ISSN 1870-4069

Predicción del éxito de proyectos de videojuegos en Kickstarter con aprendizaje automático

Salvador Nevarez-Castellanos[1], Vicente García[1], Gilberto Rivera[1], Rogelio Florencia[1], Julia Patricia Sánchez-Solís[1], Alberto Ochoa-Zezzatti[2]

[1] Universidad Autónoma de Ciudad Juárez
División Multidisciplinaria en Ciudad Universitaria, México
[2] Universidad Autónoma de Ciudad Juárez
Departamento de Ingeniería Industrial y Manufactura, México

salva.neva96@gmail.com, vicente.jimenez@uacj.mx

Resumen. Kickstarter se ha convertido en el principal medio de financiamiento para proyectos de desarrolladores independientes. En esta plataforma se han subvencionado exitosamente alrededor de 2,500 proyectos de videojuegos. Aún cuando la de campañas exitosas parece considerable, es pertinente mencionar que 12,500 campañas de esta misma categoría fracasaron en recaudar su meta. Kickstarter establece en sus políticas que el capital recaudado en una campaña debe ser devuelto a sus respectivos contribuyentes, repercutiendo negativamente a los desarrolladores. Por lo tanto, resulta importante encontrar un medio de predección del éxito/fracaso de las campañas de videojuegos. Utilizando una base de datos con datos con variables homologadas de diversas fuentes como Kaggle, Kickstarter, Facebook y otras redes sociales, en el presente artículo se propone el uso de técnicas de desabalance de clases y la regresión logística para la predicción del éxito/fracaso de las campañas. Los resultados experimentales mostraron que es posible predecir el éxito/fracaso de una campaña con una media geométrica de 0.8870.

Palabras clave: Kickstarter, regresión logística, predicción, videojuegos.

Predicting the Success of Kickstarter-Funded Video Games with Machine Learning

Abstract. Kickstarter has become Independent Game Developers' main means of projects financing. This platform has allowed the founding of around 2,500 videogame projects. Despite this being a considerable

amount of successfully founded projects, around 12,500 videogame projects have failed to reach their goal. Kickstarter establishes that the money raised in failed campaign must be returned to their respective contributors, thus, affecting developers negatively. Therefore, it becomes important to find a way of predicting the failure/success of videogame campaigns. In this article, the use of logistic regression is proposed to perform this task. Using a database wich included standarized data from diverse sources such as Kagle, Kickstart, Facebook, Youtube and other social media, the present article proposes unbalanced classes and logistic regression techniques use to predict campaigns success/failure. With a geometric median of 0.8870, experiments' results showed that it is possible to predict campaigns' success/failure.

Keywords: Kickstarter, logistic regression, prediction, video games.

1. Introducción

La industria de los videojuegos tiene gran impacto económico en el mercado internacional. En el año 2017 las ganancias generadas por videojuegos sumaron un total de 121 billones de dólares a nivel mundial (presentando un incremento del 15 % anual) [1]. Dichas ganancias son una suma de aquellas generadas por empresas *triple A* y desarrolladores independientes. Las empresas independientes, también denominadas *indies*, se diferencían de las triple *triple A* por no disponer de presupuestos fijos para la producción de videojuegos. En esta categoría recaen pequeñas empresas, grupos de desarrollo informales o desarrolladores individuales. Para solventar los gastos derivados del desarrollo y exposición de sus videojuegos, las empresas *indies* se han apoyado de medios de financimiento denominados *crowdfunding*. El *crowdfunding* es un método de financiamiento de proyectos conformado pr la inversión de pequeñas cantidades de dinero por parte de un gran número de inversionistas. Existen 4 distintos modelos de financiamiento [3]: 1) basado en donaciones, 2) basados en recompensas, 3) basados en equidad y 4) basados en deudas.

Kickstarter es una plataforma perteneciente al modelo basado en recompensas, en la cual los contribuyentes reciben compensaciones o privilegios relacionados al proyecto en base a la cantidad de dinero invertida. Fue creada el 28 de abril de 2009 por Perry Chen, Yancey Strickler y Charles Adler, bajo la premisa de *ayudar a dar vida a proyectos creativos*. Actualmente, la plataforma aloja inversionistas quienes se presentan como usuarios registrados los cuales pueden consultar proyectos y hacer donaciones que parten desde 1 dólar. Los proyectos que se desean financiar se presentan en forma de *campañas* de financiamiento que conformadas por publicaciones acerca del proyecto en desarrollo o de una idea próxima a desarrollar. Según su estado de financiamiento, a las campañas se les asigna uno de los cinco posibles estados: en vivo, suspendido, cancelado, fallido o exitoso.

