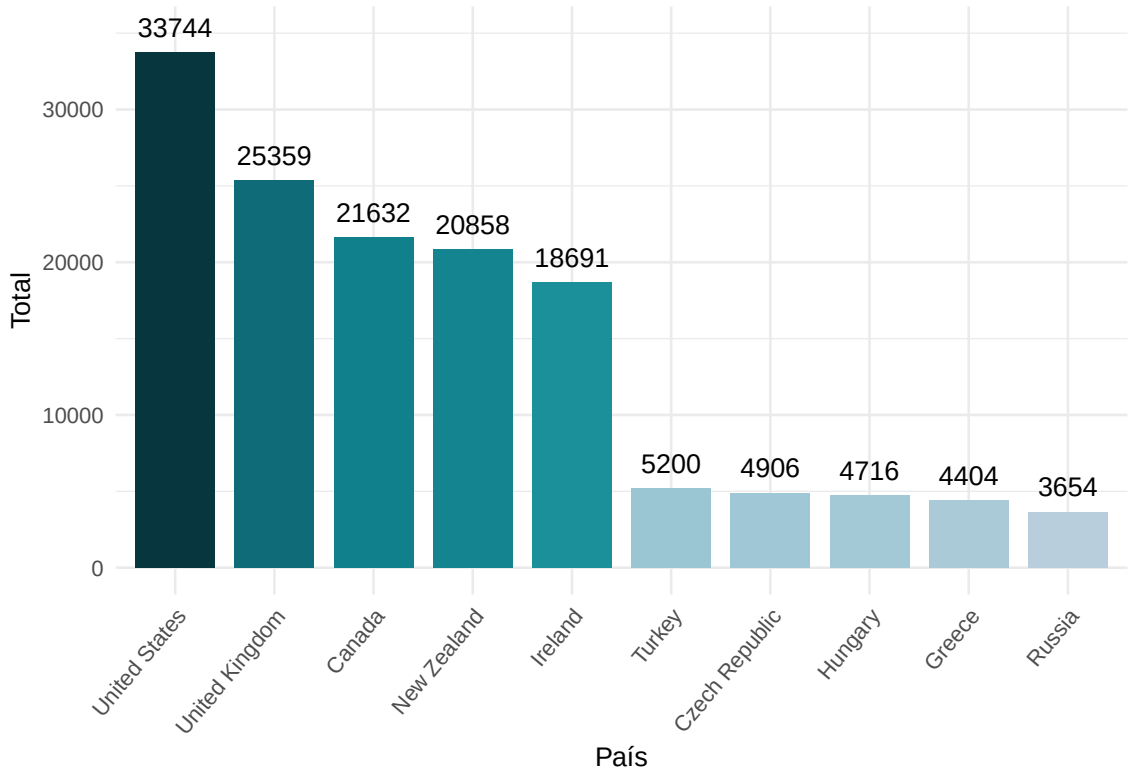


Figura 4.2: Ranking de los cinco países con más y menos contenido.

Para realizar el gráfico se han seleccionado los cincos países con más y menos contenido mediante la función *top_n* detallada en 2.2.1.2. Esta función recibe la columna que contiene el número total de series y películas, *Total*, y dependiendo de si el signo que tiene delante esta variable es negativo o no, seleccionará los cinco países con más (en el caso de signo negativo) o menos contenido. Tras ello se representa con las funciones del paquete *ggplot*.

Como se aprecia en el gráfico, el cuarto grupo está constituido en su mayoría por países de Europa Oriental: Polonia, Lituania, Estonia, Letonia, Hungría, República Checa y Rusia; aunque en él también se encuentran un par de países integrantes de Europa Occidental: Países Bajos y Grecia, y un país asiático: Tailandia. Por otro lado, el tercer grupo está compuesto en su mayoría por países de Europa Occidental y de América del Sur. Esta clara distinción de contenido por regiones da pie a cuestionarse si existen diferencias significativas entre distintas zonas geográficas. Con este fin, usando como muestra los cuarenta y seis países disponibles, estudiamos la siguiente tabla:

Tabla 4.1: Número de títulos disponibles por región.

Región	Nº títulos	Nº títulos/País
Asia Oriental y Pacífico	108507	12056
Europa y Asia Central	233120	9325
América Latina y el Caribe	86740	10842
América del Norte	55376	27688
Asia Meridional	12400	12400
África subsahariana	17149	17149

Se han agrupado los países según las siete regiones que define el Banco Mundial (veáse Bank): Asia Oriental y Pacífico, Europa y Asia Central, América Latina y el Caribe, América del Norte, Asia Meridional, África subsahariana y Oriente Medio y Norte de África. De esta última no se han obtenido datos para el estudio, ya que de los cuarenta y seis países de la muestra ninguno pertenece a ella.

La primera columna contiene la cifra total de títulos disponibles en cada región. Por ende, en valor absoluto, Europa y Asia Central es la región con mayor número de contenidos, seguida de Asia Oriental y Pacífico. Sin embargo, estas cifras no son del todo fiables, ya que algunas regiones están integradas por más países que otras: América del Norte, por ejemplo, está constituida únicamente por dos países, Estados Unidos y Canadá, mientras que en Europa y Asia Central hay más de cincuenta.

Así pues, un análisis más adecuado es el que se deriva de la segunda columna: el número medio de títulos por país en cada región. A tal efecto, se advierte que la líder de este ranking, tal y como era de esperar en vista de los resultados anteriores, es América del Norte: en esta región hay de media más de veintisiete mil títulos por país.

En segunda instancia se encuentra África subsahariana, con diecisiete mil títulos de media, aunque cabe señalar que esta región es algo más peculiar, pues para el estudio solo se han usado los datos de Sudáfrica. Europa y Asia Central, que en el análisis de la primera columna se encontraba en primer puesto, ahora se posiciona en último lugar, siendo la región que menos títulos posee por país.

En definitiva, este análisis no hace más que reforzar lo evidente: las compañías de streaming centran más sus servicios en América del Norte. Europa, y en especial Europa del Este, posee un catálogo mucho inferior.

Dentro de este marco, cabe considerar si existen discrepancias importantes entre las plataformas. Así pues, vamos a examinar el contenido disponible en cada una de estas por separado.

En primer lugar se analiza el volumen del catálogo de Netflix en los cuarenta y seis países de la muestra.

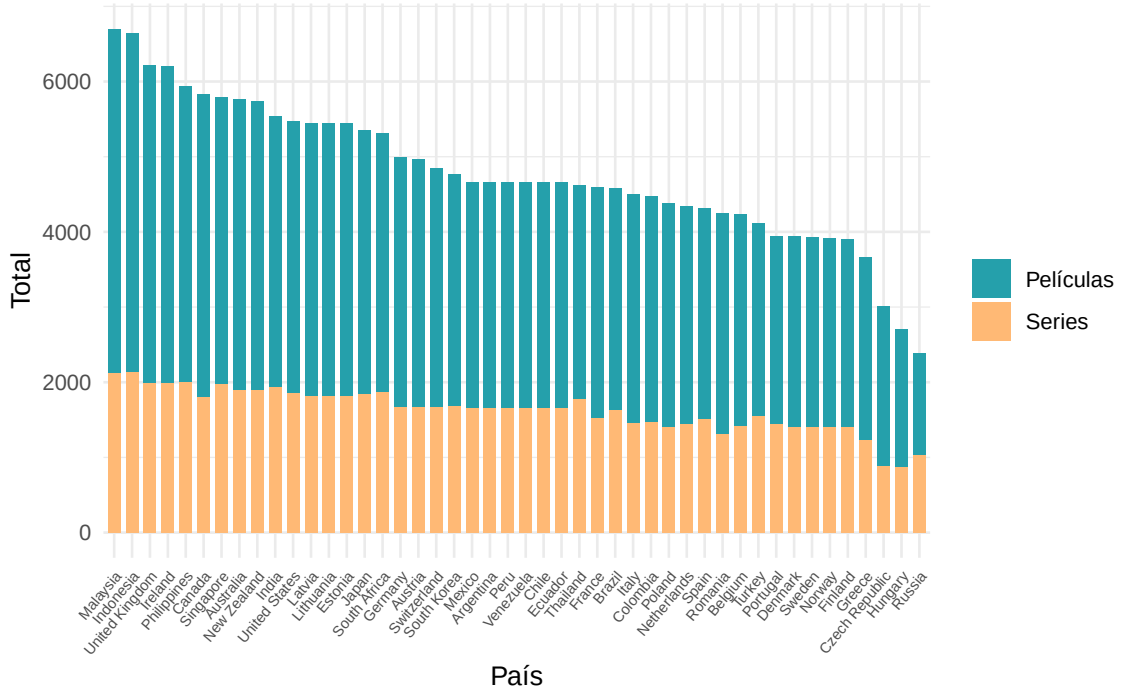


Figura 4.3: Contenido disponible en Netflix por país.

La cifra total no difiere excesivamente entre los distintos territorios: Malasia lidera este ranking con más de seis mil series y películas, mientras que Rusia se encuentra en último puesto con algo más de dos mil. Por consiguiente, el rango de valores no es muy amplio. Asimismo, la tipología de contenido está bastante equilibrada: las series representan entre un treinta y un cincuenta por ciento del contenido total.

Curiosamente, Malasia e Indonesia, que en el análisis anterior se había concluido que se encontraban entre los países que limitaban el tercer y cuarto grupo, esto es, la cantidad de contenido del que disponen es un tanto inferior a la media; ahora se encuentran encabezando este gráfico. Por consiguiente, da la impresión de que deben la mayor parte de su catálogo a esta plataforma.

Por otro lado, la mayoría de países que tenían mayor catálogo global presiden también este gráfico, aunque sorprendentemente Estados Unidos no se encuentra entre los diez primeros.

En última instancia, Estonia, Lituania y Letonia, que poseían un menor volumen de contenido total, se encuentran entre la mitad de países con más contenido en Netflix, llegando a superar incluso a España, que se encuentra entre los quince últimos.

A continuación, representemos el mismo gráfico para HBO:

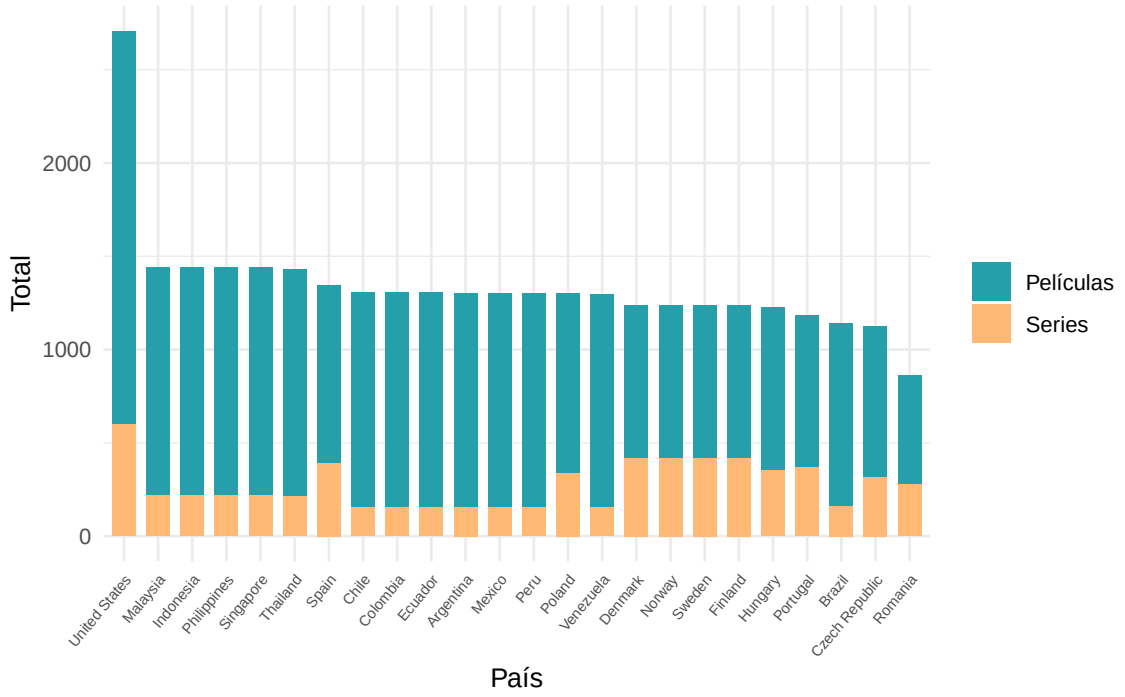


Figura 4.4: Contenido disponible en HBO por país.

El número de países para los que se ha realizado la representación es mucho menor: de los cuarenta y seis que se han tenido en cuenta, HBO solo esta disponible en veinticuatro de ellos. Entre los territorios que no tienen acceso al servicio encontramos a los dos países oceánicos, Australia y Nueva Zelanda, al mismo tiempo que una multitud de países europeos tales como Bélgica, Italia, Francia o Irlanda.

Asimismo, el catálogo es menor que el de Netflix: la mayor parte de los países cuenta con alrededor de mil títulos, y aunque se sitúa en cabeza Estados Unidos, con más de dos mil quinientos, la cifra parece bastante pequeña teniendo en cuenta que la mayoría de países puede acceder a más de dos mil títulos con Netflix.

Por último, se observa que en este caso no está tan compensado el número de series con el de películas, siendo este último notablemente superior en todos los casos.

Seguidamente, estudiaremos el caso de Amazon Prime Video:

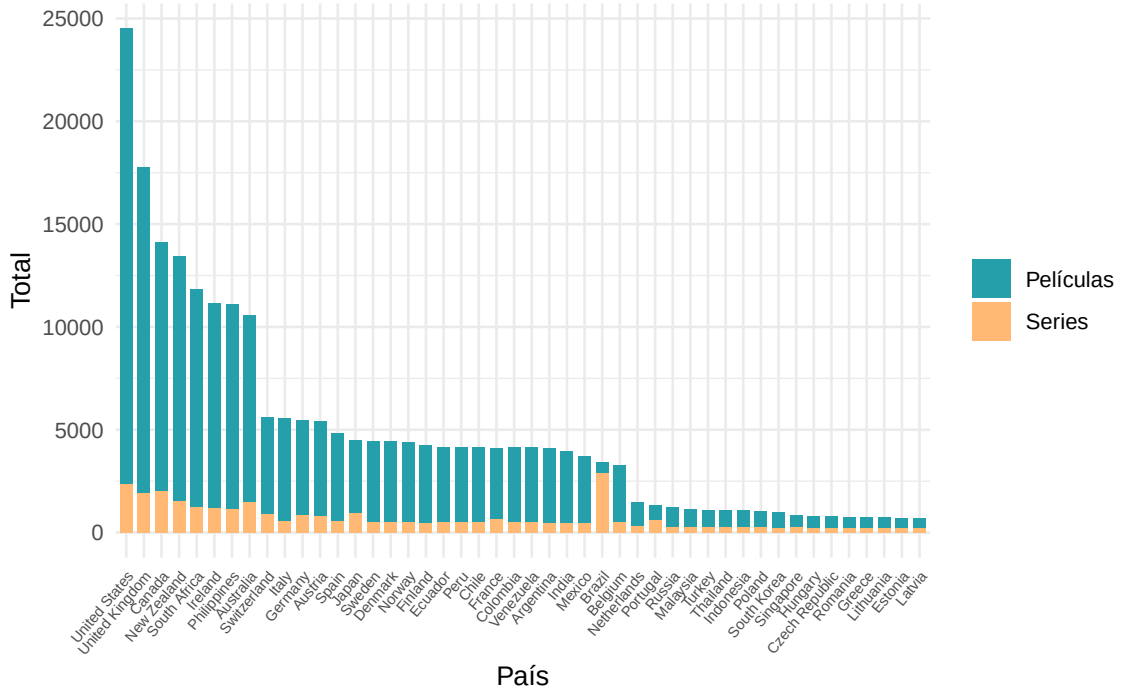


Figura 4.5: Contenido disponible en Amazon Prime Video por país.

Este gráfico difiere considerablemente de los anteriores: el rango de valores es sumamente amplio. El país líder de contenido en Netflix, Malasia, supera al último, Rusia, por aproximadamente cuatro mil títulos y el país líder de contenido en HBO, Estados Unidos, supera al último por tan solo mil. Estados Unidos vuelve a encabezar el ranking de contenido en este caso, aunque con la imponente cifra de casi veinticinco mil títulos, cantidad importante si la enfrentamos al otro extremo: Letonia se encuentra en último lugar con apenas setecientos.

Lo cierto es que las diferencias entre países son abrumadoras. Los primeros ocho coinciden con los del análisis total de contenido, e incluso el orden es casi idéntico: la única diferencia es que en este caso Sudáfrica se encuentra en quinto puesto. Cabe destacar, que aunque Reino Unido se encuentre en segundo lugar posee siete mil títulos menos que Estados Unidos, distancia que no pasa desapercibida.

El volumen de contenido de estos ocho países es superior a diez mil pero menor que quince mil, exceptuando a Reino Unido. Tras este conjunto nos encontramos otro más uniforme, constituido en su mayoría por países de Europa Occidental y América del Sur cuyo catálogo se sitúa alrededor de los cinco mil títulos, lo cual es de nuevo una diferencia considerable si comparamos con el grupo anterior. Por último, se tiene una última colección de países liderados por Bélgica, que posee algo más de tres mil series y películas y concluyendo con Letonia, que como se ha afirmado antes, consta de casi setecientos.

Para finalizar, observamos que a pesar de ser el rango total muy amplio, el de las series es todo lo contrario: el país que más series tiene es inesperadamente Brasil, con cerca de tres mil, seguido, como era de esperar, de Estados Unidos, y el que menos Grecia, con

algo más de doscientas. Sin embargo, al existir diferencias tan grandes entre las cifras de contenido total, la proporción de contenido que representan las series es muy diferente: en Estados Unidos son apenas un diez por ciento, mientras que en Letonia son casi la mitad.

Para terminar este análisis, estudiamos en último lugar Disney+:

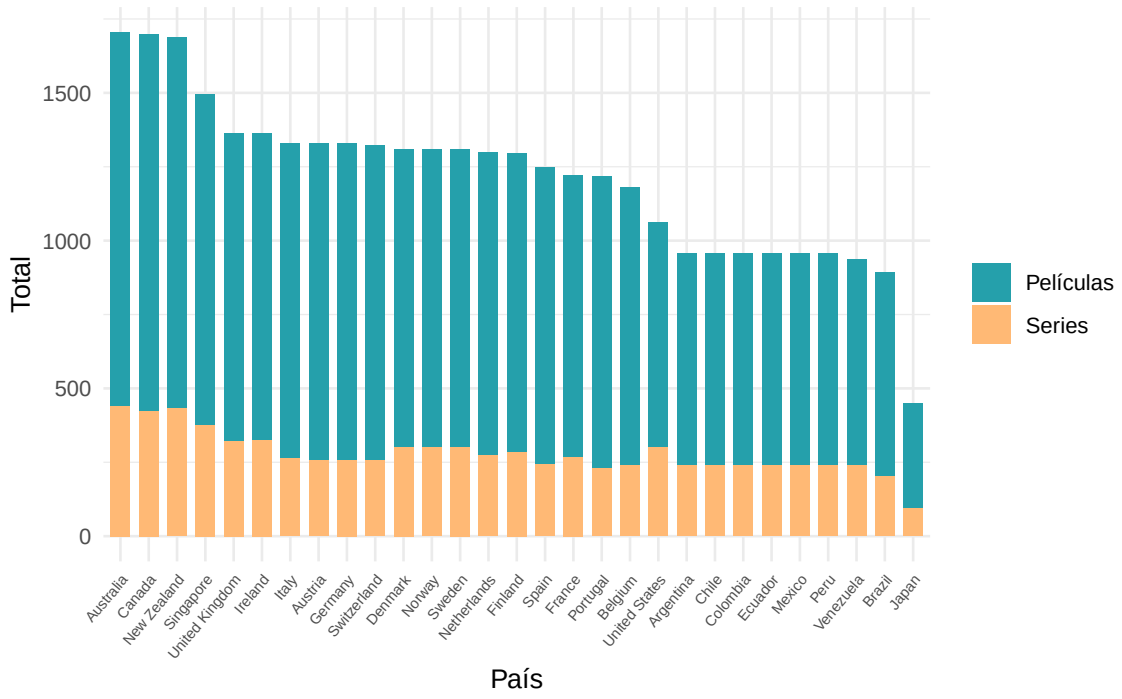


Figura 4.6: Contenido disponible en Disney+ por país.

Antes de comentar el gráfico, hay que recalcar que para el estudio de Disney+ han sido excluidas la India e Indonesia. Esto es debido a que en estos países Disney+ se encuentra integrado junto a otra compañía denominada Hotstar, existiendo por tanto un servicio unificado: Disney + Hotstar. En consecuencia, las cifras de contenido obtenidas para Disney+ en estos territorios son en realidad las cifras de la unión de ambas plataformas, por lo que la cantidad de contenido disponible en dichos países no es comparable con la del resto. Así pues, se han descartado los mismos en esta parte del estudio.

Se observa que el volumen de contenido de esta plataforma es semejante entre los distintos países: el contenido medio se sitúa en torno a mil. Asimismo, el volumen parece mantenerse uniforme por zonas: los países europeos se encuentran en el centro, con alrededor de mil títulos de media, y los países sudamericanos se encuentran al final, con algo menos de mil.

En cabeza nos encontramos a Australia, seguida muy de cerca de Canadá y Nueva Zelanda. Cabe destacar que Estados Unidos se encuentra al mismo nivel que los países sudamericanos, con apenas algo más de mil títulos. En último lugar, con un contenido bastante inferior al resto, se encuentra Japón.

Finalmente, merece la pena subrayar que la variación de contenido entre países en Disney+ se debe casi en su totalidad al volumen de películas, ya que el número de series es prácticamente el mismo en los diferentes territorios.

4.2. Estudio del precio

Otra diferencia entre países, ya comentada, es el precio. Las plataformas no tienen el mismo precio en todos los países, incluso algunas difieren entre los que se encuentran en el mismo continente. A fin de analizar si este cambio es significativo, se ha recogido para cada uno de los cuarenta y seis países anteriores el precio de las cuatro plataformas, si es que se encuentran disponibles en dicho país.

Un factor que se ha tenido en cuenta a la hora de reunir los costes es que algunas plataformas presentan diferentes precios en función del plan que escoja el usuario. Aunque las diferencias entre un plan y otro pueden deberse a causas relacionadas con la calidad de reproducción o con la capacidad de poder descargar contenido, la principal desigualdad es el número de pantallas en las que se puede reproducir contenido al mismo tiempo. Así, Netflix da a elegir entre tres planes, optando entre ver una, dos o cuatro pantallas al mismo tiempo, mientras que HBO, Amazon Prime Video y Disney+ ofrecen un número de pantallas cerrado: dos, tres y cuatro respectivamente. Como el número de pantallas no es comparable, pero las tres últimas plataformas no permiten modificarlo, se ha realizado el estudio desde el punto de vista de un individuo que no sabe por qué plataforma de streaming decantarse. Si el individuo tuviese que seleccionar un plan de Netflix escogería el básico, ya que solo necesita una pantalla. Por el contrario, si el individuo contratase HBO, Amazon Prime Video o Disney+ tendría pantallas que no usaría, pero no puede optar por otro plan. Por lo tanto para realizar el análisis se ha usado el precio del plan básico de Netflix (una única pantalla), y el único precio disponible para las demás.

Tras reunir los precios de cada plataforma en los distintos países, para poder realizar comparaciones se han convertido todos a una moneda común, en este caso el euro. Por la misma razón, para esta parte del estudio solo se han considerado aquellos países que tienen acceso a las cuatro plataformas, resultando en un total de trece, tal y como se muestra a continuación:

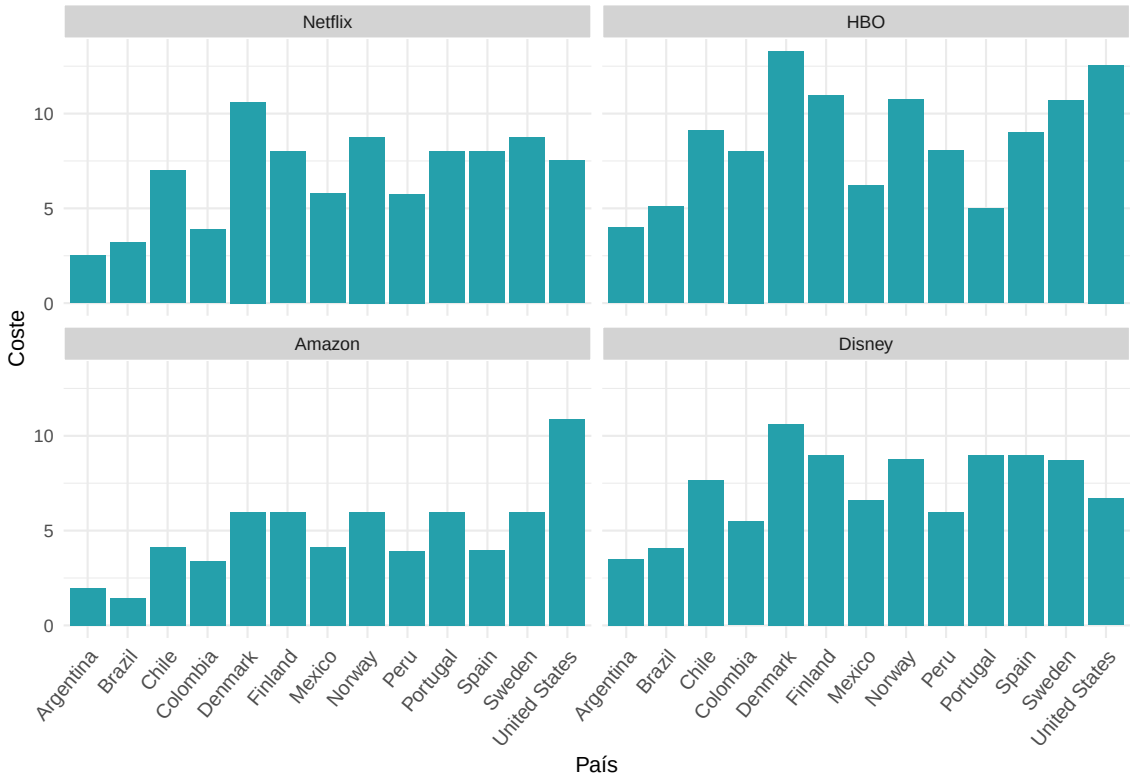


Figura 4.7: Comparación de los precios de cada plataformas por país.

Como se aprecia en el gráfico, los precios de las plataformas muestran un patrón de comportamiento similar entre los distintos países de la muestra, asignando los costes inferiores a Argentina y Brasil. Estados Unidos y Dinamarca suelen destacar por sus elevados precios, lo cual no es raro dado su alto nivel de vida, aunque bien es cierto que se puede hablar de forma similar del resto de los países nórdicos y estos no presentan este comportamiento. Por otra parte, se observa que HBO tiene los mayores precios, en contraposición a Amazon Prime Video, que posee los menores.

