Índice

1. Definición del problema

En este capítulo se abordará de forma general el contexto del estudio realizado. En primera instancia se tratará el tema de los difusores de contenido audiovisual a través de internet, y cómo este concepto se ve relacionado con **Kikvi**, empresa de la cual se extrajeron los datos para los posteriores capítulos. En segundo lugar, se abordará el problema actual por el que pasa la empresa mencionada, y finalmente los objetivos, principales y secundarios, de este estudio. Se espera durante este capítulo dar un marco general del estudio e informar sobre los motivadores que llevaron a realizarlo.

1.1. Difusores de contenido audiovisual

Desde la masificación de las redes sociales, en el pasado con plataformas como MySpace, y hoy con plataformas muy masificadas como Facebook o Instagram, se ha buscado la forma de sacar provecho de esta interacción. Una gran cantidad de negocios se ha formado en torno a este concepto, principalmente relacionados con la publicidad. De la misma manera, la publicidad se ha ido adaptando a este nuevo entorno, haciéndose cada vez más sutil en algunos casos (*product placement*[1]), y aún más invasiva en otros, como es el caso de portales de contenido viral con *banners* y *popups*[2]. Con el paso del tiempo, los usuarios de internet se han vuelto reacios y menos susceptibles al segundo tipo de publicidad,

[1] Conocido en español como "publicidad por emplazamiento", consiste en la inserción de un producto, marca o mensaje dentro de la narrativa de un programa, en este caso en particular, fotografías o videos compartidos en redes sociales. Es un tipo de publicidad sutil indirecta. [Fuente: Wikipedia.org]

[2] Un banner es un formato publicitario característico de internet. Consiste en incluir una pieza publicitaria dentro de un sitio web, cuyo objetivo es atraer tráfico al vínculo correspondiente al banner. Un *popup* es una ventana emergente dentro de un sitio web, que por lo general cuenta con un *banner*, formulario o alguna forma de captura de información dentro de él. Su función final suele ser la misma que el *banner* pero de una forma más agresiva. – [Fuente: Wikipedia.org]

ocurriendo fenómenos como la llamada **ceguera del banner**[3], lo que hace necesaria una forma de publicidad más sutil, o más atractiva, que un *banner* tradicional. Es en torno a esto que se generan los difusores de contenidos.

Un difusor de contenido, como dice su nombre, cumple la función de **propagar y difundir contenido especializado a través de redes sociales**. La mayoría de los difusores aún enfocan su modelo de negocios en *banners*, usando la difusión en redes sociales de contenidos altamente atractivos para traer finalmente visitas a sus portales web. Kikvi es un difusor de contenidos audiovisuales, pero no enfoca su esfuerzo en *banners*; en él se explora la segunda posibilidad comentada de publicidad en internet, la publicidad sutil. A continuación, se revisará la historia y funcionamiento en detalle de Kikvi.

1.2. Kikvi

Kikvi nace como un proyecto de un grupo de estudiantes en la Feria de Creación de Software de la Universidad Técnica Federico Santa María en el año 2012. En aquel entonces, bajo otro nombre (privado), la idea del producto consistía en una red de usuarios dividida en 2 grupos principales: **creadores** y **publicadores**. El primer grupo estaba compuesto por personas generadoras de contenido audiovisual, que habían invertido tiempo y dinero en esto, y necesitaban rentabilizarlo. Su desafío se presentaba al momento de difundir dicho contenido, tomando como vía principal las redes sociales, pero sin contar con el "*peso"* suficiente para llegar a una masa suficientemente atractiva de público objetivo. Es aquí donde entraría en participación el segundo grupo usuario de la plataforma, los **publicadores**. Este conjunto estaba compuesto por una serie de personas

[3] Este fenómeno se hace presente con la masificación de los *banners*. El primer *banner* recordado de la historia de internet, tuvo una conversión del 44%, esto quiere decir que de cada 100 personas que visitaron el sitio del banner, 44 hicieron *click* sobre él. Hoy, la conversión, en el mejor de los casos, llega a 2%. [Fuentes: Red de *Display* de Google]

experimentadas en el uso de redes sociales y con alto índice de influencia en sus círculos. La función de los publicadores consistía en compartir los contenidos generados por los creadores en sus respectivas redes sociales, siendo dinero la motivación. En el ideal del modelo de negocios, el grupo de **creadores** compraría una cierta cantidad de vistas de "calidad", ya que se seleccionaba a los usuarios **publicadores** a través de un proceso de filtrado, pagando una cantidad de dinero fijo por vista. Esta cantidad se dividiría entre los **publicadores** que participaran de la campaña (consiguiendo parte de ese dinero por cada vista que consiguieran) y la empresa.

El concepto tuvo un éxito relativo en su fase inicial, consiguiendo financiamiento a través de la incubadora **3IE** de la UTFSM.

Poco tiempo después de este hito, hubo diferencias de opiniones entre los fundadores del proyecto, lo que resultó en la separación de los mismos. Parte de los involucrados siguieron con el proyecto inicial y el resto emprendió nuevos caminos, agregando nuevos integrantes al equipo y formando **Playgue**, plataforma que tenía la misma idea mencionada anteriormente. Luego de poco tiempo de funcionamiento, se hicieron claras las falencias del modelo de negocios, respaldándose además en el bajo éxito del proyecto seguido por el otro grupo de socios originales:

- La idea de "ganar dinero" en internet por poco esfuerzo fue una idea que se explotó mucho en el pasado, lo que hoy genera una fuerte desconfianza por parte del usuario.

- Los clientes (creadores) eran muy reacios sobre el origen de las vistas y su legitimidad. Esto se veía potenciado por servicios de países asiáticos que ofrecían una gran cantidad de vistas a muy bajo costo.

- El grupo de publicadores no tardó en encontrar la forma de optimizar su tiempo y sistema, generando grupos de “ayuda” en Facebook donde, entre ellos mismos, cada uno veía repetidas veces los videos publicados por el resto, generando ganancias para todos (menos para el objetivo real del negocio y los creadores).

A estas alturas se tomó la decisión de modificar la manera en la que se estaba abordando el negocio. El concepto de “dinero” en internet producía rechazo, por lo que se adoptó una metodología de puntos. Además, se decide cambiar la dinámica del contenido del sitio, complementando los videos de marcas y clientes con otros altamente atractivos, pero cuya recompensa de puntos era considerablemente menor a la obtenida por los “auspiciados”. Es junto con estos cambios que se hace un fuerte trabajo de diseño de interfaces y la plataforma toma su nombre actual: **Kikvi**.

Los puntos obtenidos a través del portal podían utilizarse para canjear sobre un catálogo de productos, partiendo de cosas simples como entradas dobles al cine, y llegando hasta productos de alto valor como consolas de videojuegos y cámaras para deportes extremos. Este acercamiento provocó gran revuelo y consiguió la participación de muchos usuarios. El desafío consistía en mantener un catálogo de productos constante sin gastar más dinero del que ingresaba en la empresa.

Fue al poco tiempo después que se decidió incluir concursos en la plataforma, lo que en teoría solucionaría dos aristas en las que se estaba teniendo problemas:

- Los canjes solían ser por una gran cantidad de puntos, lo que desmotivaba fuertemente a los usuarios.

- Los canjes significaban una gran inversión de dinero (para ser atractivos).

La inclusión de concursos al sistema significó un seguimiento mucho más cercano de los usuarios a la plataforma, interactuando de forma activa por períodos de tiempo (o al menos esto se creía). Al poco tiempo los concursos habían tomado gran fuerza en la plataforma, desplazando a los canjes inmediatos.

Surge a estas alturas la necesidad de entender de mejor manera el negocio, los usuarios y los procesos de la plataforma. Hasta el momento se estaba avanzando a ciegas: funcionando en base a prueba y error. El hecho de tener información de los procesos y funcionamiento se vuelve una herramienta atractiva y poderosa, y se toma la decisión de explotarla.

1.3. El problema

Kikvi funcionó durante largo tiempo a ciegas, sin mucho conocimiento de un mercado muy poco explotado y sin respaldos ni casos de éxito cercanos para seguir. Entender el negocio, sus procesos y sus usuarios se convierte en un foco de atención, para poder mejorar la experiencia y percepción general sobre el producto.

Con el avance del tiempo se hace insostenible mantener una metodología de prueba y error, y es necesario tomar los pasos precisos en la dirección correcta. Además, es importante poder definir qué es lo que se considerará un caso de éxito en la plataforma, tanto en relación a un usuario como a un video en particular, para de esta forma poder potenciar y emular este tipo de comportamientos.

Kikvi comienza desde los cimientos sin financiamiento, lo que limita las posibilidades de contar con personal especializado para áreas como marketing o análisis de datos. Esto lleva a que la plataforma funcione de acuerdo a estipulaciones e hipótesis, sin tener claro si el camino emprendido o la forma de abordar el

problema que se pretende resolver con Kikvi, mejorando la difusión de campañas audiovisuales a través de redes sociales, sean los correctos.

En el escenario de hoy, la empresa se ve limitada al momento de comunicarse y trabajar con nuevos clientes; hay diversas interrogantes que son recurrentes en torno a esto, como por ejemplo:

- **¿Qué define un caso de éxito?**: esta pregunta es, de forma implícita, recurrente al momento de comunicarse con nuevos clientes. El cliente quiere saber de casos de éxito anteriores, quiere saber si hay algún referente en la plataforma, algo que indique que su inversión va a dar frutos. Es entonces que surge la siguiente pregunta "*¿Podríamos ver casos de éxito?*", que no es posible responder si no se tiene una concepción de lo que define un caso de éxito dentro de la plataforma.

- **¿Cuántos usuarios hay?**: otra pregunta recurrente hace referencia a la cantidad de usuarios registrados en la plataforma. Aparentemente, su respuesta es simple, pues una consulta a la base de datos puede responderla sin dificultad. Si bien el número de usuarios se puede saber con total certeza, este número no es útil para el objetivo; lo que realmente importa es la cantidad de usuarios activos, o sea los usuarios que efectivamente se encuentran interactuando con la plataforma en un intervalo de tiempo. De nada sirve tener un sitio con cientos de miles de usuarios registrados, si sólo un porcentaje mínimo de ellos efectivamente es activo en la plataforma. Se hace necesario entonces saber reconocer un usuario activo, para así poder estudiar las variables del ambiente, y del usuario mismo, que lo hacen entrar en esta categoría.

Estas dos preguntas, totalmente válidas para un cliente que espera saber si vale la pena invertir o no parte de su presupuesto de marketing en **Kikvi**, actualmente se

encuentran sin respuesta. Es necesario entonces poder entender cómo se desenvuelven los usuarios en la plataforma, cómo interactúan con los videos, cuáles son las variables que hacen que un video sea exitoso, etc. El éxito de una campaña de un potencial cliente se ver estrictamente restringido por la respuesta de los usuarios de la plataforma ante ella, entonces ¿cómo puede **Kikvi** apoyar esta campaña?

Es preciso que la plataforma se desarrolle de tal manera que optimice estos aspectos, que llame al usuario a mantenerse activo e interesado en las campañas existentes. Se deben descubrir los motivadores correctos y los escenarios ideales para obtener la mejor respuesta posible de la comunidad usuaria frente a las campañas de clientes. Además, es indispensable entender qué es lo que define y, aún más importante, cómo conseguir usuarios comprometidos con la plataforma, para así convertirla en una opción atractiva de inversión al momento de evaluar opciones de marketing digital.

1.4. Objetivos

1.4.1. Objetivo principal

Mejorar la percepción y experiencia usuaria de Kikvi para incrementar el éxito y penetración de campañas de clientes en un difusor de contenidos audiovisuales (Kikvi).

1.4.2. Objetivos específicos

Para poder lograr el objetivo principal propuesto es necesario, en primera instancia, realizar una serie de pasos relacionados con los datos:

- Hacer distinción de casos de éxito dentro de la plataforma, para así poder analizarlos y replicarlos, aumentando por un lado la satisfacción real del cliente, y por el otro la percepción del usuario.

- Descubrir qué indicadores son de interés para videos, usuarios y la plataforma en general, con el fin de tener una percepción de dónde enfocar esfuerzos y recursos para afectar de manera positiva al negocio. De la misma manera, ver cómo afectan estos indicadores a la percepción usuaria, para así poder mejorarla.

- Mejorar el porcentaje de rebote de usuarios en la plataforma; esto quiere decir que se pretende que los usuarios (visitas) no vengan con un objetivo específico a la plataforma y se vayan, si no que se distraigan, interactúen y exploren Kikvi. De esta manera, una visita no empieza y termina con la vista de un video, si no que puede significar un apoyo a otras campañas y, aún mejor, una adquisición de un nuevo usuario.

2. Estado del arte

En este capítulo se revisarán conceptos y conocimientos específicos que darán contexto a las herramientas y procedimientos utilizados durante este estudio.

Se espera entonces informar sobre las herramientas actuales para abordar esta clase de problemas, ahondando en las que fueron utilizadas para llevar a cabo este estudio.

