# Características más relevantes que determinan el rating de una aplicación móvil

Juan Carlos Aguilar Alfaro[1], Andrés González Fallas[1], Daniel Núñez Vargas[1]

juan.aguilaralfaro@ucr.ac.cr, andres.gonzalezfallas@ucr.ac.cr, daniel.nunezvargas@ucr.ac.cr

## RESUMEN

Las aplicaciones móviles se han vuelto parte del día a día de las personas. Es por eso que se pretende realizar una investigación sobre cuales factores afectan la calificación de estas y también que tan buenos son prediciendo una calificación los modelos propuestos. Para este trabajo, se utilizaron datos de aplicaciones móviles obtenidas de la Google Play Store. Se obtuvo la información de 7 421 aplicaciones y 15 variables. Además, se utilizaron dos modelos estadísticos para contrastar los resultados obtenidos por cada uno, estos modelos fueron regresión lineal múltiple y árboles de decisión. Los resultados en cuanto al modelo de regresión lineal indican que los factores más significativos de acuerdo con los p-valores del modelo son la antigüedad, si es gratuito o de pago y la cantidad de reseñas de la aplicación, en ese orden. Mientras que, para el modelo de árbol de decisión, este arrojó que los factores que son más significativos para la calificación de la aplicación son la cantidad de reseñas, la antigüedad y el tamaño de la aplicación. Para la capacidad predictiva de los modelos, se llegó a mejores indicadores por parte del modelo de la regresión lineal pues obtuvo un error cuadrático medio de 0.96 y una correlación entre el dato real y el predicho de 0.20. En contraste, el modelo de árbol de decisión obtuvo un error cuadrático medio de 0.98 y una correlación de 0.16. Se concluye gracias a ambos modelos que los factores que más impacto tienen sobre la calificación de una aplicación móvil son la cantidad de reseñas y la antigüedad de la aplicación. También, para futuras investigaciones considerar métodos de clasificación de variable ordinal u otros métodos de árboles más complejos.

**PALABRAS CLAVE:** rating, aplicación móvil, características, regresión lineal, árboles de decisión

---

[1] Estudiantes de Escuela de Matemática y Ciencias Actuariales

## INTRODUCCIÓN

Actualmente, se vive en una era donde el factor predominante en la vida cotidiana es la tecnología. Las aplicaciones han sido parte de esa evolución y llegaron para simplificarle la vida a sus usuarios. Prácticamente para cualquier tópico o actividad existe una aplicación que puede facilitar su elaboración hasta el punto que hoy en día se puede decir que los dispositivos inteligentes se utilizan más por sus aplicaciones que por su característica elemental que era hacer llamadas. Es bajo esta lógica que se propone una investigación para intentar encontrar una relación entre el rating de una aplicación con sus características.

Para llevar a cabo dicha investigación se utilizarán datos que se obtuvieron del sitio web Kaggle, en donde se menciona que para recolectar los datos utilizaron el método de "scraping" de la Google Play Store. La tabla cuenta con 10841 observaciones y 13 variables. Cada observación o fila de la tabla es una aplicación dentro de la Play Store. Analizando los datos se puede inferir que los datos fueron recolectados entre el 2018 y el 2019. En cuanto al contexto espacial, es bien sabido hay aplicaciones que solo están disponibles en ciertas áreas regiones y los datos fueron obtenidos inicialmente en la India por lo que es de esperarse que haya aplicaciones que no estén disponibles en Costa Rica.

La pregunta central de investigación será: ¿Cuáles son las variables más relevantes para explicar el rating o calificación de una aplicación móvil en la Google Play Store? La relevancia de un proyecto como este yace en el hecho de lo importante que pueden llegar a ser las aplicaciones móviles en la vida cotidiana, por lo que tiene una relevancia social bastante importante, ya que la sociedad hoy en día se basa de muchas formas en aplicaciones móviles.

## METODOLOGÍA

La tabla de datos se obtuvo del sitio web kaggle, en donde se menciona que para obtener los datos utilizaron el método de "scraping" de la Google Play Store. La tabla cuenta con 10841 observaciones y 13 variables. Cada observación o fila de la tabla es una aplicación dentro de la Play Store. La base tenía observaciones duplicadas, así como valores transpuestos o *NA*, por lo que se tuvo que realizar un proceso de depuración para eliminar todos los datos que no se pudieran usar reduciendo las observaciones a 7421.