Señalar el código empleado para realizar esta gráfica, en el que se ha usado la función *facet_wrap* del paquete *ggplot* para realizar un gráfico diferente por cada una de las cuatro plataformas. Previamente a la representación se han transformado los datos para presentarlos en el formato que requiere *ggplot* utilizando para ello la función *gather*. Así pues, se pasa de un conjunto de datos de cinco columnas donde hay una para el precio de cada plataforma, a otro en el que tan solo hay tres: una indicando el país, otra que contiene los precios y otra que indica a que plataforma pertenece dicho precio.

```
datos %>%
  drop_na() %>%
  transmute(`Country` = as.factor(Country),
         `Netflix` = Netflix_Cost_EUR,
         `HBO` = HBO_Cost_EUR,
         `Amazon` = Amazon_Cost_EUR,
         `Disney` = Disney_Cost_EUR) %>%
```

```
  gather(key = "Plataforma", value = "Coste", -c(Country)) %>%
  arrange(desc(Country)) %>%
  mutate(Plataforma = factor(Plataforma,
                             levels = (c("Netflix", "HBO",
                                         "Amazon", "Disney")))) %>%
  ggplot(aes(x = Country, y = Coste)) +
  geom_col(fill = color1) +
  facet_wrap( ~ Plataforma, ncol = 2) +
  labs(x = "País") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 50, size = 10,
                                   hjust = 1, vjust = 1),
        strip.background = element_rect(colour="white",
                                        fill="lightgrey")) +
  ggtitle("\n")
```

Sin embargo, lo cierto es que la representación anterior por sí sola no aporta más información. No pasa desapercibido que en el estudio del punto anterior se había concluido que algunas plataformas tienen mayor volumen que otras, siendo estas diferencias, en algunos casos, abismales. En vista de ello, para examinar si existe relación entre lo que se paga y el contenido disponible, se calcula el ratio entre el contenido y el precio, obteniéndose los siguientes resultados:

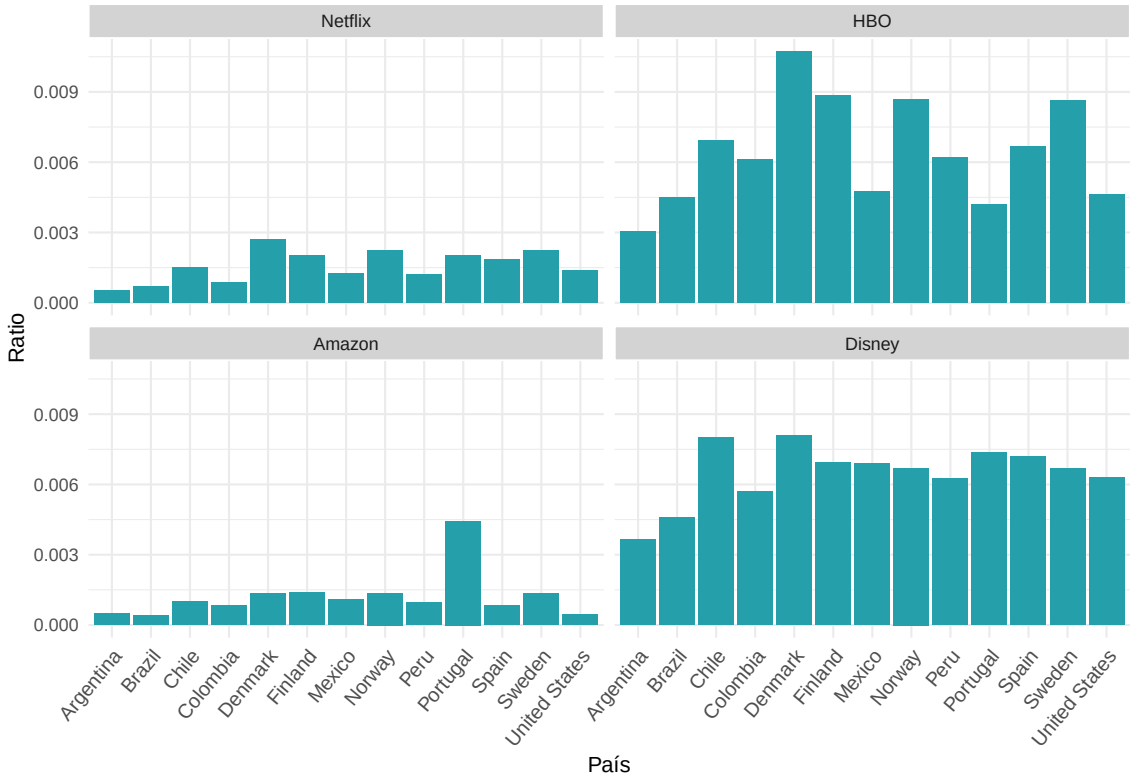


Figura 4.8: Comparación de la relación contenido/precio de cada plataforma por país.

Este gráfico resulta más ilustrativo que el anterior. De igual forma que en la representación previa, Dinamarca, a excepción del caso de Amazon Prime Video donde Portugal presenta los mayores precios, sigue siendo el país de la muestra donde más se paga por las plataformas de streaming. Además, se había concluido que Amazon Prime Video era la plataforma que poseía menores precios y HBO la que mayores, hecho que se sigue manteniendo en este caso. Por el contrario, el comportamiento de Netflix en el gráfico anterior era análogo al de Disney+, señalando que ambas plataformas tienen precios similares, mientras que en esta representación vemos que presentan un comportamiento bastante diferenciado. Por consiguiente, se concluye que a pesar de tener precios similares, como Netflix posee un volumen superior a Disney+, resulta más barato, más rentable en cuanto a la relación entre volumen de contenidos y precio, posicionándose junto a Amazon Prime Video, entre las plataformas en la que menos se paga por título, en contraposición a HBO y Disney+ donde los precios llegan a triplicarse.

Se puede observar también (más débilmente en el caso de HBO) que las plataformas tratan de equilibrar precio y contenido, de forma que la relación entre ambos se mantenga similar en los distintos países. Así pues, por ejemplo, en la primera sección se había concluido que las diferencias de contenido entre los distintos territorios en Amazon Prime Video eran bastante significativas, siendo en particular la diferencia entre la cifra de contenido en Estados Unidos y México de más de veinte mil títulos, pero aquí se observa que el precio está compensando, pagándose en ambos prácticamente lo mismo por título.

Evidentemente, el volumen de contenido que posea una plataforma no es el único determinante de su precio. Por ejemplo, se ha visto que los servicios en Dinamarca tienen un precio superior al de otros países, pero Dinamarca es conocida por ser relativamente cara; por consiguiente, que tenga precios superiores en las plataformas tampoco es insólito, como ya se comentó a principio del capítulo. En consecuencia, esto nos lleva a considerar que el nivel de vida de un país también influye en los precios.

Por otro lado, existen numerosas páginas en las que los usuarios se encargan de calificar las películas y las series, algunas tan consolidadas como Filmaffinity e IMDb. Parece natural pensar que las plataformas que contengan películas mejor calificadas, aquellas que ofrezcan una mayor calidad, tendrán un precio superior. Así pues, concluimos que la calidad que ofrezca una plataforma es también una componente a tener en cuenta.

Por ende, analicemos estos dos factores.

4.2.1. Precio y nivel de vida

Con objeto de analizar la relación existente entre el precio de las plataformas y el nivel de vida, se necesita, en primer lugar, un indicador para cuantificar esto último. Con este fin, emplearemos el producto interior bruto (World Development Indicators [2019]), generando los datos de la siguiente forma:

```
gdp = read_xlsx("datos/gdp.xlsx")[1:46, c(3,5)]

gdp = datos %>%
select(Country, Netflix_Total, HBO_Total,
       Amazon_Total, Disney_Total,
         Netflix_Cost_EUR, HBO_Cost_EUR,
```

```
      Amazon_Cost_EUR, Disney_Cost_EUR) %>%
  full_join(gdp, by = c("Country" = "Country Name")) %>%
  rename(`GDP` = "2019 [YR2019]",
         `Netflix_Cost` = "Netflix_Cost_EUR",
         `HBO_Cost` = "HBO_Cost_EUR",
         `Amazon_Cost` = "Amazon_Cost_EUR",
         `Disney_Cost` = "Disney_Cost_EUR") %>%
  mutate(`GDP` = 0.8343*GDP) %>%
  pivot_longer(-c(Country, GDP),
               names_to = c("Plataforma", "Type"),
               names_pattern = "(.*)\\_(.*)") %>%
  mutate(Plataforma = factor(Plataforma,
                             levels = (c("Netflix", "HBO", "Amazon", "Disney")))) %>%
  drop_na()
```

En primer lugar se lee el conjunto de datos que contiene el producto interior bruto para los distintos países. Después se seleccionan aquellas variables que nos interesan del conjunto de datos inicial: la variable que indica el país y las referentes a los precios y contenidos. Tras ello, se renombran para hacer más sencillo su uso. Con ayuda de la función *mutate* pasamos el producto interior bruto de dólares estadounidenses a euros, utilizando el cambio correspondiente.

En el conjunto hay diez variables: una indicando el país, otra para el producto interior bruto, cuatro para el precio y cuatro para el contenido. El objetivo es quedarnos con tan solo cinco, que señalen país, producto interior bruto, precio, contenido y plataforma.

Teniendo en cuenta que el nombre de las variables es de la forma *Plataforma_Tipo*, donde el tipo puede ser el precio o el contenido, usamos la función *pivot_longer* con el argumento *names_pattern*. En él precisamos el patrón que siguen los nombres de las variables, indicando así cómo separarlas en las dos columnas deseadas. Por último, cambiamos la variable a tipo factor y eliminamos las filas con valores perdidos.

Para empezar los análisis de esta sección, estudiamos la correlación existente entre ambas variables distinguiendo por plataformas y presentando los resultados en forma de tabla. A fin de poder establecer comparaciones con el estudio anterior, añadiremos a esta la correlación existente entre el precio y el contenido. Los resultados obtenidos se muestran a continuación:

Tabla 4.2: Correlación entre precio y contenido disponible por plataforma.

Plataforma	PIB	Contenido
Netflix	0.6144403	-0.0622532
HBO	0.8292337	0.3408117
Amazon	0.5544640	0.3222643
Disney	0.7463375	0.2515726

La correlación existente entre precio y contenido no es muy alta. Aún así, en todos los casos (excepto en el de Netflix donde es casi nula) es positiva, indicando que a mayor

volumen de contenidos, mayor precio. Por otra parte, observamos cómo la correlación que existe entre el precio y el producto interior bruto es siempre superior. El caso de HBO es el más notable, poseyendo la mayor correlación lo cual indica que esta plataforma tiene precios superiores en aquellos países que son más ricos.

Seguidamente, representamos para cada plataforma el precio en función del producto interior bruto, agregando las rectas de regresión correspondientes junto a los respectivos R^2 ajustados. Se obtienen las siguientes nubes de puntos:

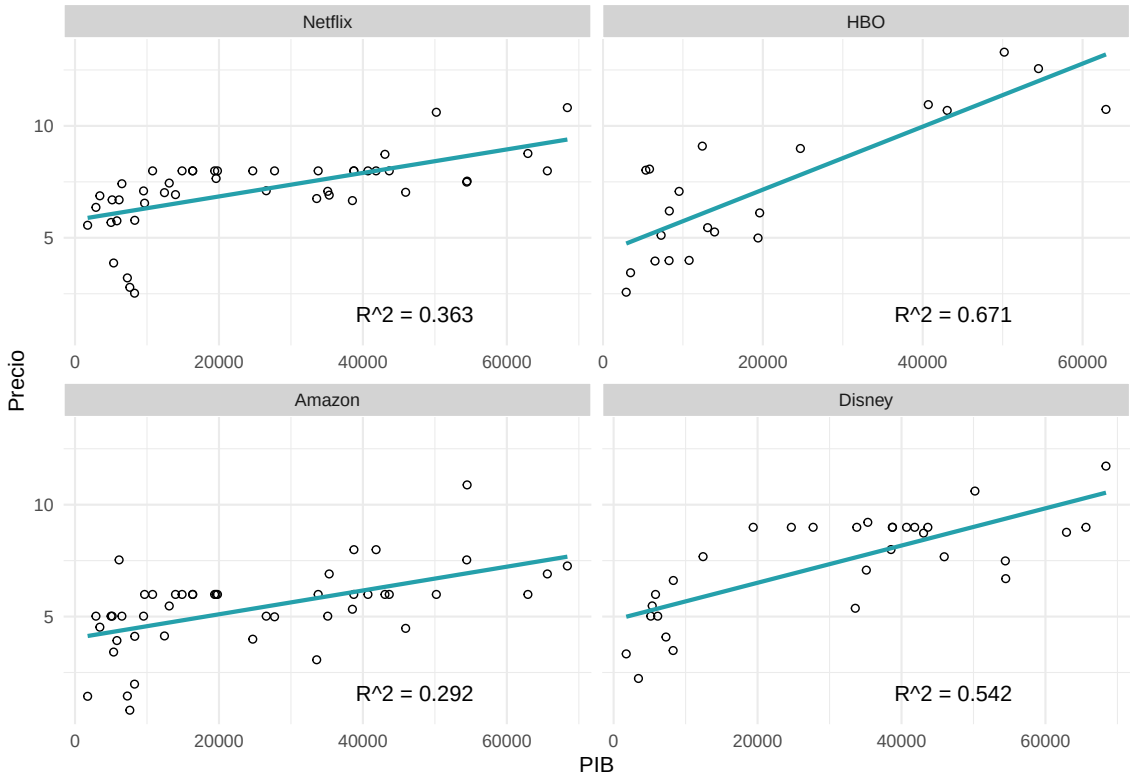


Figura 4.9: Relación entre el PIB y el precio de cada plataforma.

Como era de esperar tras el análisis de las correlaciones, el mejor ajuste lo obtenemos en el caso de HBO, con $R^2 = 0.67$ y el peor para Amazon Prime Video y Netflix.

Por tanto concluimos que el nivel de vida sí influye en las compañías a la hora de determinar el precio de los servicios que estas ofrecen, aunque en algunas tiene un peso mayor que en otras.

4.2.2. Precio y calidad

Finalmente, para analizar la relación entre la calidad de una plataforma y el precio de la misma, se han combinado dos datasets. El primero de ellos contiene los datos de quinientas películas disponibles en HBO en Estados Unidos, dando a conocer para cada una de ellas, además del título correspondiente, el año en el que fueron estrenadas, la clasificación por edades y su puntuación en IMDb (Kumar [2020]). El segundo contiene

datos relativos a más de diecisiete mil películas. Entre ellos nos encontramos el título, una lista de los países donde está disponible, y tres variables binarias que indican si la película se encuentra en Netflix, Amazon Prime Video y/o Disney+, además de las mismas variables que en el primer dataset: título, año de estreno, clasificación por edades y puntuación en IMDb (Bhatia [2020]).

Para fusionarlos correctamente, si bien es cierto que en el resto de las secciones se han usado datos relativos a todos los países, para este estudio solo ha sido posible encontrar la lista de títulos y puntuaciones de HBO en Estados Unidos. Por tanto, en este apartado solo usamos los datos relativos a este país, seleccionando del segundo dataset únicamente aquellas películas disponibles en este territorio.

Previamente, se muestra un resumen del pretratamiento que han recibido los datos, usando las funciones del paquete tidyverse, para poder ser empleados con el fin de sacar conclusiones.

Se ha empezado leyendo el conjunto de datos referente a HBO, seleccionando las variables que se van a usar y renombrándolas de forma que su designación coincida con las del otro dataset, para hacer más sencilla la unión posterior de ambos. Tras esto, con la función *mutate*, se ha creado una nueva variable, HBO, simulando a las variables binarias del segundo conjunto de datos: es una variable binaria que servirá para identificar si el título se encuentra o no en HBO. En el caso de los títulos pertenecientes a este dataset, siempre valdrá uno. Se muestra a continuación el fragmento de código utilizado:

```
hbo_movies = read.csv("datos/HBO-ratings.csv", encoding = "UTF-8") %>%
  select(1:4) %>%
  rename(‘Title‘ = movie, ‘Year‘ = year,
         ‘Age‘ = rating, ‘IMDb‘ = imdb_rating) %>%
  mutate(HBO = 1)
```

Antes de cargar el siguiente dataset, se han modificado algunos títulos de este: aquellos cuyo nombre tiene tilde. Estas, en el conjunto de datos original vienen con signo de interrogación. Con ayuda de la función *grep*, se han buscado aquellos títulos que contenían dicho símbolo y se han modificado manualmente.

```
hbo_movies[grep("\\?", hbo_movies$Title),]$Title =
  c("Amélie", "Pokémon Detective Pikachu", "Alien 3",
    "My Dinner with Hervé","3 1/2 Minutes, 10 Bullets",
    "What’s Your Number?")
```

Después, se ha leído el segundo conjunto, tomando las variables de interés. Se han seleccionado, con la función *filter*, aquellas películas que se encuentran en Estados Unidos. Para ello se ha hecho uso de la función *grepl*: dada una observación, mira si en la variable país se encuentra Estados Unidos devolviendo en caso afirmativo ‘Verdadero’, por lo que *filter* la selecciona. También prestamos especial atención a la variable que muestra la clasificación por edades: en el anterior dataset, las películas de las que no se poseía información acerca de esta variable completaron el campo con N/A, y en este dataset, las películas de las que no se tiene información acerca de esta variable dejan el campo vacío. Por tanto, para fusionar correctamente los dos conjuntos, cambiamos en este los campos vacíos de la variable Age por N/A haciendo uso de las funciones *mutate* y *replace*.

Por último, con *full_join* usando las variables comunes (título, calificación por edades, año de estreno y puntuación en IMDb) se han unido ambos datasets, reteniendo todas las observaciones. Para las películas que se encuentran en los dos no hay ningún problema: los campos de todas las variables se encuentran completos. Sin embargo, para las que no, habrá campos vacíos: si la película solo se encontraba en el segundo dataset tendrá vacío el campo de HBO, y si solo se encontraba en el primero tendrá el de Netflix, Prime Video y Disney+. Los campos vacíos indican que las películas no se encuentran en dichas plataformas, por lo que los reemplazamos con un cero usando *mutate*. Mostramos el código utilizado:

```
ratings = read.csv("datos/Movies-ratings.csv",
                   encoding = "UTF-8")[,-c(1,7,9,12)] %>%
  filter(grepl("United States", Country)) %>%
  select(2:8) %>%
  rename(`Disney` = Disney.) %>%
  mutate(Age = replace(Age, Age == "", "N/A")) %>%
  full_join(hbo_movies) %>%
  mutate(Netflix = replace(Netflix, is.na(Netflix) , 0),
         HBO = replace(HBO, is.na(HBO) , 0),
         Prime.Video = replace(Prime.Video, is.na(Prime.Video) , 0),
         Disney = replace(Disney, is.na(Disney), 0))
```

Tras esto, obtenemos un único dataset con más de diez mil películas, en el que, además de las cuatro variables comunes utilizadas para combinarlos, nos encontramos cuatro variables binarias que indican si el título pertenece a Netflix, HBO, Amazon Prime Video y/o Disney+.

Con el fin de estudiar la calidad de las plataformas, se ha obtenido la puntuación media de las películas que contienen. Asimismo, se han incorporado las puntuaciones máximas y mínimas junto con la varianza de dichas calificaciones, para medir la fluctuación de estos valores. Por otro lado, para calibrar la modernidad de cada plataforma, se ha agregado la media de los años de estreno de las películas que contienen. Así pues se obtiene la siguiente tabla:

Tabla 4.3: Resumen de la calidad ofrecida por cada plataforma.

	Media	Varianza	Máximo	Mínimo	Año medio
Netflix	6.25	1.19	8.8	1.6	2013
HBO	6.99	0.47	8.8	5.2	2005
Amazon Prime Video	5.64	2.03	9.3	0.0	1999
Disney+	6.43	1.06	8.7	1.6	1998

A la vista del análisis, los mejores resultados han sido obtenidos por HBO: la puntuación media de las películas de esta plataforma es de casi siete puntos, con una varianza muy reducida, sinónimo de que el rango de valores de las calificaciones de las películas no es muy amplio. Netflix y Disney+ están muy igualadas: la media no es muy inferior a la de HBO, pero tienen una varianza superior. La última posición en este estudio

la ocupa Amazon Prime Video: las películas de esta plataforma tienen, de media, una puntuación poco superior al cinco y medio y, además, la varianza de esta plataforma es también la mayor: el rango de puntuaciones es muy amplio. Esto se ve ya reflejado con las calificaciones máxima y mínima ilustradas en la tabla: Amazon Prime Video posee tanto la película con mayor nota de las cuatro plataformas, como la de menor.

En relación al año medio de las producciones se deduce que Netflix es la plataforma más actualizada. Esto puede deberse a la gran cantidad de películas originales que estrena cada año. Por otra parte, las plataformas con un contenido más antiguo son Amazon Prime Video y Disney+: esto último no sorprende, Disney+ contiene todas las películas antiguas de la compañía.

Por tanto, Amazon Prime Video, que es la plataforma que más contenido posee, es también la que menor calidad tiene, es decir, su catálogo es bastante amplio pero nos encontramos con películas de todo tipo, desde mejor hasta peor calidad. En cambio, con HBO sucede justo lo contrario: es la plataforma con menos contenido, pero la de mayor calidad. Esto también cuadra con el análisis de precios que se hizo anteriormente: HBO era la plataforma más cara y Amazon Prime Video la más barata.

Así pues, gracias a este estudio se concluye lo siguiente: la plataforma que posee, de media, más contenido es Amazon Prime Video, llegando a ser el número total de títulos en algunos países más de doce veces mayor que en otras plataformas. Este hecho tiene consecuencias tanto buenas como malas: debido a la gran cantidad de títulos que posee, es la plataforma más barata si nos fijamos en el ratio contenido/precio; pero, también por esta razón, es la que más varianza tiene en cuanto a calificaciones de los títulos que posee, contiene películas con muy buena puntuación, pero también otras con calificaciones muy bajas.

Por otro lado, HBO es todo lo contrario: se encuentra, igualada con Disney+, entre las plataformas que menos contenido tiene, y además, está disponible en un número inferior de países. Sus precios son más altos, llegando a pagar más por título, aunque compensándolo según el nivel de vida del país, haciendo que los países con mayor nivel económico paguen más por los contenidos que ofrece la compañía. Sin embargo, de las cuatro, es la plataforma que ofrece más calidad: las películas que en ella se encuentran tienen una puntuación alta, y el rango de las mismas no varía demasiado.

Netflix y Disney+ se encuentran en el medio de las dos anteriores. Netflix se asemeja más a Amazon Prime Video, aunque con un menor volumen de contenido. Los precios por título en esta plataforma son de los más bajos de las cuatro estudiadas, y la calificación media de las películas de la misma se mantiene sobre un seis, superior a Amazon Prime Video, pero no tan buena como la de HBO. Disney+ se parece más a HBO: se paga más por título, aunque no tanto como en esta última, y el contenido es bastante menor, estando también disponible en menos países.

En vista de todo esto, un individuo que quisiera elegir entre una de estas cuatro plataformas debería sopesar detenidamente los pros y los contras, decidiendo, teniendo su país en cuenta, que plataforma le conviene más: quizás quiera tener acceso a más contenido, pagando menos por título, y se decante por Amazon Prime Video, o tal vez prefiera contentarse con menos contenido, pero de mayor calidad, eligiendo HBO. Una opción más equilibrada sería Netflix, que compensa precio por título, contenido y calidad.

Parte III

Modelos predictivos

En este bloque se ha conjugado el estudio exhaustivo de las variables desde el punto de vista estadístico con el empleo de técnicas propias de la Estadística, Matemática e Inteligencia Artificial para el diseño de los modelos y los workflows correspondientes para el desarrollo y evaluación de los mismos.

Capítulo 5

Pretratamiento y análisis descriptivo de los datos

Las plataformas de streaming de vídeo, además de incorporar películas y series ya existentes en su catálogo, también producen su propio contenido. A pesar de que este no era el motivo central por el que fueron creadas, lo cierto es que se ha convertido en una de las actividades principales de las mismas. Netflix, por ejemplo, desde el lanzamiento de su primera serie original en 2013, House of Cards, ha producido más de mil elementos de contenido propio.

Naturalmente, las compañías esperan obtener beneficios del contenido que producen. Por tanto, antes de realizar una serie o película, deben plantearse si esta les otorgará algún tipo de ganancia. Dicho de otro modo, necesitarán predecir el éxito de la producción. Esta necesidad no solo la tienen estas empresas, sino cualquier productora a la hora de realizar una nueva serie o película.

La cuestión entonces es: ¿cómo se puede predecir el éxito que va a tener una nueva producción? Aun cuando hay muchos factores que no pueden ser controlados, existen otros que sí. El éxito de una producción puede verse influenciado por elementos como el director, los actores protagonistas o el género. Asimismo, otro aspecto importante a tener en cuenta, es cómo medir dicho éxito. Pese a que se puede considerar como una dimensión, en parte, subjetiva, existen ciertas medidas que nos permiten cuantificarlo: desde la puntuación que posee en alguna base de datos relevante hasta el número de galardones con los que cuenta. También el dinero recaudado, cuando hablamos de taquilla, pero en el caso de las plataformas de streaming en muchos casos no van al cine sino que se estrenan y persisten en su plataforma.

En este capítulo se va a estudiar cómo predecir el éxito de una película de dos formas posibles: considerando su puntuación en el IMDb, o bien a través de la posibilidad de ganar un premio Oscar. Para ello, se emplearán diferentes técnicas estadísticas y del ámbito de la inteligencia artificial cuyo marco teórico viene detallado en el capítulo 2.

5.1. Pretratamiento de los datos

Para realizar los análisis antes descritos, se han usado los datos que comparte el IMDb de forma gratuita (IMDb). Esta aplicación, cuyas siglas son la abreviatura de *Internet*

Movie Database, es una base de datos de series y películas que, además de recopilar información de las mismas, les asigna una puntuación calculada a través de las votaciones realizadas por los usuarios. La compañía, actualmente adquirida por Amazon, posee millones de suscriptores, siendo una de las páginas de reseñas cinematográficas más relevantes en todo el mundo.

IMDb proporciona, sin costes, parte de la información que almacena diariamente, distribuida en siete datasets. Para este estudio, se usarán cuatro de estos conjuntos. En ellos, los títulos se identifican con un código alfanumérico único, denominado *tconst*.

El primero, el archivo *title.basics*, proporciona la información básica para cada título: nombre, tanto original como principal; géneros, duración, año de inicio y si está clasificado como contenido para adultos o no. También proporciona la tipología, indicando si se trata de una película, una serie, un corto, un concurso de televisión..., así como el año de finalización en el caso de las series.

Por otro lado, se encuentran los archivos *title.crew*, que contiene los identificadores de los directores y los guionistas de cada título, y *title.principals*, que contiene los identificadores del resto de participantes: actores, actrices, compositores, directores de fotografía... En último término, el archivo *title.ratings* posee la información relativa a las calificaciones de cada título: puntuación y número de votos.

Para poder llevar a cabo los análisis indicados en la introducción, habrá que tratar previamente estos datos, con objeto de asegurar la calidad de los resultados. Con este fin, se dedicará esta sección a la explicación detallada del procedimiento de la limpieza de datos o *Data Cleaning*, indicando de forma precisa cada uno de los pasos y decisiones tomadas en el proceso. Para ello, usaremos las funciones de el paquete *tidyverse* de R, especificadas en 2.2.1.

5.1.1. Lectura de datos

Antes que nada hay que leer los datos. Una primera idea sería cargar los cuatro conjuntos al mismo tiempo para después combinarlos, pero, para optimizar el proceso, se leerán todos excepto el archivo *title.principals*. Este último tiene un gran tamaño, por lo que no se usará hasta que sea necesario para no ralentizar el proceso.

Con objeto de acotar el estudio, este se limitará tan solo a películas. Así pues, eliminaremos del archivo *title.basics* todas aquellas observaciones que no correspondan a películas, pudiendo prescindir, por tanto, de dos de las variables: tipología y año de finalización. Ninguna aporta información relevante: la tipología de todas las observaciones es ahora la misma, y el año de finalización tenía solo sentido en las series.

Por tanto, el primer paso será unir los tres conjuntos de datos siguientes: *title.basics*, *title.crew* y *title.ratings*. Para ello haremos uso de alguna de las funciones que combinan datasets detalladas en 2.2.1.5. En este caso la adecuada es *left_join*, utilizando para enlazar el identificador del título, *tconst*, y siendo el primer conjunto el referente a *title.basics*. Esto último no es una decisión tomada a la ligera: en este conjunto solo tenemos los identificadores de las películas, mientras que en los otros dos tendremos también identificadores referentes a series, cortos u otro contenido. Uniéndose de esta forma lo que haremos es añadir a las películas la información del equipo y las puntuaciones de las

mismas, ignorando los datos referentes al equipo y puntuaciones de otra tipología de contenido.

Por último, recalcar que usaremos únicamente aquellas observaciones que tengan todos los campos completos, por lo que eliminaremos aquellas que no los tengan mediante la función *drop_na*.

En el siguiente fragmento de código se realiza el proceso descrito previamente:

```
titles2 = as.data.frame(data.table::fread(
  'datos/IMDb/title2.tsv', encoding = 'UTF-8', na.strings = "\\N")) %>%
  mutate(titleType = as.factor(titleType), isAdult = as.factor(isAdult),
         startYear = as.numeric(startYear), endYear = as.numeric(endYear),
         runtimeMinutes = as.numeric(runtimeMinutes)) %>%
  filter(titleType %in% c('movie', 'tvMovie')) %>%
  select(-titleType, -endYear)
crew = as.data.frame(data.table::fread('datos/IMDb/crew.tsv'),
                     na.strings = "\\N")
ratings = as.data.frame(data.table::fread('datos/IMDb/ratings.tsv'),
                        na.strings = "\\N")
datos = titles2 %>%
  left_join(ratings) %>%
  left_join(crew) %>%
  filter(directors != '\\N', writers != '\\N') %>%
  drop_na()
```

5.1.2. Directores y guionistas

Al unir los tres datasets obtenemos un único conjunto con once variables, resumidas a continuación:

Tabla 5.1: Descripción de las variables del conjunto de datos.