2.1. Minería de datos

De forma general, la minería de datos consiste en el proceso de analizar datos[4] de múltiples fuentes, desde diferentes perspectivas, con el fin de resumirla en información[5] útil, o sea, la que pueda ser utilizada para aumentar ganancias, disminuir costos, mejorar procesos, etc. Entonces, un software de minería de datos es una herramienta analítica para datos.

Las herramientas de minería de datos permiten a sus usuarios analizar datos recopilados desde muchas dimensiones o ángulos diferentes, resumiéndolo todo en una serie de relaciones identificadas entre las variables estudiadas. Por lo general, la minería de datos se utiliza para encontrar correlaciones o patrones entre docenas de variables, o para encajar en el contexto, campos de una gran base de datos relacional. A pesar de que la minería de datos es un término relativamente nuevo, la tecnología no lo es. Las compañías han utilizado por mucho tiempo computadores de alto rendimiento para iterar sobre grandes volúmenes de datos con el fin de generar reportes de interés para análisis durante años. Sobre estos escenarios, la innovación continua sobre herramientas computacionales como procesadores, discos de

[4] Hecho, número, o texto que puede ser procesado por un computador.

[5] Los patrones, asociaciones o relaciones entre datos pueden generar información. A diferencia de los datos, la información tiene uso, utilizad.

almacenamiento y software estadísticos, ha logrado incrementar dramáticamente la precisión de los análisis, mientras disminuyen los costos y tiempos de realizarlos.

De una forma muy simplificada, la minería de datos consiste en la identificación de patrones en conjuntos de datos, generalmente de grandes dimensiones, con el fin de adquirir algún conocimiento[6].

Los avances en métodos de captura de datos, procesamiento, transmisión de datos y almacenamiento, permiten hoy a las organizaciones integrar sus bases de datos en lo que se conoce como *data warehouses*. *Data warehousing* se define como el proceso de administrar y recuperar datos centralizados; representa la idea de mantener un repositorio central con todos los datos de una entidad. Esta práctica es necesaria para maximizar el acceso y posibilidades de análisis de los usuarios.

La minería de datos permite determinar relaciones tanto entre variables internas como externas de las compañías. Además, permite descubrir factores de retroalimentación como por ejemplo el impacto de una campaña en ventas, satisfacción de los clientes, etc. Con los resultados de un trabajo de minería de datos, un vendedor podría refinar el mercado objetivo de un producto para enfocar sus esfuerzos de campaña en esa dirección y lograr alta respuesta de clientes.

Un proyecto de minería está compuesto de 5 etapas principales:

- Extraer, transformar, y cargar datos en el *data warehouse (ETL[7])*.

- Almacenar y administrar los datos en un sistema de bases de datos multidimensional.

[6] La información puede ser transformada en conocimiento sobre patrones históricos o modas futuras.
[7] Por su sigla en inglés: *extract, transform, load.*

- Dar acceso a los datos a analistas del negocio y profesionales de TI.

- Analizar los datos con aplicaciones especializadas.

- Presentar la información en formatos útiles, como gráficos o tablas.

En relación a los niveles de análisis de la cuarta etapa mencionada, hay una serie de algoritmos y/o métodos utilizados comúnmente, como:

- Reglas de inducción: conjunto de reglas *if-then* útiles, basadas en significancia estadística.

- Árboles de decisión: estructuras en forma de árboles que representan una línea de diferentes resultados a través de una serie de decisiones. Estas decisiones generan reglas de clasificación para un conjunto de datos.

- Vecino más cercano: técnica que clasifica a cada registro en base a una combinación de sus *k* vecinos más cercanos[8].

- Redes neuronales artificiales: modelos predictivos no lineales que aprenden a través de entrenamiento.

- Algoritmos genéticos: técnicas evolutivas que usan procesos como combinaciones genéticas, mutaciones, y selección natural en un diseño basado en los conceptos de evolución natural.

[8] Su nombre en inglés es *k-nearest neightbours,* o por su sigla, *KNN*.

- Visualización de datos: interpretación visual de relaciones complejas en datos multidimensionales.

2.2. Procesos de minería de datos

En minería de datos hay variedad de procesos estándares para alcanzar los objetivos de la disciplina. A continuación, se revisarán los procesos más ampliamente utilizados.

2.2.1. Proceso de descubrimiento del conocimiento (KDD[9])

Recibe este nombre el proceso que tiene por entrada la base de datos y sus versiones modificadas, y tiene como salida el subconjunto de patrones que se transformarán en conocimiento, luego de la aplicación de minería de datos.

De acuerdo con *[1]*, *KDD* es el proceso de usar métodos de minería de datos para extraer lo que es considerado conocimiento de acuerdo a una serie de medidas y umbrales, usando bases de datos en conjunto con cualquier pre-procesamiento necesario, extracción de muestras o transformación. El proceso cuenta con 5 fases fundamentales: Selección, Pre procesamiento, Transformación, Minería de datos, Interpretación/Evaluación.

En la **ilustración 2.1** se aprecian los 5 pasos del KDD. Este es precedido por el desarrollo de un entendimiento del área de aplicación, cualquier conocimiento previo relevante y los objetivos del usuario final. Es un proceso iterativo e interactivo, e involucra numerosos pasos con muchas decisiones tomadas en el camino por el usuario.

[9] *Knowledge discovery in databases*

1) **Selección**: esta etapa consiste en definir un conjunto de datos, o enfocar los esfuerzos en una serie de variables de los mismos. En esta, es fundamental contar con un conocimiento previo del negocio, que ayudará a definir cuáles variables son relevantes para el estudio y cuáles no lo son. Por ejemplo, si se desea descubrir qué clientes son más susceptibles a un esfuerzo de marketing, casi con certeza el nombre del cliente no será una variable importante para el estudio, pero sí el segmento económico o el nivel de ingresos del mismo.

2) **Pre-procesamiento**: durante esta etapa se busca *limpiar* los datos. Este quiere decir que se tomará una serie de acciones para que los datos no cuenten con inconsistencias u observaciones faltantes/inválidas. Durante esta etapa se realiza una limpieza de los datos:

   - **Faltantes**: en torno a esta situación se pueden tomar una serie de acciones, como ignorar datos con observaciones faltantes, llenarlos manualmente, usar una variable global para llenarlos (como N/A, -inf, etc), poner la media del atributo con respecto a todos los datos, usar la media del atributo considerando sólo los datos de la misma clase o usar el valor más probable del dato.

   - **Datos *ruidosos***: un dato ruidoso es una observación que tiene un error aleatorio en una variable medida.

   - **Datos inconsistentes**: los datos inconsistentes se generan principalmente por variaciones al momento de ingresarlos, como el uso de diferentes capitalizaciones o faltas de ortografía. Una inconsistencia puede ser, por ejemplo, si en una observación de persona, su ciudad de residencia es “Santiago”, mientras que en otra es “stgo” o “Sanitago”. Se entiende que todas las observaciones hacen referencia a la misma ciudad, pero por errores o decisiones humanas, tienen un valor diferente.

3) **Transformación**: en esta etapa se realizan todas las transformaciones necesarias a los datos para que puedan ser interpretados de mejor manera por los algoritmos de minería de datos. Dependiendo de los algoritmos a aplicar, se requiere aplicar uno o más tipos de transformación, siendo algunas de ellas:

   - **Normalización**: consiste en representar los valores de las observaciones en un intervalo definido; por ejemplo, normalizar los datos para que sus valores estén dentro del rango [0,1]. Este método es de particular importancia cuando se planea utilizar técnicas de *clustering* basadas en distancia, ya que al no aplicarse, se desbalancea la importancia de diferentes variables por culpa de las unidades de medidas usadas. Por ejemplo, de distorsionará la distancia, dándole más importancia a una variable de mayor magnitud, como podría ser el ingreso per cápita de una base de datos de clientes (orden de los cientos de miles y millones) respecto de la edad.

   - **Agregación**: utilizada cuando se desea agrupar variables. Por ejemplo, pasar una serie de registros de ingreso mensual a una cantidad más reducida de registros de ingreso anual.

   - **Generalización**: como dice su nombre, consiste en generalizar variables. Se trata de reemplazar datos de variables de bajo nivel por un dato de niveles más altos. Por ejemplo, reemplazar datos de ciudades por regiones (Santiago a Región Metropolitana, Temuco a IX región, etc).

4) **Minería de datos**: esta fase consiste en la búsqueda de patrones de interés en alguna forma particular de representación, dependiendo del objetivo final de la minería.

5) **Interpretación/Evaluación**: en esta etapa final, se interpretan y evalúan los patrones encontrados, con el fin de juzgar su utilidad para el objetivo final o negocio, además de su asertividad.

Ilustración 2.1: Proceso del descubrimiento del conocimiento (KDD)

(https://www.researchgate.net)

2.2.2. SEMMA (Sample, Explore, Modify, Model and Assess)

SEMMA es una serie de etapas secuenciales que guía a la implementación de aplicaciones de minería de datos. Su nombre es acrónimo para *Sample*, *Explore*, *Modify*, *Model & Assess*, lo que hace referencia a cada una de las fases del proceso:

1) *Sample* (**Muestreo**): consiste en la selección de un conjunto de datos para modelar. El desafío recae en que esta muestra debe ser lo suficientemente grande para que sea representativa, y lo suficientemente pequeña como para ser manejada de forma eficiente.

2) *Explore* (**Exploración**): durante esta fase se visualizan los datos, con el fin de entenderlos al descubrir relaciones, anticipadas como no anticipadas, entre las variables en ellos, además de la detección de anomalías.

3) *Modify* (**Modificación**): en esta etapa del proceso se realiza cualquier acción para seleccionar, crear y/o transformar datos con el fin de prepararlos para el modelo.

4) *Model* (**Modelado**): el objetivo de esta fase es aplicar varias técnicas de modelo sobre las variables preparadas con el fin de crear modelos que puedan posiblemente generar los resultados esperados.

5) *Assess* (**Evaluación**): última etapa de *SEMMA*, consiste en la evaluación de los modelos desarrollados, con el objetivo de juzgar si son suficientemente confiables y útiles.

Una crítica que se hace comúnmente a este proceso, es que deja los aspectos propios del negocio afuera del análisis, a diferencia de otros procesos como *CRISP-DM*[10], que cuentan con fases[11] enfocadas en estos aspectos.

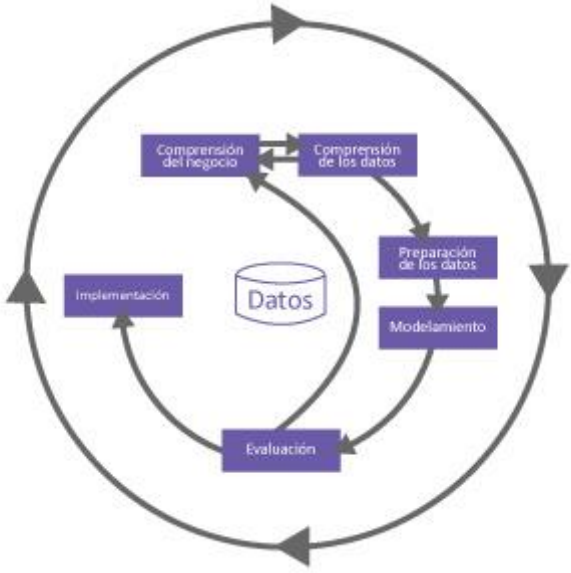
2.2.3. CRISP-DM (Cross-Industry Standard Process for Data Mining)

CRISP-DM recibe su nombre del acrónimo en el título (en español, Proceso estándar multi-industria para minería de datos), y consiste en un ciclo compuesto de 6 etapas:

[10] ***C****ross-****I****ndustry* ***S****tandard* ***P****rocess for* ***D****ata* ***M****ining*, o en español, Proceso estándar multi-industria para minería de datos.
[11] Fase *Business Understanding phase* (Fase de entendimiento del negocio) de *CRISP-DM*.

Ilustración 2.2: Ciclo de vida de CRISP-DM

(http://www.function1.com)



1) **Entendimiento del negocio**: en la primera etapa de *CRISP-DM*, se busca comprender los objetivos y requerimientos del proyecto desde el enfoque del negocio, para luego transformarlo en un problema de minería de datos y un plan preliminar para alcanzar los objetivos.

2) **Entendimiento de los datos**: comienza con un conjunto de datos inicial, y se busca familiarizarse con ellos, identificar problemas de calidad, descubrir una primera mirada o subconjuntos interesantes con el fin formular una hipótesis para información escondida.

3) **Preparación de los datos**: esta fase comprende todas las actividades necesarias para generar el set de datos final a partir de los datos en bruto.

4) **Modelo**: es la aplicación de varias técnicas de modelo, calibrando sus parámetros a valores óptimos.

5) **Evaluación**: los modelos obtenidos son juzgados y los pasos para construirlos son evaluados con el fin de concluir con seguridad que efectivamente cumple con los objetivos del negocio.

6) **Despliegue**: el término del modelo por lo general no significa el fin del proyecto. El conocimiento obtenido luego debe ser organizado y desplegado de forma que el cliente final pueda utilizarlo.

De forma gráfica, se aprecia en la **ilustración 2.2** la serie de etapas que componen el proceso.