El hecho de que Kickstarter sea preferido por los desarrolladores de videojuegos independientes se debe a que hay alrededor de 15 millones de contribuyentes registrados en la plataforma. Asimismo, a partir de esta plataforma se han financiado videojuegos que tuvieron altos índices de ventas al momento de su lanzamiento, como: Shovel Knight (311 mil dólares recaudados de una meta de 75 mil) y Divinity: Original Sin (944 mil dólares de una meta de 400 mil) [5]. A pesar del evidente apoyo hacia proyectos de videojuegos, estadísticas disponibles en Kickstarter.com muestran que se han financiado exitosamente 150,000 de 418,000 campañas pertenecientes a todas las categorías [3]. Hasta el año 2017, del total de las campañas, 13,619 pertenecen a la categoría de videojuegos, siendo exitosas 2,539 (alrededor de un 18 %) [3]. Esto supone un problema para la financiación de proyectos de videojuegos, ya que, de no alcanzarse la meta establecida, la plataforma emplea la política de devolución del dinero recaudado a sus contribuyentes. Por ello, el objetivo principal de este artículo es predecir el éxito y/o fracaso, a partir de datos específicos obtenidos de las campañas de videojuegos y otras fuentes de información, considerando únicamente atributos tales como: meta de la campaña, gráficos del videojuego, título de la campaña. números de comentarios, actualizaciones realizadas, entre otros. El conjunto de datos conformado permitirá tener una descripción completa de la situación más apegada a la realidad de las campañas.

El artículo se estructura de la manera siguiente: Sección 2 presenta una breve revisión de los trabajos relacionados con la predicción del éxito en proyectos de Kickstarter., En la Sección 3 se describe el proceso de recolección, integración y preprocesamiento de los datos. Sección 4 se define el diseño experimental adoptado en este trabajo. Posteriormente, en la Sección 5 se muestran y discuten los resultados. Finalmente, en la Sección 6 se encuentran las conclusiones y el trabajo futuro.

2. Trabajos relacionados

En la literatura se pueden encontrar diversos trabajos orientados a predecir o determinar los factores que influyen en el fracaso o éxito de las campañas. Por ejemplo, Antonio Uribe, fundador de la empresa de desarrollo de videojuegos Hyper Beard, en su artículo sobre la situación del "Desarrollo de Videojuegos en México", expone que existe un apoyo mínimo por parte de la comunidad mexicana hacia los productos de sus compatriotas [6], debido a la preferencia del público en adquirir videojuegos desarrollados en Estados Unidos, sugiriendo que un posible factor de éxito es la preferencia de videojuegos desarrollados en ciertos países. Asimismo, menciona que otro posible factor que influye negativamente es la cantidad monetaria que se establece como meta de la campaña. Ello puede observarse en las campañas de Mulaka (videojuego sobre la cosmogonía tarahumara), que no logró su meta de 77,000 dólares [7], y Neon City Riders, videojuego mexicano, que cumplió su meta de 8,335 dólares [8].

Otra forma de abordar el problema de determinar los factores que han sido determinantes en el éxito, es mediante el uso de técnicas de minería de

datos. Ejemplo de ello, es el trabajo de Michal Trněný [9], en el cual se analizó la información de 4,634 juegos con la finalidad de identificar los factores que predicen el éxito de un videojuego publicado, definiendo el éxito como el número de ventas tales que cubran los costos de desarrollo del producto. La base de datos empleada se obtuvo mediante las API públicas de Steam, considerando para ello sólo juegos creados por compañías AAA (Por ejemplo, Ubisoft). Los atributos que se tomaron en cuenta fueron el género del videjuego, el precio, el desarrollador, así como el alcance en Internet. Para el análisis utilizó máquinas de soporte vectorial, árboles de decisión y redes bayesianas. Trněný [9] concluyó que existe una correlación entre el éxito y la experiencia del desarrollador, número de jugadores concurrentes en los primeros dos meses y los atributos descriptivos.