Nombre	Tipo	Descripción
tconst	Cadena	Identificador del título
primaryTitle	Cadena	Título principal
originalTitle	Cadena	Título original
isAdult	Binaria	Clasificación para adultos (sí/no)
startYear	Numérica	Año de estreno
runtimeMinutes	Numérica	Duración
genres	Cadena	Géneros principales (máximo tres)
averageRating	Númerica	Puntuación media
numVotes	Numérica	Número de votos
directors	Cadena	Identificador de los directores
writers	Cadena	Identificador de los guionistas

En primer lugar, vamos a centrarnos en dos de ellas: directores y guionistas. La cuestión es cómo incorporar estas variables, que son cadenas, en los análisis planteados. Una solución posible sería tratarlas como factores, pero el problema principal de esto radica en la cantidad de niveles que habría y en cómo usarlos para técnicas de regresión: hacer one hot econding o usar variables dummy implicaría trabajar con un número de variables desmesurado. Otro inconveniente que presenta este planteamiento es que hay películas con más de un director o más de un guionista, lo cual haría que el número de niveles fuese aún más amplio y haría diferentes, por ejemplo, a dos películas que comparten director, si en una de ellas ha dirigido alguien más. Para ilustrar esto último, mostramos a continuación como vienen presentadas las películas con más de un director, usando como ejemplo West Side Story, película de 1961 dirigida por Jerome Robbins y Robert Wise, con identificadores nm0730385 y nm0936404, respectivamente.

Tabla 5.2: Ejemplo película con más de un director.

Identificador	Título	Directores
tt0055614	West Side Story	nm0730385,nm0936404

Con el fin de solucionar este asunto y poder emplear estas variables de alguna manera, se ha tomado la siguiente decisión: para cada director, se calculará su nota media, siendo esta la nota media de las películas que ha dirigido. Así, para cada película se creará una nueva variable, que será la nota media de los directores que la dirigen, siendo la nota de cada director la calculada anteriormente. Esta nueva variable es la que se usará en los análisis posteriores.

Seguidamente se muestra y explica el código empleado para crear esta nueva variable para el caso de los directores.

```
datos = datos %>%
  select(tconst, directors, averageRating) %>%
  separate(directors, sep = ",", into = paste0('Director', 1:84)) %>%
  pivot_longer(cols = paste0('Director', 1:84),
               names_to = 'Director_number',
               values_to = 'Director_code') %>%
  drop_na() %>%
  group_by(Director_code) %>%
  summarise(tconst = tconst, Nota = mean(averageRating))  %>%
  group_by(tconst) %>%
  summarise(NotaDirector = mean(Nota)) %>%
  right_join(datos)
```

Para crear la nueva variable se han seguido cinco pasos. Antes bien, se han seleccionado únicamente aquellas variables con las que se va a trabajar, para hacer el proceso más rápido y eficiente.

El primer paso consiste en crear una variable para cada director. Mediante la función *separate* indicamos que separe el contenido de la variable directors, a través de la coma, en ochenta y cuatro variables. El número de variables ha sido escogido mediante un análisis

previo, en el que se observó que la película que más directores poseía en el conjunto, era World of Death, de 2016, con ochenta y cuatro. Un ejemplo de como se tendrían ahora los datos, en particular para una película con dos directores, es el siguiente:

Tabla 5.3: Ejemplo limpieza de datos directores (1).

Película	Director 1	Director 2	Director 3	...	Director 84
Película 1	D1	D2	NA	...	NA

El segundo paso consiste en utilizar la función *pivot_longer*, para pasar de las ochenta y cuatro columnas anteriores a dos. Las observaciones tendrían ahora la siguiente forma:

Tabla 5.4: Ejemplo limpieza de datos directores (2).

Película	Número director	Código director
Película 1	Director 1	D1
Película 1	Director 2	D2
Película 1	Director 3	NA
Película 1	...	...
Película 1	Director 84	NA

En tercer lugar, eliminamos las filas con valores perdidos. Así, tendremos un resultado parecido al anterior pero en el que cada película aparece tantas veces como directores tenga. De este modo la película de nuestro ejemplo, *Película 1*, aparecería dos veces, una por el director *D1* y otra por el director *D2*.

El cuarto paso consiste en calcular la nota media de cada director. Agrupamos por la columna que contiene el código de los directores mediante *group_by* y hacemos la media de las notas de las películas que ha dirigido usando *summarise*. Añadimos los códigos de las películas, teniendo así el mismo conjunto que en el paso anterior con una nueva columna que indica la nota media del director. Siguiendo con el ejemplo:

Tabla 5.5: Ejemplo limpieza de datos directores (3).

Película	Código director	Nota director
Película 1	D1	7.6
Película 1	D2	8.9

Estamos ya en condiciones de realizar el último paso: calculamos la nueva variable, denominandola notaDirector. Agrupamos por el identificador de la película, usando de nuevo *group_by*, y calculamos para cada una la media de los directores que han participado en ella. Así, para la película usada como ejemplo se tendrá el siguiente resultado:

Tabla 5.6: Ejemplo limpieza de datos directores (4).

Película	Nota media directores
Película 1	8.25

Por último, unimos esta nueva variable al conjunto de datos de partida. El razonamiento con los guionistas es análogo.

5.1.3. Géneros

Con la variable géneros se tiene un problema que guarda cierta similitud con el anterior: se trata, de nuevo, de una variable tipo cadena. En la descripción de los datos realizada por el IMDb se especifica como, en su base de datos, una película puede tener hasta tres géneros diferentes, y, además, al haber eliminado todas aquellas películas con campos incompletos, sabemos que cada título está clasificado con al menos uno.

Los géneros vienen expresados como cadena, separando unos de otros mediante comas. Una vez más, hay que plantearse cómo usar estos datos para los estudios posteriores. Nuevamente, tratar los datos como factores acarrearía problemas: no sería comparable una película catalogada como drama con otra catalogada con drama y terror.

La solución que se ha tomado es la siguiente: mediante un estudio previo, se ha observado que el número de géneros diferentes es algo superior a treinta, por lo que, como el número no es muy elevado, se crearán nuevas variables, tantas como géneros distintos haya. Estas variables serán binarias, indicando para cada película si esta es o no del género que se señala.

A continuación se expone y describe el código utilizado:

```
datos = datos %>%
  select(tconst, genres) %>%
  separate(genres, ",", into = c("Genero1", "Genero2", "Genero3")) %>%
  pivot_longer(cols = c("Genero1", "Genero2", "Genero3"),
               values_to = "Genero") %>%
  select(-name) %>%
  drop_na() %>%
  mutate(valor = 1) %>%
  pivot_wider(names_from = Genero, values_from = valor, values_fill = 0) %>%
  mutate(across(where(is.character) &
                  !tconst, as.factor)) %>%
  full_join(datos)
```

Los dos primeros pasos son análogos a los del estudio anterior: una vez seleccionadas las variables que se van a utilizar, se emplea la función *separate* para separar la variable género en tres nuevas columnas, una por cada género posible. Tras esto, con la función *pivot_longer* pasamos de estas tres columnas a dos, de las cuales nos interesa únicamente la segunda, que contiene el nombre de los géneros. Por último, eliminamos aquellas observaciones incompletas, apareciendo de esta forma cada película tantas veces como

géneros tenga. Siguiendo el mismo ejemplo que antes, suponiendo que la *Película 1* esta clasificada como drama y terror, obtendremos lo siguiente:

Tabla 5.7: Ejemplo limpieza de datos géneros (1).

Película	Género
Película 1	Drama
Película 1	Terror

El próximo paso será usar la función *mutate* para crear una nueva columna de unos. Tras esto, usaremos la función *pivot_wider* utilizando esta columna y la columna que contiene los géneros: se crearán tantas variables como géneros distintos haya, y cada título tendrá un uno únicamente en aquellas columnas correspondientes a los géneros que posean. El resto de columnas tendrá valores perdidos, que se sustituyen por cero con el argumento *values_fill*. El resultado será el siguiente:

Tabla 5.8: Ejemplo limpieza de datos géneros (2).

Película 1	Drama	Comedia	Terror	...	Acción
Película 1	1	0	1	...	0

Por último, usando de nuevo *mutate* cambiaremos estas variables a tipo factor. Finalmente los unimos al dataset anterior, consiguiendo el resultado que deseamos.

Antes de continuar, destacar que al realizar estas modificaciones se observa cómo aparecen cuatro géneros que no son lógicos en una película: *Talk-Show*, *Short*, *Reality-TV* y *Game-Show*. Como los títulos que pertenecen a estos géneros son una minoría, se eliminan del estudio:

```
datos = datos %>%
  filter(`Talk-Show` == 0, Short == 0,
         `Reality-TV` == 0, `Game-Show` == 0) %>%
  select(-`Talk-Show`, -Short, -`Reality-TV`, -`Game-Show`)
```

5.1.4. Archivo principals

Realizados todos los pasos anteriores, ya estamos en posición de leer el archivo *title.principals*. Este archivo no fue cargado con el resto debido a su magnitud, pero, ahora, al haber simplificado nuestro conjunto inicial, al leerlo seleccionaremos únicamente aquellas observaciones correspondientes a películas que se encuentran en los datos anteriores, reduciendo significativamente su tamaño.

Teniendo presente que el archivo hace referencia al equipo principal de cada película, notemos que consta, para cada observación, de tres variables: una indicando la película, mediante *tconst*; otra para el código identificativo de la persona y otra para la función que desempeña dicha persona en la película. Así, si en particular para la película del

ejemplo, *Película 1*, se tienen guardados dos actores y un compositor, se mostraría como sigue:

Tabla 5.9: Ejemplo distribución de datos del equipo principal.

Película	Persona	Función
Película 1	P1	Actor
Película 1	P2	Actor
Película 1	P3	Compositor

De nuevo, cada persona viene identificada por un código único que es una cadena. Para trabajar con estos datos e incorporarlos a los análisis, se usará el mismo razonamiento que con los directores y guionistas, calculando primero para cada persona su nota media, siendo esta la media de las notas de las películas en las que ha participado. En este caso hay que recalcar un aspecto significativo: una misma persona puede haber participado en películas con roles distintos. Por ejemplo, un individuo puede haber protagonizado alguna película y dirigido otras. Surge entonces la siguiente cuestión: ¿debe tener para ambas funciones la misma calificación? Teniendo en cuenta que una misma persona puede ser muy buena desempeñando un trabajo y pésima para otro, en este estudio se ha considerado que la respuesta a dicha pregunta es negativa: un individuo tendrá diferente nota según la función que desempeñe.

Por tanto, siguiendo en la línea anterior, para cada persona y función se calculará su nota media, pudiendo tener un mismo individuo más de una nota, dependiendo de los roles que haya ejercido a lo largo de su carrera. Tras esto, para cada película se determinarán una serie de notas: la nota de los actores, que será la media de las notas de los actores que participan en la película; la nota de los compositores, que será la media de las notas de los compositores que intervienen en la película, etc.

Para realizar estos pasos, se ha usado en primer lugar el siguiente código, modificando el conjunto del equipo principal. Se muestra y se detalla a continuación:

```
principals = principals %>%
  left_join(ratings) %>%
  group_by(nconst, category) %>%
  summarise(nota = mean(averageRating, na.rm = T)) %>%
  full_join(principals) %>%
  group_by(tconst, category) %>%
  summarise(nota = mean(nota)) %>%
  pivot_wider(names_from = category, values_from = nota)
```

Las fases seguidas son similares a las anteriores: primero, tras unir con las puntuaciones, se agrupa usando *group_by* por persona y categoría, para calcular, para cada par, la nota media indicada anteriormente mediante *summarise*. Tras esto, tenemos para cada persona y función su nota correspondiente. En segundo lugar, volvemos a enlazar con el archivo que contiene todos los datos, teniendo así el conjunto inicial con una nueva columna correspondiente a las notas medias, como se indica a continuación:

Tabla 5.10: Ejemplo limpieza de datos equipo principal (1).

Película	Persona	Función	Nota
Película 1	P1	Actor	6.8
Película 1	P2	Actor	7.3
Película 1	P3	Compositor	5.6

Por último, agrupamos por películas y categorías, y calculamos la media de las notas, teniendo ya el objetivo: para cada película la nota media de los actores, la de los compositores, etc. Con la función *pivot_wider* pasamos estos datos de filas a columnas, obteniendo el siguiente resultado:

Tabla 5.11: Ejemplo limpieza de datos equipo principal (2).

Película	Actor	Actriz	...	Compositor	Dir. fotografía
Película 1	7.05	NA	...	5.6	NA

El problema de estos datos es la gran cantidad de valores perdidos. Por ejemplo, solo para treinta y cuatro películas se tienen recogidos los datos de la categoría *archive_sound*. Teniendo en cuenta que para realizar los análisis no vamos a trabajar con observaciones incompletas, hay que decidir con qué categorías quedarse. Finalmente estas son cuatro: actores, actrices, directores de fotografía y compositores, quedándonos en último término con un total de 105655 observaciones.

Con objeto de perder menos datos, fusionamos las categorías actores y actrices en una, haciendo la media de ambas variables. Eso sí, teniendo en cuenta que puede influir que los actores principales de una película sean solo mujeres o sólo hombres, se crea una nueva variable categórica, *sex*, con tres niveles: *both*, si hay protagonistas de ambos sexos en la película; *man*, si estos solo son hombres; y *woman* en caso contrario. Realizamos estos cambios y fusionamos con el conjunto que posee todas las variables:

```
datos = principals %>%
  mutate(actors = mean(c(actor, actress), na.rm = T),
         sex = 'both', sex = replace(sex, is.na(actress), 'man'),
         sex = replace(sex, is.na(actor), 'woman')) %>%
  filter(!is.na(actors), !is.na(composer), !is.na(cinematographer)) %>%
  select(tconst, actors, sex, cinematographer, composer) %>%
  left_join(datos)
```

5.2. Análisis exploratorio y descriptivo de los datos

Antes de realizar los estudios propuestos en la introducción se va a efectuar un análisis exploratorio de los datos. Además de las variables mostradas en la tabla 5.1 contamos con las nuevas calculadas en la sección anterior: veinticuatro de ellas corresponden a las variables binarias de los géneros y cinco a las notas del equipo de la película. Por último,

tenemos la nueva variable categórica *sex*, que indica la presencia o ausencia de ambos sexos en los papeles protagonistas de la película. Así, tenemos en total más de cuarenta variables. Sin embargo, hay que tener en cuenta que algunas de ellas no nos servirán en los estudios posteriores, pues son solo descriptivas; tal es el caso del título de la película, por lo que, realmente, disponemos de treinta y cuatro.

Con el fin de realizar el análisis descriptivo previo de las mismas, se explorarán en primer lugar las variables categóricas y posteriormente las numéricas.

5.2.1. Variables categóricas

En el conjunto hay un total de veintiséis variables categóricas. Entre ellas se encuentran la variable *isAdult*, que indica si una película es o no para adultos, y la variable *sex*, ya descrita. Para visualizar como están distribuidas estas variables entre las observaciones se realiza la siguiente representación:

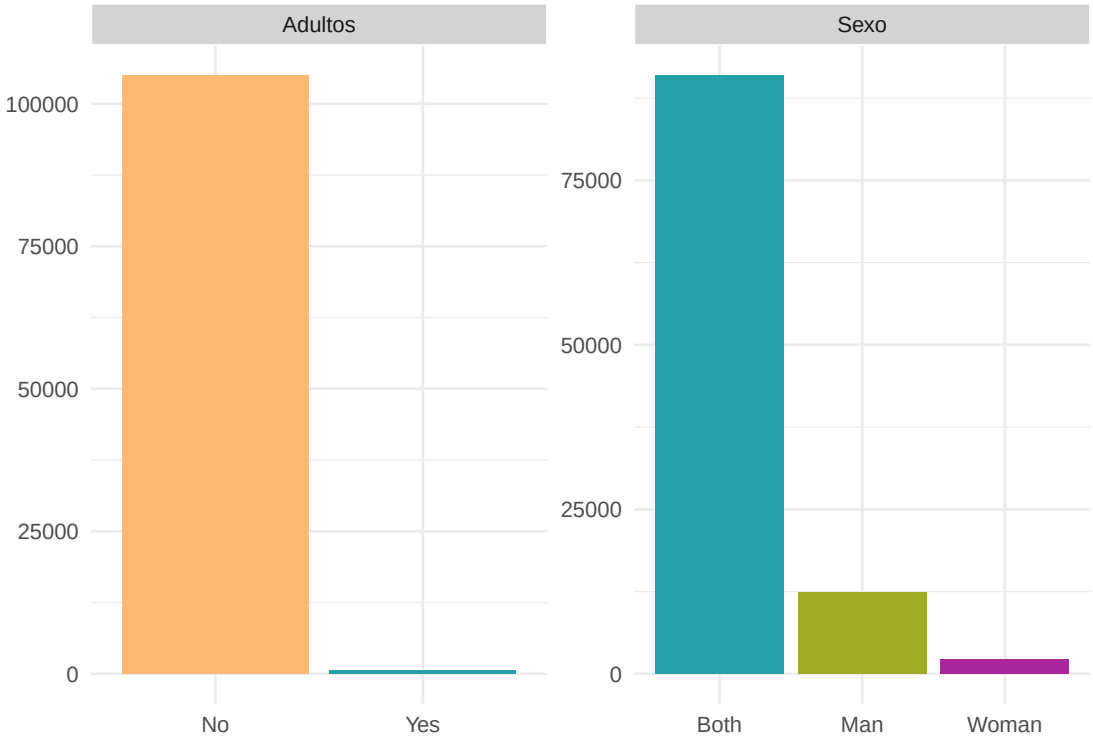


Figura 5.1: Distribución de la variable sexo y la variable mayor de edad.

Se observa que el porcentaje de películas calificadas como contenido para adultos es mínimo, representando menos del 1 % de las observaciones. Por otro lado, examinando el gráfico de la derecha, comprobamos que la mayor parte de las películas están protagonizadas por personas de ambos sexos. Pese a todo, la cantidad de películas protagonizadas únicamente por hombres es casi seis veces superior a la cantidad de películas protagonizadas únicamente por mujeres, lo cual parece significativo y resulta por tanto interesante destacar.

Las veinticuatro variables restantes corresponden, como se ha mencionado al principio de esta sección, a los géneros. Seguidamente, realizamos la misma representación gráfica que antes, poniendo de relieve cuantas películas hay de cada género:

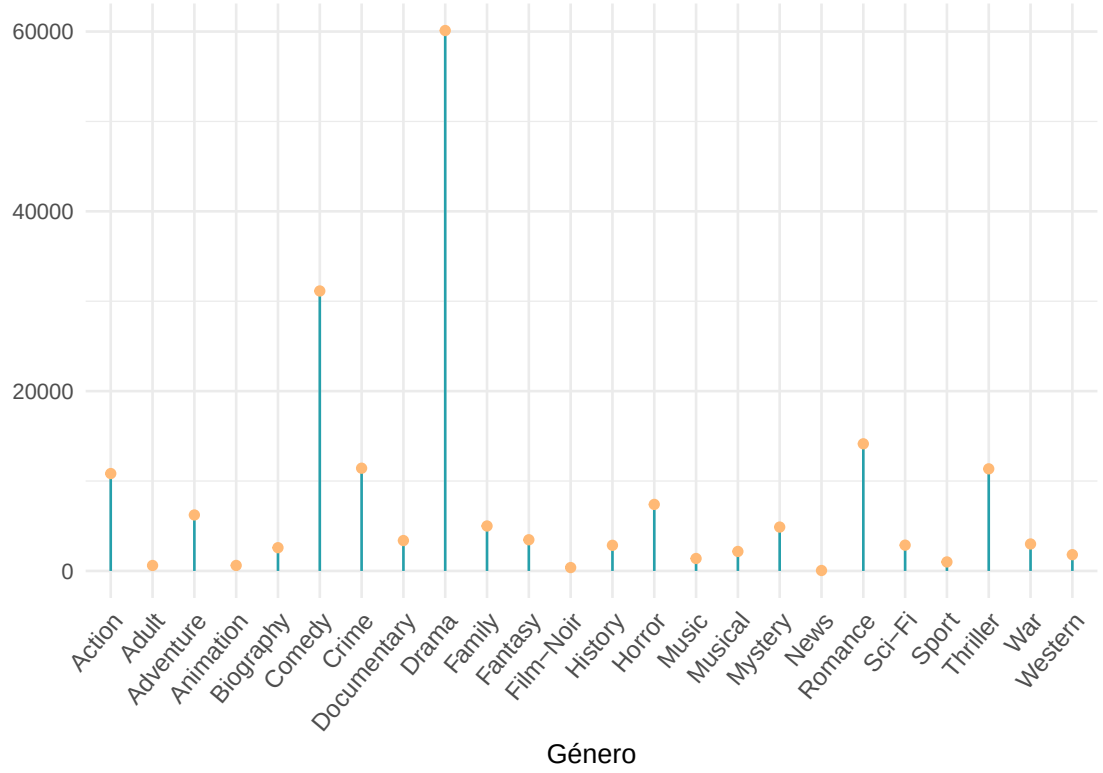


Figura 5.2: Distribución de los géneros.

El género que más se repite, y con diferencia, es el dramático, seguido de comedia, romance, acción y thriller. Tenemos por tanto que más de la mitad de películas del conjunto de datos están clasificadas como drama, dato que no pasa desapercibido. Con todo, no debemos olvidar que una misma película puede pertenecer a más de un género a la vez. Esto nos lleva a plantearnos si existe algún tipo de patrón en el comportamiento de los géneros siendo más común encontrar a algunos juntos que a otros.

En primer lugar, estudiemos qué tríos de géneros se repiten con más frecuencia. Para ello haremos uso de el paquete *tidytext* y en particular de la función *unnest_tokens*, ambas detalladas en 2.2.2. Representando los resultados se obtiene lo siguiente:

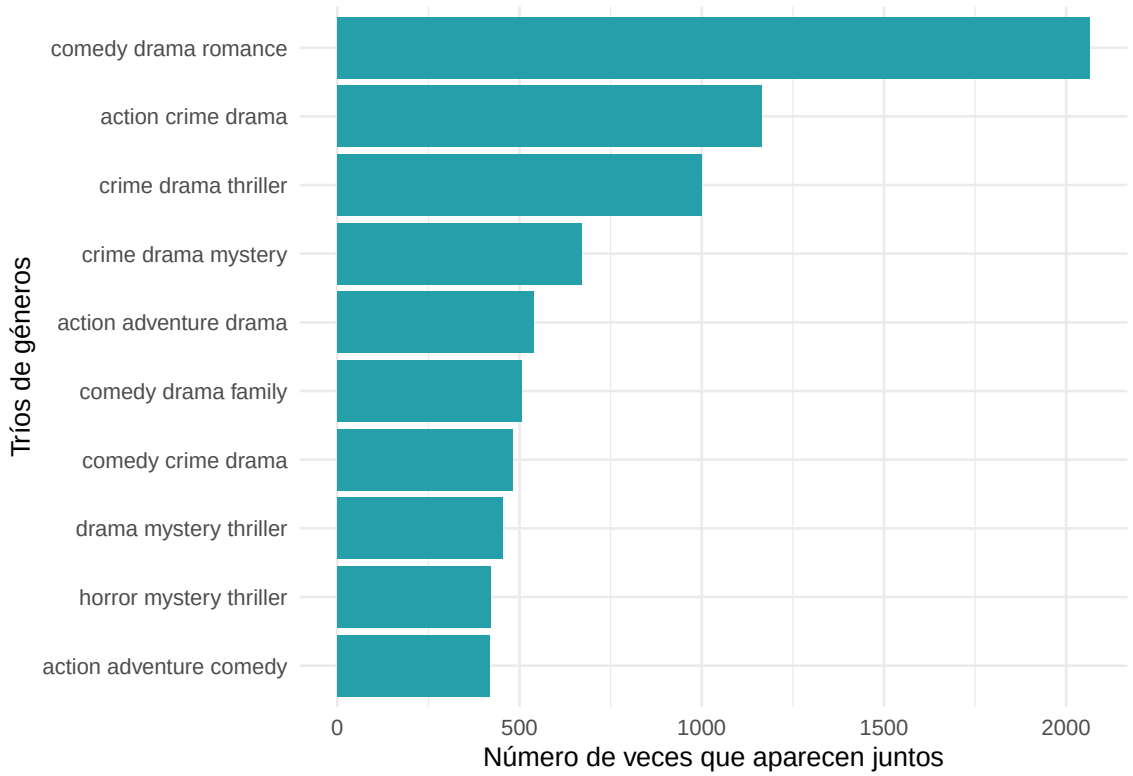


Figura 5.3: Ranking de las diez tríadas de géneros más frecuentes.

La tríada de géneros que más veces se repite es comedia, drama y romance, contando con más de dos mil películas con dicha clasificación. Posteriormente, se encuentran acción, crimen y drama, seguido de crimen, drama y thriller. Subrayamos en primer lugar que entre estos tres casos se encuentran los cinco géneros más usuales, y en segundo como en los diez casos destacados, exceptuando los dos últimos, se encuentra el género dramático, el más frecuente de todos.

Lo próximo será estudiar qué pares de géneros aparecen juntos con más frecuencia. De nuevo, haremos uso de la función *unnest_tokens* del paquete *tidytext*, pero esta vez habrá que hacer alguna modificación. La función *unnest_tokens*, al recibir el parámetro n igual a dos, busca bigramas, es decir, pares de palabras que salen juntas. El problema es que si esta función recibe tres géneros, digamos crimen, drama y romance, solo cuenta los pares de palabras que aparecen literalmente juntas, es decir, en este caso solo contaría que crimen aparece junto a drama y drama junto a romance, pero no tendría en cuenta el par formado por crimen y romance, que también nos atañe. Por tanto, habrá que solucionar este inconveniente de alguna forma. El código usado para solventarlo y su explicación se muestra a continuación:

```
datos %>%
  select(genres) %>%
  mutate(genres = gsub("-", "_", genres), genres = tolower(genres)) %>%
  tibble() %>%
  separate(genres, ",", into = c("one", "two", "three")) %>%
  drop_na() %>%
  select(-two) %>%
```

```
  unite("bigram", one, three, sep = " ") %>%
  rbind(estudio_generos %>%
          unnest_tokens(bigram, genres, token = "ngrams", n = 2) %>%
          drop_na()) %>%
  count(bigram)
```

Tras transformar el conjunto de datos en un *tibble*, objeto de R similar a los *dataframes* que facilita el uso de las funciones del paquete *tidytext*, nos centraremos primero en el problema comentado anteriormente. Para solventar el hecho de que *unnest_tokens* en las películas con tres géneros ignora el par formado por el primer y el último, usamos en primer lugar la función *separate* para separar la columna *genres* en tres, una por cada género posible. Después, mediante la función *drop_na* eliminamos aquellas observaciones incompletas, es decir, nos quedamos únicamente con aquellas películas que tengan información en las tres columnas creadas, esto es, que tengan exactamente tres géneros. Por último, eliminamos la segunda columna, y unimos las dos restantes en una mediante la función *union*. De este modo ya estamos en posesión de los pares de géneros formados por el primero y el último de cada trío.