2.3. Tareas de minería de datos

En esta sección se revisarán los diferentes tipos de tareas de minería de datos. Además, se describirán cada una de las subcategorías pertenecientes a dichos tipos.

2.3.1. Tareas descriptivas

En este tipo de tareas el objetivo es describir los datos existentes. Busca proporcionar información entre las relaciones existentes en los datos y sus características. En el contexto, teóricamente se podría llegar a una afirmación como por ejemplo: el que un estudiante tenga actividades extraprogramáticas en el primer semestre, implica que también tendrá en el segundo.

Visualización

La tarea de visualización consiste en revisar los datos de forma mecánica, para encontrar cualquier relación entre variables que se pueda apreciar en primeras

instancias. Para facilitar esta tarea hay una gran cantidad de software, de donde destaca *Tableau*[12], que si bien se trata de una herramienta de procesamiento analítico de datos (*OLAP*), puede ser utilizada también para visualización.

Correlaciones y factorizaciones

Esta tarea consiste en desplegar los datos y evaluar si se encuentra alguna correlación entre las variables pertenecientes al estudio. La correlación puede ser lineal o ser de otra manera. Esta tarea solo tolerará valores numéricos, debido a su naturaleza.

Asociación

La asociación es una tarea descriptiva, no supervisada, que hace referencia a reglas que son capaces de describir los datos en base a ocurrencias en las variables; en otras palabras, describe el comportamiento de una variable en base al de otra (u otras). Por ejemplo, una regla de asociación sería "si el año de ingreso de un estudiante es igual a 2008, ha tomado actividades extra programáticas y estudia arquitectura, entonces su colegio es subvencionado". Las reglas de asociación sólo pueden aplicarse sobre variables nominales (todas las involucradas). La asociación se presentará en una de dos maneras:

- Reglas de asociación: son asociaciones recíprocas, o sea, que hay una implicancia doble, describiendo cada una de las variables relacionadas a la asociación a la otra.

- Dependencias: a diferencia del caso anterior, este tipo de asociaciones son direccionales, o sea, el cumplimiento de una serie de condiciones implica que se cumplirán otras, y no al revés.

[12] *http://www.tableausoftware.com*

Los algoritmos de búsqueda de asociaciones tienen la particularidad de que la mayoría se puede descomponer en dos fases. La primera consta de la búsqueda de un conjunto de ítems frecuentes con un soporte[13] mayor o igual al deseado, o sea, que se buscan conjuntos de elementos que cuenten con cierto criterio establecido, sin separarlos aún. Luego, en la segunda fase, se hacen particiones de los conjuntos de ítems, calculando la confianza[14] de cada una, y reteniendo las reglas que tengan confianza mayor o igual a la deseada.

Segmentación (Agrupamiento)

Este tipo de tareas consisten en agrupar los datos en diferentes subconjuntos, o clases, de acuerdo a la relación entre ellos. Se busca que todos los elementos presentes en un grupo definido tengan propiedades parecidas, o sea, similares entre sí y diferentes a los de otros grupos. La segmentación es una técnica de aprendizaje no supervisado, que utiliza el término de *distancia* para describir a los elementos; dos elementos tendrán poca distancia entre ellos si son parecidos (similares). Análogamente, tendrán harta distancia entre ellos si no son similares. En base a estos términos, la segmentación busca minimizar la distancia de los elementos pertenecientes a un mismo grupo y maximizar la distancia entre los grupos.

La definición de distancia puede variar de acuerdo a la fórmula que se utilice para calcularla:

- Distancia de Minkowski:

$$d_r(x,y) = \left(\sum_{j=1}^{J} \left|x_j - y_j\right|^r\right)^{\frac{1}{r}}, r \geq 1$$

[13] Medida para cuantificar los casos en los que e antecedente se hace verdadero. Puede ser número de casos o porcentaje.

[14] Número de casos en que, habiéndose cumplido el antecedente de la regla, se cumple el consecuente.

- Distancia de Manhattan (Minkowaki, con r = 1)

$$d_1(x, y) = \sum_{j=1}^{J} |x_j - y_j|$$

- Distancia Euclideana (Minkowaki, con r = 2)

$$d_2(x, y) = \sqrt{\sum_{j=1}^{J} (x_j - y_j)^2}$$

- Distancia de Chebyshev (Minkowaki, cuando $r \rightarrow \infty$)

$$d_\infty(x, y) = \max_{j=1..J} |x_j - y_j|$$

Los algoritmos de segmentación pueden ser clasificados por diferentes tipos, los cuales se listan a continuación:

- Particionamiento: estos métodos construyen *k* particiones de un conjunto de datos, representando cada una de éstas un grupo. *K* debe ser menor o igual al número de elementos del conjunto de datos mencionado. Ejemplos: *K-Means, K-Medoids.*

- Jerárquicos: genera una descomposición jerárquica del conjunto de grupos, en otras palabras, crea una serie de subconjuntos de datos, en los que algunos engloban a otros. Ejemplos: *BIRCH, CURE*.

- Basados en densidad: de acuerdo a *[5]*, se define la pertenencia de cada elemento a un *cluster* si dicho elemento contiene una cantidad establecida de vecinos, dentro de una vecindad definida de radio mayor a 0. Ejemplos: *DBSCAN, OPTICS.*

- Basados en grilla: separa el espacio de datos en una grilla (finita), para luego realizar operaciones de agrupamiento sobre ella. Ejemplos: *STING, CLIQUE*.

- Basados en modelo: se utiliza un potencial de modelo para cada uno de los grupos, ajustando los datos a dichos modelos. Ejemplos: *COBWEB*, *CLASSIT* (estadísticos), mapas autorganizados (redes neuronales).

Detección de anomalías

La detección de valores e instancias anómalas es una tarea necesaria al momento de realizar minería de datos. En todo conjunto de datos se presentarán registros que se escapen de todo patrón o tendencia, y es importante poder reconocerlos para no considerarlos como un patrón común, si no como comportamientos anómalos como fraudes, fallas u *outliers*. Informalmente, un *outlier* es cualquier valor de dato que pareciera estar fuera de lugar con respecto al resto de los datos. De acuerdo a *[2]*:

"La definición intuitiva de un *outlier* sería 'una observación que se desvía tanto de las otras observaciones como para despertar sospechas de que fue generada por un mecanismo diferente'"[15]

2.3.2. Tareas predictivas

Las tareas predictivas, son problemas en los que se hace necesario predecir un (o varios) valores para un grupo de datos. La salida de una tarea predictiva es una categoría (a la que pertenece uno o más datos) o un valor numérico relacionado con el o los datos en cuestión. A continuación, se revisarán algunos de los tipos de tareas predictivas.

[15] Textual en inglés: "The intuitive definition of an outlier would be 'an observation which deviates so much from the other observations as to arouse suspicions that it was generated by a different mechanism'"

Clasificación

Las tareas de clasificación buscan definir un modelo que sea capaz de predecir la clase de un objeto que no la tiene definida. Además, pueden ser utilizadas para estimar un valor que se encuentre perdido o que no se tenga a priori. Estas tareas son supervisadas, por lo que se debe contar con un conjunto de datos de entrenamiento, que ya se encuentran clasificados. El proceso de generación de un modelo predictivo consta de 3 pasos:

1) División de los datos en dos conjuntos: entrenamiento y prueba.

2) Utilización del subconjunto de entrenamiento para la construcción del modelo.

3) Utilización del subconjunto de prueba para validar el modelo conseguido en el punto anterior, si el porcentaje de casos exitosos es aceptable, se valida el modelo (útil para clasificar otros casos).

A continuación, se revisarán algunos algoritmos de clasificación.

Árboles de clasificación

Los árboles de clasificación son un método en el cual se somete un dato a una serie de condiciones, que lo clasifican de acuerdo a los valores de las variables relacionadas con el mismo. Por ejemplo, se somete primero a un dato a la evaluación de una variable: “si el alumno tiene un promedio mayor a 55, entonces se pregunta la variable *año de ingreso*; si no, se consulta la variable *plan de carrera”*, con el fin de predecir alguna variable en particular. Cabe destacar que no se trata de árboles binarios, si no que se pueden considerar numerosos intervalos o valores para cada variable para generar la clasificación.

Su estructura es similar a un diagrama de flujo, donde cada vértice simboliza una condición a la que se somete el dato a predecir. El último nivel del árbol, los nodos hoja, representan las clases. Su construcción suele llevarse a cabo con estrategias del tipo "dividir y conquistar"[16], empezando con todos los elementos del grupo de entrenamiento en la raíz, y continuando dividiéndolos en el atributo que se elija para ramificarlo.

Inducción de reglas de clasificación

Los métodos de inducción de reglas tienen las mismas propiedades que los métodos de árboles de decisión, describiendo una serie de condiciones *if-then* para llegar a la clasificación deseada. La obtención de dichas condiciones, o reglas, puede ser a partir de un árbol de decisión, a través de algoritmos específicos como *STAR* o *Ripper*, o a partir de reglas de asociación. Además, es posible extraer reglas de clasificación desde una red neuronal, a través del algoritmo *MofN*, propuesto por *[3]*. En particular, permite la extracción de reglas desde una red neuronal multicapa, a través de agrupamiento, extracción de reglas, agrupación de reglas y poda de reglas.

Métodos Bayesianos

Estas herramientas estadísticas son capaces de predecir las probabilidades de que un elemento en cuestión pertenezca a una clase en particular. Asumen que el valor de cada una de las propiedades es independiente de los valores de las otras (en un mismo elemento), llamada independencia condicional de clases. Los métodos bayesianos se basan en el teorema de Bayes, y pueden ser utilizados tanto para fines descriptivos como predictivos. En el primer caso, se usan para descubrir relaciones de independencia y/o relevancia para poder realizar un estudio más profundo a través de inferencias estadísticas. En el segundo caso, se utilizan como clasificadores.

[16] Conocido por su traducción en inglés *divide & conquer*.

Métodos basados en casos y vecindad

Se caracterizan por utilizar el conjunto de entrenamiento para clasificar nuevos datos. En esta categoría hay presentes técnicas para segmentación, como *K-Means*, y para clasificación, como *LVQ*. Además, se utilizan métodos de ensamblaje, que combinan varios modelos con el objetivo de conseguir una mejor precisión final en el clasificador

Regresión estadística

A través de esta tarea, se busca generar una función matemática que sea capaz de estimar el valor de alguna variable de interés a partir del resto de las variables relacionadas con un dato en particular. La regresión estadística puede ser utilizada únicamente para valores numéricos, y la función se puede calcular a través de interpolación, estimación o logística.

2.4. Herramientas de minería de datos

Hay una amplia gama de herramientas de minería de datos a disposición. Cada herramienta cuenta con implementaciones diferentes de una porción de los algoritmos más utilizados en los procesos de minería. Además, muchas de las herramientas cuentan con interfaces usuarias para facilitar el proceso de minería para quienes no tienen un conocimiento base de líneas de comando o programación. Algunas de las herramientas más utilizadas se listan a continuación.

- *Rapid Miner*: escrita en Java, esta herramienta de minería de datos funciona en torno a interfaces gráficas avanzadas, por lo que el usuario final requiere escribir muy poco código. Cabe destacar que esta herramienta se ofrece como servicio, más que como software local. Además, proporciona funcionalidades

de pre-procesamiento y visualización, análisis predictivo, esquemas de aprendizaje y algoritmos de scripts de R. Esta potente herramienta es de código abierto, bajo licencia AGPL[17] (http://www.rapidminer.com).

- *Angoss*: enfocada principalmente para organizaciones involucradas con ventas, marketing y análisis de riesgo, esta herramienta cuenta con una interfaz gráfica avanzada además de un asistente amplio para sus procedimientos. Si bien las interfaces y asistente podrían ser restrictivos para usuarios avanzados, *Angoss* implementa soporte total de línea de comando en R, satisfaciendo así a los usuarios que prefieren personalización por sobre facilidad de uso. Una buena característica de esta herramienta es que tiene una amplia gama de representaciones gráficas de datos. Si bien cuenta con variedad de implementaciones de los algoritmos más conocidos, no tiene suficientes herramientas para personalizar los procesos, por lo que no es la opción para quienes prefieran un ambiente fácilmente extensible (http://www.angoss.com).

- *KNIME (Konstanz Information Miner)*: esta herramienta nace como una solución para farmacéuticas a nivel empresarial. Los desarrolladores crearon un producto escalable, modular y de código abierto, teniendo la flexibilidad necesaria para adaptarse rápidamente a las demandas de un campo de estudio en crecimiento como es la minería de datos. Siguiendo su éxito en la industria farmacéutica, otras industrias siguieron la tendencia y utilizan *KNIME* para sus procesos de *CRM*[18] e inteligencia de negocios. Otra ventaja considerable que tiene esta herramienta, es que cuenta con una comunidad activa tanto de desarrolladores como de usuarios. No requiere conocimientos de

[17] Para más información sobre la licencia de tipo AGPL, referirse a http://www.gnu.org/licenses/agpl-3.0.html.

[18] *CRM*, o *Customer Relationship Management*, es el proceso que gira en torno a pulir y mejorar las relaciones con los clientes, a través de campañas de marketing específicas, servicio personalizado al cliente y gestión del equipo de ventas.

programación para ser usada, ya que cuenta con interfaces intuitivas y de fácil uso (http://www.knime.org).