Por su parte, Li et al.[10] emplearon diversos modelos probabilísticos para la predicción de la posible fecha de éxito de una campaña de Kickstarter mediante un análisis de 18,093 proyectos registrados en la plataforma y que hayan alcanzado su meta de crowdfunding. Los datos fueron obtenidos de Kickspy, página que se dedica a recopilar datos de campañas de la mencionada plataforma, así como utilizar las API de Twitter y Facebook para complementar su colección de datos. Consideraron métricas de la plataforma (tiempo de inicio, tiempo de término, meta, total recaudado) complementadas con estadísticas de publicidad en Twitter y Facebook. Posteriormente, obtuvieron dos modelos, donde uno de ellos consideró únicamente los casos de éxito mientras que el otro consideró los datos en su totalidad. Concluyeron que las variables que determinan la fecha de éxito incluyen: la existencia de publicidad en redes sociales; y la fecha de inicio de la campaña. Asimismo, reconocieron que el modelo que contempla casos de éxito y fracaso presenta un error de predicción menor al 5 %.

Kindler et al. [11], a partir de un ánalisis realizado a una base de datos con datos del 2013, sugieren que es posible predecir en una etapa temprana el éxito de un campaña en Kickstarter, en donde, la comunidad de contribuyentes influye en que se alcance la meta deseada por la campaña.

Los trabajos sobre la predicción del éxito de campañas de Kickstarter, han empleado como fuente de información todos los atributos que describen a las campañas. Sin embargo, esto puede implicar que se utilicen atributos que describen exclusivamente a otros tipos proyectos enfocados al cine, el arte, la moda, el diseño, el teatro, entre otros.

3. Conjunto de datos

El conjunto de datos utilizado en este artículo contiene información de las campañas de videojuegos comprendidas entre los años 2010 y 2017. El proceso de recolección fue realizado de manera manual y automática, donde en este último caso se desarrollaron scripts en PERL y Ruby utilizando las librerías kickscraper de Mark Olson [12] y WWW::Kickstarter del usuario de CPan Ikegami [13], respectivamente.

3.1. Obtención de los datos

Para la conformación del conjunto de datos, se tomó como referencia la base de datos proporcionada por Kaggle [14], la cual está constituida por 378 mil registros de campañas de Kickstarter creadas entre octubre del 2009 y junio del 2018. Las campañas contenidas en la base de datos pertenecen a las 15 categorías de campañas soportadas por la plataforma. Asimismo, contempla un total de 14 atributos para cada campaña. Para cumplir con el objetivo de sólo emplear datos relacionados a los videojuegos se eliminaron registros de campañas que no pertenecieran a las subcategorías *Video Games* y *Mobile Games* de la categoría principal *Games*. Asimismo, registros cuyo estado de campaña fueran *live*, *paused* o *canceled*. Finalmente, todos aquellos registros cuyo año de inicio y finalización estuvieran fuera del rango 2010 a 2017.

Con la finalidad de enriquecer el conjunto de datos inicial con otros atributos, se llevó a cabo un proceso de extracción automática de información sobre la plataforma Kickstarter. Para ello, se apoyó en script desarrollados en PERL y RUBY, con los cuales se pudieron agregar 12 nuevos atributos:

- *Staff pick*: Variable dicotómica con valores [true, false] que indica si una campaña fue marcada como destacada dentro de Kickstarter.
- *Rewards levels*: Número de niveles de recompensas definido por el autor de una campaña.
- *Rewards Min*: Aporte necesario para adquirir el nivel de recompensa de menor valor (en divisa indicada).
- *Rewards Max*: Aporte necesario para adquirir el nivel de recompensa de mayor valor (en divisa indicada).
- *Project count at 2018*: Número de campañas creadas por el autor de la campaña hasta junio de 2018.
- *Project first*: Variable dicotómica con valores [true, false] utilizada para indicar si la campaña fue la primera del usuario.
- *Project count at creation*: Número de campaña que representa la instancia de campaña de Kickstarter.
- *Blurb*: Descripción corta ubicada al inicio de una campaña.
- *Rewards physical*: Indica la existencia de recompensas físicas dentro de los niveles de recompensa definidos.
- *Updates count*: Número de actualizaciones realizadas sobre el contenido de la campaña por parte del autor.
- *Comments count*: Número de comentarios que publicaron los usuarios de Kickstarter y el autor de la campaña a lo largo del tiempo de vida de la campaña.
- *Has video*: Variable dicotómica con valores [true, false]. Indica si la campaña contiene un vídeo como portada. No aplican videos embebidos en el contenido de la campaña.