Tras esto, unimos este resultado al obtenido de aplicar la función *unnest_tokens* al conjunto de datos original, teniendo ya por tanto todos los pares posibles. Los diez pares de géneros que más se repiten se muestran a continuación:

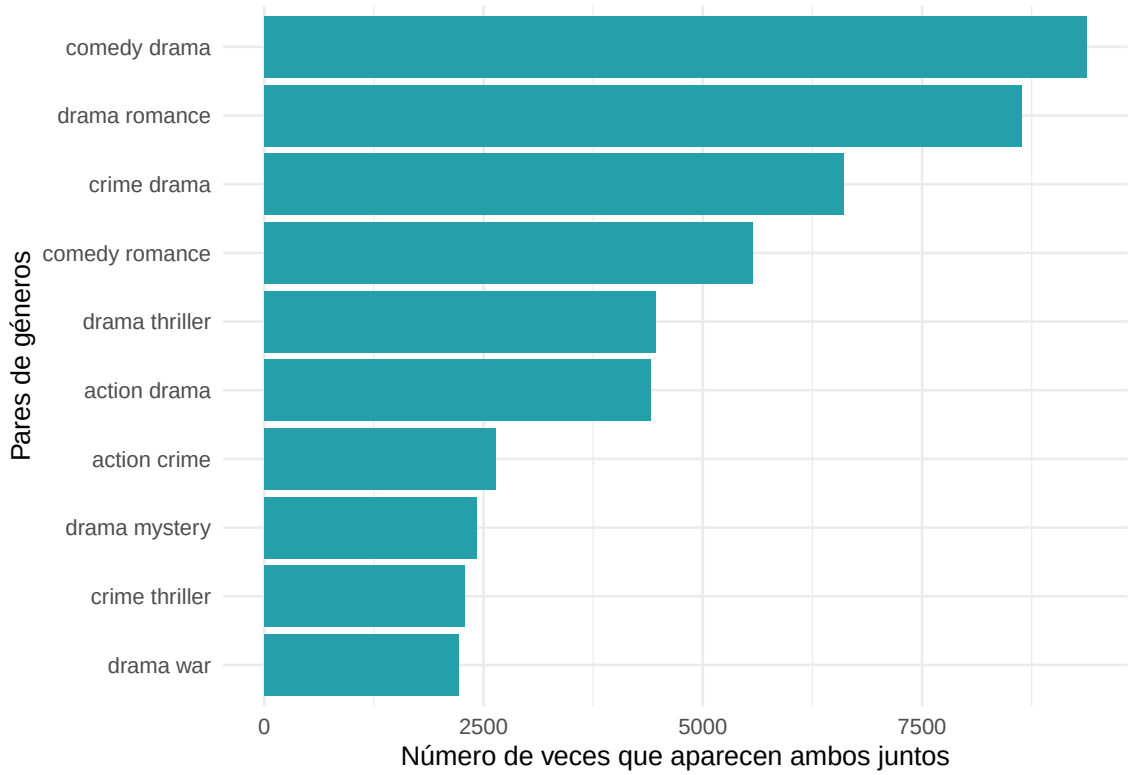


Figura 5.4: Ranking de los diez pares de géneros más frecuentes.

Comedia y drama es el par más frecuente, seguido muy de cerca por drama y romance. Por otro lado, drama y thriller aparecen juntos aproximadamente las mismas veces que

drama y acción, y, además, este último sale junto a crimen en un número considerable de ocasiones.

No obstante, a pesar de que este análisis nos aporta información relevante, más interesante sería saber cuál es la proporción de veces que un género sale junto a otro. Por ejemplo, sabemos de la gráfica anterior que comedia y drama salen juntos más de siete mil veces, pero teniendo en cuenta que hay más de sesenta mil películas catalogadas como drama, llegamos a la conclusión de que drama sale junto a comedia tan solo en un 13 % de sus apariciones, porcentaje bastante bajo. Teniendo esto en mente, observamos la siguiente tabla:

Género 1	Género 2	Proporción (%)
film_noir	drama	88.27
news	documentary	87.76
film_noir	crime	76.00
war	drama	73.93
biography	drama	72.04
history	drama	69.75
sport	drama	63.74
romance	drama	61.07
music	drama	58.41
crime	drama	57.79

En la tabla se muestra el porcentaje de veces que aparece el Género 1 junto al Género 2. Por ejemplo, más del 88 % de las películas clasificadas como cine negro lo estan también como drama y más del 76 % como de crimen. Recalcar que la aparición de drama en la segunda columna es casi absoluta, excepto en la segunda instancia. Estos resultados pueden ser interesantes de cara a los análisis posteriores, teniendo en cuenta que hay géneros que pueden estar muy relacionados.

Una vez estudiadas las variables categóricas, nos centramos en las numéricas.

5.2.2. Variables numéricas

Para empezar, mostramos una tabla conteniendo el resumen básico de cada una de las variables numéricas involucradas en el estudio:

Tabla 5.12: Resumen estadístico de las variables numéricas.

Variable	Mínimo	Media	Máximo	Desviación típica
Año	1906	1991.073	2021	22.655
Duración de la película	22	95.819	999	22.161
Nota actores	1	6.030	10	0.813
Nota compositor	1	6.063	10	0.868
Nota dir. de fotografía	1	6.068	10	0.843
Nota director	1	6.040	10	0.980
Nota guionista	1	6.044	10	0.962
Nota película	1	6.030	10	1.236
Número de votos	5	2862.074	2387673	29027.417

La película más antigua data de 1906 y se corresponde con *The Story of the Kelly Gang*, película basada en la novela *The Kelly Gang* de Arnold Denham, mientras que las más nuevas son de este mismo año. Cabe destacar que la media de los años se corresponde con el final de los noventa, hecho nada sorprendente si se tiene en cuenta que antes se producían una menor cantidad de películas puesto que los recursos para realizarlas no eran tan accesibles. Por otro lado, en cuanto a la duración, observamos que la película más corta consta tan solo de veintidós minutos, mientras que la más larga, *Mystrio (Uno... dos... tres pilyos!)*, consta de más de dieciséis horas. La duración media de las películas es algo superior a una hora y media.

Las notas obtenidas por los miembros del equipo son similares: la nota media oscila en torno a seis y hay desde puntuaciones muy bajas hasta puntuaciones muy altas. Lo mismo ocurre para las notas de las películas, que muestran más variación que las anteriores.

Por último, recalcar que el número de votos es la variable con mayor desviación típica y, observando los valores mínimo, medio y máximo, esto no es de extrañar: el número de votos que recibe una película, de media, es de algo menos de tres mil, pero nos encontramos con películas con menos de diez votos y con películas con más de dos millones. Es más, la mediana de esta variable, la cual no ha sido mostrada en la tabla anterior, es ochenta y siete indicando que la mitad de las películas cuentan con menos de ochenta y siete votos. Esta diferencia abismal entre media y mediana señala que la variable número de votos esta muy sesgada a la izquierda, siendo notablemente menor la cantidad de películas con un mayor número de votos. Liderando este ranking se encuentra *The Shawshank Redemption*, *Cadena Perpetua* en español, que viene apareciendo desde hace muchos años de las primeras en los rankings en cuanto a nota otorgada por los usuarios en las distintas plataformas.

Como se ha mencionado en el párrafo anterior, cada nota media de los integrantes del equipo oscila entorno a seis. Una pregunta natural que surge a raíz de este hecho es si el equipo de una película tiene notas similares o no. Para estudiar este supuesto, calculamos los coeficientes de correlación entre las notas de los integrantes del equipo:

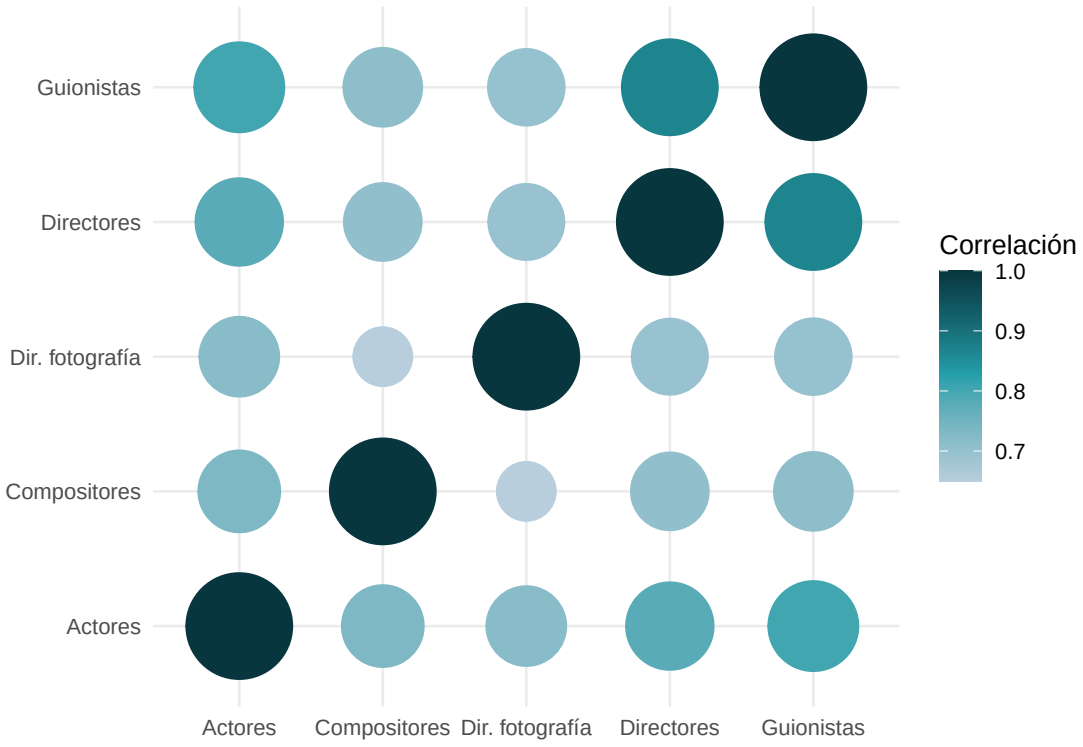


Figura 5.5: Correlación de las notas de los integrantes del equipo.

La respuesta a la pregunta anterior parece ser un sí: las diferentes notas están altamente correladas, y además positivamente. La correlación más intensa se produce entre directores y guionistas: a mejor nota del director, mejor del guionista y viceversa. Por el contrario, la más débil, aunque también positiva, es entre la nota de los compositores y de los directores de fotografía. Aún así, destacar que esta sigue siendo superior a 0.6, cifra considerable aunque no tan significativa. La alta correlación existente es relevante de cara a los estudios posteriores: quizás haya que aplicar alguna técnica para reducir su impacto.

Una vez descritas las características básicas de las variables, estamos en condiciones de realizar el primer estudio propuesto: predecir el éxito de una película, medido a través de su puntuación, a partir del resto de información disponible sobre la misma.

Capítulo 6

Modelos de regresión

Para predecir el éxito de una película a través de su puntuación lo que se hará es predecir esta variable, es decir, *averageRatings*, a través de las diferentes variables explicativas detalladas en la sección anterior mediante el uso de diferentes técnicas de regresión.

En el apartado previo se ha realizado un análisis descriptivo de las variables explicativas que usaremos en esta sección, tanto las categóricas como las numéricas. Ahora bien, antes de aplicar las técnicas de regresión sería interesante ver cómo se comportan estas variables en función de la variable objetivo, por lo que agregamos seguidamente unos breves análisis descriptivos.

6.1. Breves análisis descriptivos

En primer lugar usamos los gráficos boxplot para representar la variable correspondiente a las puntuaciones en función de los diferentes géneros:

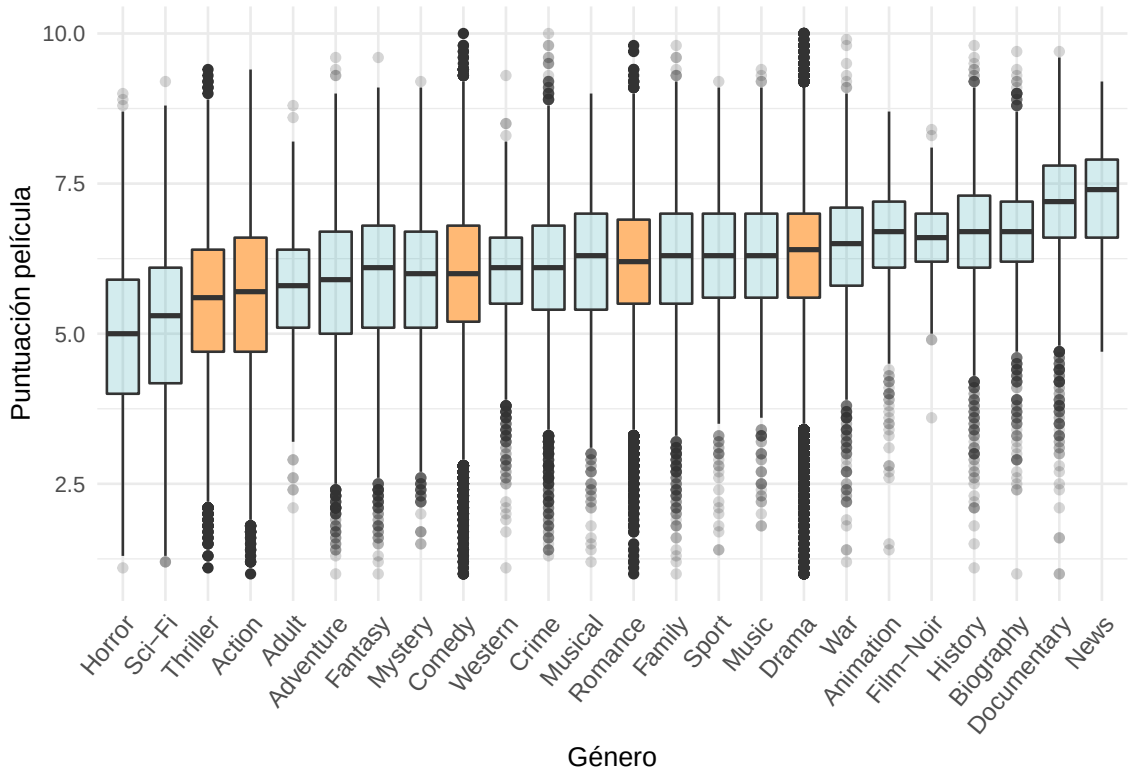


Figura 6.1: Estudio de la puntuación de una película en función de su género.

Como se aprecia en la representación superior, se ha concedido un color diferente a cada uno de los cinco géneros más comunes, determinados en los análisis descriptivos previos. Los diferentes *boxplot* (diagrama de caja y bigotes) están ordenados en orden creciente en función de la mediana, siendo los géneros con menores calificaciones terror y ciencia ficción y los géneros con mayores calificaciones noticias y documental.

En lo que respecta a los cinco géneros más comunes, observamos como el dramático es el que obtiene calificaciones superiores, y thriller y acción los que menores. Comedia y romance se encuentran muy cerca, situándose en la mitad del gráfico. Destacar que el número de outliers es considerable en todas las categorías, correspondiendo principalmente a películas con puntuaciones inferiores. En particular, en el género dramático y romántico la media se sitúa en torno a seis, pero hay una cantidad considerable de producciones con notas inferiores al tres. Los géneros que encabezan o finalizan el gráfico son los que menos outliers poseen.

En definitiva, se concluye que aunque las puntuaciones no difieran demasiado para los diferentes géneros, sí que se observan algunas desigualdades que pueden tener algún tipo de repercusión posterior.

Seguidamente estudiamos las puntuaciones de las películas en función de las notas de los diferentes integrantes del equipo:

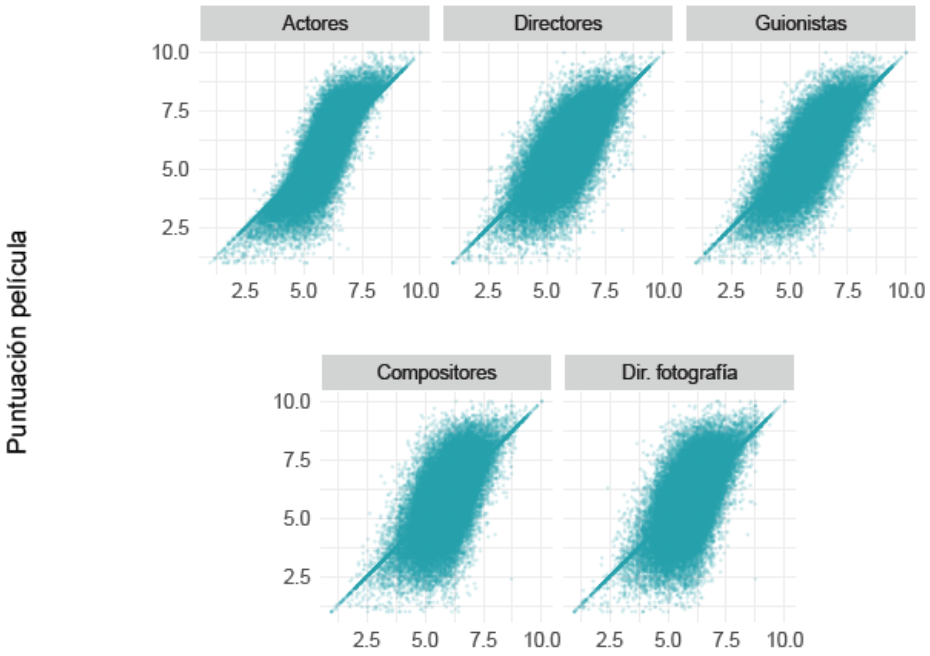


Figura 6.2: Puntuación de la película en función de las notas de los miembros del equipo.

En cada gráfico se ha representado en el eje de abscisas la nota de los miembros del equipo correspondiente y en el de ordenadas la de las películas.

La representación obtenida para los compositores y los directores de fotografía es bastante similar, produciendo muy buenos resultados en los extremos. Lo mismo les ocurre a los gráficos correspondientes a directores y guionistas, que ofrecen visualizaciones casi análogas. En estos dos últimos los puntos se encuentran más sobre la recta $y = x$ que en los anteriores, indicando que la relación entre las notas de estos integrantes y la de las películas es más sólida. Cabe destacar que el caso de los guionistas parece ser a simple vista el mejor: su representación es la más achatada, indicando mayor proximidad de los puntos a la recta.

Por otro lado, el gráfico de los actores es el más diferente, presentando una forma ondulada. En él se observa que la nota de los actores es similar a la de las películas cuando ambas se sitúan alrededor al seis. En el extremo superior se aprecia cómo los actores con notas superiores a siete participan en películas con notas más altas, mientras que en el inferior se observa cómo los actores con notas inferiores a cinco producen películas con menor nota aún.

Finalmente, se estudia cómo se comportan las puntuaciones en función de las variables referentes al número de votos y a la duración:

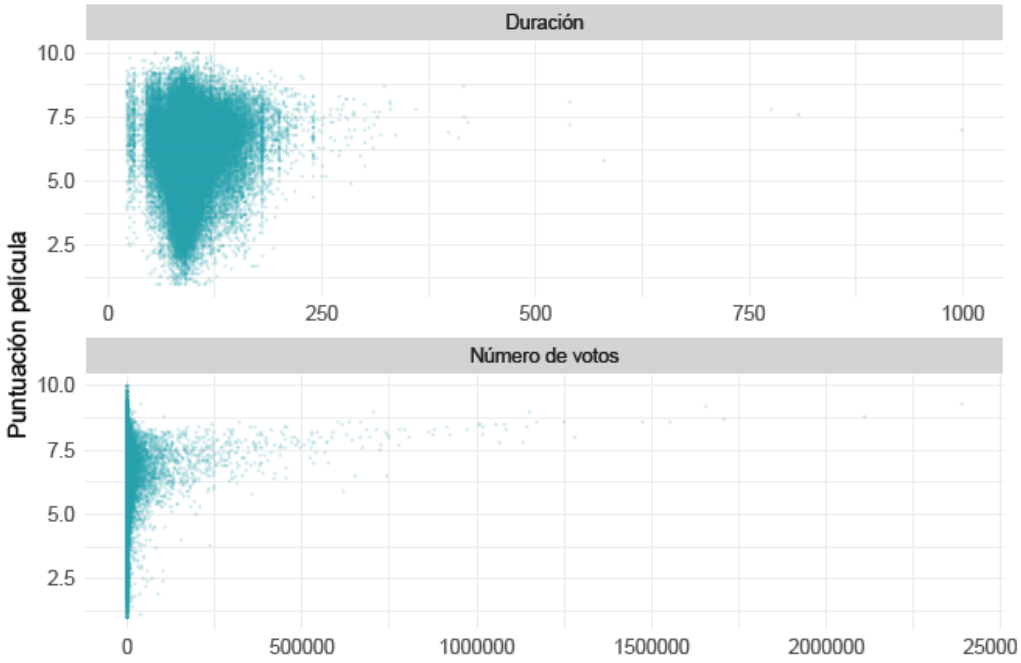


Figura 6.3: Puntuación de la película en función de la duración y el número de votos.

Las representaciones anteriores ponen de manifiesto lo comentado en la sección dedicada al análisis descriptivo de las variables: la variación entre sus valores es muy amplia, especialmente en el número de votos. En este caso se observa cómo los outliers correspondientes a las películas más votadas tienen notas altas, superiores al siete y medio en la mayor parte de los casos. En lo referente a la duración no se observa nada especial: la mayor parte de las películas duran menos de tres horas, pero no parece que la duración afecte a la nota de la película.

Al ser tan amplio el rango de estas variables y poseer bastantes valores atípicos, lo mejor es realizar alguna transformación para poder sacar mejores conclusiones de la visualización de los datos. A continuación se expone la misma gráfica que antes, aplicando la transformación logarítmica al eje de abscisas:

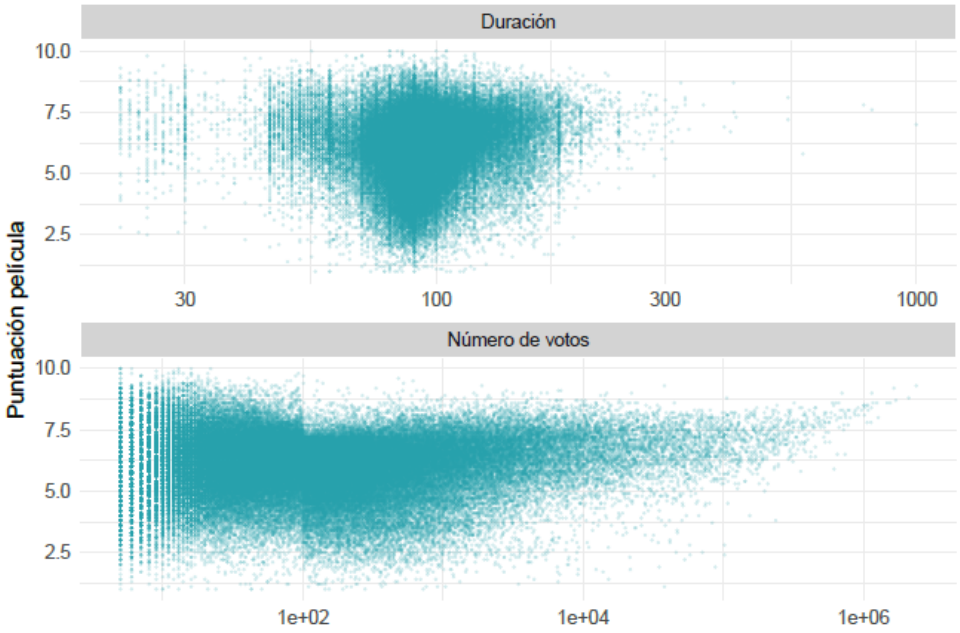


Figura 6.4: Puntuación de la película en función de la duración y el número de votos. Transformación logarítmica.

Las películas con una duración superior están calificadas con notas entre seis y nueve. Por otro lado, las que duran menos presentan todo tipo de puntuaciones: desde altas hasta bajas. Lo mismo ocurre con las que se encuentran en la media. Sin embargo, se observa cómo la nube de puntos se estira a medida que la duración aumenta: parece ser que las películas con duración superior adquieren mayor puntuación.

Por su parte, el comportamiento de la puntuación en función del número de votos se asemeja a lo que se comentaba en la otra gráfica: las películas con mayor número de votos son las que mejor nota tienen. Es más, a partir de un cierto número de votaciones es posible acotar la puntuación que tendrá la película. Antes de dicha cifra el número de votos no parece influir en la nota, encontrando películas que han sido calificadas con notas tanto muy altas como muy bajas.

Una vez estudiado cómo se comporta la variable objetivo en función de las variables explicativas más representativas, podemos iniciar los análisis de regresión. Para ello, aplicaremos dos técnicas: regresión lineal múltiple, escogida por su simplicidad pero también por su gran potencia en muchos contextos; y random forest, seleccionada por su capacidad de alcanzar una mejor precisión y estabilidad del modelo.

Para aplicar estas técnicas usaremos el paquete *tidymodels* y algunos de los paquetes asociados a él: *rsample*, *recipes*, *tune*... detallados en 2.2.3. Con objeto de hacer los dos modelos escogidos comparables, se lleva a cabo la división del conjunto de datos en conjunto de entrenamiento y conjunto test, necesaria para evaluar adecuadamente el rendimiento de cada modelo; será la misma para ambos, así como las particiones generadas por la técnica de validación cruzada. La creación de dichos conjuntos se obtiene con el código aportado a continuación:

```
data_split = initial_split(datos, prop = 0.7, strata = averageRating)
train_data = data_split %>% training()
test_data = data_split %>% testing()

data_folds = vfold_cv(train_data, strata = averageRating, v = 8)

n1 = nrow(train_data)
n2 = nrow(test_data)
```

Recalcar que al recibir en el argumento *strata* la variable que contiene la puntuación de las notas, indicamos que se realice estratificación por dicha variable, esto es, que los valores de las notas estén compensados en los conjuntos entrenamiento y test. Por su parte, señalar que para la validación cruzada se han tomado ocho subconjuntos en ambos modelos. Por otro lado, ambos modelos utilizan la misma receta, elaborada con el paquete *recipe*:

```
receta = recipe(averageRating ~ ., data = train_data) %>%
  update_role(tconst, primaryTitle, originalTitle,
              genres, directors, writers, new_role = "ID") %>%
  step_dummy(sex)
```

En ella detallamos el objetivo: predecir el valor de *averageRatings*, las calificaciones de las películas, en función del resto de variables. Además, como comentábamos al inicio de la sección 5.2 existen variables que aportan información que sólo es descriptiva y que no podemos usar en un análisis estadístico, como es el caso del título de la película. Con la orden *update_role* y el argumento *ID* indicamos que incluya estas variables en los análisis, pues aportan información valiosa, pero que no las use para realizar predicciones. Por último, recalcar que la variable *sex* es una variable categórica con tres niveles: *both*, *man* y *woman*, por lo que para poder emplearlas en los análisis de regresión se crearán dos variables ficticias o variables *dummy* mediante el paso *step_dummy*.

Tras haber realizados estos pasos, podemos realizar el primero de los análisis.

6.2. Regresión lineal

A continuación vamos a señalar y comentar los pasos seguidos y los resultados obtenidos aplicando el modelo de regresión lineal, teniendo en cuenta que la descripción teórica de esta técnica viene detallada en el 2.1.1.

El primer paso será definir el modelo a utilizar: en nuestro caso será regresión lineal utilizando el modelo de regresión lineal múltiple.

```
lr_mod =
  linear_reg() %>%
  set_engine("lm") %>%
  set_mode('regression')
```

En segundo lugar, habrá que definir el *workflow*, proporcionando la receta junto al modelo seleccionado:

```
lr_wflow =
  workflow() %>%
  add_model(lr_mod) %>%
  add_recipe(receta)
```

Una vez realizados los dos pasos anteriores, podremos aplicar el modelo en el conjunto de entrenamiento, esto es, entrenar el modelo.

```
lr_fit = lr_wflow %>%
  fit(data = train_data)
```

Ahora bien, con el fin de ver cómo se han ajustado los datos en el conjunto de entrenamiento, analizamos algunas medidas de bondad del ajuste.

Para empezar, estudiemos algunos de los coeficientes más importantes a la hora de determinar cómo de bueno es el ajuste del modelo:

Tabla 6.1: Estadísticos de bondad del ajuste para la regresión lineal múltiple.

R^2 ajustado	AIC	BIC
0.7783359	129785	130125.8

Observando el valor del R^2_{adj} se deduce que con este modelo queda explicada un 77.84 % de la variabilidad total, por consiguiente, podemos considerar que el modelo tiene buena capacidad explicativa. Los valores de los estadísticos AIC y BIC se usarán para la comparación con los modelos posteriores.

En segunda instancia, estudiemos qué variables han resultado significativas en el modelo y cuáles no lo han sido. Recordemos, que al realizar el modelo de regresión lineal múltiple, dada una variable objetivo Y y n variables explicativas $X_j, j = 1 \dots n$, se planteaba el modelo:

$$Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_n X_n$$

Así, para ver si la variable X_j influye o no en el el mismo, se realizaba el siguiente contraste:

$$\begin{cases} H_0: & \beta_j = 0 \\ H_1: & \beta_j \neq 0 \end{cases}$$

Por tanto, a un nivel de significación de 95 % podemos concluir que una variable no es significativa si el p-valor de dicho contraste es inferior a 0.05. A continuación mostramos las variables que no han resultado significativas en el modelo:

Tabla 6.2: Variables menos significativas del modelo de la regresión lineal múltiple.