- *R:* más que una herramienta para minería de datos, R es un lenguaje de programación y ambiente para computación estadística y análisis. Es esta la razón que hace a R una potente herramienta de minería de datos. Bajo licencia GPL[19], y de código abierto, R puede ser personalizado abiertamente, sin restricción, lo que se traduce en una cantidad inigualable de algoritmos e implementaciones desarrolladas por usuarios alrededor del mundo, lo que se traduce en una herramienta flexible, escalable y extremadamente personalizable. Por otro lado, a pesar de que hay algunas interfaces para tratar con este lenguaje, se requiere conocimientos de programación (y del lenguaje en sí) para sacar el máximo provecho de esta herramienta (http://www.r-project.org).

[19] Más información sobre licencia de R: https://www.r-project.org/COPYING

3. Diseño de la solución

Para este estudio se decide usar R, en primera instancia por las ventajas mencionadas en el párrafo anterior, y en segunda instancia porque la herramienta ya es conocida por quien realiza el estudio.

Para el proceso de desarrollo de la solución, se decide utilizar *CRISP-DM* debido a su integración y consideración de reglas del negocio.

Como primer acercamiento al alcance de los objetivos propuestos, se requiere identificar propiedades del negocio. Se ha detectado que el período de actividad de los usuarios no es satisfactorio para las proyecciones de funcionamiento de la plataforma. Se hace necesario entonces generar cambios que puedan mejorar la experiencia de los mismos y aumentar los niveles de interacción que tienen con el producto. Se abordarán en este capítulo las dos primeras etapas de la metodología *CRISP-DM.*

3.1. Entendimiento del negocio

Se estudiará entonces cómo maximizar la cantidad de **usuarios activos** en un instante de tiempo, evaluando los escenarios en los que esta variable alcanza valores satisfactorios para poder replicar dichos escenarios. Por otro lado, se intentará también encontrar patrones e indicadores en los escenarios en lo que la variable mencionada se encuentre en sus valores más bajos, para así poder evitarlos.

La finalidad de buscar el aumento de usuarios activos en la plataforma recae en que, para los clientes del servicio, éste se vuelve más atractivo entre más usuarios activos posea. Es común que un negocio de este segmento cuente con una gran

cantidad de usuarios registrados, pero éstos no son los que tienen valor para el cliente final, ya que no necesariamente se encuentran actualmente en interacción con la plataforma. Por ejemplo, es más atractivo tener 10.000 usuarios, pero con un 50% de ellos activos (5.000 potenciales clientes), que tener 1.000.000 de usuarios, pero sólo con 1% de ellos activos (1.000 potenciales clientes).

Otra variable de particular interés para el negocio (y que se relaciona con la variable recién mencionada) es la **penetración** de un video en particular. Este indicador, a diferencia de los usuarios activos que busca patrones de la plataforma en un instante de tiempo, se enfoca en el video mismo en cuestión (principalmente), como la duración. No se descarta que haya variables y patrones en la plataforma (externos al video mismo) que afecten en su penetración.

Una mayor penetración significa un mayor éxito en la campaña de los clientes de Kikvi, ya que se traduce en que el video de la campaña alcanzó una mayor cantidad de personas, que es justamente lo que ellos buscan en un servicio como este: llegar a la mayor cantidad de personas con la campaña en cuestión.

Tomando como enfoque principal los usuarios, se hace necesario poder identificar de forma eficiente la calidad de estos mismos, en base a su forma de interactuar con la plataforma en el tiempo. De esto se desprende una nueva variable de interés: **calidad usuaria**. Este indicador tiene un particular interés para quienes administran la plataforma, ya que el saber qué es lo que define a un usuario de calidad puede llevar al conocimiento de cómo se genera o consigue dicho usuario, para luego poder replicar el proceso y construir una base de usuarios de calidad para los objetivos finales del cliente, entregando finalmente un mejor servicio. Además, el entender cada una de las clases usuarias servirá para enfocar esfuerzos en la dirección correcta al hacer campañas de publicidad para conseguir usuarios nuevos.

Tabla 3.1: Indicadores de interés

Indicador	Descripción
Usuarios activos	Cantidad de usuarios activos (que han tenido actividad con la plataforma en la última semana) en un instante de tiempo.
Penetración	Proporción de usuarios activos que comparten un video en particular.
Calidad usuaria	Hace referencia a la frecuencia, dimensión y extensión de la forma en que un usuario interactúa con la plataforma.

Tabla 3.2: Variables de entrada, caso usuarios

Variable	Descripción
puntos_historicos	Variable que hace referencia a la cantidad de puntos totales que ha acumulado un usuario durante su actividad.
genero	Sexo del usuario.
puntos_gastados	Cantidad total de puntos que el usuario ha gastado en la plataforma.
shares_totales	Cantidad total de veces que el usuario compartió algún video a través de la plataforma.
recruitments	Cantidad de usuarios que fueron reclutados por el usuario en cuestión.
fecha_afiliacion	Fecha en la que un usuario se registró en la plataforma.
uni	Universidad en la que estudia (o estudió) un usuario.
nacimiento	Fecha de nacimiento de un usuario.
puntos_gastados	Cantidad de puntos gastados del usuario.
categoria_dominante	Categoría de videos preferida por usuario (en base a su interacción)
concursos_participados	Cantidad de concursos (diferentes) en los que participó un usuario.
premios_canjeados	Cantidad de premios canjeados por un usuario.
tickets_canjeados	Cantidad de tickets canjeados por un usuario.
difference_last_and_first_share	Diferencia de tiempo entre la primera y la última vez que un usuario compartió un video.
difference_last_raffle_first_share	Diferencia de tiempo entre la primera vez que un usuario compartió un video y la última vez que canjeó un ticket.

Los indicadores que finalmente serán de interés para este estudio se pueden apreciar de manera resumida en la **tabla 3.1**.

Habiendo definido los indicadores de interés, se muestran en las **tablas 3.2 y 3.3** las variables de entrada, cuya variación podría afectar o no a los indicadores de interés. Estas variables son extraídas de la base de datos de la plataforma, de las tablas de usuarios, videos y de sus interacciones).

Tabla 3.3: Variables de entrada, caso videos.

Variable	Descripción
duración	Duración en segundos del video en cuestión
release_difference	Diferencia de lanzamiento entre el video en su fuente original (*youtube*) y su publicación en la plataforma.
total_views	Cantidad total de vistas que el video en cuestión consiguió durante el período de estudio.
shares_first_day	Cantidad de veces que un video fue compartido durante el primer día de su publicación.
shares_first_week	Cantidad de veces que un video fue compartido durante la primera semana de su publicación.
shares_first_month	Cantidad de veces que un video fue compartido durante el primer mes de su publicación.
total_shares	Cantidad total de veces que un video fue compartido durante el período de estudio.
active_raffles	Cantidad de concursos al momento de publicación del video en cuestión.
active_canjes	Cantidad de canjes al momento de publicación del video en cuestión.

3.2. Entendimiento de los datos

En esta sección se revisará cómo se distribuyen y relacionan en una primera instancia las variables mencionadas en el punto anterior. Se presentarán gráficos y descripciones sólo de variables relevantes para el estudio, es decir, que aporten a los objetivos planteados o sean razón de toma de decisiones.

Los datos de la plataforma estudiada se encuentran en una base de datos relacional detrás de un motor *MySQL*, cuyo modelo de datos parcial se puede apreciar en el **anexo 1**. Cabe destacar que se incluyeron sólo las tablas y variables más significativas en el diagrama.

La ventana de tiempo contemplada en el estudio es desde el 27 de junio 2013[20], hasta el 9 de enero 2015[21]. Se cuenta para este período con 2669 registros de usuarios y 921 registros de videos.

Una primera mirada a los datos deja en evidencia los siguientes problemas:

- Para el caso de datos de usuario, la variable **uni** (*universidad_id* en tabla *users*, referencia a la universidad en la que estudia o estudió el usuario en cuestión) tiene una gran cantidad de datos perdidos (nulos o vacíos); aproximadamente un 85% de los registros tiene falencias en este aspecto. Se decide no considerar esta variable para el estudio.

- La variable **release_difference** (o diferencia de lanzamiento desde la fecha en que se sube un video en su fuente original, y se publica en la plataforma), junto con otras variables[22] relacionadas con tiempo, se encuentran en unidades poco intuitivas para ser dimensionadas por el estudio. Se cambiarán estas variables a unidades más adecuadas en cada caso.

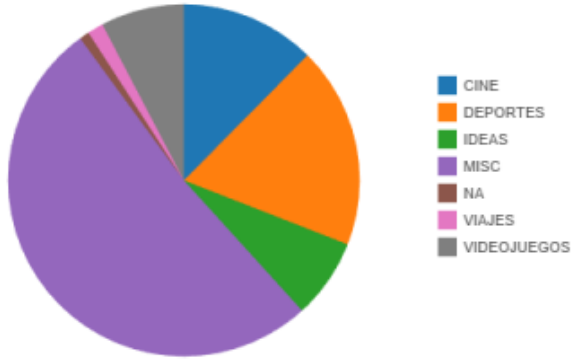
- Es necesario cambiar la fecha de nacimiento de los usuarios por una variable más comparable, o sea, por la edad de los mismos al momento del estudio.

[20] Fecha de inicio de actividades de la plataforma.
[21] Fecha en que se comienza a realizar este estudio.
[22] Otras variables: *difference_last_and_first_share, difference_last_raffle_first_share*

3.2.1. Visualización de datos usando *Tableau* (OLAP)

A continuación, se revisará en primera instancia cómo se distribuyen y relacionan las variables de la plataforma. Para estos fines se utilizó la herramienta *Tableau*, principalmente por su variedad de posibilidades, facilidad de uso, y familiarización con quien realiza el estudio.

Ilustración 1.4: Distribución categorías de videos (921 registros)

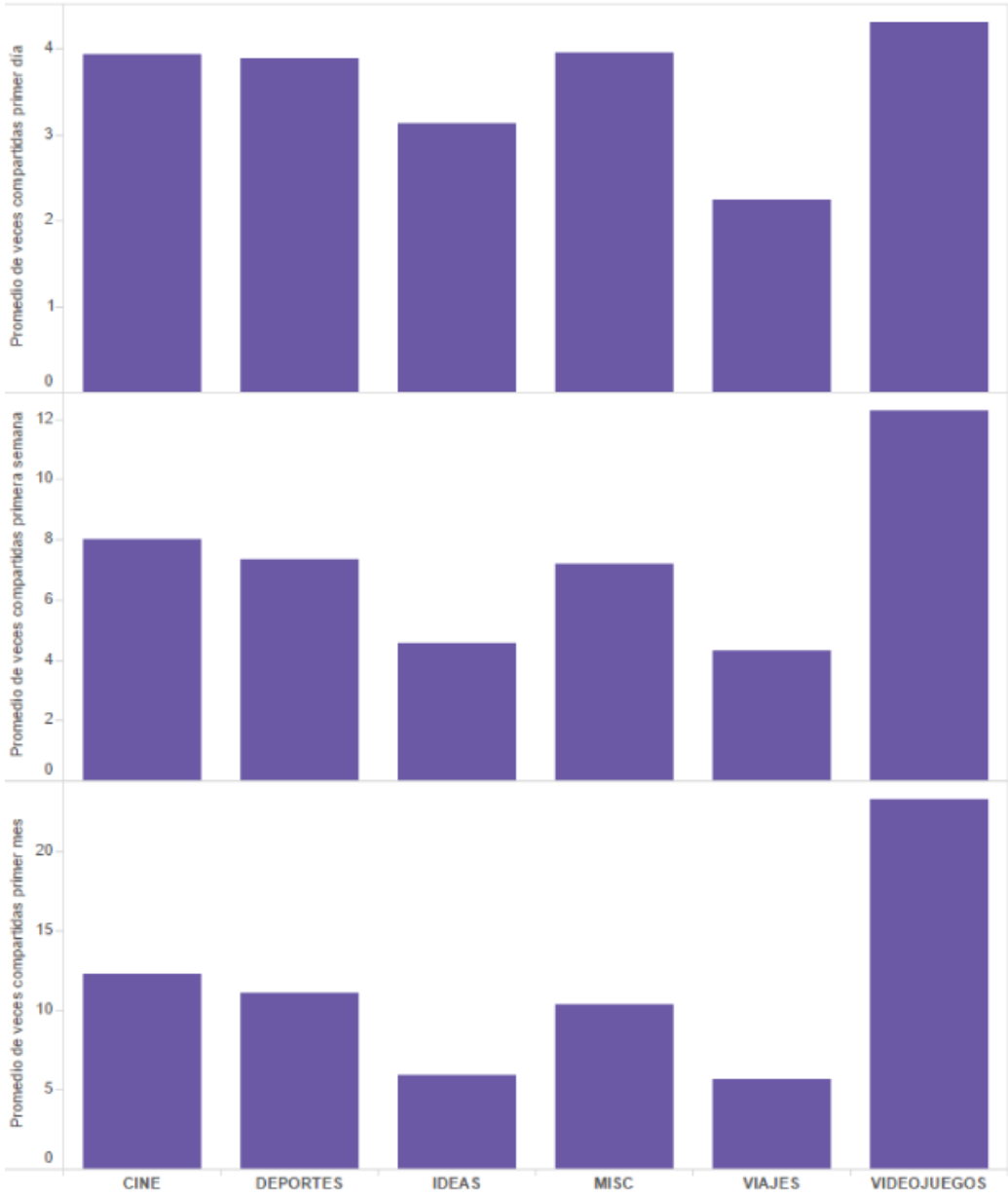


En la **ilustración 3.4** se muestra la distribución de videos de acuerdo a su categoría. Aproximadamente un 50% de ellos pertenecen a *MISC*, que es una categoría relativamente general para definir a todos los videos que no pertenecen al resto de los grupos, lo que le resta valor a esta variable en particular. En las siguientes ilustraciones se revisará cómo se comporta esta variable en relación a otras.