En cuanto al contexto temporal, la tabla no cuenta con el dato específico, pero al observar una columna que muestra la última actualización registrada de la aplicación al momento de obtener los datos, se puede inferir que los datos fueron recuperados entre el 2018 y el 2019. En cuanto al espacio temporal

de la tabla, es sabido que en estas tiendas móviles algunas aplicaciones pueden estar disponibles o no de acuerdo a la región donde se ingrese, en este caso tampoco se menciona si los datos tienen alguna restricción de este tipo o cual versión de la tienda móvil se utilizó. Sin embargo, el dataset fue proporcionado por una estudiante de la India por lo que se puede considerar a futuro si esto sugiere si hay aplicaciones distintas a las que se pueden encontrar en la tienda de Costa Rica u otros países.

La base de datos cuenta con las siguientes columnas que vendrían siendo nuestras variables de estudio para responder la pregunta de investigación:

- **App:** Nombre de la aplicación
- **Category:** Categoría de la aplicación, donde se cuentan con 33 diferentes categorías entre las cuales se distinguen: *Eventos, Deportes, Juegos, aplicaciones de citas o "dating",* entre otras.
- **Rating:** Calificación de la aplicación donde el intervalo se extiende de 1 a 5, el 1 siendo la calificación más bajo y 5 el más el alto.
- **Reviews:** Cantidad de calificaciones para la aplicación, corresponde a una variable numérica entera
- **Size:** Tamaño de la aplicación en megabytes, esta es una variable categórica donde hay 414 posibilidades
- **Installs:** Un aproximado de la cantidad de instalaciones que tiene la aplicación, es una variable categórica ordinal que consta de 19 niveles
- **Type:** Si es de pago o es gratis, corresponde a una variable categórica
- **Price:** El precio de la aplicación en caso de que sea de pago
- **Content Rating:** La calificación de la aplicación de acuerdo al contenido de madurez o grupo de edad a la cual está dirigida
- **Genres:** Aparte de la categoría principal una aplicación puede pertenecer a diferentes géneros o subcategorías, hay 115 posibilidades
- **Last Updated:** Última actualización que recibió la aplicación
- **Current Version:** Última versión de la aplicación

- **Android Version:** Versión mínima de sistema operativo Android que puede ejecutar la aplicación

Aparte de estas variables se creó uno nuevo, que mide la diferencia en días de la última que vez que la aplicación fue actualizada con respecto a la fecha más reciente, esto pues no se tiene certeza de la fecha de la obtención de datos, pero gracias a esto se obtiene una idea de cuándo fue la actualización de la aplicación al momento de recolección de datos.

Cabe recalcar que todos estos datos están sujetos a la fecha de recopilación de la información. Además, se define la población de estudio como las aplicaciones móviles de la Google Play Store mientras que la muestra observada aquellas aplicaciones que se obtuvieron por medio de *web scraping* en una fecha comprendida entre el año 2018 y 2019. Bajo la misma línea, se define la unidad estadística como la aplicación móvil que estuvo en la plataforma de Google Play Store entre los años 2018 y 2019. A su vez, este proyecto fue realizado mediante el lenguaje de programación R en su versión 4.1.3, lenguaje utilizado para implementar los modelos y cálculos que permitieron el desarrollo de la investigación.

Como apoyo teórico se va a tomar en cuenta la investigación realizada por Suleman, Malik y Hussain en su trabajo que forma parte de una serie de artículos publicados y argumentados en la Conferencia Internacional de Ciencia de Datos del 2019 y fue realizado por un grupo de expertos en el tema, los cuales se propusieron comparar y diferenciar distintos algoritmos y métodos de Machine Learning para averiguar cual tenía mejor rendimiento a la hora de predecir la calificación de aplicaciones móviles. Debido al hecho de que utilizaron variables similares a las que se van a usar en esta investigación, se permite tener una base de la cual partir a la hora de escoger un método sobre otro. Entre los métodos que se utilizaron en este estudio está la regresión lineal que fue inicialmente el primer intento de inferencia que se utilizó para contestar la pregunta de investigación.

Una regresión lineal es una técnica que permite expresar la relación entre un conjunto de variables $X$ y una variable $Y$. La ecuación de regresión múltiple se utiliza para obtener valores de la variable dependiente $Y$ dados valores de la variable independiente $X$. Esta ecuación puede ser escrita de manera matricial mediante la ecuación:

$$y = X\beta + \epsilon$$

donde ahora $y$ y $\epsilon$ son un vector de dimensión $n \times 1$ donde y $n$ denota el número de observaciones. La matriz $X$, que contiene las observaciones de las variables predictoras, tiene un dimensión de $n \times p$ donde $p$ denota la cantidad de variables predictoras.