En una segunda etapa, de forma manual se exploraron otras fuentes de información ligadas a los proyectos tales como Facebook, YouTube y páginas Web.

Adicionalmente se extendió dicha exploración a la plataforma de Kickstarter para obtener aquellos datos que los script no pudieron extraer. De este proceso se obtuvieron 15 atributos adicionales que son:

- *Release console*: Variable dicotómica con valores [true, false], indica si el videojuego que se intenta financiar se encontrará disponible en consolas.
- *Release PC*: Indica si el juego estará disponible en computadora (mediante navegador web, archivo ejecutable o por tiendas virtuales).
- *Release Mobile*: Indica si el videojuego se encontrará disponible en dispositivos móviles con sistema operativo Android (incluye OUYA), iOS, Windows Phone.
- *Facebook*: Variable dicotómica con valores [true, false], indica si existió publicidad del videojuego y/o la campaña en dicha red social.
- *Twitter*: Variable dicotómica con valores [true, false], indica la existencia de publicidad del videojuego y/o la campaña en dicha red social.
- *YouTube*: Variable dicotómica con valores [true, false]. Muestra si existió publicidad del videojuego y/o la campaña en dicha red social.
- *Social landing page*: Variable dicotómica con valores [true, false]. Indica si existió un sitio web del grupo desarrollador que detalle información sobre el videojuego.
- *Press kit*: Variable dicotómica con valores [true, false]. Indica si existió un kit de prensa para el videojuego de una campaña alojado en un sitio web o como un conjunto de archivos descargables.
- *Gameplay*: Variable dicotómica con valores [true, false]. Si la campaña contiene imágenes o videos de las mecánicas del videojuego. Incluye demos y prototipos del juego.
- *Game art style*: Variable categórica que asume los valores [3D, 2D, ND] según el tipo de gráficos del videojuego que se desea financiar.
- *Existing remake*: Variable dicotómica con valores [true, false]. Indica si el juego que se intenta financiar es una versión renovada de otro existente, de la cual no se tiene la licencia o es la modificación.
- *Campaing Target Audience*: Variable categórica que toma uno de los valores [Everyone, Adults, Kids] acorde al tipo de público para el cual se dedican la campaña y el videojuego.
- *Game VR compatibility*: Variable dicotómica con valores [true, false] utilizada para indicar si el juego es compatible con realidad virtual (VR).
- *Multiplayer*: Variable dicotómica con valores [true, false], indicadora de si el videojuego permite cualquier tipo de interacción con personas que dispongan del mismo juego.
- *Backed Projects*: Número de proyectos que el autor de la campaña ha apoyado.

Dado que el objetivo es predecir a priori el éxito o fracaso de una campaña de videojuego, se eliminaron los atributos *usd pleadge real* y *Backers*. Estos dos atributos pertenecen al primer conjunto de datos que se obtuvo de Kaggle. El primero de ellos describe la cantidad monetaria recaudada a lo largo del tiempo de vida de la campaña. El segundo, representa el número de contribuyentes que apoyaron a la campaña.

3.2. Preprocesamiento de los datos

Una vez integrada la información adicional a la base de datos inicial, se codificaron algunos atributos. Siguiendo los hallazgos de la investigación de Downs y Ghauri (investigación en la que demuestra la importancia de la longitud del nombre de una campaña) [15], se crearon nuevos atributos numéricos enfocados a representar el nombre de la campaña (*name*) y la descripción de la campaña (*Blurb*) en términos de la longitud de caracteres (*name chars*, *Blurb chars*) y del número de palabras (*name words*, *Blurb words*).

En el caso de los atributos de tipo fecha, se optó por crear 3 nuevos atributos que representan el mes, el día y el día de la semana. En el caso de los atributos de tipo moneda, debido a que los proyectos utilizan diferentes tipos de cambio, se decidió utilizar el dólar como la moneda estándar. Por lo que se realizó la conversión a dicha moneda, de acuerdo al tipo de cambio que existía en la fecha en la que se estableció la meta de la campaña.