Variable	P-valor	Variable	P-valor
News1	0.8455893	Thriller1	0.2699744
Comedy1	0.7528598	sex_woman	0.2022561
War1	0.7289192	Mystery1	0.1878850
startYear	0.4728105	Action1	0.1755498
Documentary1	0.4417282	Animation1	0.1390731
Fantasy1	0.3763578	isAdult1	0.0691304

Se concluye que hay nueve géneros que no influyen en la nota final de la película o para los cuáles al menos no se puede descartar la hipótesis nula, que asume que la posible variable predictora no influye en la variable objetivo. Entre ellos están acción, comedia y thriller, que se encontraban entre los géneros más comunes. El último caso, isAdult1 puede ser cuestionable, ya que está cerca del valor umbral planteado de 0.05. El año de la película tampoco ejerce influencia, indicando que la nota de la película no depende del año en el que ésta haya sido estrenada.

Por otro lado, para ver qué variables han sido las más relevantes, usamos la función *vip*, del paquete del mismo nombre, que nos proporciona el siguiente gráfico:

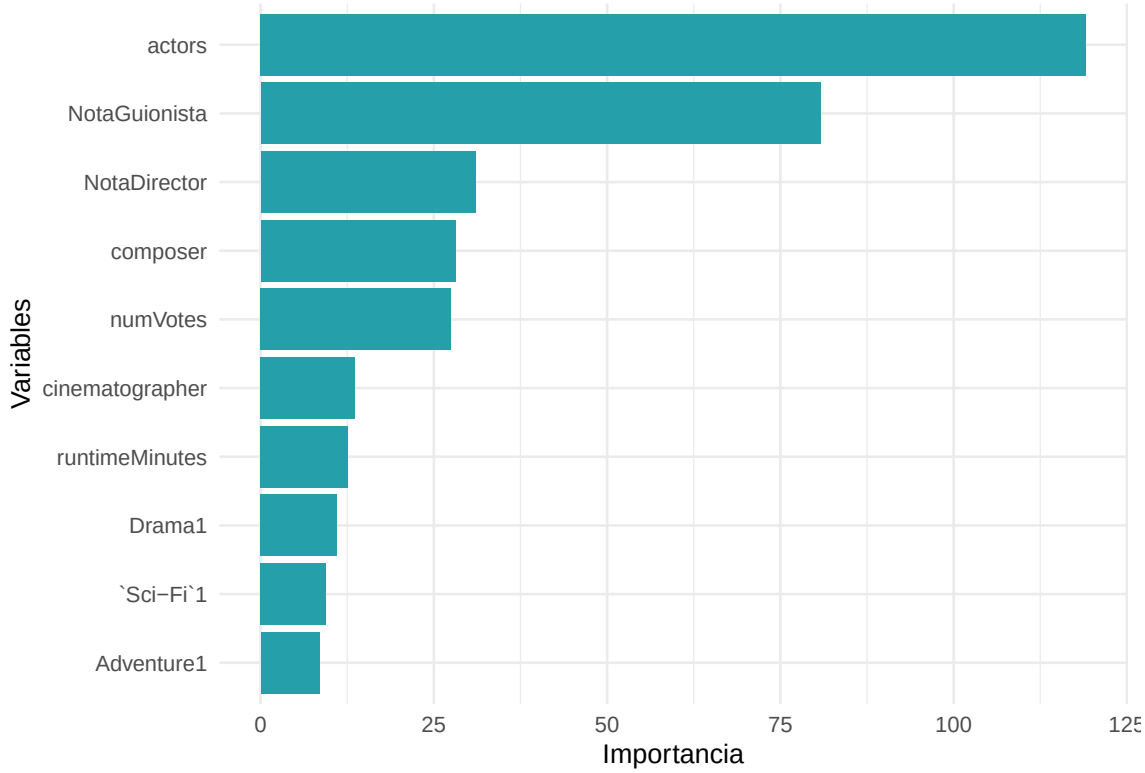


Figura 6.5: Variables más significativas en el modelo de regresión lineal múltiple.

Las variables que más influyen son las asociadas a las notas del equipo de la película, siendo la más influyente la calificación correspondiente a los actores. Le sigue, aunque con cierta distancia, la nota del guionista, y después la del director, siendo por tanto estas

las tres variables más significativas en el modelo. Por otro lado, destacar que los géneros que más influyen son el dramático, ciencia ficción y aventura. Además, los coeficientes β_j asociados al género de ciencia ficción y al de aventura son negativos, indicando que si una película está clasificada en alguno de estos géneros es probable que tenga menor puntuación, mientras que el dramático es positivo, indicando lo contrario. Esto coincide con el gráfico 6.1, en el que se observaba cómo las películas de ciencia ficción y aventura tenían menor calificación, mientras que las de drama tenían notas más elevadas.

Una vez analizados los resultados en el conjunto de entrenamiento, pasamos a estudiar el rendimiento del modelo en el conjunto test. Para ello, realizamos las predicciones de las puntuaciones en dicho conjunto usando la función *predict*. Además, se añade una columna con los valores reales de las puntuaciones, para poder obtener conclusiones acerca del ajuste obtenido.

```
lr_pred =
  predict(lr_fit, test_data) %>%
  bind_cols(test_data %>% select(averageRating))
```

Con objeto de analizar la bondad del modelo, calculamos el error absoluto medio, *MAE*; la raíz del error cuadrático medio, *RMSE* y el R^2 ajustado:

Tabla 6.3: Estadísticos de bondad del ajuste en la predicción para la regresión lineal múltiple.

MAE	RMSE	R^2 ajustado
0.4357412	0.5830311	0.7782978

Se observa como el R^2_{adj} es similar al obtenido en el conjunto de entrenamiento, indicando que no se ha producido *overfitting*, es decir, no hay sobreajuste. Por otro lado, teniendo en cuenta que el rango de las notas de las películas va desde cero hasta diez, se observa como el *MAE* y el *RMSE* son valores pequeños, lo cual indica que el ajuste obtenido es bastante razonable. El valor del *MAE* nos indica que, con datos distribuidos de manera similar a los que se tienen en el conjunto test, se espera un error de ± 0.44 en la media de las calificaciones de las películas.

Por último, para observar gráficamente la bondad del ajuste, representamos las calificaciones reales frente a las predicciones, incorporando la recta $y = x$ a la representación:

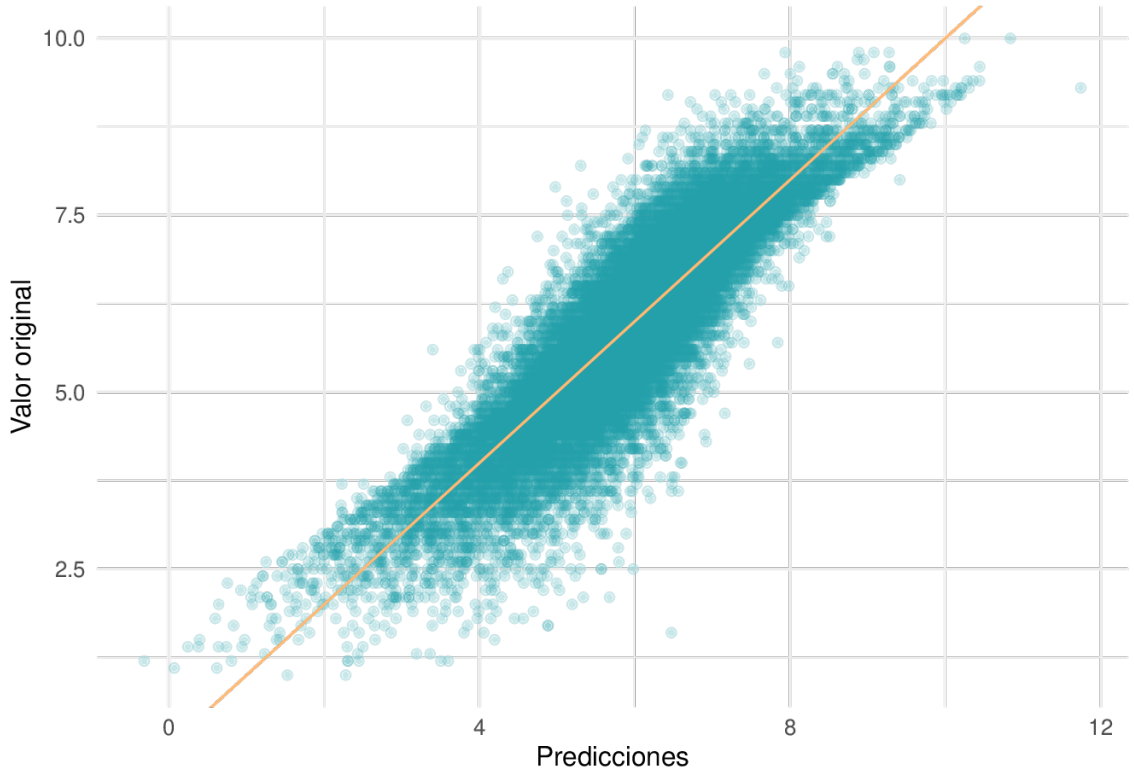


Figura 6.6: Puntuación de las películas en función de la predicción mediante regresión lineal múltiple.

En los extremos es donde el ajuste es mejor y en los valores centrales donde peor, aunque la mayoría de los puntos se sitúa a poca distancia de la recta, indicando un buen ajuste. Si se recuerda el gráfico 6.2, se observa que la primera representación, correspondiente a los actores, es parecida a la que se acaba de analizar. Esta similitud puede deberse al hecho de que la nota de los actores es la variable que más influye en el modelo, como se vio en el gráfico 6.5.

Se debe señalar que para realizar el ajuste del modelo se han asumido una serie de hipótesis. Para verificar la credibilidad de las conclusiones obtenidas, estas hipótesis deben ser ciertas, por lo que debemos realizar un diagnóstico del modelo. Para ello, se analizan los residuos, comprobando que cumplen cuatro hipótesis: normalidad, homocedasticidad, independencia y multicolinealidad.

En primer lugar, estudiemos la hipótesis de normalidad, realizando para ello un gráfico Q-Q. Estos gráficos nos permiten observar cuán cerca está la distribución de un conjunto de datos a una distribución deseada, que en este caso es la distribución normal. Para ello se representa en el eje de abscisas los cuantiles teóricos y en el de ordenadas los de la muestra. Si la nube de puntos obtenida se sitúa en torno a la recta $y = x$ se concluye que la distribución de la muestra coincide con la deseada. En nuestro caso, se obtiene lo siguiente:

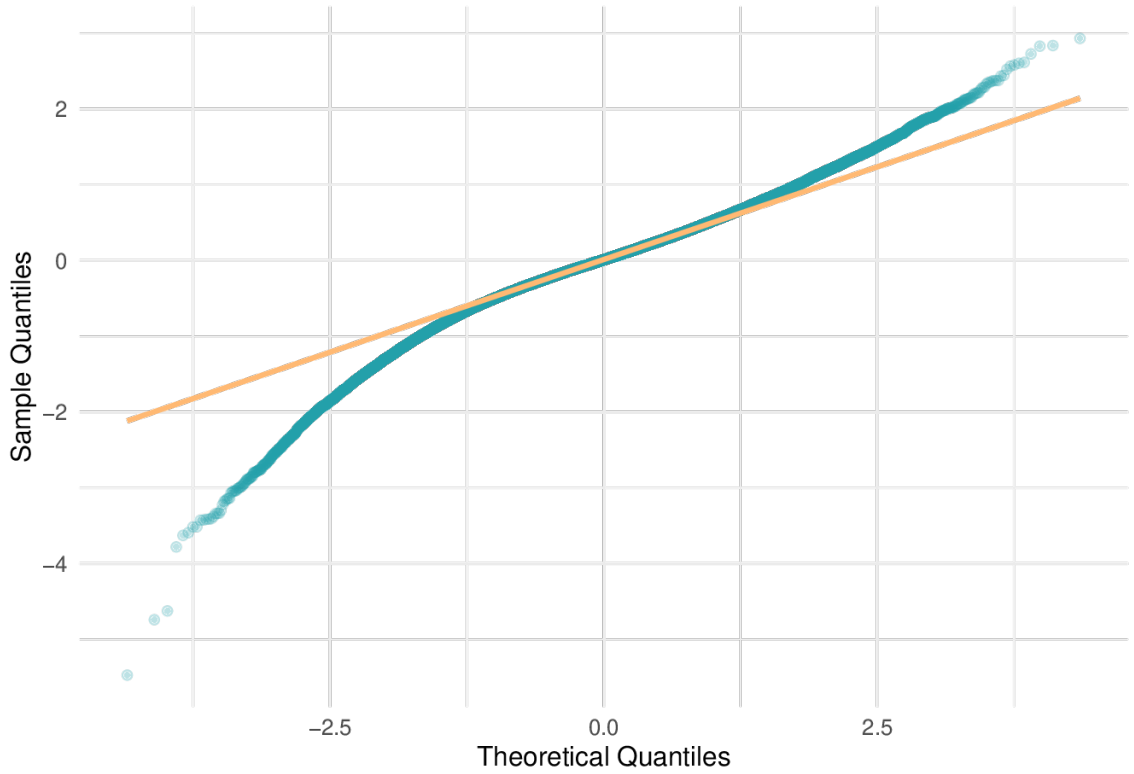


Figura 6.7: Hipótesis de normalidad de los residuos en el modelo de regresión lineal múltiple.

Los valores centrales se sitúan sobre la recta, pero no ocurre lo mismo con los de los extremos, debido a las observaciones atípicas. Por tanto, aunque parece ser que la hipótesis de normalidad es cierta, para asegurarnos, se realiza el test de Anderson-Darling:

```
    Anderson-Darling normality test
Estadístico = 283.9571
P-valor = 3.7e-24
```

El p-valor obtenido es casi cero por lo que podemos aceptar la hipótesis de normalidad.

En segundo lugar, estudiamos la hipótesis de homocedasticidad. Para ello, realizamos un gráfico de los residuos frente a los valores ajustados:

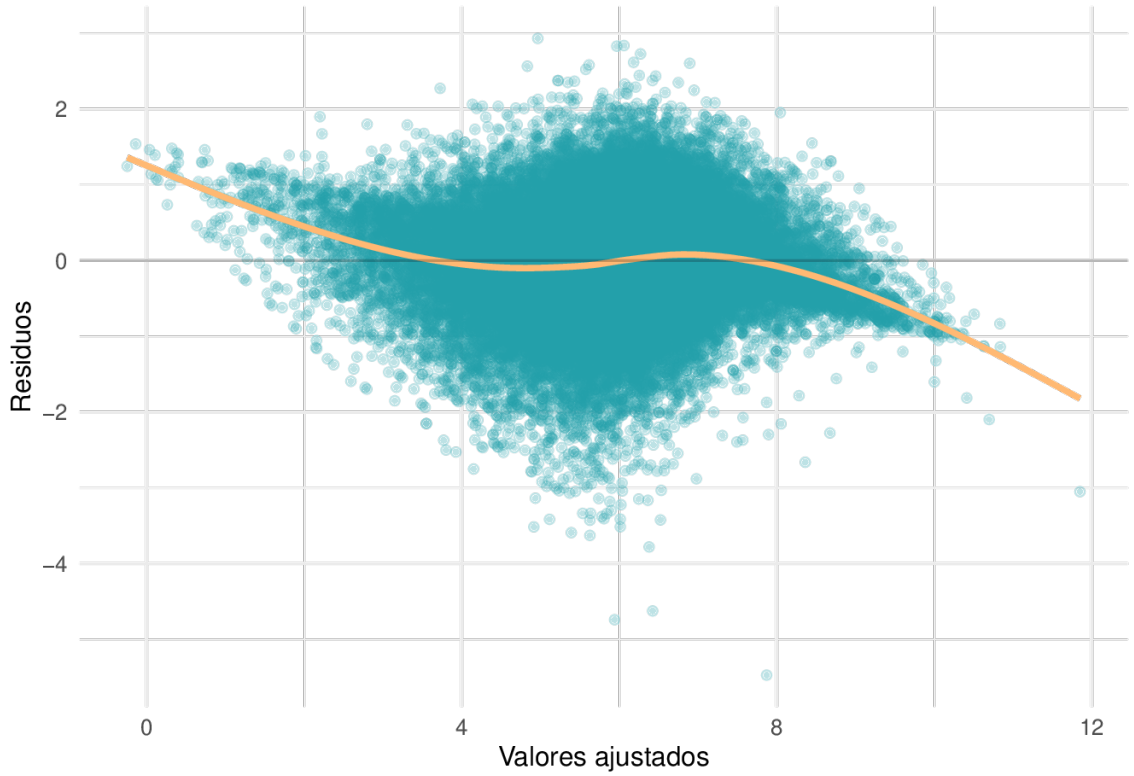


Figura 6.8: Hipótesis de homocedasticidad de los residuos en el modelo de regresión lineal múltiple.

No parece haber una tendencia creciente o decreciente en los residuos, por lo que se podría considerar que los errores tienen varianzas constantes. Sin embargo, sí que se produce un cierto abombamiento en la zona central de la representación, lo cuál dificulta la interpretación. Como el gráfico no nos permite obtener información más concisa, se realiza el test de Breusch-Pagan:

```
    studentized Breusch-Pagan test
Estadístico = 1274.749
P-valor = 3.491887e-245
```

El p-valor es muy cercano al cero, por lo que se acepta la hipótesis de homocedasticidad.

En tercer lugar, para verificar la independencia de los residuos se realiza el test de Durbin-Watson:

```
    Durbin-Watson test
Estadístico = 1.50059
P-valor = 0
```

El p-valor es cero y por tanto aceptamos la hipótesis de que los residuos están incorrelados.

La última hipótesis que queda por comprobar es la multicolinealidad. Una de las medidas más utilizadas para su estudio es el factor de inflación de varianza, *VIF*. Con la

función *vif* del paquete *car* se obtiene el índice VIF asociado a cada variable. Esta medida toma valores entre uno e infinito, y se suele decir que valores superiores a cinco son indicio de multicolinealidad. A continuación, vamos a calcular los índices para cada una de las variables del modelo aunque, debido al gran número de variables involucradas, analizaremos solo aquellas cuyo índice sea mayor a dos, esto es, aquellas que presenten indicios de ser variables colineales:

Tabla 6.4: Factor de inflación de la varianza para cada variable en el modelo de regresión lineal múltiple.

Variable	VIF
actors	3.753825
cinematographer	2.403009
composer	2.530415
Adult	9.116069
NotaGuionista	5.034463
NotaDirector	4.614622
isAdult	9.114609

Los dos valores más altos, cercanos a diez, son los asociados a las variables *isAdult* y *Adult*, donde la primera indicaba si una película estaba calificada como para mayores de edad y la segunda si uno de los géneros de la película es adulto. El resto de variables son las correspondientes a las notas del equipo de la película.

El hecho de que las notas del equipo de la película presenten un índice *VIF* alto no es nada nuevo: ya vimos en las secciones previas que estas variables estaban altamente correladas. Sin embargo, la correlación entre las otras dos variables no ha sido calculada en ningún momento. Como se trata de variables binarias habrá que utilizar el coeficiente Phi, similar al coeficiente de correlación de Pearson. Para ello se necesita la tabla de contingencia de ambas variables.

	Adult	
isAdult	0	1
0	104979	70
1	3	603

```
El coeficiente phi es 0.94
```

El coeficiente Phi es casi uno, indicando que ambas variables están correladas y además, positivamente. Por tanto tiene sentido que el índice *VIF* fuese alto.

Una forma de solucionar este problema es mediante un análisis de regresión escalonada o *stepwise regression*, por medio de la función *step*. Estos métodos evalúan todos los modelos posibles teniendo en cuenta todas las combinaciones de las variables predictoras. El problema principal de estos radica en que tardan un tiempo considerable, al tener en cuenta todos los modelos, lo cual, considerando nuestro caso en el que tenemos un número

muy elevado de variables, no parece buena idea. Además, no incluir alguna variable en el modelo puede suponer una pérdida de información.

Con el fin de solucionar el problema de las variables muy correladas, se va a usar el método de componentes principales para aplicar posteriormente regresión lineal múltiple. Gracias a este procedimiento seremos capaces de trabajar con un número reducido de variables que además estén incorreladas entre sí, solucionando el problema de la colinealidad.

6.3. Regresión lineal con PCA

Seguidamente, aplicaremos regresión habiendo antes realizado un análisis de componentes principales. Esta técnica, explicada más en detalle en 2.1.4, pretende reducir la dimensión de un conjunto de datos descrito por un número elevado de variables interrelacionadas entre sí.

La idea es describir la estructura de varianzas del conjunto de variables a través de otro conjunto de variables más reducido, definidas como combinaciones lineales de las anteriores. Cada componente principal explica un porcentaje de la variabilidad total, por lo que finalmente nos quedaremos con las componentes cuya proporción de la variabilidad total explicada sea alta.

Por tanto, el primer paso será seleccionar el número de componentes adecuado y ver cómo se relacionan estas con las variables originales. Para ello, habrá que realizar el análisis de componentes principales correspondiente. Siguiendo la línea anterior y usando por tanto el paquete *tidymodels*, modificamos la receta previa, incorporando nuevos pasos:

```
receta_pca_prep = receta %>%
  step_normalize(all_numeric(), -sex_woman, -sex_man) %>%
  step_pca(all_numeric_predictors(), -sex_woman, -sex_man)
```

Antes de nada, como algunas variables como el número de votos tienen una varianza muy elevada respecto a las demás, se han estandarizado las variables numéricas mediante la función *step_normalize* haciendo que estas tengan media cero y desviación típica uno. Después, se ha aplicado el análisis de componentes principales mediante la función *step_pca*. Notamos que las dos variables dummy creadas a partir de la variable original *sex* han sido eliminadas en ambos pasos, pues el programa las considera variables numéricas a pesar de que son binarias, y no pueden ser usadas entonces en el análisis de componentes principales, reservado a variables continuas.

Una vez modificada la receta, antes de aplicar el modelo de regresión lineal, se estudiará el comportamiento de las componentes principales. Para ello usaremos la orden *prep*:

```
prep_pca = prep(receta_pca_prep)
```

Las componentes principales son combinaciones lineales de las variables originales, por lo que cabe esperar que exista correlación alta entre ellas, es más, esta correlación será la que permita averiguar el significado de cada componente. Así pues, si para cada componente estudiamos la correlación con las variables originales obtenemos lo siguiente:

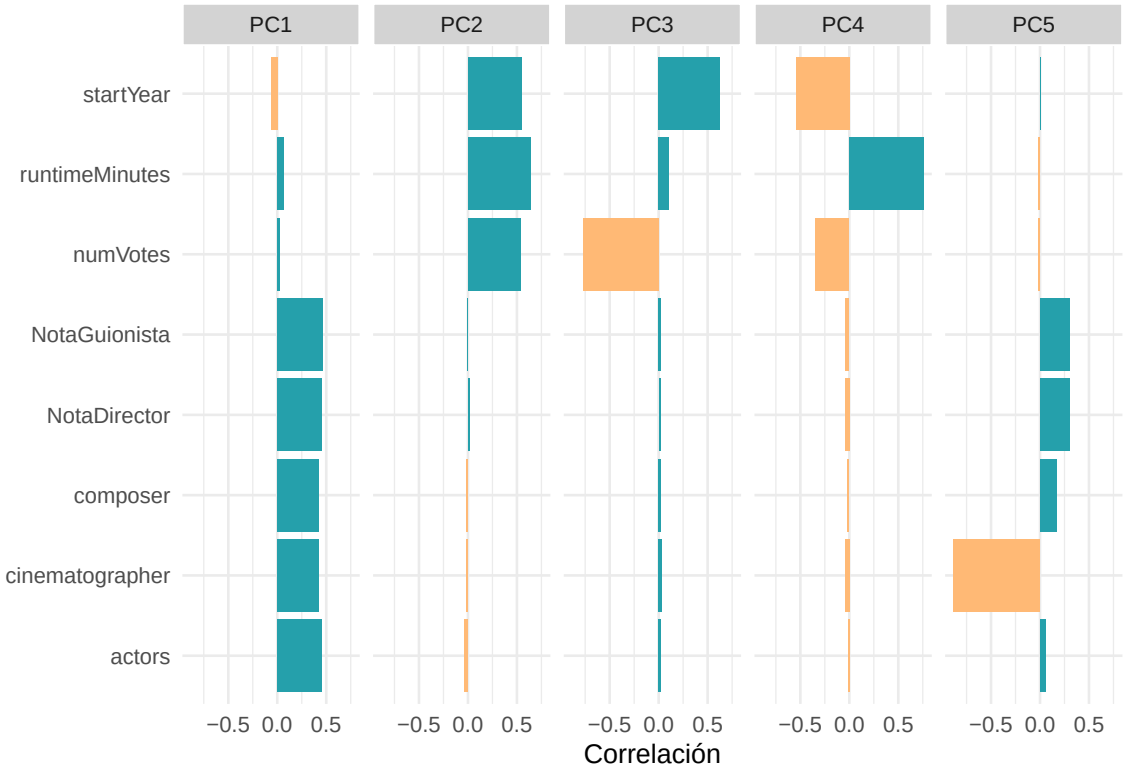


Figura 6.9: Correlación de las variables originales con las cinco primeras componentes principales.

Como se observa en el gráfico, en el cual solo se han tenido en cuenta las cinco primeras componentes, la primera está altamente correlada con las notas de los distintos integrantes del equipo y, además, positivamente; mientras que la segunda está muy correlada, y también positivamente, con el resto de variables numéricas: año de comienzo de la película, duración y número de votos. Las componentes tres, cuatro y cinco también tienen correlaciones altas con las variables originales, tanto positiva como negativamente, pero no siguen un patrón concreto.

Podemos establecer otro gráfico, esta vez con las seis primeras componentes, en el que mostremos los valores de estas correlaciones en valor absoluto:

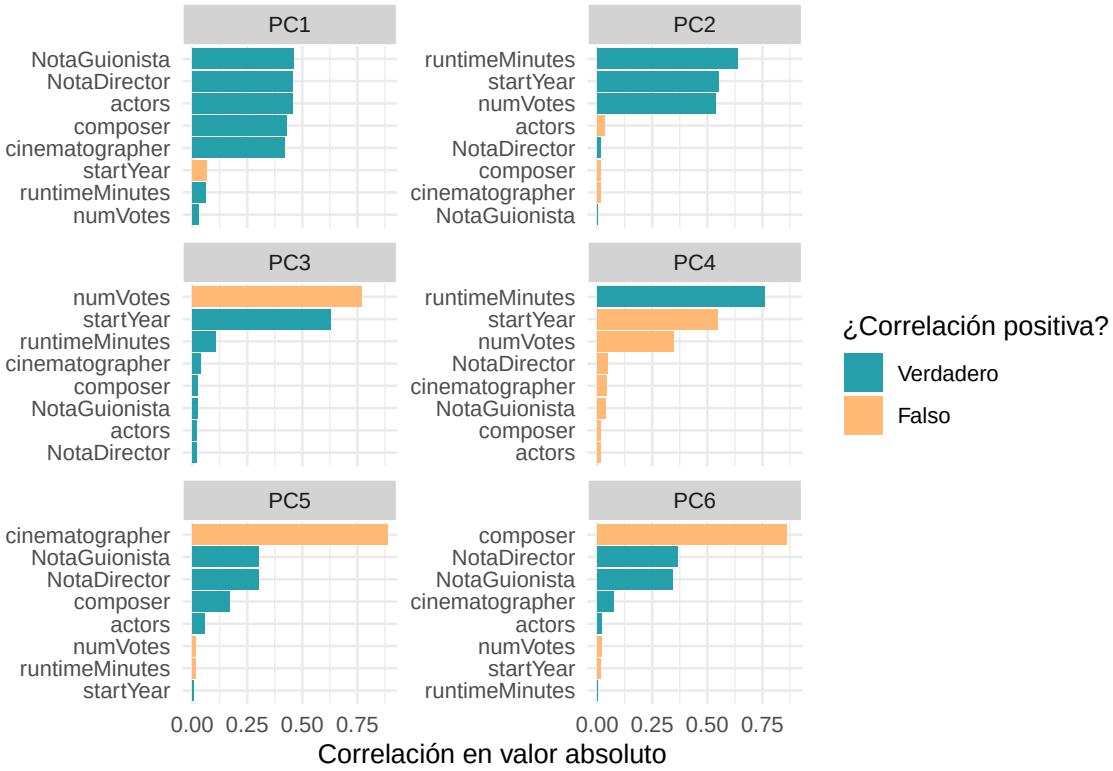


Figura 6.10: Correlación en valos absoluta de las variables originales con las seis primeras componentes principales.