En las **ilustraciones 3.5a, 3.5b y 3.5c** se muestra cómo se relaciona la categoría de un video con la cantidad de veces que ésta se comparte el primer día, la primera semana, y el primer mes (luego de su publicación). Para el primer caso (4a), no se aprecia una diferencia muy significativa. Dejando de lado *viajes*, el resto parece comportarse de forma similar. Pero cuando se revisan los siguientes gráficos,

que relacionan las categorías con las veces que se comparte en la primera semana (4b) y mes (4c) respectivamente, se aprecia que *videojuegos* tiende a mantener su popularidad, mientras que el resto de las categorías parecen dejar de ser interesantes para el usuario.

Ilustraciones 3.5a, 3.5b y 3.5c: Categoría de video vs promedio de veces que se comparte en intervalos



3.5a

3.5b

3.5c

En base a esta potencial relación, se decide estudiar cómo se comportan cada una de las categorías a largo plazo en función de las veces que se comparten. Además, se decide estudiar si este comportamiento se ve reflejado o no en una mayor cantidad de vistas para dichas categorías al largo plazo. El resultado se aprecia en la **ilustración 3.6**, donde se confirma que la hipótesis es correcta y que en el tiempo la categoría de *videojuegos* pareciera seguir siendo la más compartida. Por otro lado, al revisar la relación con el promedio del total de vistas, la categoría *Misc* parece ser la que más destaca. Una mirada a los datos de este gráfico permite ver que hay *outliers*[23] presentes en esta categoría en particular. Se prueba quitando estos datos para ver las nuevas proporciones, pero la tendencia se mantiene (aunque un poco menos marcada). Finalmente se concluye que *misc* parece ser la categoría más vista dentro de las planteadas, aunque no la más compartida.

En base al comportamiento indicado por las **ilustraciones 3.5a, 3.5b y 3.5c**, se decide revisar si existe alguna relación entre la cantidad de vistas que consigue un video y la cantidad de veces que se comparte. Esta relación se aprecia en la **ilustración 3.7**, donde se ve que una gran cantidad de los videos se encuentran en el cuadrante de menores valores del gráfico, por lo que se decide estudiar en detalle el cuadrante definido por videos con hasta 10.000 vistas, y hasta 200 veces compartido. Este cuadrante se aprecia en la **ilustración 3.8** donde, nuevamente, no se aprecia ningún tipo de tendencia o relación entre la cantidad de veces que se comparte un video y la cantidad de vistas que consigue. Iteraciones siguientes de filtrado no reflejaron cambios significativos, por lo que se decide no incluir los gráficos correspondientes.

Para la base de datos de usuarios, se presenta en la **ilustración 3.9** la distribución de las observaciones en relación a la edad. Se deduce de este gráfico que la mayor cantidad de usuarios de la plataforma tiene entre 20 y 26 años. También se aprecia la presencia de muy pocos usuarios sobre 30 años de edad.

[23] En este caso, un video con más de 50.000 vistas, y tres videos con más de 20.000.

Ilustración 3.6: Categoría de video vs Promedio de vistas y veces que se comparte

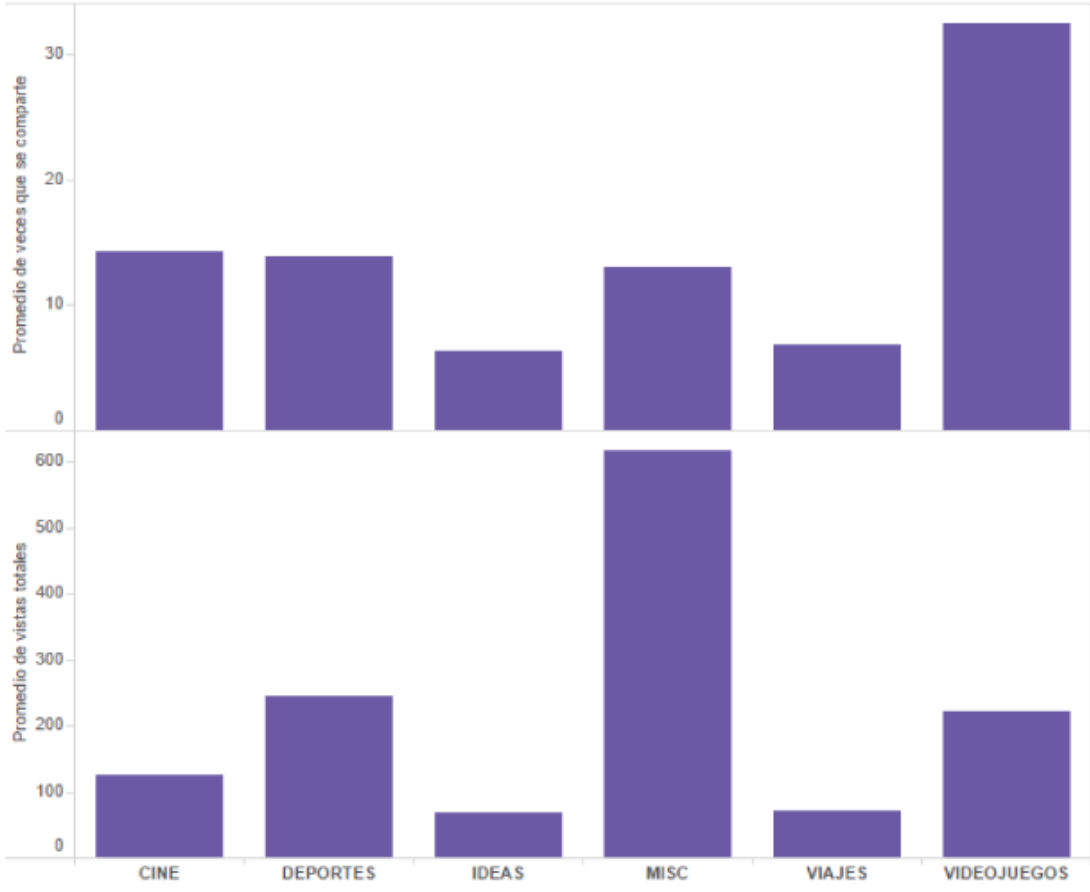


Ilustración 3.7: Veces que se comparte vs Veces que se ve

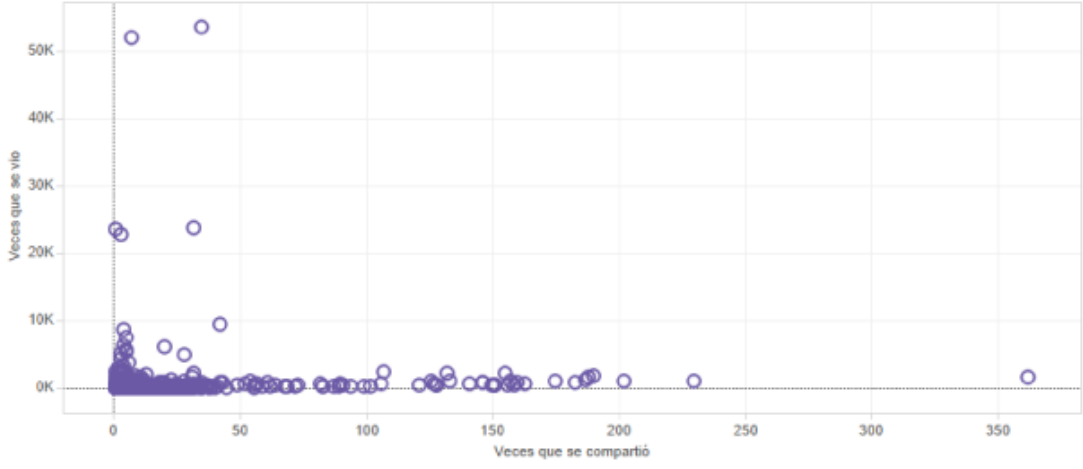


Ilustración 3.8: Veces que se comparte vs veces que se ve (filtrado)

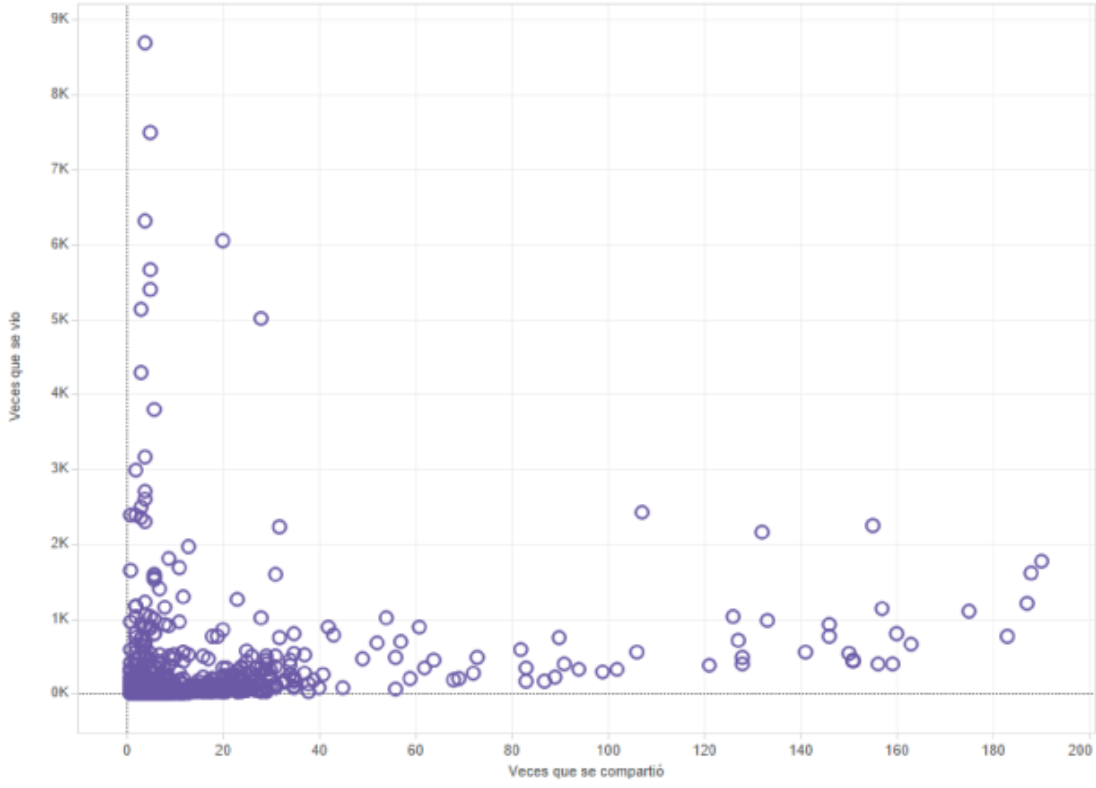
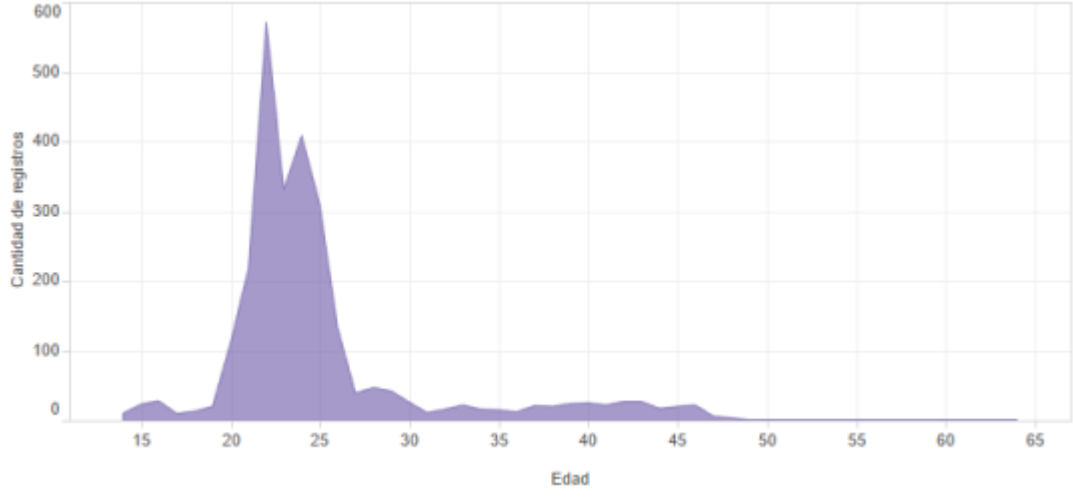


Ilustración 3.9: Distribución etaria de grupo usuario



A continuación, se presenta el análisis en torno a la edad de los usuarios de la plataforma; en la **ilustración 3.10** se aprecia la relación entre esta variable y la cantidad total de veces que un usuario compartió videos. Rápidamente salta a la vista la presencia de *outliers*: una observación de 31 años de edad con más de 500 videos compartidos, y registros con edades superiores a 45 años de edad con muy pocos videos compartidos. En la **ilustración 3.11** se presenta nuevamente la relación mencionada, esta vez sin los *outliers* identificados.