De los 14 atributos originales, se eliminaron atributos que se consideraron irrelevantes para el objetivo de estudio como son: *category*, *main category* y *PID*. El primer y segundo atributo indican la categoría principal y subcategoría a la que pertenece la campaña. Sólo fueron útiles para extraer los datos de las campañas pertenecientes a videojuegos. Finalmente, en el caso de *PID*, es un identificador numérico de la campaña.

El conjunto final está conformado por 40 atributos (uno de ellos es la etiqueta de clase), de los cuales 12 son cuantitativos (numéricos) y el resto nominales o categóricos. Del número de instancias, un total de 1,079 son de la clase éxito y 3,043 de la clase fracaso.

3.3. Variables ficticias

Los modelos de regresión tratan todos los atributos (variables dependientes) como números. Por lo que, los atributos categóricos o nominales (no numéricos) fueron transformados a números con el fin de poder emplear el algoritmo de regresión. Por ejemplo, el atributo *Game art style* que puede tener uno de los tres valores 2D, 3D o ND (no especifica el tipo de gráfico), se utilizaron los números 1, 2 y 3 para representar cada una de los tipos de gráficos, respectivamente. No obstante, asignar un valor numérico a una variable categórica no representa que el 3 sea mayor que 2 ó que se puedan restar/sumar. Para solventar este problema, cada uno de los atributos categóricos fueron codificados a variables ficticias o *dummy*, la cual es un atributo artificial que toma un valor de 0 o 1, para indicar la ausencia y presencia del valor, respectivamente [16]. Si un atributo tiene n valores diferentes, entonces se crearán $n-1$ nuevos atributos ficticios [17]. En la Tabla 1 se muestra un ejemplo para la codificación del atributo *Game art style*.

4. Diseño experimental

El objetivo de este artículo es predecir y determinar los factores de éxito en proyectos de videojuego en Kickstarter.

Tabla 1. Creación de variables ficticias para *Game art style*.

Game art style	Nueva Variable 1	Nueva Variable 2
1 (2D)	1	0
2 (3D)	0	1
3 (ND)	0	0

Para ello, los experimentos se realizaron sobre el conjunto de datos conformado por diversas fuentes de información. La Fig. 1 muestra el diagrama de flujo del proceso empleado para la creación del modelo de predicción del éxito/fracaso de las campañas de videojuegos. Los primeros dos pasos incluyeron las actividades de preparación de la base de datos inicial; y la eliminación, transformación y creación de variables. Posteriormente, se realizaron actividades de construcción y evaluación del modelo, en las cuales se aplicó una validación cruzada de 5 particiones en su versión estratificada, por lo que se preservaron las probabilidades a priori y la independencia estadística entre las particiones de entrenamiento y test.

En el aprendizaje automático, uno de los métodos de clasificación más utilizados en problemas de dos clases es la regresión logística [18,19,20], el cual es modelo clásico que proviene del área de la estadística [19]. En este trabajo el modelo fue tomado de la librería de R denominada caTools [3].

4.1. Métricas de evaluación

La evaluación de un clasificador se suele basar en un tabla de confusión, la cual es una matriz, cuyas entradas (i, j) contienen el número las predicciones correctas/incorrectas [21]. La Tabla 2 muestra una matriz de 2×2 para un problema de dos clases, donde las columnas representan la salida estimada por parte del clasificador y las filas indican las clases reales. Los elementos en la diagonal principal contienen el número correcto de predicciones en la clase positiva y negativa, mientras que el resto de las entradas son los errores.

Tabla 2. Matriz de confusión para un problema de dos clases.

	Predicción Positivos	*Predicción Negativos*
Real Positivos	Verdaderos Positivos (TP)	Falsos Negativos (FN)
Real Negativos	Falsos Positivos (FP)	Verdaderos Negativos (TN)

Por lo general, la métrica empleada para evaluar la efectividad de un clasificador es la exactitud ($A = (TP + TN)/(TP + FN + TN + FP)$).