Se observa lo ya comentado en el gráfico anterior: la primera componente hace referencia a las notas mientras que la segunda al resto de variables numéricas. Además, se advierte que la tercera componente está muy negativamente correlada con el número de votos, mientras que la cuarta está muy correlada, pero positivamente, con la duración de la película.

Ahora bien, se debe escoger el número de componentes principales adecuado, tomando un número no muy elevado, pues el objetivo es reducir el grupo de variables originales; pero que explican una proporción total de la variabilidad considerable. Teniendo esto en cuenta, se muestra a continuación la variabilidad explicada por cada una de las componentes, así como la variabilidad total acumulada:

Tabla 6.5: Variabilidad explicada por cada componente principal.

Componente	Variabilidad explicada	Variabilidad acumulada
PC1	0.4974748	0.4974748
PC2	0.1410594	0.6385342
PC3	0.1195200	0.7580542
PC4	0.1113036	0.8693578
PC5	0.0446754	0.9140331
PC6	0.0423525	0.9563856
PC7	0.0273037	0.9836893
PC8	0.0163107	1.0000000

Con tan solo la primera componente principal se explica casi un 50 % de la variabilidad total. Las componentes dos, tres y cuatro explican cada una un 10 % de la variabilidad total, mientras que a partir de la quinta se explica menos de un 5 %. Como se observa en la última columna, con tan solo cuatro componentes queda explicada más de un 86 % de la variabilidad total, mientras que con cinco más del 90 %. En vista de estos resultados, se considera adecuado escoger entre cuatro o cinco componentes principales.

Otra forma de averiguar cuál es el número óptimo de componentes principales, es probar el modelo deseado, la regresión lineal múltiple, con todas las cantidades posibles de componentes y analizar posteriormente los resultados. A tal efecto, se realiza un ajuste del modelo, *model tuning*, evaluando los resultados mediante validación cruzada, usando para ello los ocho pliegues definidos al comienzo de la sección. Por tanto, habrá que especificar en la receta que no se sabe a priori el número de componentes principales que se desean obtener si no que se va a realizar *tuning* sobre este valor. Con este fin, modificamos la receta anterior de la siguiente forma:

```
receta_pca_tuning = receta %>%
  step_normalize(all_numeric(), -sex_woman, -sex_man) %>%
  step_pca(all_numeric_predictors(), -sex_woman, -sex_man,
           num_comp = tune())
```

Una vez realizado este cambio en la receta, se creará un grid de posibles valores para el número de componentes, que en este caso son todos los números naturales entre el uno y el ocho, y se ajustará el modelo evaluandolo mediante validación cruzada. Para ello, se necesitará en primer lugar definir el flujo de trabajo correspondiente, teniendo en cuenta que hay que proporcionar una receta y un modelo. El modelo teórico que se usará sigue siendo la regresión lineal múltiple, por lo que se utilizará el modelo definido en el punto previo y como receta, usamos la anterior:

```
lr_pca_wflow_tuning =
  workflow() %>%
  add_model(lr_mod) %>%
  add_recipe(receta_pca_tuning)
```

Una vez definido el *workflow*, se realiza el ajuste del modelo, obteniendo las siguientes métricas:

```
pca_grid = expand.grid(num_comp = 1:8)

pca_components = tune_grid(
  lr_pca_wflow_tuning,
  resamples = data_folds,
  grid = pca_grid
)
```

Tabla 6.6: Métricas obtenidas para las diferentes componentes principales.

Nº componentes	RMSE	R^2 ajustado
8	0.4709597	0.7782214
7	0.4724921	0.7767752
6	0.4854354	0.7643745
5	0.4884666	0.7614176
4	0.4947240	0.7552660

Como era de esperar, cuanto mayor es el número de componentes mejor es el resultado, obteniendo menores valores para la raíz del error cuadrático medio y superiores para el R^2_{adj}. Sin embargo, las diferencias de estos valores son mínimas: tomando ocho componentes obtenemos un *RMSE* de 0.47, mientras que tomando cuatro de 0.49. Así mismo, tomando ocho componentes el R^2_{adj} es del 77 %, mientras que tomando cuatro es del 75 %.

A la vista de estos resultados, y teniendo en cuenta que con cuatro componentes principales queda explicada más del 86 % de la variabilidad total, tomamos esta cantidad para realizar el análisis de regresión lineal múltiple, analizando, en primer lugar, el rendimiento del modelo en el conjunto de entrenamiento:

Tabla 6.7: Estadísticos de bondad del ajuste para la regresión lineal múltiple usando componentes principales.

Modelo	R^2 ajustado	AIC	BIC
Regresión y PCA	0.7553676	137072.7	137376.7
Regresión	0.7783359	129785.0	130125.8

Como ya se había observado se tiene un R^2_{adj} algo superior al 75 %, indicando que queda explicada más de un 75 % de la variabilidad total. Por otro lado, los valores del AIC y el BIC, que hacen comparables unos modelos con otros, son superiores en este caso, indicando que este modelo es más adecuado que el anterior. Esto puede deberse a que, a pesar de que los resultados obtenidos con el mismo sean peores, el número de variables que utiliza es bastante menor, lo cual señala la principal ventaja de tomar tan solo cuatro componentes. Aún así, señalar que seleccionando seis componentes ya estaríamos en unos valores de error prácticamente iguales a los obtenido con la totalidad de variables.

A continuación estudiamos cuales son las variables más significativas en el modelo:

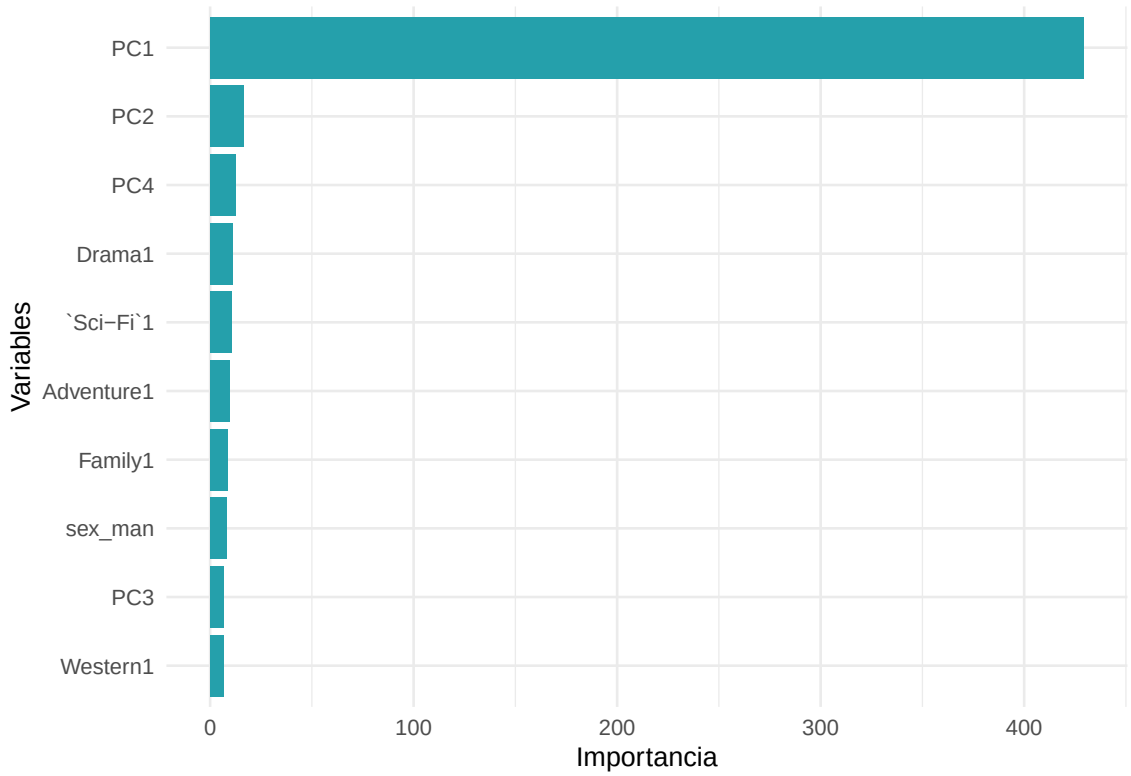


Figura 6.11: Variables más significativas en el modelo de regresión lineal múltiple usando componentes principales.

En el anterior modelo, las variables correspondientes a las cinco calificaciones del equipo eran las más importantes, por lo que ahora la variable primordial es la correspondiente a la primera componente, que estaba altamente correlacionada con estas notas. Le siguen las componentes dos, relacionada con el número de votos, la duración y el año; y la número cuatro. Los géneros drama, ciencia ficción y aventura ya se encontraban entre las variables más sustanciales en el análisis previo, a los que se añaden los géneros familia y cine del oeste.

Una vez analizada la eficiencia del modelo en el conjunto de entrenamiento, pasamos a estudiarla en el conjunto test. Para ello, realizamos las predicciones, y calculamos las mismas medidas de bondad del ajuste que en el caso anterior:

Tabla 6.8: Estadísticos de bondad del ajuste en la predicción para la regresión lineal múltiple usando componentes principales.

Modelo	MAE	RMSE	R^2 ajustado
Regresión y PCA	0.4596704	0.6123930	0.7554172
Regresión	0.4357412	0.5830311	0.7782978

El *RMSE* y el *MAE* son superiores en este caso que en el anterior, pero levemente. Lo mismo pero al contrario sucede con el R^2_{adj}: es inferior, pero ligeramente.

Si por último representamos las calificaciones reales frente a las predicciones obtenidas en este modelo obtenemos lo siguiente:

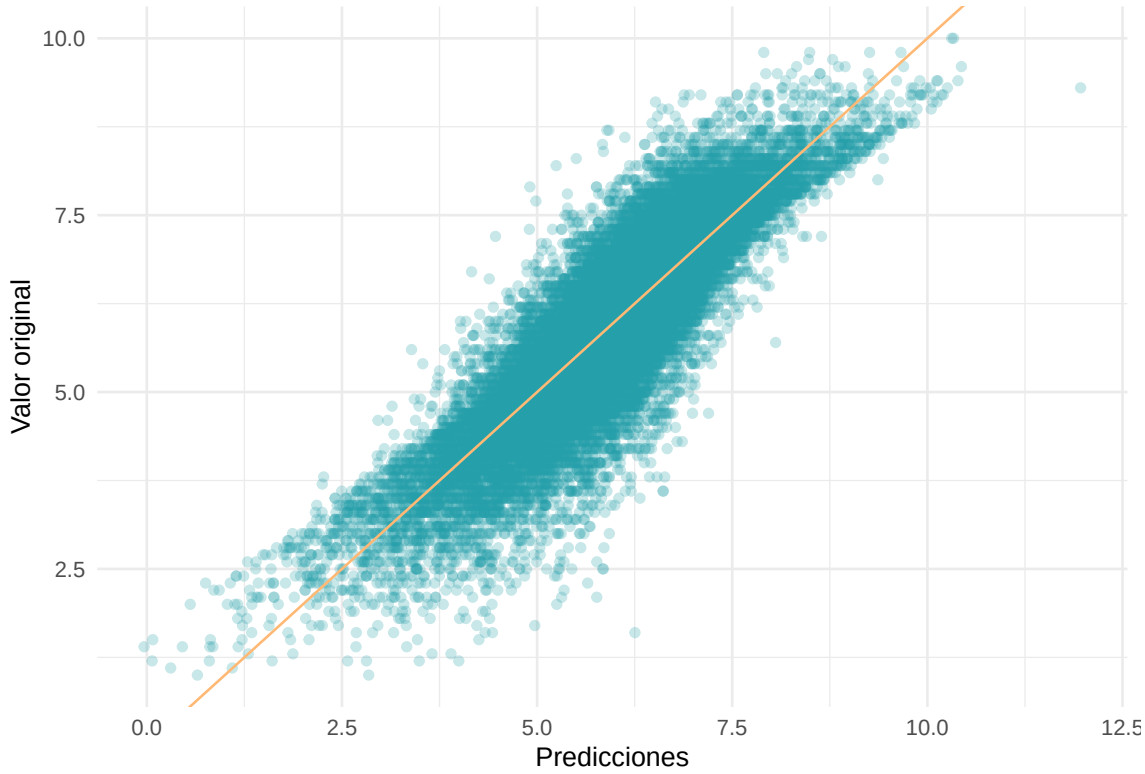


Figura 6.12: Puntuación de las películas en función de la predicción mediante regresión lineal múltiple usando componentes principales.

El gráfico es muy similar al gráfico 6.6, pero menos aplanado, indicando un peor ajuste en este caso, que era de esperar en vista de las diferencias obtenidas en las medidas de bondad del ajuste. Aún así, la diferencia es muy leve.

En conclusión, en vista de los resultados obtenidos en ambos apartados se deduce que el modelo de regresión lineal múltiple obtenido usando componentes principales es mejor que el obtenido sin usarlas ya que, a pesar de que las medidas del rendimiento del modelo son peores, lo son mínimamente y esto se ve compensado con el hecho de trabajar con un menor número de variables.

Para terminar esta sección, incorporaremos una última técnica para predecir la puntuación de las películas: Random Forest. Un resumen teórico detallado de esta técnica viene en 6.4.

6.4. Random Forest

Para aplicar esta técnica, al igual que pasó con la regresión lineal múltiple, lo primero será definir el modelo a utilizar: usaremos la técnica de bosques aleatorios o Random Forest y el algoritmo Ranger. Los parámetros seleccionados para el modelo serán los predeterminados: quinientos árboles y cinco variables seleccionadas para realizar cada división. Este algoritmo destaca por su rapidez, virtud que necesitamos en vista del gran número de observaciones.

```
rf_mod = rand_forest() %>%
  set_engine("ranger", importance = 'impurity') %>%
  set_mode("regression")
```

Una vez definido el modelo, al igual que en el caso anterior, habrá que definir el *workflow*, fusionando el modelo anterior junto con la receta creada al principio de la sección:

```
rf_wflow = workflow() %>%
  add_model(rf_mod) %>%
  add_recipe(receta)
```

El siguiente paso será entrenar el modelo, usando para ello el mismo conjunto de entrenamiento que en los anteriores casos con objeto de obtener medidas que permitan realizar comparaciones con los dos modelos obtenidos anteriormente. Se obtiene lo siguiente:

Tabla 6.9: Comparación estadísticos de bondad del ajuste para los diferentes modelos.

Random Forest	Reg. lineal múltiple	Reg. lineal múltiple (PCA)
0.8105481	0.7783359	0.7553676

El R^2_{adj} es notablemente superior en este caso, quedando con el modelo de Random Forest explicada más de un 81 % de la variabilidad total. El AIC y el BIC no han sido proporcionados en este caso debido a que el criterio de información de Akaike y el criterio de información bayesiano son calculados usando el número de parámetros del modelo, que es un concepto impreciso en este caso, pues no queda señalado si hace referencia al número de árboles, al número de particiones o a la profundidad del árbol.

Por otro lado, se puede analizar cuáles han sido las variables determinantes a la hora de realizar el modelo:

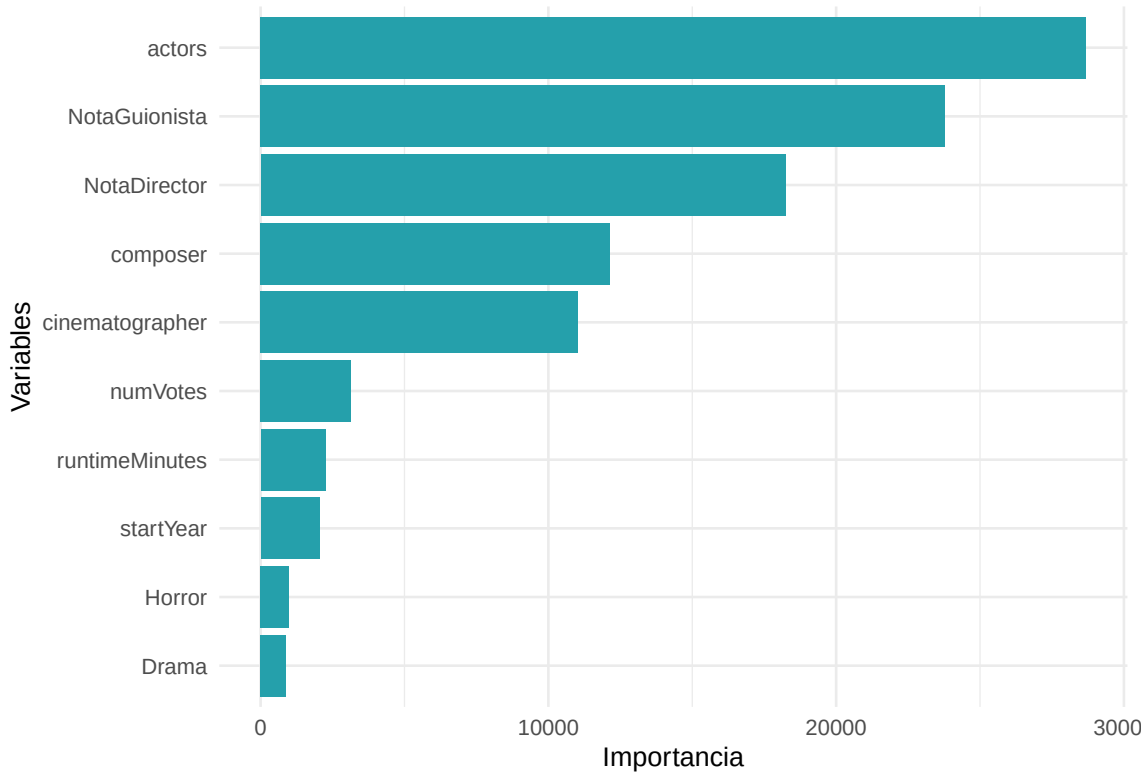


Figura 6.13: Variables más significativas en el modelo de random forest.

Las variables más importantes son casi las mismas que las obtenidas en el análisis de regresión lineal múltiple inicial, expuestas en el gráfico 6.5. Es más, las tres principales coinciden en ambos casos. Este modelo, al contrario que el otro, considera más importante la nota del director de fotografía que el número de votos de la película y además también otorga valor al año de estreno de la película, lo cuál el otro modelo no hacía. En este caso drama también es uno de los géneros más importantes, pero al contrario que en el modelo inicial, donde destacaban los géneros de *ciencia ficción* y *aventura*, en este caso *horror* es más relevante.

Por último, para terminar de evaluar correctamente el rendimiento del modelo, analicemos su comportamiento sobre los datos del conjunto test:

Tabla 6.10: Estadísticos de bondad del ajuste en la predicción para random forest.

Modelo	MAE	RMSE	R^2 ajustado
Random Forest	0.3892719	0.5398713	0.8112078
Regresión y PCA	0.4596704	0.6123930	0.7554172
Regresión	0.4357412	0.5830311	0.7782978

Las medidas de bondad del ajuste obtenidas con este modelo son mucho mejores que las obtenidas con las anteriores. Además, observamos como el R^2_{adj} del conjunto test es similar al obtenido en el conjunto de entrenamiento, por lo que se concluye que no se ha producido un sobreajuste de los datos a lo cual suelen tender los algoritmos de Random Forest. Así pues, con datos distribuidos de manera similar a los que se tienen en el conjunto test, con

este modelo se espera un error de ± 0.38 en la media de las calificaciones de las películas, menor que el esperado con regresión, o con regresión junto a componentes principales.

Representando las calificaciones reales frente a las predicciones obtenidas con este modelo se obtiene la siguiente representación:

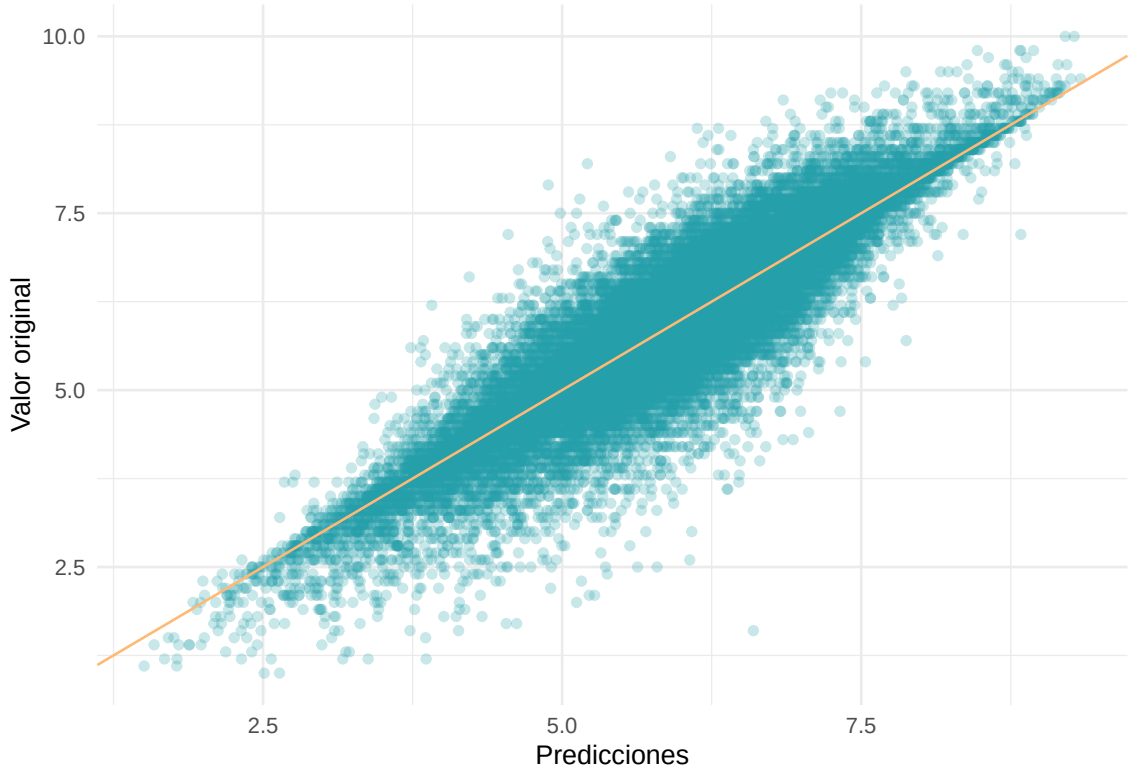


Figura 6.14: Puntuación de las películas en función de la predicción mediante random forest.

La nube de puntos obtenida en este caso es más plana que las anteriores, situándose, como era de esperar en vista de las medidas de bondad del ajuste obtenida, una mayor proporción de puntos en torno a la recta $y = x$.

Resumiendo lo planteado en esta sección, se han estudiado y comparado tres modelos diferentes: regresión lineal múltiple, regresión lineal múltiple usando anteriormente análisis de componentes principales, y random forest. A raíz de los resultados obtenidos, se deduce que el modelo que mejor ajusta, y que por lo tanto mejor predice las calificaciones de las películas, es random forest. Pese a que es el modelo más complejo de todos, el coste de implementarlo es mínimo si se tienen en cuenta los resultados. En conclusión, por dicho motivo, el mejor modelo de los considerados en este apartado es random forest.

Capítulo 7

Modelos de clasificación

Tal y como se expuso en la introducción del capítulo, una de las formas de medir el éxito de una película es a través del número de galardones que posee. En esta sección se intentará predecir, por medio de las variables explicativas ya descritas, si una película ganará o no algún premio Oscar. Evidentemente, el número de películas que ha ganado un premio Oscar es excesivamente inferior al que no, por lo que nos encontramos ante un problema de clasificación con datos desbalanceados, lo cuál se intentará resolver empleando diferentes técnicas.

Con objeto de realizar el análisis propuesto, se carga en primer lugar un archivo conteniendo las películas que han ganado algún premio Oscar desde la primera ceremonia en el año 1929 (Fontes [2020]). Tras ello, se debe fusionar dicho dataset con el conjunto de datos con el que se ha estado trabajando hasta ahora, añadiendo por tanto para cada película una nueva variable binaria que indique si ha ganado o no un premio Oscar. El problema es que al limpiar al inicio del capítulo el conjunto de datos procedente del *IMDb*, se suprimieron un gran número de observaciones correspondientes a películas con información incompleta; por lo que ahora nos encontramos con una gran cantidad de películas que aparecen en el conjunto de los Oscar pero no en nuestro conjunto de datos.

A fin de solucionar este hecho, se realizará más adelante una imputación sobre los campos vacíos de las películas que hayan ganado algún premio Oscar. Eliminar dichas observaciones del conjunto de datos no sería adecuado en este caso, ya que debido a la escasa cantidad de películas ganadoras de algún premio es preferible preservar el mayor número de observaciones posibles.

Para ello, se fusiona el archivo que recoge los premios con los conjuntos de datos originales (*title.principals*, *title.basics*, *title.crew* y *title.ratings*) y se siguen los mismos pasos que en la sección 5.1 dedicada al pretratamiento de los datos, pero sin eliminar las observaciones incompletas. De esta manera, obtendremos para cada película que ha ganado un premio Oscar los datos que se usarán para la predicción, y estaremos en condiciones de fusionarlos con el conjunto de datos que se ha usado en las secciones anteriores, para realizar después el proceso de imputación.

Antes que nada, al igual que en la sección dedicada al análisis de regresión, se va a realizar un breve análisis descriptivo para estudiar la relación entre las variables explicativas y la variable objetivo.

7.1. Breve análisis descriptivo

En primer lugar, analicemos cuáles son los géneros más frecuentes en las películas galardonadas mediante la siguiente representación:

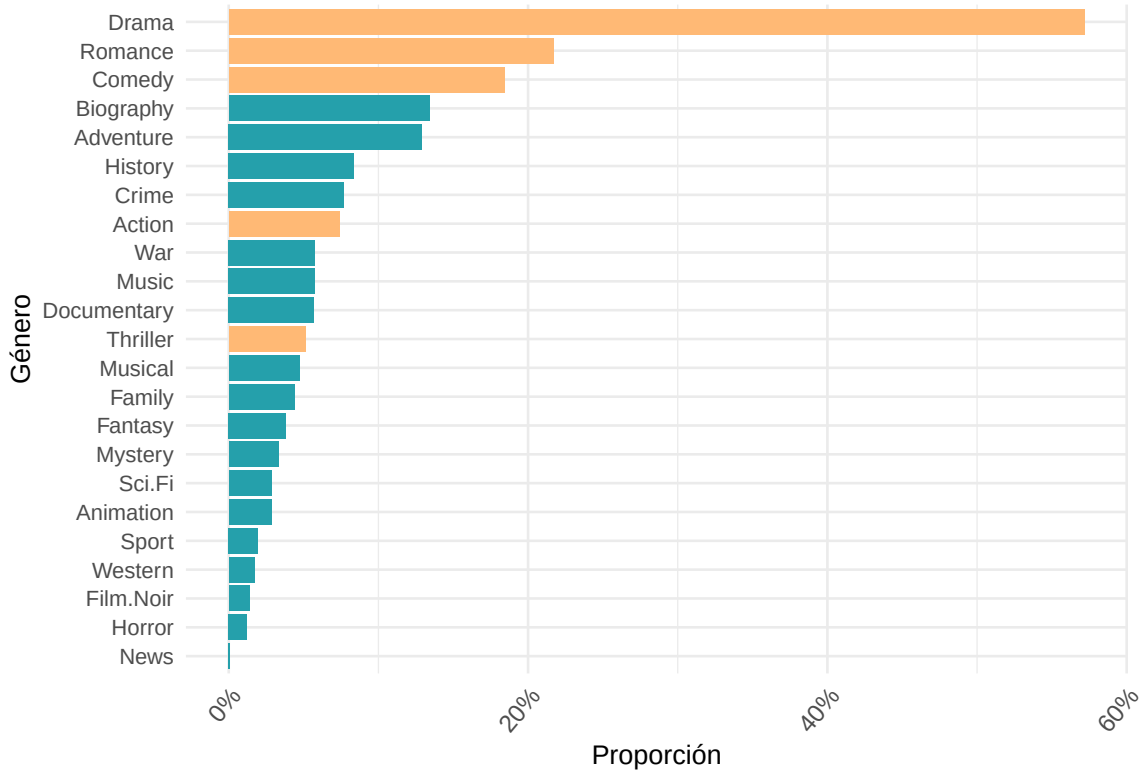


Figura 7.1: Estudio de los Oscars en función del género.