Ilustración 3.10: Edad vs Veces que compartió

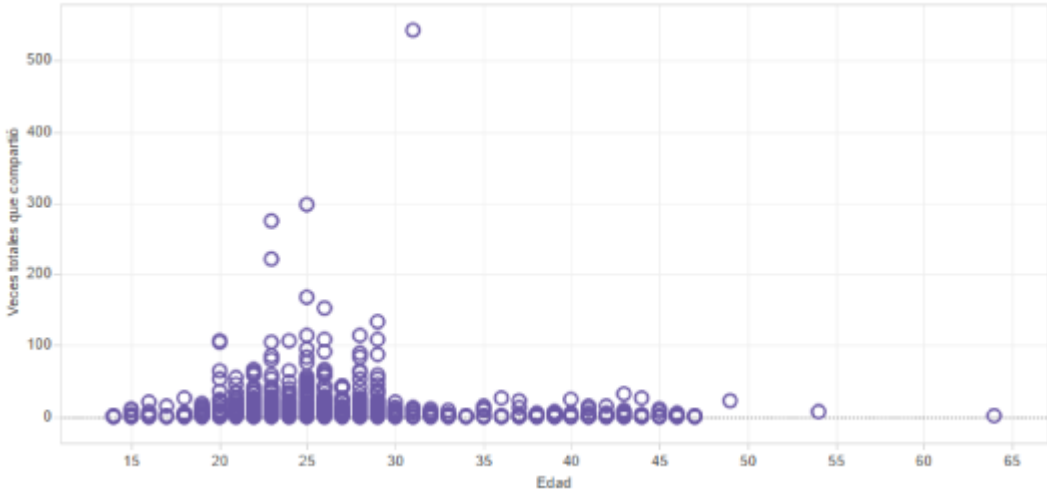
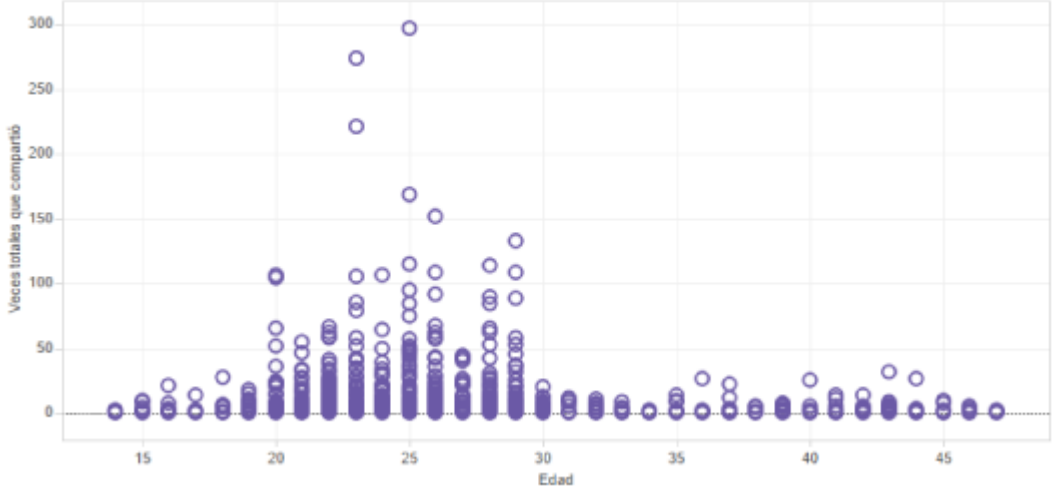


Ilustración 3.11: Edad vs Veces que compartió (filtrado)



En la **ilustración 3.11** aún se aprecian registros que se escapan de la norma, pero se puede identificar que el grupo etario más activo en relación a compartir videos se encuentra entre los 20 y los 29 años.

Como último análisis en torno a la edad del grupo usuario de Kikvi, se presenta en la **ilustración 3.12** la relación entre esta variable y la frecuencia promedio en días con la que los usuarios de dicha edad comparten contenido de la plataforma. Para fines del negocio, un valor menor de frecuencia es mejor, ya que significa que el usuario comparte más veces en un período de tiempo definido. Esto genera un problema ya que hay una gran cantidad de registros que tienen frecuencia con valor 0, que en la práctica quiere decir que no compartieron contenido de la plataforma en ningún momento, pero que en el gráfico (y al promediarse con otros valores) se verá reflejado como algo positivo (disminuyendo la frecuencia promedio en la edad a la que pertenezcan); por esta razón se decide filtrar los registros cuya frecuencia sea igual a 0. El resultado de este proceso se aprecia en la **ilustración 3.13**, de donde se puede deducir que los usuarios que comparten más frecuentemente se encuentran entre las 14 y los 28 años de edad.

Ilustración 3.12: Edad vs Frecuencia en que comparte

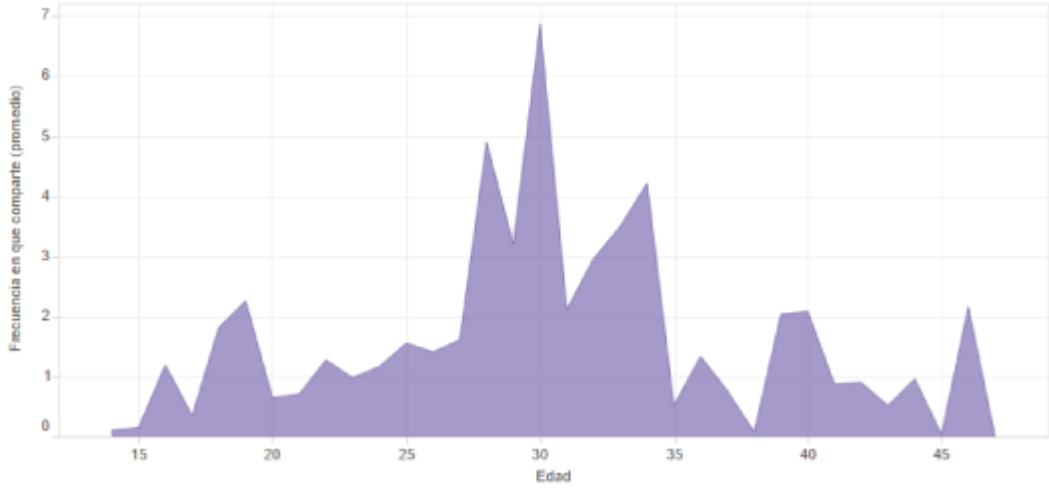
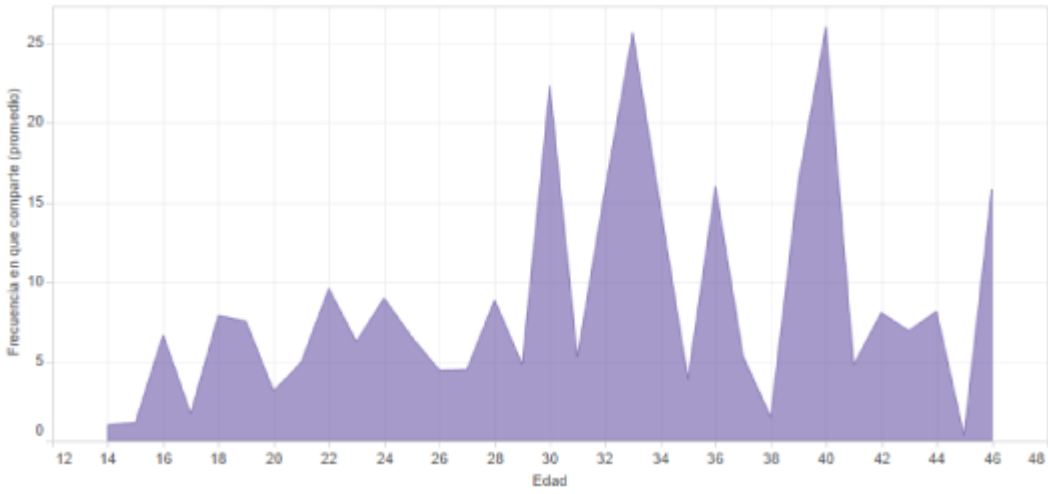


Ilustración 3.13: Edad vs Frecuencia en que comparte (filtrado)



Finalmente, se deduce del entendimiento de los datos que el grupo usuario de la plataforma se ve mejor representado por personas entre 20 y 26 años, y que a pesar de que hay fuera de este rango, estos no son muy activos en la plataforma, por lo que no aportan gran valor al negocio. Además, esto se complementa con que los videos que son más atractivos para estos usuarios son los pertenecientes a la categoría de videojuegos, ya que son los más compartidos tanto inicialmente como en el tiempo, aunque esto no significa que sean los más vistos, lo que podría explicarse en que las redes sociales en las que comparten dichos videos no comparten necesariamente sus mismos intereses por esta categoría de videos.

En relación a la actividad, esta primera mirada a los datos revela que los usuarios entre 14 y 28 años son los que más frecuentemente interactúan con el sitio, ya sea compartiendo videos, participando en concursos, o canjeando premios.

4. Desarrollo de la solución

En este capítulo se abordará el resto de las etapas pertenecientes al proceso *CRISP-DM*. En una primera instancia se aplicarán acciones sobre los datos, con el fin de adecuarlos para procesos posteriores. Luego se aplicarán y compararán una serie de modelos, y finalmente se evaluarán y desplegarán los resultados obtenidos.

4.1. Preparación de los datos

Durante esta etapa se realizó una serie de operaciones sobre los datos, con el fin de mitigar por un lado la falta de datos mencionada en el punto anterior, y por otro descubrir variables que no se encuentran inicialmente en el conjunto de datos, pero que pueden ser calculadas en base a la existentes.

Con el fin de mitigar la ausencia de datos de usuarios y de videos que podrían ser interesantes para el estudio, se utilizaron las herramientas (*API*) de *Facebook* y de *YouTube* para buscar información faltante de los usuarios y videos respectivamente. El proceso se automatizó utilizando los *scripts* en *PHP* y *Ruby on Rails* presentes en los **anexos 2 y 3** respectivamente.

Para poder satisfacer necesidades del negocio tratadas en puntos anteriores, se hace necesario definir los posibles valores que tomará el indicador de **calidad usuaria**. Teniendo como consideración que el enfoque actual de Kikvi es la difusión de contenidos, se establecen ocho niveles de calidad usuaria, usando como parámetros de entrada la frecuencia en la que el individuo comparte, la cantidad de veces que comparte, y el período total de actividad del mismo. En la **tabla 4.1** se muestran los criterios para asignación de clases de esta variable.

Tabla 4.1: Calidad usuaria

Calidad	Descripción
No interesado, No entendió	Usuario se registró en la plataforma, pero nunca tuvo interacción[24] con la misma
No capturado	Usuario se registró e interactuó en la plataforma durante un único día.
Perdido	Usuario se registró en la plataforma e interactuó con ella por un período menor a una semana.
Diario semanal	Usuario se registró en la plataforma e interactuó con ella diariamente, por un período de entre 7 y 29 días.
Diario mensual	Usuario se registró en la plataforma e interactuó con ella diariamente, por un período mayor a 30 días.
Diario constante	Usuario se registró en la plataforma e interactuó con ella diariamente, por un período mayor a 60 días.
Semanal mensual	Usuario se registró en la plataforma e interactuó con ella semanalmente (al menos), por un período de entre 1 y 2 meses.
Semanal constante	Usuario se registró en la plataforma e interactuó con ella semanalmente (al menos), por un período mayor a 2 meses.

Habiéndose definido el primero de los indicadores clave para el negocio mencionados en la **tabla 3.1**, en la **tabla 4.2** se explica cómo fueron calculados los dos restantes.

Tabla 4.2: Cálculo de indicadores clave

Variable	Descripción y cálculo
Usuarios activos	Cantidad de usuarios activos al momento en que un video fue publicado en la plataforma. Se define un usuario activo como un usuario que interactuó con la plataforma durante una semana definida [4]. Para este caso en particular, una interacción se entiende como la acción de compartir un video, realizar un canje, reclutar otro usuario, ver un video o participar en un concurso. Entonces, en un momento dado, la cantidad de usuarios activos estará dada por el número de usuarios que interactuaron con la plataforma en los últimos 7 días.