Sin embargo, se ha demostrado que dicha métrica no es apropiada cuando el conjunto de datos no está balanceado, por lo que, se suelen utilizar otras

[3] https://cran.r-project.org/web/packages/caTools/index.html

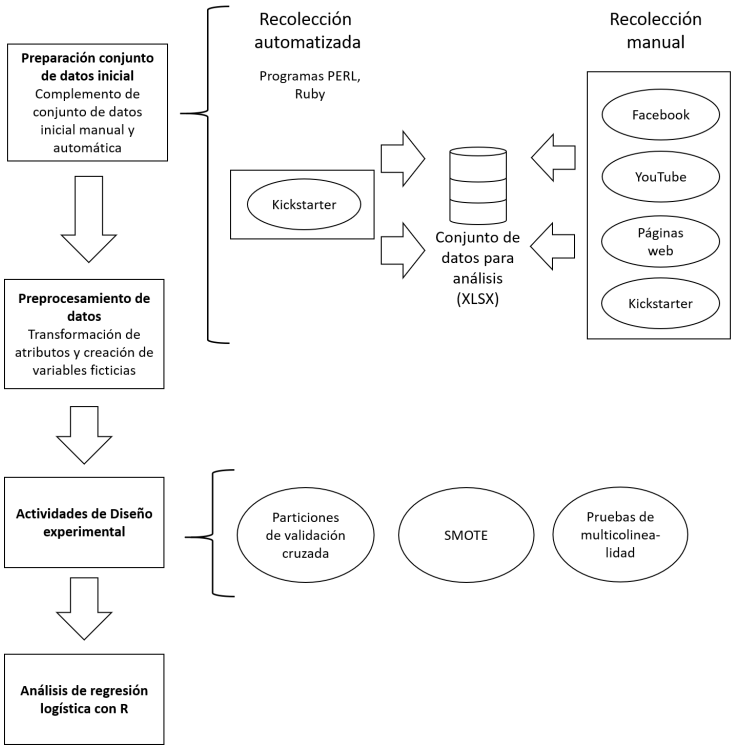

Fig. 1. Diagrama de flujo del proceso experimental.

alternativas [21]. Un ejemplo de esto son las métricas que evaluan la efectividad en cada clase como la tasa de verdaderos positivos ($TPr = TP/(TP + FN)$) y la tasa de verdaderos negativos ($TNr = TN/(TN + FP)$). Una tercera métrica que engloban a las dos anteriores es la media geométrica, la cual, maximiza la efectividad individual de cada clase mientras se mantienen balanceadas ($GM = \sqrt{TPr \times TNr}$) [22]. En este trabajo utilizaremos estas tres últimas métricas.

4.2. Clases no balanceadas

Uno de los mayores obstáculos en la construcción de modelos de aprendizaje automático, es el problema denominado de clases no balanceadas. Un conjunto de dos clases se dice que es no balanceado, si una de las clases (mayoritaria) tiene mayor presencia que la otra (minoritaria). Este tipo de complejidad de los datos puede conducir, en métodos tradicionales de clasificación, a aprendizajes sesgados en perjuicio de la clase minoritaria que, usualmente, contiene los casos de mayor interés [23]. En este trabajo la clase de mayor interés es la denominada como éxito y cuyo número de registros es de 1,079 registros, mientras que en el caso de la clase fracaso los ejemplos son 4,122. Esto supone un ratio de clases no balanceadas ($negativas/positivas$) igual a 3.82.

En la literatura existen diversas estrategias para tratar con el problema. Una de las más populares es aquella que incrementa el tamaño de la clase minoritaria ya sea creando de forma aleatoria copias de las instancias minoritarias o nuevas instancias artificiales a partir de heurísticas. Se ha demostrado que crear copias de los instancias puede conducir a un sobreajuste del modelo. Para evitar este problema algunos métodos crean ejemplos sintéticos, como es el caso de la técnica SMOTE (Synthetic Minority Oversampling TEchnique), la cual genera nuevos datos mediante la interpolación de instancias existentes de la misma clase [24]. Para ello, primero se obtienen los K vecinos más cercanos de la clase minoritaria. Posteriormente, las nuevas instancias son generadas en la dirección de algunos de estos vecinos. La Fig. 2 ejemplifica la interpolación de instancias con SMOTE.

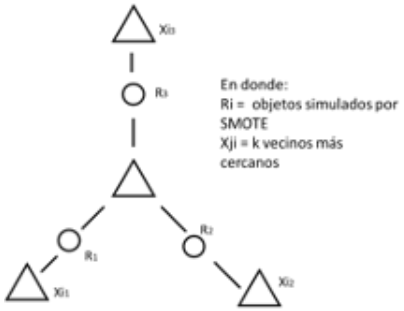

Fig. 2. Interpolación de valores con SMOTE.