Los cinco géneros más comunes en el conjunto de datos se encuentran resaltados y, exceptuando thriller, se hallan entre los diez géneros más premiados. Casi un 60 % de las películas ganadoras están catalogadas como drama, que es también, como ya se ha señalado a lo largo de todo el capítulo, el género más frecuente. Después de este claro liderazgo, los dos géneros más usuales son comedia y romance, representando cada uno aproximadamente un 20 % de las películas premiadas. Por último, señalar que las películas catalogadas como horror o cine negro no obtienen apenas reconocimiento.

No obstante, el número de películas clasificadas como horror es notablemente inferior al de las clasificadas como drama, por lo que los resultados anteriores pueden no ser del todo sólidos, encontrándonos con que hay un porcentaje superior de películas ganadoras de algún premio Oscar pertenecientes al género dramático por el mero hecho de que hay más películas catalogadas como tal. En vista de esto, se calcula a continuación la proporción de películas de cada género ganadoras de algún premio Oscar y la probabilidad de, siendo una película de un género determinado, ganar algún premio, esto es, la probabilidad condicionada. Se obtiene en primer lugar lo siguiente:

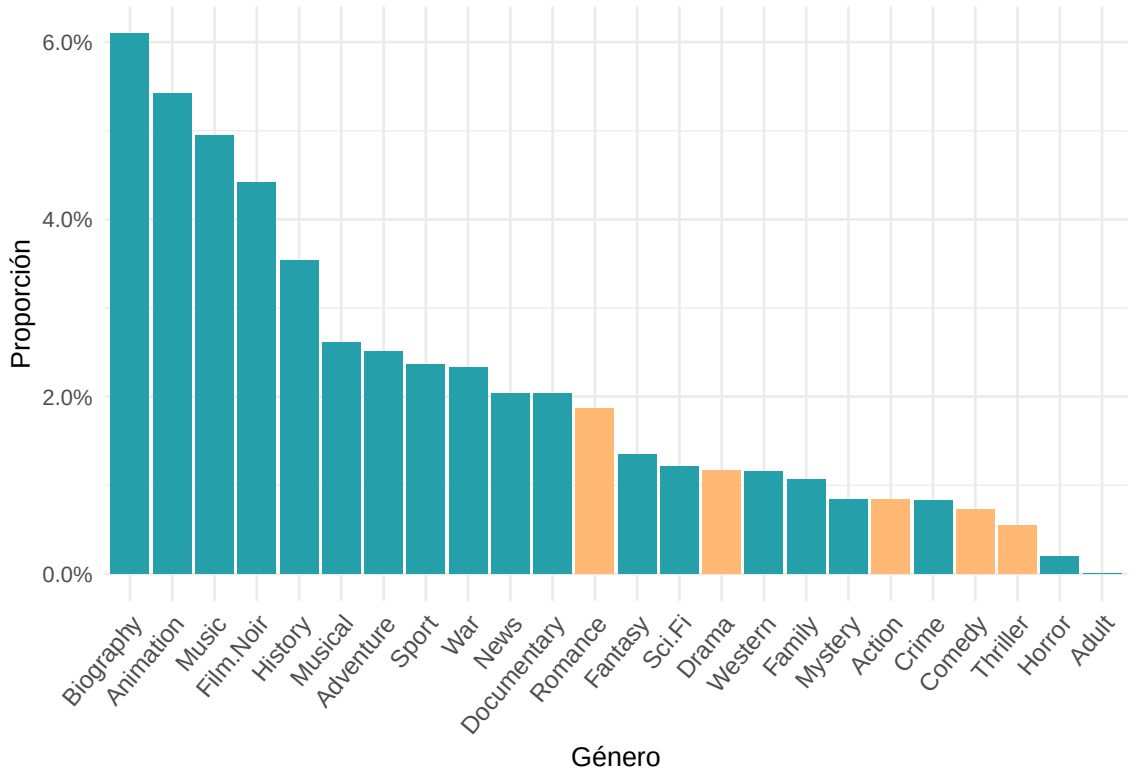


Figura 7.2: Proporción de películas premiadas de cada género.

La proporción de películas de animación ganadoras de un premio Oscar se encuentra en segundo lugar, indicando que, a pesar de que en el gráfico anterior nos encontrábamos con que este género estaba entre los menos premiados, esto era debido a que la cantidad de películas del mismo es inferior. El género dramático tiene el efecto contrario: habíamos visto que las películas ganadoras suelen pertenecer a dicho género, y en este gráfico se observa como la proporción de películas catalogadas como drama y que son ganadoras es muy pequeña, lo cual puede ser debido a la gran cantidad de películas clasificadas como tal.

Tabla 7.1: Probabilidad de ganar un premio Oscar según el género

Género	Probabilidad condicionada
Biography	9.4e-05
Drama	7.7e-05
Romance	4.7e-05
Adventure	3.7e-05
History	3.4e-05
Music	3.3e-05
Animation	1.8e-05
War	1.6e-05
Comedy	1.5e-05
Musical	1.4e-05
Documentary	1.3e-05
Action	7e-06
Crime	7e-06
Film.Noir	7e-06
Fantasy	6e-06
Family	5e-06
Sport	5e-06
Sci.Fi	4e-06
Mystery	3e-06
Thriller	3e-06
Western	2e-06
Horror	0
News	0

En vista de los resultados, las películas biográficas tienen una mayor probabilidad de ganar un premio Oscar. Por otro lado, la tabla reivindica lo señalado anteriormente: la proporción de películas catalogadas como drama y ganadoras es pequeña debido a la gran cantidad de películas clasificadas como tal ya que, en la tabla se observa como las películas de este género tienen una posibilidad superior de ganar algún premio.

En segundo lugar, se van a examinar las calificaciones del equipo en función de si las películas están o no premiadas. Para ello, se utiliza la siguiente gráfica:

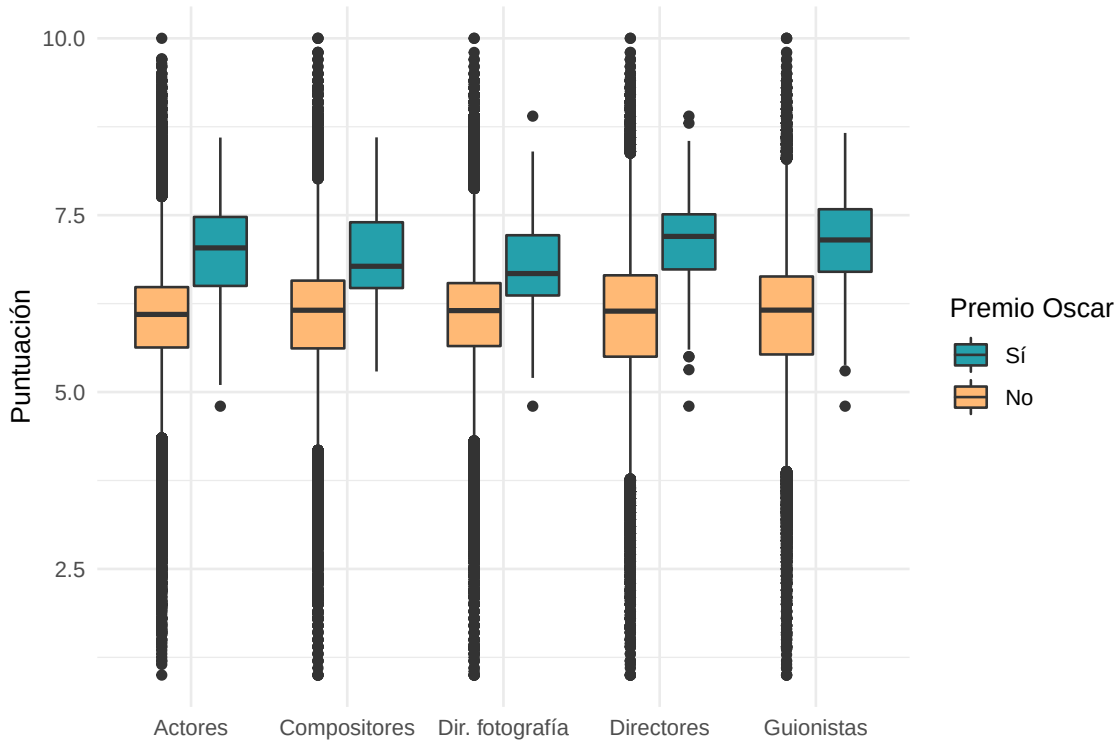


Figura 7.3: Estudio de los Oscars en función de las notas del equipo.

Se observa que las películas ganadoras de algún premio Oscar tienen un equipo mejor calificado que las que no han sido ganadoras de ninguno. Así pues, mientras que la media de las películas no galardonadas se sitúa en torno al seis, la de las que sí lo han sido esta alrededor del siete. Cabe destacar que las notas correspondientes al equipo de las películas ganadoras presentan muy pocas observaciones atípicas, al contrario de lo que ocurre en la otra categoría, donde el número de outliers es muy elevado, debido en parte a la gran cantidad de observaciones con dicha clasificación.

A continuación, estudiamos la duración y el número de votos de las películas premiadas:

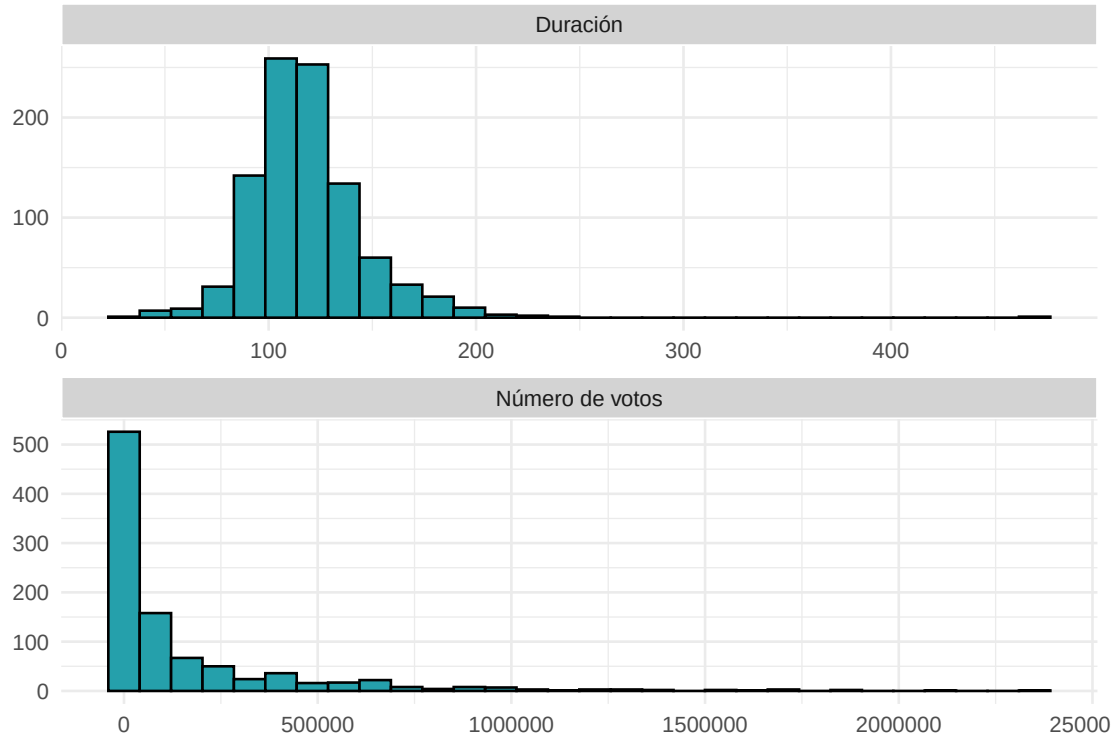


Figura 7.4: Estudio de los Oscars en función de la duración y el número de votos de la película.

Al contrario de lo que se podría pensar, la mayor parte de las películas premiadas han sido votadas pocas veces, habiendo aún así un número considerable que han sido votadas en más de medio millón de ocasiones. Las tres películas más votadas con al menos un premio Oscar son *El caballero oscuro*, *Origen* y *Pulp Fiction*. En lo referente a la duración, lo más común es que sea de dos horas, encontrándonos con casos excepcionales como *O.J.: Made in America*, película documental con casi ocho horas de duración.

Por último, analicemos la relación con la variable objetivo de la sección anterior: la puntuación de la película.

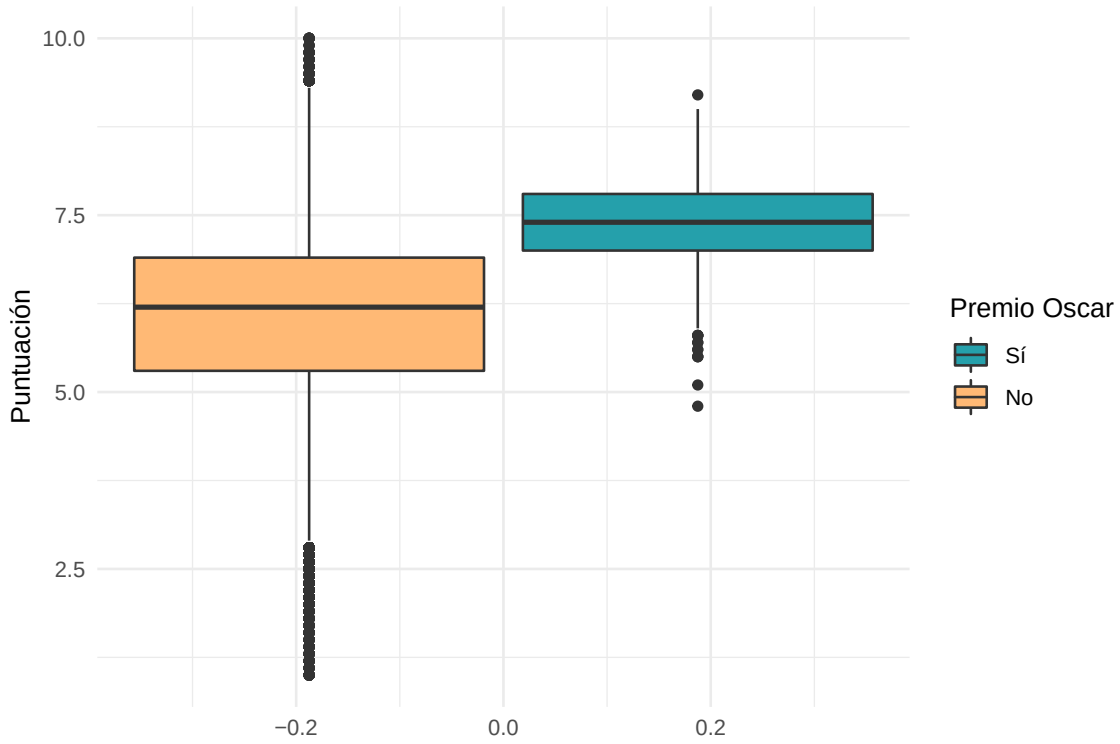


Figura 7.5: Estudio de los Oscars en función de la nota de la película.

Como era de esperar, las películas premiadas tienen de media una puntuación superior a las que no. Mientras que la nota media de estas últimas se sitúa en torno a seis, la de las películas galardonadas es de siete y medio, más de un punto superior. Además, la variación es menor: tan solo el 25 % de los datos tiene una puntuación inferior al siete.

Una vez expuesta la relación entre la variable objetivo y las variables explicativas, se procede a realizar los análisis antes mencionados.

7.2. Clasificación

El principal inconveniente del problema de clasificación planteado es la diferencia del número de observaciones en cada categoría, siendo mucho menor en el grupo correspondiente a las películas premiadas. Para ilustrar este hecho, se muestra la siguiente tabla:

Tabla 7.2: Proporción de películas con premio Oscar.

Ganadora	Número	Proporción
No	105289	98.84 %
Sí	1231	1.16 %

Tal y como se había previsto, la clase correspondiente a ganar un premio Oscar representa apenas un 1 % del total de las observaciones. Hay más de cien mil películas que no han ganado ningún premio Oscar, mientras que tan solo mil doscientas lo han obtenido.

A raíz de este hecho, se afirma que estamos ante un problema de clasificación con datos desbalanceados. La cuestión principal de este dilema es qué técnicas aplicar para obtener buenos resultados, pues si se aplica algún método de clasificación sin incorporar ningún paso previo para solventar dicho problema, lo más probable es que nos encontremos con una especificidad muy alta y una sensibilidad muy baja. Esto se debe a que clasificando todas las películas como no premiadas se incurrirá en una tasa de falsos negativos muy baja, al haber muy pocos datos correspondientes a películas que sí lo estan. Por tanto, se concluye que antes de aplicar alguna técnica de clasificación se debe llevar a cabo un proceso adecuado para solventar el problema de los datos desbalanceados.

Dos de las técnicas aplicadas frecuentemente en problemas de este tipo son *dowsampling* y *upsampling*, descritas en 2.1.7. Mientras que *downsampling* soluciona el problema seleccionando una muestra del tamaño del conjunto de observaciones de la clase minoritaria de las observaciones pertenecientes a la clase dominante, *upsampling* aumenta la representación de la clase minoritaria, generalmente a través de la generación de datos ficticios. En este estudio se ha optado por *upsampling* puesto que usar *downsampling* implicaría trabajar solo con dos mil observaciones, lo cual no parece conveniente debido a la diversidad de los datos.

Ahora bien, antes de tratar los datos desbalanceados hay que lidiar con el problema de las observaciones con campos vacíos, que se solventará imputando dichos valores. Para ello, usaremos *step_impute_knn* que aplica la técnica de los k vecinos más cercanos, conocida como kNN, para realizar la imputación de los valores perdidos. Añadiremos este paso a una receta aplicada al conjunto de datos total, y después usaremos las funciones *prep*, para estimar los parámetros, y *juice* para obtener el conjunto de datos con los valores ya imputados, que será el que usemos más adelante. El código utilizado se muestra a continuación:

```
impute_rec = recipe(Oscars ~ ., data = datos_premios) %>%
  update_role(tconst, primaryTitle, originalTitle,
              genres, directors, writers, new_role = "ID") %>%
  step_impute_knn(all_predictors())

datos_premios_im = prep(impute_rec) %>% juice()
```

Una vez que se tienen todas las observaciones del conjunto de datos completas, sin ningún campo vacío, se puede estudiar ya el problema de clasificación planteado, haciendo hincapié en el problema de los datos desbalanceados. Con este fin, aplicaremos tres técnicas diferentes de *upsampling*, prediciendo posteriormente mediante el modelo de regresión logística. Las tres técnicas se compararán para ver cuál proporciona mejores resultados.

En primer lugar, tal y como se hizo en la sección anterior, dividiremos el conjunto en dos: conjunto de entrenamiento y conjunto de prueba (test). En esta ocasión la estratificación se realizará en la variable binaria objetivo, para que la proporción de películas premiadas sea similar en ambos conjuntos. Además, se generarán diez particiones del conjunto de entrenamiento mediante validación cruzada, que serán usadas con posterioridad.

```
oscars_split = initial_split(datos_premios_im, strata = Oscars)
oscars_train = oscars_split %>% training()
```

```
oscars_test = oscars_split %>% testing()

oscars_folds = vfold_cv(oscars_train, strata = Oscars)
```

En segundo lugar, se define el modelo de regresión logística ya que será el que se usará para las tres técnicas de *upsampling*:

```
glm_mod = logistic_reg() %>%
  set_engine("glm")
```

El siguiente paso será evaluar el modelo para cada una de las tres técnicas de *upsampling* seleccionadas. Los tres algoritmos seleccionados y sus resultados se exponen en los siguientes apartados.

7.3. Algoritmo ROSE

El primer algoritmo que se usará para solucionar el problema de los datos desbalanceados es el algoritmo ROSE, descrito en 7.3. Este algoritmo selecciona una observación perteneciente a la clase deseada y genera nuevos ejemplos en su vecindad. Para implementarlo, se añade *step_rose* a la receta.

Una vez creada la receta y añadida al flujo de trabajo correspondiente, se entrena el modelo usando para ello las particiones generadas por validación cruzada:

```
doParallel::registerDoParallel()
glm_rose_rs = glm_rose_wflow %>%
  add_model(glm_mod) %>%
  fit_resamples(
    resamples = oscars_folds,
    metrics = metric_set(roc_auc, accuracy, sensitivity, specificity),
    control = control_resamples(save_pred = TRUE)
  )
```

Se han añadido todas las métricas que aportan información provechosa para evaluar el modelo, así como se ha indicado que devuelva las predicciones realizadas. Una vez completado este paso, el resultado es el modelo entrenado usando las particiones generadas por el método de validación cruzada, sobre las que se ha aplicado el algoritmo de *upsampling*.

Para evaluar la precisión del modelo, se representa en primer lugar la curva ROC. Como se ha realizado el entrenamiento mediante los diez conjuntos generados por validación cruzada, se representará una curva para cada uno de ellos. De igual forma, se proporcionan las diferentes métricas indicadas en el paso anterior, teniendo en cuenta que son el resultado de la media de las métricas obtenidas en los diez subconjuntos. Los resultados son:

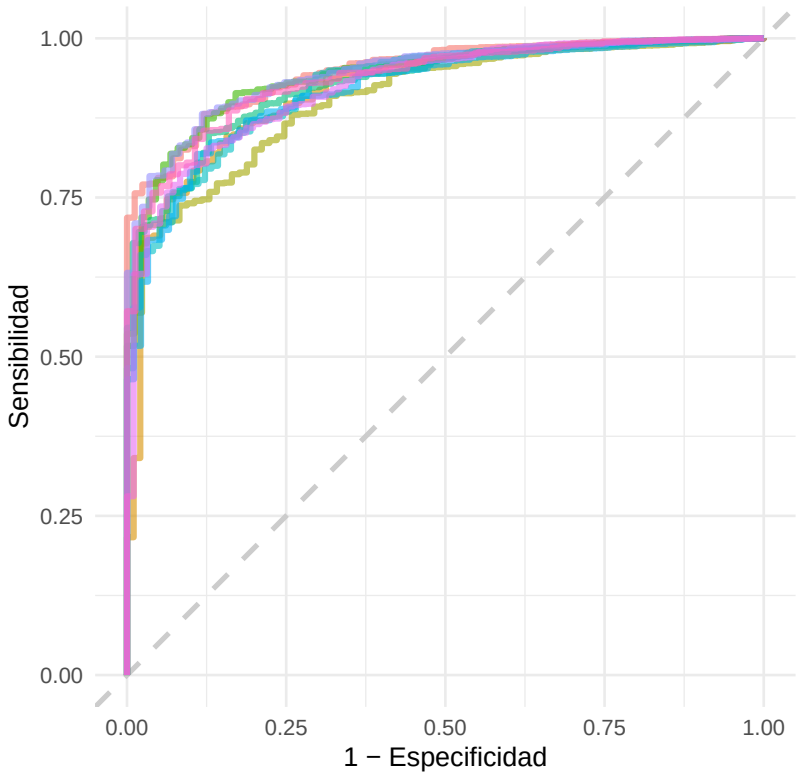


Figura 7.6: Curva ROC para la regresión logística con el algoritmo ROSE.

Tabla 7.3: Estadísticos de bondad del ajuste para la regresión logística con el algoritmo ROSE.

Precisión	Área bajo la curva	Sensibilidad	Especificidad
0.8313806	0.9296211	0.8309641	0.8681936

Recordemos que los resultados son mejores cuanto más cerca se encuentre la curva del punto (0,1), el cual indicaría una especificidad y una sensibilidad del 100 %. Parece que la curva muestra buenos resultados, aún así no perfectos. Las métricas devueltas señalan una especificidad y sensibilidad altas, de más del 80 %, así como la exactitud o *accuracy*. Por su parte, el área bajo la curva es mayor del 90 %.

Analizando también la matriz de confusión obtenida:

Tabla 7.4: Matriz de confusión para la regresión logística con el algoritmo ROSE.

	Real	
Predicción	0	1
0	6544	12.1
1	1353	79.6

Señalar de nuevo que, como se ha entrenado el modelo usando los subconjuntos proporcionados por la validación cruzada, las celdas de la matriz de confusión anterior muestran

la media de las celdas de cada una de las diez matrices de confusión creadas. Observamos como la clase problemática, categorizada como la clase del uno, se comporta adecuadamente dadas las circunstancias: de una media de cien observaciones en cada pliegue clasificadas como ganadoras de un premio Oscar, aproximadamente el 80 % se clasifican correctamente. Sin embargo, hay un número considerable de observaciones clasificadas como ganadoras cuando en realidad no lo son, es decir, falsos positivos.

Para terminar de evaluar la bondad del modelo, lo aplicamos al conjunto de prueba, estudiando las diferentes métricas en dicho caso. En primer lugar, observamos los coeficientes obtenidos en la regresión logística:

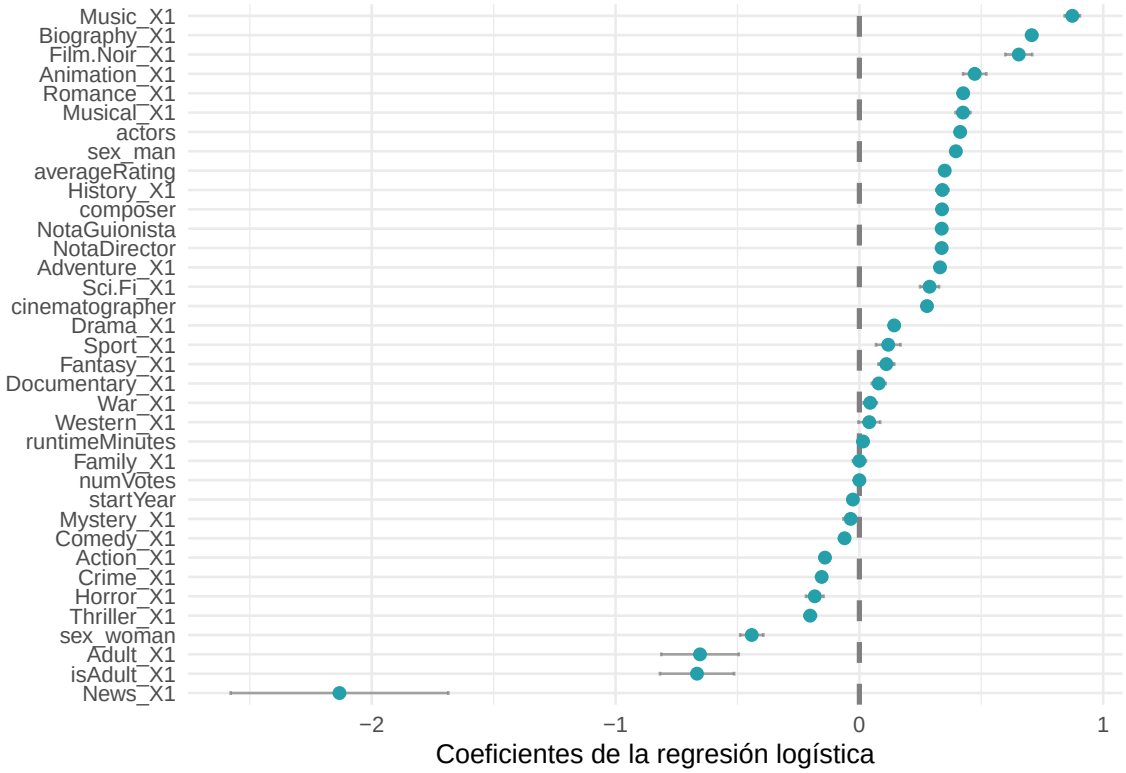


Figura 7.7: Coeficientes de la regresión logística usando el algoritmo ROSE.

Las variables con coeficientes en la parte positiva del eje de abscisas están relacionadas con ganar un premio Oscar. Estas son, entre otras, los géneros musical, biografía, animación y romance, tres de los cuales se encuentran entre los géneros con mayor número de premios. Las variables con coeficientes en la parte negativa del eje de abscisas están relacionadas con no ganar un premio Oscar. Por ejemplo, nos encontramos con el género noticias, que, recordando los análisis descriptivos previos, es uno de los géneros con menos premios. Por ende, los resultados coinciden con lo esperado del análisis descriptivo.