Al conjunto de respuestas $y_i$ se les conoce como variables dependientes o variables respuesta. El parámetro $\beta$ es un vector cuya dimensión corresponde al número de variables predictoras. Este parámetro $\beta$ se estima mediante una técnica llamada mínimo cuadrados que consiste en determinar la línea recta del diagrama de dispersión que minimice la suma de los cuadrados de las distancias de todos los puntos a la recta.

Dicho de manera más simple, la técnica de mínimos cuadrados se utiliza para encontrar la recta de regresión que minimiza los residuos, donde se entiende como residuo las diferencias entre los valores reales y los estimados por la recta de regresión lineal. Para ello, y considere las distancias para cada una de las observaciones del modelo

$$S(b) = \sum_{i=1}^{n} (y_i - x_i \cdot b)^2 = (y - Xb)^T (y - Xb)$$

Entonces, el parámetro $\beta$ se estima encontrando el $b$ que minimiza esos residuos. Se puede denotar de forma matricial el parámetro $\hat{\beta}$ como:

$$\hat{\beta} = (X^T X)^{-1} X^T y$$

Para estimar el error $\hat{\epsilon}$ se hace:

$$\hat{\epsilon} = y - X\hat{\beta}$$

De esta forma ya se puede calcular las valores $y_i$ resolviendo el sistema

$$\hat{y} = X\hat{\beta} + \hat{\epsilon}$$

Otro de los algoritmos que usan los autores Suleman, Malik y Hussain son los árboles de decisión. Para explicar la metodología que hay detrás de esta técnica se utilizará como principal base teórica el libro *An Introduction to Statistical Learning*, en donde se detalla que hay dos pasos principales a seguir:

1. Dividir el espacio predictor, que son los valores $X_1, \dots X_j$, en J regiones distintas y disjuntas, $R_1, \dots R_j$

2. Para cada observación que caiga sobre la región $R_j$, se hace la misma predicción, que es la media de los valores de respuesta para los valores en $R_j$

Estas regiones podrán tener cualquier forma. Se buscan regiones $R_j$ que minimicen el RSS que se puede escribir como:

$$RSS = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{j=1}^{J} \sum_{i \in R_j} \left( y_i - \hat{y}_{R_j} \right)^2$$

No es computacional eficiente considerar todas las particiones del espacio predictor $(X_1, \dots, X_j)$ en $J$ cajas, por lo que se usa una división binaria de arriba hacia abajo y codiciosa. Esto significa que cada división que origina una variable $X_j$ divide el espacio de variables en dos regiones, que corresponden a dos ramas. A la hora se realizar el árbol de decisión se consideran todas las variables predictoras $X_1, \dots, X_p$ con todos los valores de división $s$ para cada variable. Ahora basta considerar la variable $X_j$ y el punto $s$ que minimice el RSS y esto genera el un nodo raíz que a su vez genera dos ramas. Se continua hasta que se alcance un criterio de parada.

De manera formal se intenta resolver el siguiente problema de optimización, según:

$$min_{j,s}\{min_{c_1} \sum_{x_i \in R_1(j,s)} (y_i - c_i)^2 + min_{c_2} \sum_{x_i \in R_2(j,s)} (y_i - c_i)^2\}$$

Mediante greedy search se resuelve el problema anterior, que genera dos regiones:

$$R_1(j,s) = \{X \big| X_j \leq s\} \text{ y } R_2(j,s) = \{X \big| X_j > s\}$$

En particular para efectos de comparabilidad sobre la relevancia de las variables tenemos que la magnitud de las betas de la regresión nos da la relevancia, es por esta razón que antes de estimar el modelo se ha procedido a estandarizar las variables numéricas a fin de que diferencias en escalas o unidades de medición no incidan en el valor y por ende la magnitud del beta estimado.

Por otro lado, mediante un muestreo aleatorio usando 60 por ciento para entrenamiento se procede a dividir el set de datos con el fin de poder calcular las métricas de naturaleza más predictiva.