En este trabajo se utilizó SMOTE hasta lograr que las clases estuvieran balanceadas, con valor de $K = 8$. Para evitar resultados resultados sobre ajustados o demasiado optimistas, el sobremuestreo de la clase minoritaria se realizó en el conjunto de entrenamiento en cada iteración de la validación cruzada [25].

5. Resultados y discusión

La Tabla 3 muestra los promedios de clasificación con la regresión logística, utilizando el conjunto original (sin los atributos *usd pledge real* y *backer*). Como se puede observar, debido a la problemática de las clases no balanceadas, la exactitud en la clase menos representada (clase éxito) presenta una tasa de 0.7465, mientras que el caso de la clase mayoritaria (clase fracaso) el resultado está por arriba de 0.94. A pesar de esto el resultado de la media geométrica alcanza el 0.8503. Es importante mencionar que, en este conjunto no se consideraron aquellos atributos que puende influir en el éxito de la compaña.

Al llevar a cabo los experimentos sobre el conjunto sobremuestreado por medio de SMOTE, se observa, que existe un incremento en la exactitud de la clase minoritaria (TPr), pasando de 0.7645 a 0.8712. Asimismo, se observe un ligero decremento en la exactitud en la clase mayoritaria .

Tabla 3. Resultados promedios de TPr, TNr y GM usando el conjunto no balanceado.

	TPr	TNr	GM
Original	0.7645	0.9457	0.8503

Con respecto al resultado de la media geométrica, el valor se incrementa hasta llegar a 0.8712 (vea Tabla 4).

Tabla 4. Resultados promedios de TPr, TNr y GM usando el conjunto balanceado con SMOTE para K=8.

	TPr	TNr	GM
Balanceado	0.8378	0.9059	0.8712

Una de las condiciones para usar la regresión logística es la ausencia de colinealidad entre las variables independientes (atributos de entrada) [26]. Por consiguiente, se aplicó una prueba de multicolineadad para detectar aquellos atributos que presenten dicha característica. En total se detectaron los siguientes atributos: *currency*, *deadline weekday*, *launched weekday*, *country*, *game art style*, *campaign target audience*. Existen diferentes formas de resolver la problemática de la multicolinealidad: eliminar las variables, cambiar la escala o combinarlas en una medida única [26]. En este trabajo por sencillez se optó por eliminar los atributos.

En la Tabla 5, se muestran los resultados de clasificación en términos de TPr, TNr y GM. Como se puede observar, el eliminar el problema de la multicolinealidad, los resultados de exactitud en la clase minoritaria se incrementan pasando de 0.8378 (resultados con el conjunto balanceado) a 0.8646. Esto ocasiona que también se de un incremento en la media geométrica hasta llegar a 0.8870

Tabla 5. Resultados promedios de TPr, TNr y GM usando el conjunto balanceado con SMOTE para K=8 y sin colinealidad.

	TPr	TNr	GM
Balanceado S/Col.	0.8646	0.9099	0.8870

6. Conclusiones y trabajo a futuro

El presente trabajo tuvo como objetivo el construir un modelo de regresión logística, para predecir el éxito/fracaso de campañas de videojuegos en Kickstarter. Con este fin, se construyó un conjunto de datos a partir de diversas fuentes tales como YouTube, Facebook, páginas web.

De esta manera, se pudo integrar diversos atributos que proveen información sobre los comentarios, el tipo de videojuego, las redes sociales, niveles de recompensa, entre otros. Posteriormente, se aplicaron diversos preprocesamientos a los datos, tales como convertir de datos nominales a numéricos, crear variables artificiales, y eliminar datos con multicolinealidad. Asimismo, debido a la presentación de clases no balanceadas se sobremuestreo el conjunto para tratar dicha problemática.

Los resultados obtenidos, en términos de la media geométrica, sugieren que la construcción de la regresión logística con la base de datos preprocesada obtiene los mejores resultados de clasificación, con valores cercanos al 0.8870 en términos de la media geométrica. Teniendo, en cuenta este resultado se puede concluir que es posible construir un modelo que sea capaz de predecir el éxito/fracaso de un proyecto de videojuego considerando una base de datos con atributos diversos. Como trabajo futuro, planteamos el uso de otros clasificadores com árboles de decisón, basados en regla, dada la naturaleza del conjunto de datos de tener una mezcla de atributos. Asimismo, aplicar técnicas de selección de características y de balanceo de clases.