Por otra parte, analizamos la matriz de confusión obtenida así como las medidas de bondad del ajuste:

Tabla 7.5: Matriz de confusión para la predicción en el modelo de regresión logística con el algoritmo ROSE.

	Real	
Predicción	0	1
0	21747	44
1	4558	281

Tabla 7.6: Estadísticos de bondad del ajuste para la predicción en el modelo de regresión logística con el algoritmo ROSE.

Precisión	Sensibilidad	Especificidad	Área bajo la curva
0.8271874	0.826725	0.8646154	0.9189612

La exactitud es del 83 %, indicando que el 83 % de las observaciones han sido clasificadas correctamente. Por otro lado, el área bajo la curva es alta, al igual que ocurría con la obtenida de los datos correspondientes al conjunto de entrenamiento. La matriz de confusión muestra que la clasificación de las observaciones pertenecientes a la categoría asociada con ganar un Oscar es buena, clasificando únicamente 44 casos mal de 325, es decir, apenas un 12 %. Por otro lado, la cifra de falsos positivos es de cuatro mil observaciones, indicando que se han clasificado incorrectamente un 17 % de las películas no premiadas. Estos datos se reflejan en la especificidad y la sensibilidad, que muestran un buen rendimiento.

Una vez analizada la efectividad de este algoritmo, aplicamos el segundo de ellos para comparar resultados.

7.4. Algoritmo SMOTE

El segundo algoritmo que se va a aplicar para el *upsampling* es el algoritmo SMOTE. Este genera nuevas instancias de la clase minoritaria interpolando los valores de las observaciones minoritarias más cercanas a una dada.

Para implementarlo en el estudio, se añade el paso *step_smote* a la receta y se incorpora esta junto al modelo de regresión logística en un flujo de trabajo. A continuación, como se hizo en el caso anterior, se entrena el modelo usando los diez subconjuntos creados mediante validación cruzada a partir del conjunto de entrenamiento.

```
doParallel::registerDoParallel()
glm_sm_rs = glm_sm_wflow %>%
  add_model(glm_mod) %>%
  fit_resamples(
    resamples = oscars_folds,
    metrics = metric_set(roc_auc, accuracy, sensitivity, specificity),
```

```
    control = control_resamples(save_pred = TRUE)
)
```

Para evaluar el rendimiento del modelo se usarán las mismas métricas que antes, comenzando por visualizar la curva ROC y las medidas de bondad del ajuste obtenidas:

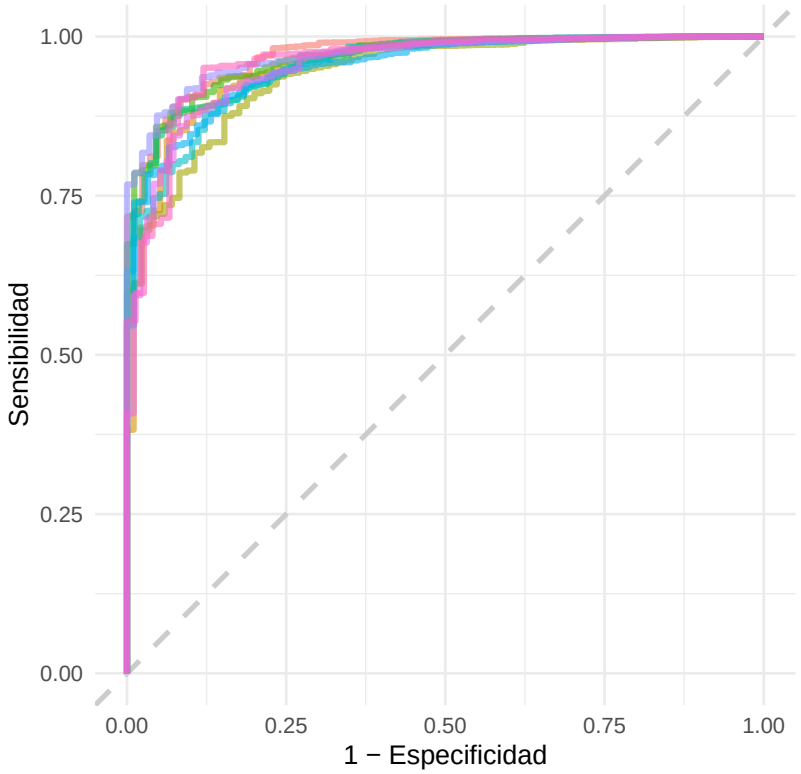


Figura 7.8: Curva ROC para la regresión logística con el algoritmo SMOTE.

Precisión	Área bajo la curva	Sensibilidad	Especificidad
0.8921392	0.9581063	0.8923444	0.8759678

Observando únicamente las curvas ROC se puede inferir que este modelo proporciona mejores resultados que el anterior, pues las curvas obtenidas están más cercanas al punto óptimo que cualquiera de las previas. Analizando las medidas de bondad del ajuste obtenidas, se deduce gracias al valor de la exactitud o *accuracy*, que con este modelo quedan correctamente clasificadas alrededor de un 90 % de las observaciones, casi un 10 % más que en el anterior. El resto de valores también son superiores a los obtenidos anteriormente, en especial la sensibilidad, que se ha incrementando considerablemente.

Por otro lado, analizando la matriz de confusión obtenida por medio de la validación cruzada:

Tabla 7.7: Matriz de confusión para la regresión logística con el algoritmo SMOTE.

	Real	
Predicción	0	1
0	7050.2	11.4
1	848.2	79.2

La tasa de falsos positivos ha disminuido considerablemente, lo que explica el aumento de la sensibilidad. Por otro lado, la tasa de falsos negativos también ha sufrido una mejora, aunque muy leve.

A la vista de estos resultados, parece ser que esta técnica de *upsampling* proporciona mejores resultados que la anterior. Para comprobarlo definitivamente, analizamos la matriz de confusión y las medidas de bondad del ajuste en el conjunto test:

Tabla 7.8: Matriz de confusión para la predicción en el modelo de regresión logística con el algoritmo SMOTE.

	Real	
Predicción	0	1
0	23445	47
1	2860	278

Tabla 7.9: Estadísticos de bondad del ajuste para la predicción en el modelo de regresión logística con el algoritmo SMOTE.

Precisión	Sensibilidad	Especificidad	Área bajo la curva
0.8907247	0.8911614	0.8553846	0.9489014

En este caso hay tres películas más que en el anterior se clasifican como no premiadas cuando sí que lo están. Sin embargo, el número de películas no premiadas clasificadas como que sí, es decir, la tasa de falsos positivos, es menor en este caso, como se vio en el conjunto de entrenamiento y como desvela la sensibilidad, que es superior en este caso. En vista de estos resultados, de los dos algoritmos escogidos para realizar el sobremuestreo, el método *SMOTE* es el que mejores resultados proporciona.

Para terminar, estudiemos los resultados que se obtienen aplicando la última técnica.

7.5. Sobremuestreo aleatorio

La última técnica de *upsampling* que se va a emplear es quizás la más simple, pues lo que hace es crear observaciones ficticias en la clase minoritaria mediante muestreo con reemplazamiento. Para implementarla, basta añadir en la receta *themis::step_upsampling*.

Al igual que en los dos casos previos se creará un *workflow* combinando la receta junto al modelo de regresión logística y se entrenará el modelo usando los diez subconjuntos generados por validación cruzada:

```
doParallel::registerDoParallel()
glm_up_rs = glm_up_wflow %>%
  add_model(glm_mod) %>%
  fit_resamples(
    resamples = oscars_folds,
    metrics = metric_set(roc_auc, accuracy, sensitivity, specificity),
    control = control_resamples(save_pred = TRUE)
  )
```

Para evaluar su rendimiento, observamos lo siguiente:

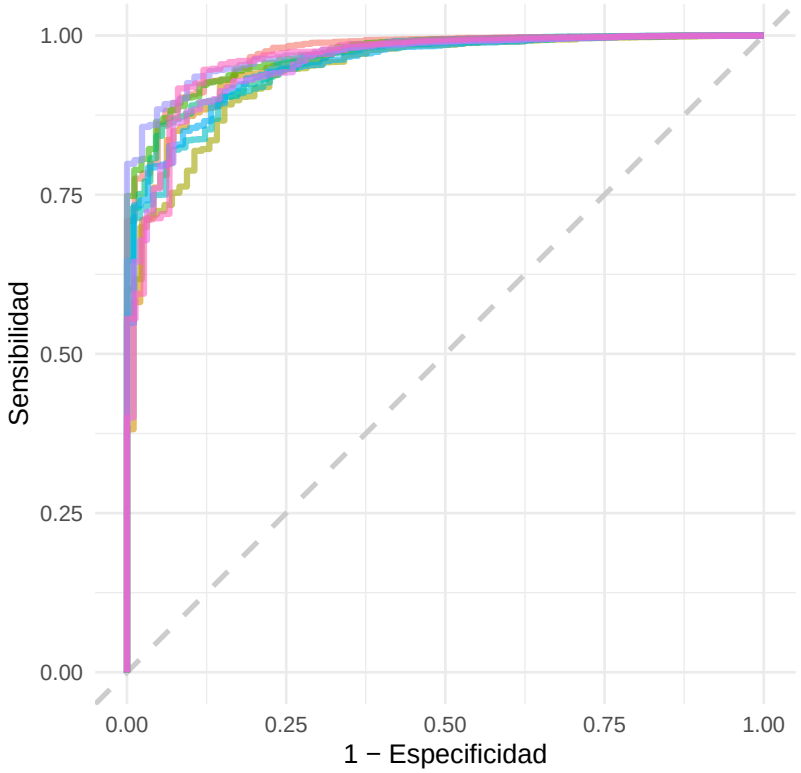


Figura 7.9: Curva ROC para la regresión logística con upsampling.

Tabla 7.10: Estadísticos de bondad del ajuste para la regresión logística con upsampling.

Precisión	Área bajo la curva	Sensibilidad	Especificidad
0.8884967	0.9597102	0.8884825	0.8908471

La curva ROC es parecida a la anterior, esto es, mejor que la primera que se obtuvo mediante el algoritmo ROSE. Las medidas de bondad del ajuste obtenidas también son similares, aunque en este caso la sensibilidad es levemente inferior, equiparandose con la

especificidad. La exactitud es ligeramente inferior, estando aproximadamente el 89 % de las observaciones bien clasificadas.

Por su parte, la matriz de confusión no hace más que corroborar estos hechos:

Tabla 7.11: Matriz de confusión para la regresión logística con upsampling.

	Real	
Predicción	0	1
0	7016.3	9.7
1	882.1	80.9

La clasificación de la clase de películas ganadoras es mejor que la realizada con el algoritmo anterior, pero a costa de incrementar el número de falsos positivos, como podía inferir al haberse visto reducida la sensibilidad. Por último, analizamos los resultados en el conjunto test:

Tabla 7.12: Matriz de confusión para la predicción en el modelo de regresión logística con upsampling.

	Real	
Predicción	0	1
0	23308	42
1	2997	283

Tabla 7.13: Estadísticos de bondad del ajuste para la predicción en el modelo de regresión logística con upsampling.

Precisión	Sensibilidad	Especificidad	Área bajo la curva
0.8852047	0.885421	0.8676923	0.9501106

Los resultados son similares a los obtenidos en el conjunto de entrenamiento.

Se concluye por tanto que los dos últimos modelos ofrecen mejores resultados que el primero. A pesar de que la técnica anterior proporciona una mejor clasificación de la clase negativa, esta última iguala los valores de la especificidad y la sensibilidad, produciendo por tanto los mejores resultados.

Para terminar esta sección, en la que se ha tratado el problema de clasificación de datos desbalanceados a través de tres técnicas diferentes, se va a usar la mejor técnica de sobremuestreo obtenida, esto es, la última, junto a un modelo diferente al de la regresión logística. Debido a que en la sección previa, dedicada a la regresión, el modelo que mejor resultados proporcionaba era *Random Forest*, se va a implementar este modelo para el problema de clasificación con objeto de analizar si es de nuevo el más efectivo.

7.6. Random Forest

Para aplicar Random Forest, se define primero el modelo a utilizar:

```
rf_mod = rand_forest() %>%
  set_mode("classification") %>%
  set_engine("ranger")
```

Se va a utilizar el algoritmo *ranger*, al igual que se hizo en la sección anterior, pero esta vez indicando que se trata de un problema de clasificación. Una vez definido el modelo, este se combina junto a la receta de *upsampling* usada en último lugar, para crear el flujo de trabajo correspondiente. Como en los casos anteriores, se entrena el modelo usando los diez conjuntos generados mediante validación cruzada:

```
rf_up_wflow = workflow() %>%
  add_recipe(oscars_up_rec)

rf_up_rs = rf_up_wflow %>%
  add_model(rf_mod) %>%
  fit_resamples(
    resamples = oscars_folds,
    metrics = metric_set(roc_auc, accuracy, sensitivity, specificity),
    control = control_resamples(save_pred = TRUE)
  )
```

Representando la curva ROC y calculando las medidas de bondad del ajuste usuales:

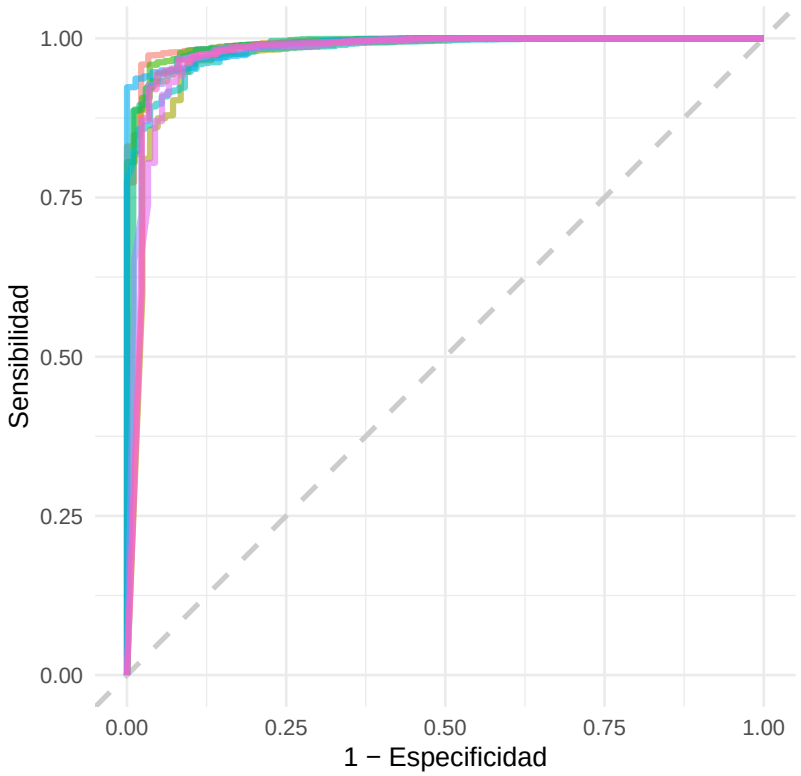


Figura 7.10: Curva ROC para random forest.

Tabla 7.14: Estadísticos de bondad del ajuste para random forest.

Precisión	Sensibilidad	Especificidad
0.9937164	0.9993541	0.5027743

Si observamos únicamente la curva junto a las medidas de exactitud, y sensibilidad concluiríamos que la clasificación es perfecta, pues el modelo clasifica correctamente en casi el total de los casos. Sin embargo, la especificidad delata el bajo rendimiento del modelo, pues esta es muy baja comparada con los anteriores. Esto es debido a que el modelo ha clasificado la mayoría de las observaciones como no ganadoras de un premio Oscar, incurriendo en el problema que se comentaba al principio de esta sección. Este hecho puede corroborarse analizando la matriz de confusión:

Tabla 7.15: Matriz de confusión para random forest.

	Real	
Predicción	0	1
0	7893.5	45.1
1	5.1	45.3

Como se observa, el modelo clasifica casi a la perfección la clase del cero, pero solo estan bien clasificadas el 50 % de las observaciones de la clase del uno, como se podía

deducir de la especificidad. Si comprobamos el rendimiento del modelo en el conjunto test:

Tabla 7.16: Matriz de confusión para la predicción en el modelo de random forest.

	Real	
Predicción	0	1
0	26280	23
1	175	152

Tabla 7.17: Estadísticos de bondad del ajuste para la predicción en el modelo de random forest.

Precisión	Sensibilidad	Especificidad
0.9925648	0.9991256	0.4648318

La matriz de confusión se comporta igual que en el conjunto de entrenamiento, clasificando perfectamente la clase del cero y mal la del uno. Clasifica correctamente todas las películas que no han ganado un premio Oscar excepto 23, siendo la tasa de falsos positivos ínfima. Sin embargo, clasifica incorrectamente más del 50 % de las películas que sí han ganado un Oscar, siendo por tanto la tasa de falsos negativos superior al 50 %. Como en el conjunto de entrenamiento, todas las medidas son muy buenas excepto la correspondiente a la especificidad, siendo por tanto peor que todos los modelos anteriores.

En resumen, englobando lo planteado en esta sección, se ha estudiado la predicción del éxito de una película a través de inferir si ganará o no un premio Oscar. El principal inconveniente surge de la desproporción en los datos: el número de películas galardonadas con un premio Oscar es notablemente inferior al que no ha ganado ninguna. Tras probar, usando regresión logística, tres técnicas diferentes de sobremuestreo para solucionar dicho problema de clasificación con datos desbalanceados, se ha llegado a la conclusión de que mediante el remuestreo aleatorio con reemplazamiento se obtienen los mejores resultados, igualando los valores de especificidad y sensibilidad. Por otra parte, probando otro modelo más complejo como Random Forest, cuyo tiempo de ejecución es mayor, se llega a la conclusión de que no por más complejo que sea el modelo significa que este ofrezca mejores resultados, ya que, como se ha visto en este caso, los resultados obtenidos con el modelo de regresión logística eran considerablemente mejores.

Parte IV

Conclusiones y futuras líneas de investigación

En este bloque se presentan las conclusiones principales obtenidas tras la realización del estudio. Además, se aportan algunas posibles líneas de investigación futuras que pueden enriquecer el estudio desarrollado.

Capítulo 8

Conclusiones y trabajo futuro

8.1. Conclusiones

Tal y como se ha manifestado a lo largo de todo el estudio, el mundo del cine y en particular la industria de las plataformas de streaming no escapan a la influencia de las matemáticas, la estadística y la inteligencia artificial, siendo todas estas herramientas útiles para estudiar su evolución y predecir su comportamiento a largo plazo.

Se ha estudiado el comportamiento de estas plataformas a lo largo de los años, analizando sus particularidades en distintos países. Las conclusiones obtenidas de esta parte son claras: las plataformas de streaming de video están en pleno auge, aumentando cada año considerablemente el número de suscriptores y por ende, sus ingresos. España, en particular, no se ha resistido a esta novedad y cada vez son más las personas de nuestro país abonadas a las mismas.

Al mismo tiempo, se ha visto cómo la desigualdad existente entre países se produce también en esta industria, encontrándonos con diferencias abismales de contenido entre unos países y otros, además de las vigentes en el precio y la calidad. Se han explorado las características principales de las cuatro plataformas seleccionadas, estableciendo cuál debería escoger un individuo en función de sus ventajas e inconvenientes y teniendo en cuenta su país: Amazon Prime Video es, en general, la opción más asequible relación contenido/precio y HBO posee un contenido de mayor calidad.

Destacar el principal reto en esta parte del estudio, el cual es uno de los obstáculos principales a la hora de trabajar con un caso real: la búsqueda exhaustiva de fuentes de información. La mayoría de los datos financieros oficiales de las diferentes plataformas de streaming son difíciles de localizar, por lo que cabe señalar el tiempo invertido en la búsqueda de los mismos.

Por otro lado, se han estudiado diversos modelos para predecir el éxito de las producciones. Al contrario de lo que se podría pensar, lo cierto es que el éxito de una producción está, dentro de unos límites, condicionado de antemano por las características que ésta posee. Se ha observado en los análisis cómo seleccionado el género o los integrantes del equipo principal de la película se tiene ya definido parte de su éxito, no queriendo decir con esto que no quede una componente debida al factor humano u otros aspectos ajenos a estas u otras variables.

Señalar que antes de evaluar los modelos, los datos utilizados han sido limpiados y tratados. Se han tomado distintas decisiones sobre cómo lidiar con ellos, con el objeto de facilitar tanto la implementación de los modelos como las conclusiones obtenidas.

Las diferentes técnicas aplicadas han proporcionado buenos resultados, obteniendo con un sencillo modelo de regresión un buen ajuste y mejorandolos con random forest. No ha sido así para el problema de clasificación desbalanceada propuesto, donde el algoritmo de random forest no proporcionaba mejores resultados. Aún así, la parte más importante de esta parte han sido las técnicas de sobremuestreo empleadas, que han proporcionado resultados notables.

8.2. Trabajo futuro

Una vez desarrollado el estudio, se han encontrado líneas de investigación con las que se puede mejorar y ampliar el espectro del problema.

En primer lugar, notar que la falta de información ha sido un constante inconveniente en el trabajo, en especial en lo referente al segundo bloque donde encontrar información acerca de los ingresos y suscriptores de las plataformas de streaming no ha sido sencillo. Por tanto, cualquier información nueva que se encuentre sobre este ámbito, ya sea porque no se ha dado con ella en este estudio como por que aún no se tenga, aportará nuevas y más enriquecedoras conclusiones.

Resaltar que una de las principales ideas antes de iniciar el estudio era tratar de predecir la evolución de las distintas plataformas, pero en vista de la escasez temporal de los datos disponibles para casi todas ellas, esto queda como posible trabajo futuro cuando se pueda contar con esta información.

En segundo lugar, cabe destacar que la cantidad de modelos estadísticos existentes es bastante amplia, y que los problemas de clasificación y regresión planteados en el tercer bloque se podrían haber abordado con otros completamente diferentes, obteniendo resultados quizás más interesantes. Naturalmente, se ha empleado una dedicación muy considerable en aspectos desde la búsqueda de información hasta todo el desarrollo, análisis estadístico y modelización, siempre justificando el proceso llevado a cabo y reflexionando sobre sus implicaciones, pero por más empeño que hemos puesto el tiempo no es infinito y hay una extensión y unas fechas que no pueden extenderse indefinidamente. A partir de aquí, si se dispusiera de más tiempo para continuar trabajando en esta línea se podrían probar distintos modelos con distintas fortalezas y debilidades, pero debido a las limitaciones existentes se concluye que cualquier otro modelo que se aplique al mismo conjunto de datos será objeto digno de estudio y comparables con los aquí descritos.

Por otro lado, cualquier conjunto más amplio de películas también aportará nuevos resultados y quizás mejore los obtenidos. Además, parte del tratamiento que han recibido los datos se ha realizado de forma subjetiva, por lo que cualquier otro enfoque sería de gran utilidad.

Bibliografía

ATT. Investor relations | att. URL https://investors.att.com/.

World Bank. Regiones. URL https://www.bancomundial.org/es/about/annual-report-2015-copy/annual-report1/regions.

Jeff Bezos. Amazon letter to shareholders, 2018. URL https://s2.q4cdn.com/299287126/files/doc_financials/annual/Amazon_Shareholder_Letter.pdf.

Ruchi Bhatia. Movies on netflix, prime video, hulu and disney+, Mayo 2020. URL https://www.kaggle.com/ruchi798/movies-on-netflix-prime-video-hulu-and-disney.

CNMCData. Cnmcdata - comision nacional de los mercados y la competencia. URL http://data.cnmc.es/datagraph/.

Daniel Peña Sánchez de Rivera. *Estadística : modelos y métodos. vol.2, Modelos Lineales y series temporales*. Alianza, 1987.

Disney. Investor relations - stock information, events, reports, financial information, shareholder information - the walt disney company. URL https://thewaltdisneycompany.com/investor-relations/#reports.

Raphael Fontes. The oscar award, 1929-2020, Febrero 2020. URL https://www.kaggle.com/unanimad/the-oscar-award.

Wolfgang Karl. Härdle and Léopold. Simar. *Applied Multivariate Statistical Analysis*. Springer International Publishing, 5th ed. 2019. edition, 2019. ISBN 3-030-26006-2. doi: 10.1007/978-3-030-26006-4.

IMDb. Imdb datasets. URL https://www.imdb.com/interfaces/.

JustWatch. Justwatch - the streaming guide. URL https://www.justwatch.com/.

Max Kuhn and Hadley Wickham. *Tidymodels: a collection of packages for modeling and machine learning using tidyverse principles.*, 2020. URL https://www.tidymodels.org.

Max Kuhn and Hadley Wickham. *tidymodels: Easily Install and Load the Tidymodels Packages*, 2021. URL https://CRAN.R-project.org/package=tidymodels. R package version 0.1.3.

Rajat Kumar. Hbo 500 movies rating, Octubre 2020. URL https://www.kaggle.com/rajatkumar30/hbo-500-movies-rating.

Rocío Espinar Lara and Rafael Pino Mejías. Modelos de clasificación con datos no balanceados, 2018.

Pedro L. Luque-Calvo. *Escribir un Trabajo Fin de Estudios con R Markdown*, 2017.

Pedro L. Luque-Calvo. *Cómo crear Tablas de información en R Markdown*, 2019.

Netflix. Netflix investors, 2010-2020a. URL https://ir.netflix.net/ir-overview/profile/default.aspx.

Netflix. Netflix financials - quarterly earnings, 2010-2020b. URL https://ir.netflix.net/financials/quarterly-earnings/default.aspx.

Laura Perez and Marcos Mendez. Amazon, netflix, hbo... y más: qué ofrecen y cuánto cuestan las 12 plataformas en españa, Noviembre 2019. URL https://vertele.eldiario.es/noticias/comparacion-plataformas-television-streaming-netflix-apple-disney-hbo_1_7427175.htmlhttps://es-statista-com.eu1.proxy.openathens.net/estadisticas/1067435/coste-mensual-de-la-suscripcion-a-las-diferentes-plataformas-audiovisuales-de-pago-espana/.

Cristian Fernando. Tellez Piñerez and Mario Alfonso Morales Rivera. *Modelos estadísticos lineales : con aplicaciones en R.* Ediciones de la U, 2016. ISBN 9789587624786. Contiene bibliografía.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2016. URL https://www.R-project.org/.

David Robinson and Julia Silge. *tidytext: Text Mining using dplyr, ggplot2, and Other Tidy Tools*, 2021. URL https://github.com/juliasilge/tidytext. R package version 0.3.1.

Julia Silge and David Robinson. tidytext: Text mining and analysis using tidy data principles in r. *JOSS*, 1(3), 2016. doi: 10.21105/joss.00037. URL http://dx.doi.org/10.21105/joss.00037.

Todd Spangler. Amazon has more than 100 million prime subscribers, jeff bezos says - variety, Abril 2018. URL https://variety.com/2018/digital/news/amazon-prime-100-million-subscribers-jeff-bezos-1202757832/.

Statista. Netflix: número de suscriptores en todo el mundo, 2011-2020. URL https://es.statista.com/estadisticas/598771/numero-de-suscriptores-netflix-en-streaming-en-todo-el-mundo/.

Statista. Netflix: number of subscribers worldwide, 2013-2020. URL https://www.statista.com/statistics/250934/quarterly-number-of-netflix-streaming-subscribers-worldwide/.

Statista. Statista global consumer survey, 2018-2020. URL https://es.statista.com/global-consumer-survey?from=%252Fglobal-consumer-survey%252Fsurveys.

Time Warner. Form 10-k annual report pursuant to section 13 or 15(d) of the securities exchange act of 1934 for the fiscal year ended delaware 13-4099534 securities registered pursuant to section 12(b) of the act: Title of each class name of each exchange on which registered, 2017. URL www.timewarner.com.

Hadley Wickham. *tidyverse: Easily Install and Load the Tidyverse*, 2021. URL https://CRAN.R-project.org/package=tidyverse. R package version 1.3.1.

Hadley Wickham, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, Alex Hayes, Lionel Henry, Jim Hester, Max Kuhn, Thomas Lin Pedersen, Evan Miller, Stephan Milton Bache, Kirill Müller, Jeroen Ooms, David Robinson, Dana Paige Seidel, Vitalie Spinu, Kohske Takahashi, Davis Vaughan, Claus Wilke, Kara Woo, and Hiroaki Yutani. Welcome to the tidyverse. *Journal of Open Source Software*, 4(43):1686, 2019. doi: 10.21105/joss.01686.

World Bank World Development Indicators. Gdp per capita (current us dolars), 2019. URL https://data.worldbank.org/indicator/NY.GDP.PCAP.CD.