[24] Se define interacción como cualquiera de las siguientes acciones: compartir un video, visualizar un video, participar en un concurso, realizar un canje de algún producto.

Penetración	Porcentaje de usuarios activos que compartieron el video en cuestión. Para este cálculo, se utilizó la variable active_users en el tiempo, y se comparó con la cantidad de veces que se compartió un video en el mismo intervalo.

Tabla 4.3: Nuevas variables para usuarios

Variable	Descripción
edad	Edad (en años) del usuario. Esta variable se encontraba presente en el grupo de datos como fecha de nacimiento. Posibles valores: Número entero, entre 14 y 64.
sistema_registro	Hace referencia al sistema a través del cual se registró el usuario Posibles valores: *Facebook campaign*: hace referencia a los usuarios que se registraron al sitio a través de una campaña en Facebook. Cabe destacar que no se trata del uso de "Facebook Ads", si no que de un post/campaña a través de la Fan Page de Kikvi. *Normal sign in*: usuarios que se registraron al sitio sin ninguna referencia registrada. En otras palabras, llegaron al sitio y se registraron haciendo *click* en el botón "Registrarse". *Physical campaign – fair*: registros durante la ejecución de la feria de software en la que la empresa participó en sus inicios, bajo otro nombre. *Physical campaign – flyers*: hace referencia a los usuarios que se registraron luego de llegar al sitio a través de una campaña física de *flyers* realizada en universidades de Santiago. *Recruited*: usuarios adquiridos a través del sistema de reclutamiento establecido en el sitio[25]. *Through video display*: registros a través de links dispuestos en la "vitrina"[26]. Esta es la clase más común para esta variable.

[25] El sistema de reclutamiento funciona de la siguiente manera: Un usuario puede motivar a sus contactos a registrarse en la plataforma a través de un *link* personal. Esto tiene como consecuencia que, una vez que la persona reclutada junte 300 puntos o más, se regalan al reclutador 300 puntos.
[26] Vitrina es el nombre utilizado para referirse a la página en la que se muestra un video.

calidad_videos	Calidad de videos en el tiempo comprendido entre una semana antes del registro del usuario y una semana después. Posibles valores: "*High*": alta calidad de videos "*Somewhat high*": calidad de videos relativamente buena "*Regular*": videos regulares "*Somewhat low*": videos de relativamente baja calidad "*Low*": videos de baja calidad
densidad_concursos	Considera la densidad de concursos de las dos semanas siguientes al registro de un usuario. Posibles valores: "*High*": alta calidad de videos "*Somewhat high*": calidad de videos relativamente buena "*Regular*": videos regulares "*Somewhat low*": videos de relativamente baja calidad "*Low*": videos de baja calidad
densidad_videos	Variable que hace referencia a la cantidad de videos en las dos semanas siguientes al registro de un usuario. Posibles valores: "*High*": alta calidad de videos "*Somewhat high*": calidad de videos relativamente buena "*Regular*": videos regulares "*Somewhat low*": videos de relativamente baja calidad "*Low*": videos de baja calidad

Para poder tener un mejor entendimiento del grupo usuario, se definió y/o calculó una serie de variables que podrían ser significativas para explicar el comportamiento y calidad usuaria, listadas en la **tabla 4.3**.

Además, se cambiaron las unidades de una serie de variables con el fin de ser más útiles dentro de su contexto al momento de visualizar los datos.

Durante la etapa de modelado se aplicarán diferentes algoritmos de minería de datos. Algunos de éstos requieren que las variables de ingreso sean exclusivamente discretas, por lo que, en algunos casos, se hace necesaria la categorización de ellas. Para conseguir este objetivo, se utiliza la función *discretize* del paquete *arules* de R.

Se decide utilizar esta función ya que tiene la posibilidad de no dividir la información en rangos fijos, si no que considerando su posición en la escala a la que corresponde y separándolos en *clusters*. Por un lado, esto significa que algunos de los rangos generados podrían estar considerablemente mejor representados que otros, pero además tiene como consecuencia que los valores que se escapan mucho del resto de los datos quedan en sus propios rangos (debido a la separación en *clusters*, lo que podría servir para explicar o descartar de mejor manera relaciones encontradas. En la **tabla 4.4** se presentan las variables categorizadas para el caso de los registros de usuarios.

Tabla 4.4: Variables categorizadas para usuarios

Variable	Cambio aplicado	Posibles valores
puntos_historicos	Se categoriza en diferentes rangos.	*"[0, 4651)"* *"[4651, 18212)"* *"[18212, 46406)"* *"[46406,118093)"* *"[118093,299581]"*
shares_totales	Se categoriza en diferentes rangos.	*"[0.0, 17.0)"* *"[17.0, 68.4)"* *"[68.4,184.3)"* *"[184.3,403.0)"* *"[403.0,542.0]"*
recruitments	Debido a los bajos valores de usuarios reclutados, se decide utilizar esta variable como un indicador binario.	*"1"*: El usuario reclutó 1 o más usuarios. *"0"*: El usuario no reclutó a otros usuarios.
concursos_participados	Se categoriza en diferentes rangos.	*"[0.00, 1.21)"* *"[1.21, 3.37)"* *"[3.37, 5.56)"* *"[5.56, 8.72)"* *"[8.72,16.00]"*
premios_canjeados	De la misma forma que para la variable **recruitments**, se decide utilizar como un indicador binario.	*"1"*: El usuario canjeó 1 o más premios. *"0"*: El usuario no canjeó ningún premio.

tickets_canjeados	Se categoriza en diferentes rangos.	*"[0.00, 7.37)"* *"[7.37, 27.94)"* *"[27.94, 66.36)"* *"[66.36,209.18)"* *"[209.18,459.00]"*
edad	Se categoriza en diferentes rangos.	*"[14.0,21.0)"* *"[21.0,23.6)"* *"[23.6,27.8)"* *"[27.8,36.3)"* *"[36.3,64.0]"*

En la **tabla 4.5** se presentan las diferentes variables categorizadas para el caso de los registros de videos, junto con los posibles valores que cada una de ellas podrá tomar.

Tabla 4.5: Variables categorizadas para videos

Variable	Cambio aplicado	Posibles valores
active_raffles	Se categoriza en diferentes rangos.	*"2 o menos"* *"Entre 3 y 4 "* *"Entre 5 y 6"* *"Entre 7 y 9"* *"10 o más"*
active_users	Se categoriza en diferentes rangos.	*"[0, 40]"* *"(40, 80]"* *"(80, 120]"* *"(120, 160]"* *"Más de 160"*
duracion	Se categoriza en diferentes rangos.	*"30 o menos"* *"(30, 60]"* *"(60, 120]"* *"(120, 180]"* *"(180, 240]"* *"(240, 300]"* *"(300, 600]"* *"600 o más"*
release_difference	Se categoriza en diferentes rangos.	*"Menos de 6 horas"* *"Entre 6 horas y 1 día"* *"Entre 1 y 3 días"* *"Entre 3 días y 1 semana"* *"Entre 1 y 2 semanas"* *"Más de 2 semanas"*

penetracion	Se categoriza en diferentes rangos.	*"Menos del 10%"* *"Entre 10% y 20%"* *"Entre 20% y 30%"* *"Entre 30% y 40%"* *"Entre 40% y 50%"* *"Entre 50% y 60%"* *"Entre 60% y 70%"* *"Entre 70% y 80%"* *"Entre 80% y 90%"* *"Más del 90%"*
total_views	Se categoriza en diferentes rangos.	*"[1, 386)"* *"[386, 1484)"* *"[1484, 4417)"* *"[4417,20842)"* *"[20842,53557]"*
shares_first_day	Se categoriza en diferentes rangos.	*"[0.00, 4.13)"* *"[4.13, 8.14)"* *"[8.14,12.62)"* *"[12.62,19.38)"* *"[19.38,31.00]"*
shares_first_week	Se categoriza en diferentes rangos.	*"[1.00, 7.08)"* *"[7.08,16.19)"* *"[16.19,29.44)"* *"[29.44,47.96)"* *"[47.96,76.00]"*
shares_first_month	Se categoriza en diferentes rangos.	*"[1.00, 5.98)"* *"[5.98, 15.70)"* *"[15.70, 33.73)"* *"[33.73, 70.85)"* *"[70.85,173.00]"*
total_shares	Se categoriza en diferentes rangos.	*"[1.00, 7.52)"* *"[7.52, 19.94)"* *"[19.94, 54.34)"* *"[54.34,123.42)"* *"[123.42,362.00]"*
active_canjes	Se categoriza en diferentes rangos.	*"0.000"* *"[0.693,2.193)"* *"[2.193,3.612)"* *"[3.612,5.112)"* *"[5.112,6.000]"*

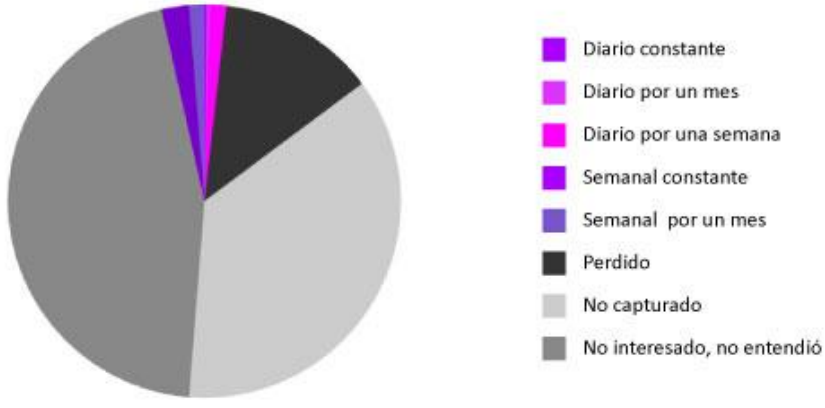
4.2. Modelado

En esta etapa de *CRISP-DM* se procede a aplicar algoritmos y modelos ya existentes y conocidos para extraer información que pueda ser valiosa para el negocio a partir de los datos tratados en el punto anterior. Además, se utilizan herramientas de procesamiento analítico para generar gráficos que pudiesen aportar valor al estudio.

4.2.1. Análisis visual

En esta sección, se presentan algunos gráficos de interés generados con *Tableau*, que a primera vista parecen ser importantes para conocer el negocio. Los gráficos mencionados buscan entender de mejor manera como se distribuyen y comportan los indicadores mencionadas en la **tabla 3.1.**

Ilustración 4.6: Distribución de la calidad usuaria (2669 registros)



En la **ilustración 4.6** se muestra cómo se distribuyen los registros de usuarios en referencia a su calidad. Para facilitar su entendimiento, se han dispuesto tonalidades de color morado para las calidades usuarias consideradas positivas, y

tonalidades de gris para las que son consideradas negativas. Se puede apreciar en la ilustración mencionada que la proporción de usuarios de calidad positiva respecto de calidad negativa es preocupante[27]. Esta distribución extremadamente dispareja también significará que cualquier regla de asociación que se pueda extraer de los datos que haga referencia a calidades positivas de usuarios, tendrá un soporte extremadamente bajo (independiente de la confianza que tenga).

Para poder utilizar técnicas de clasificación sobre este indicador se hace necesario hacer *undersampling*[28] o en su defecto *oversampling*[29] de los datos. Se decide proceder con este último ya que para algunos escenarios la cantidad de registros es muy pequeña, y una submuestra no generará suficientes datos para el estudio. Como criterio de creación de datos, se utilizan todos los valores existentes en registros conocidos de cada clase, y se asigna aleatoriamente uno de ellos al nuevo registro.

Es interesante en el desarrollo de este estudio considerar las variables del ambiente (de la plataforma) en el momento en que se registran los usuarios, para así ver si hay alguna relación, patrón o tendencia marcada para cualquiera de las clases. A continuación, se presenta la relación de dichas variables con indicadores de interés relacionados.

En la **ilustración 4.7** se compara la calidad usuaria con la calidad de los videos (recientes) al momento que dichos usuarios se registraron. En la tabla, un color más intenso hace referencia a una mayor cantidad de registros, lo que además se puede apreciar por el número en cada celda. Debido a la baja cantidad de registros para las calidades positivas, no es posible inferir a simple vista si hay alguna clase de

[27] De forma más exacta, hay 2527 (94.68%) de registros de usuario negativos contra 142 (5.32%) positivos.
[28] Submuestrear, consiste en el proceso de eliminar registros del set de datos con el fin de que todas las clases objetivo tengan una cantidad similar de ocurrencias.
[29] Sobremuestrear, análogo a submuestrear, consiste en generar datos ficticios, acorde a un criterio definido, con el fin de que todas las clases objetivo tengan una cantidad similar de ocurrencias.

patrón con referencia a la calidad de los videos al momento de registro. Por otro lado, para las calidades negativas (últimas tres columnas), pareciera haber mayor peso en los valores de calidades negativas (últimas dos filas).