En el caso del árbol de regresión se utiliza el estadístico conocido como "feature importance" que mide el aporte en el uso de una variable, básicamente para conocido el error del modelo, se calcula la diferencia entre ese error original y el error que resulta de predecir permutando los valores de la columna. Dentro de este proceso metodológico se busca en primera instancia que el modelo tenga sentido así como otras métricas estadísticas como el $R$ cuadrado en el caso de la regresión, el error cuadrático medio e

incluso la correlación que existe entre la predicción del modelo y el verdadero valor en el set de pruebas. Para efectos del árbol de regresión se optimizó el hiperparámetro de complejidad $cp$ mediante validación cruzada de manera que se minimice el error cuadrático medio (el parámetro cp es un umbral que permite splits siempre y cuando haya una mejora mínima en las métricas y va a representar el criterio de parada de este proyecto)

La validación cruzada en este caso consiste en que primero se divide la muestra en k partes, se entrena con k-1 folds y testea con el k-esimo fold, este proceso se realiza para cada uno de los k-folds, luego se estima el siguiente estimador que en esta investigación utiliza al error cuadrático medio como métrica a optimizar:

$$CV_k = \frac{1}{k}\sum_{i=1}^{k}\sqrt{\frac{\sum_{j=1}^{n}\left(y_{ij} - \hat{y}_{ij}\right)^2}{n}}$$

Se utiliza validación cruzada con K-folds (se utilizan 10 folds en esta investigación al ser un valor estándar) como método en la optimización de hiperparámetro debido a las ventajas que presentan según como que es más ventajoso computacionalmente, además de que con 5 o 10 folds se sabe que empíricamente se alcanza un equilibrio entre sesgo y varianza en el error de los estimadores, y se ampara en las consecuencias de la ley fuerte de los grandes números y el hecho de que al promediar se tiene un estimador insesgado que mejora con el tamaño de los datos.

## RESULTADOS

Ahora se presentarán los resultados de implementar los pasados métodos a los datos que ya tenemos. Con los resultados de la regresión y considerando los niveles de significancia de las variables, se puede deducir que entre las variables que aparecen con mayor significancia está las Reviews, la Edad o si es paga o no. Luego aparecen demás variables que no tienen una significancia considerable como las categorías de LifeStyle y Productivity.

A continuación, las variables más importantes para los árboles:

**Figura 1**

*Variables más importantes según el modelo de árbol sin optimizar*

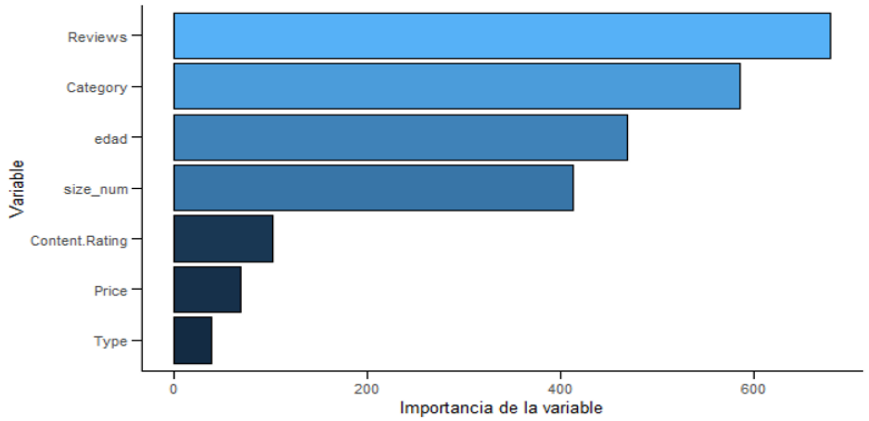
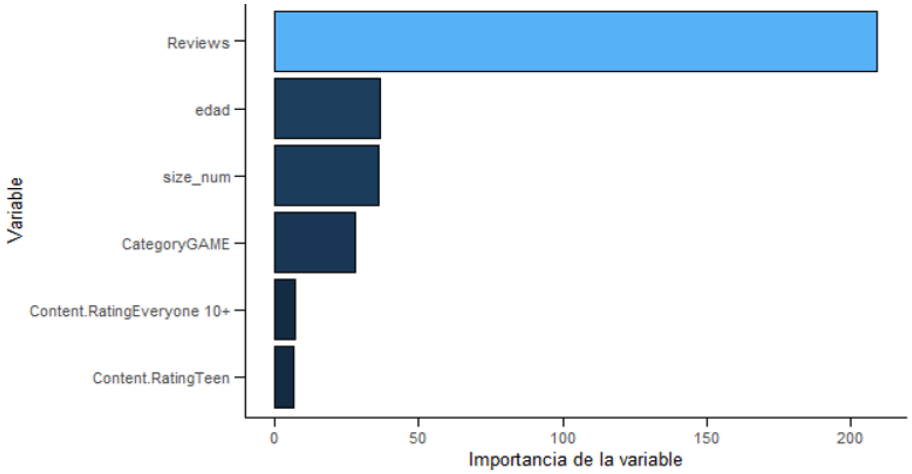



**Figura 2**

*Variables más importantes según el modelo de árbol optimizado*




Para el caso de los árboles de decisión, tanto en el caso del árbol optimizado y sin optimizar, la variable más relevante es el Reviews, sin embargo, en el caso del árbol optimizado esta variable tiene mucho mayor relevancia que en el caso del árbol sin optimizar. Otras variables que se distinguen entre las más importantes para ambos árboles son la Edad, Tamaño y el Content Rating.