Referencias

1. Wijman, T.: NewZoo, https://newzoo.com/insights/articles/global-games-market-reaches-137-9-billion-in-2018-mobile-games-take-half (2018)
2. Alonso, A.: HobbyConsolas, https://www.hobbyconsolas.com/noticias/juegos-aaa-vs-juegos-indie-cara-cara-65552 (2018)
3. Reddy, S., Heng Tan, Y.: Crowfounding: Financing Ventures in the Digitial Era. Marketing Intelligence Review 9 (1), 37–41 (2017)
4. Kickstarter Homepage, https://www.kickstarter.com/about (2018)
5. Digital Trends, https://www.digitaltrends.com/gaming/best-kickstarter-funded-video-games/ (2018)
6. Uribe, A.: Medium Corporation, https://medium.com/@Fayer/nos-hace-falta-apoyar-m %C3 %A1s-a-los-h %C3 %A9roes-locales-3eef1163a613 (2018)
7. Kickstarter PBC: Mulaka Origin, https://www.kickstarter.com/projects/lienzo/mulaka-origin-tribes (2018)
8. Kickstarter PBC: Neon City Riders: A Cyberpunk Turf Wars Action Adventure, https://www.kickstarter.com/projects/mechastudios/neon-city-riders-a-cyberpunk-turf-wars-action-adve (2018)
9. Trneny, M.: Machine Learning for Predicting Success of Video Games. Masaryk University, Faculty of Informatics (2017)
10. Li, Y., Rakesh, V., Reddy, C.: Project Success Prediction in Crowdfunding. En: 9th ACM International Conference on Web Search and Data Mining, pp. 247–256. ACM, San Francisco, California (2016)
11. Kindler, A., Golosovsky, M., Solomon, S.: Early prediction of the outcome of Kickstarter Campaigns: is the sucess due to virality. Palgrave Communications, pp. 1–19 (2019)
12. Olson, M.: API Library for Kickstarter.com, https://github.com/markolson/kickscraper (2018)
13. Ikegami: WWW-Kickstarter-v1.12.0, https://metacpan.org/release/WWW-Kickstarter (2018)

14. Mouille, M.: Kickstarter Projects Dataset, https://www.kaggle.com/kemical/kickstarter-projects (2018)
15. Downs, R., Ghauri, M.: Predicting Kickstarter Campaign Success with Classification Models (2018)
16. Fox, J.: Applied Regression Analysis and Generalized Linear Models. SAGE Publications, London (2015)
17. The Odum Institute: Learn about multiple regression with dummy variables in SPSS with data from the general social survey (2012). SAGE Publications, United Kingdom (2015)
18. Esposito, D., Esposito, F.: Introducing Machine Learning. Microsoft Press (2020)
19. Mohri, M., Rostamizadeh, A., Talwalkar, A.: Foundations of Machine Learning. The MIT Press (2018)
20. Zaki, M. J., Meira, W.: Data Mining and Machine Learning: Fundamental Concepts and Algorithms. Cambridge University Press, United Kingdom (2020)
21. Jadhav, A. S.: A novel weighted TPR-TNR measure to assess performance of the classifiers. Expert Systems with Applications 152, 1–27, (2020)
22. Kuncheva, L. I., Arnaiz-González, Á., Díez-Pastor, J., Gunn, I. A.D.: Instance selection improves geometric mean accuracy: a study on imbalanced data classification. Progress in Artificial Intelligence 8, pp. 215–228 (2019)
23. Fotouhi, S., Shahrokh, A., Kattan,M. W.: A comprehensive data level analysis for cancer diagnosis on imbalanced data. Journal of Biomedical Informatics 90, pp. 1–30 (2019)
24. Fernandez, A., Garcia, S., Herrera, A., Chawla, N. V.: SMOTE for Learning from Imbalanced: Progress and Challenges, Making the 15 year Anniversary. Journal of Artificial Intelligence 61, pp. 863–905 (2018)
25. Santos, M. S., Soares, J. P., Abreu, P. H., Araujo, H., Santos, J.: Cross-validation for imbalanced datasets: Avoiding Overoptimistic and Overfitting Approaches. IEEE Computational Intelligence Magazine 13(4), pp. 59–76 (2018)
26. Lopez-Roldan, P., Fachelli, S.: Metodología de la investigación social cuantitativa. Universitat Autonoma de Barcelona (2015)