**Tabla 1**

*Indicadores de predicción de los modelos utilizados*

| Indicador | Regresión lineal | Árbol de decisión | Árbol de decisión optimizado |
|---|---|---|---|
| Error cuadrático | 0.9579 | 1.15 | 0.9797 |
| Error relativo | 0.9779 | 1.049 | 0.9887 |
| Correlación | 0.2094 | 0.1821 | 0.1559 |

En este caso se puede observar una mejora con respecto a la capacidad predictiva del modelo anterior de árbol de decisión, sin embargo, este sigue por debajo del modelo de regresión lineal. Aunque, se puede observar que la diferencia entre la regresión y este nuevo modelo se acortó bastante, aun así, la correlación entre los valores verdaderos y los predichos de los datos de prueba (testing) bajó con respecto al árbol original que también es algo a considerar.

## CONCLUSIONES

Dada la realidad actual de las aplicaciones y la importancia que tienen en el día a día actualmente, puede considerarse de importancia conocer los parámetros que determinan el éxito que puede llegar a tener. Es por ello que se adoptó una pregunta de investigación cuyo fin fuese determinar la relación entre el rating de una aplicación con sus respectivas características.

Para llevar a cabo dicho proceso se utilizaron dos métodos estadísticos, el primero fue una regresión lineal mientras que se optó como segunda opción fue árboles de decisión. A partir de ambos métodos se lograron extraer conclusiones significativas, como, por ejemplo, a nivel predictivo la regresión lineal arroja mejores resultados que los árboles de regresión.

Además, en el caso de la regresión lineal se determina que las variables más relevantes son la cantidad de Reviews, la Edad o si es paga o no. Hay muchas categorías que no presentan una significancia considerable. Para el caso de los árboles de regresión, tanto en el caso del árbol optimizado y sin optimizar, la variable más relevante es el Reviews, sin embargo, en el caso del árbol optimizado esta variable tiene mucho mayor relevancia que en el caso del árbol sin optimizar. Otras variables que se distinguen entre las más importantes para ambos árboles son la Edad, Tamaño y el Content Rating. La

base clasifica las aplicaciones en más de 30 categorías diferentes y trabajar con tantas categorías no fue para nada eficiente porque influían significativamente en los resultados.

Entre las recomendaciones que se extienden para futuras investigaciones que quieran trabajar un problema similar se distinguen trabajar con modelos de clasificación de variable ordinal en vez de modelos de regresión ya que por sí sola la regresión no mostró una distribución lineal de sus errores.

## BIBLIOGRAFÍA

Dridi, R., Zammali, S., Alsulimani, T., y Arour, K. (2020). Effective rating prediction based on selective contextual information. Information Sciences, 510, 218-242. Descargado de https://www.sciencedirect.com/science/article/pii/S0020025519308540 doi: https://doi.org/10.1016/j.ins.2019.09.008

Gnotthivongsa, N., y Huang, D. J. (2021). Rating prediction for mobile applications via collective matrix factorization considering app categories. En Journal of physics: Conference series (Vol. 1993, p. 012034).

Hastie, T., Tibshirani, R., y Friedman, J. (2009). The elements of statistical learning: data mining, inference, and prediction. springer series in statistics. Springer New York.

INRIA. (2022). Scikit-learn course. Instituto nacional francés de las ciencias digitales. https://inria.github.io/scikit-learn-mooc/.

James, G., Witten, D., Hastie, T., y Tibshirani, R. (2013). An introduction to statistical learning (Vol. 112). Springer.

Liang et al. (2017). Mobile application rating prediction via feature-oriented matrix factorization. 2017 IEEE 24th International Conference on Web Services.

Molnar, C. (2020). Interpretable machine learning. Lulu. com.

Suleman, M., Malik, A., y Hussain, S. S. (2019). Google play store app ranking prediction using machine learning algorithm. Urdu News Headline, Text Classification by Using Different Machine Learning Algorithms 57.