Ilustración 4.7: Calidad usuaria vs calidad de videos

Calidad Videos	Diario constante	Diario por una semana	Diario por un mes	Semanal por un mes	Semanal constante	Perdido	No capturado	No interesado, no entendió
High	1			6	7		45	74
Somewhat High		8	2	5	16	18	117	123
Regular		21	4	16	26	7	173	241
Somewhat Low	2	5	1	5	8	20	444	519
Low	1	2	1	2	3	382	194	258

En el caso de los usuarios perdidos (sexta columna), para la gran mayoría de los casos la calidad de los videos al momento de registro es mala (*Low*). Además, para las clases de calidad usuaria "No capturado" y "No interesado, no entendió" (últimas 2 columnas) se ve que los valores más comunes corresponden a las calidades de videos regular, relativamente mala y mala, con mayor peso en relativamente mala.

Ilustración 4.8: Calidad usuaria vs densidad de concursos

Densidad Concursos	Diario constante	Diario por una semana	Diario por un mes	Semanal por un mes	Semanal constante	Perdido	No capturado	No interesado, no entendió
High	1	6	2	7	7	60	19	260
Somewhat High	1	4	3	6	10	51	21	192
Regular		6		3	11	34	19	107
Somewhat Low	2	11	2	11	14	117	213	364
Low		9	1	7	18	85	701	284

En la **ilustración 4.8**, siguiendo los niveles y características visuales de la tabla anterior, se presenta la distribución de calidad usuaria en comparación a la

densidad de concursos al momento de registro. Una vez más, no parece haber una tendencia en las calidades de usuario positivas, mientras que en las calidades negativas parece hacer peso por bajas densidades de videos, en especial en el caso de la clase “No capturado”, donde una notable mayoría se registró en un período de baja densidad de concursos (o al menos, relativamente baja). Para el caso de la clase “No interesado, no entendió” la distribución de valores de densidad de concursos parece ser bastante pareja.

Ilustración 4.9: Calidad usuaria vs densidad de videos

Densidad Videos	Diario constante	Diario por una semana	Diario por un mes	Semanal por un mes	Semanal constante	Perdido	No capturado	No interesado, no entendió
High	3	26	6	24	36	75	30	295
Somewhat High	1	6	2	10	23	81	31	243
Regular		3			1	42	32	134
Somewhat Low						56	200	235
Low		1				93	680	300

En la **ilustración 4.9** se aprecia la relación entre la calidad usuaria y la densidad de videos en las dos semanas siguientes a su registro. De forma diferente a las relaciones que se habían revisado hasta el momento, en este caso parece haber una tendencia en las calidades usuarias positivas. Para las cinco primeras columnas (clases consideradas positivas) pareciera que los valores de densidad de videos se concentran en los dos valores más altos: Alta y Relativamente alta. Este patrón parece además reflejarse en la calidad usuaria “Perdido”, aunque en este caso también hay peso considerable, e incluso mayor, en los valores más bajos de densidad de videos. Para el caso de usuarios de clase “No capturado” se ve una fuerte tendencia a los valores más bajos de densidad de videos. Por otro lado, para la clase “No interesado, no entendió” no pareciera haber una relación a primera vista.

En la **ilustración 4.10** se aprecia la relación entre calidad usuaria y sistema de registro. Cabe destacar que, para esta variable, los valores de las clases no tienen ningún tipo de escala. Salta a la vista que para la clase "No interesado, no entendió" el sistema de registro más común es a través de la "vitrina", lo que parece repetirse, para la clase "No capturado, acompañado por el sistema de registro normal. Otra relación que se ve notoriamente es la que comprende a las clases de calidad usuaria "Perdido" y de registro a través de "Reclutamiento". Por otro lado, para las clases positivas de calidad usuaria pareciera haber un peso mayor para los valores de sistema de registro de campaña de Facebook y campaña física de *flyers*, aunque esta tendencia no está muy marcada.

Ilustración 4.10: Calidad usuaria vs sistema de registro

Sistema Registro	Diario constante	Diario por una semana	Diario por un mes	Semanal por un mes	Semanal constante	Perdido	No capturado	No interesado, no entendió
Facebook campaign	1	15	4	18	43	11	193	28
Normal sign in	1	2	1		2	20	258	78
Physical campaign: fair					1	29	113	48
Physical campaign: flyers	2	16	1	15	13	1	101	15
Recruited		2				273	57	24
Through video display		1	2	1	1	13	251	1 014

Hasta el momento se han comparado diferentes variables con la variable de interés **calidad usuaria**. A continuación, se realizarán comparaciones en búsqueda de patrones que pudiesen influir en las variables **penetración** y **usuarios activos**, respectivamente.

En la **ilustración 4.11** se aprecia la distribución de valores presentes para penetración. En el eje horizontal se presentan los valores de penetración aproximada a dos decimales, mientras que en el vertical se refleja la cantidad de registros que cuentan con ese valor. A primera vista la variable parece estar bien distribuida en sus valores posibles, teniendo una baja considerable en sus valores más altos (80% de penetración o más).

Ilustración 4.11: Distribución de videos de acuerdo a su penetración

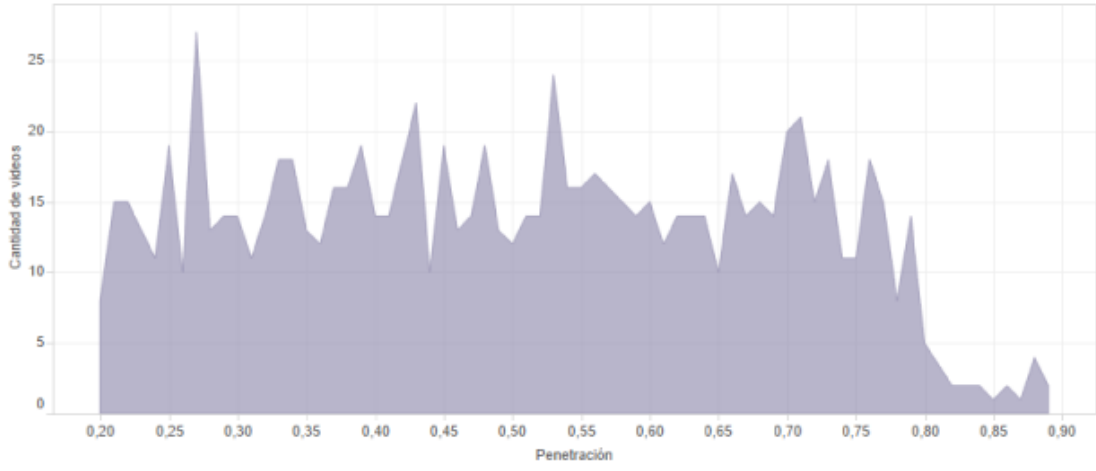
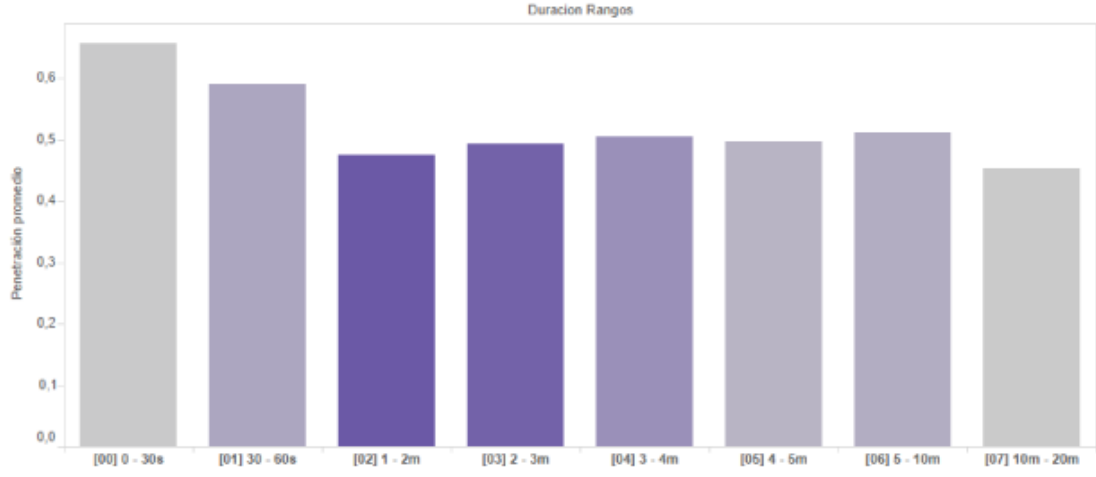


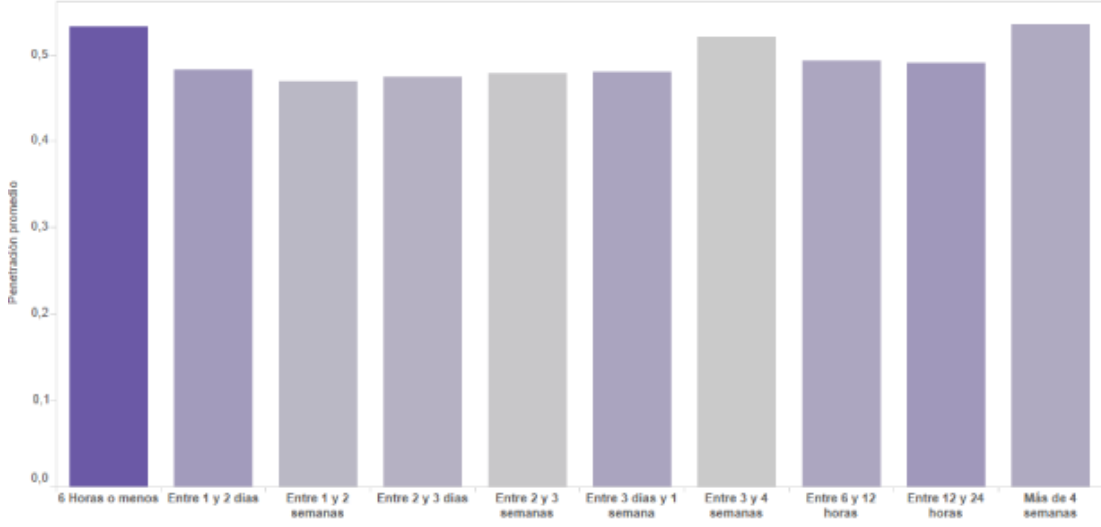
Ilustración 4.12: Penetración vs duración



En la **ilustración 4.12** se refleja la relación entre la duración de un video en rangos y la penetración promedio del mismo en ese rango. El color representa la cantidad de registros que ese rango representa (a mayor intensidad, mayor cantidad de registros). Como es de esperarse, y a pesar de que la tendencia no parece muy marcada, pareciera que un video tiene mejor penetración a menor duración.

Se teoriza que es importante ser de las primeras plataformas en difundir un contenido en particular para conseguir un buen recibimiento del grupo usuario, traducido en una penetración alta. En la **ilustración 4.13**, se busca entender una posible relación entre la diferencia de lanzamiento[30] y la penetración de un video. El eje horizontal representa la diferencia de lanzamiento como variable independiente, y el vertical la penetración promedio de ese segmento en particular. A simple vista no pareciera haber una diferencia muy marcada entre los diferentes rangos de lanzamiento, aunque el primer rango se nota considerablemente mejor representado que el resto de los rangos (la opacidad del color del gráfico hace referencia a la cantidad de datos que pertenecen en cada segmento).

Ilustración 4.13: Penetración contra diferencia de lanzamiento



Como exploración adicional, se decide estudiar conjuntamente la influencia de ambas variables en la penetración. En la **ilustración 4.14** se aprecia la exploración propuesta. En una primera interpretación pareciera que los videos en el primer rango de diferencia de lanzamiento y primeros rangos de duración alcanzan una mejor penetración. Llama la atención la penetración considerablemente marcada del primer

[30] Diferencia de tiempo entre que el contenido es publicado en YouTube en comparación a su publicación en la plataforma.

rango de duración (entre 0 y 30 segundos) y la diferencia de lanzamiento *"Entre 1 y 2 semanas"*, pero al revisar los datos detrás de estos rangos se descubre que hay un único registro en ellos, concluyéndose que se trata de un *outlier*.

Ilustración 4.14: Relación penetración vs diferencia de lanzamiento y duración

Duracion Rangos	6 Horas o menos	Entre 6 y 12 horas	Entre 12 y 24 horas	Entre 1 y 2 dias	Entre 2 y 3 dias	Entre 3 dias y 1 semana	Entre 1 y 2 semanas	Entre 2 y 3 semanas	Entre 3 y 4 semanas	Más de 4 semanas
[00] 0 - 30s	0,7317	0,5600	0,5650	0,3900		0,6350	0,8000			
[01] 30 - 60s	0,8097	0,3775	0,4750	0,4900	0,5817	0,4354	0,3800	0,5200	0,4900	0,5100
[02] 1 - 2m	0,4793	0,4935	0,4793	0,4463	0,4506	0,4583	0,4633	0,4100	0,6450	0,5158
[03] 2 - 3m	0,5020	0,4333	0,4728	0,5187	0,4645	0,4616	0,4123	0,5550	0,5000	0,5766
[04] 3 - 4m	0,5279	0,5568	0,5016	0,4540	0,4908	0,5119	0,4982	0,5180	0,4900	0,4525
[05] 4 - 5m	0,4878	0,4850	0,5143	0,5067	0,4267	0,6233	0,4600		0,4600	0,5550
[06] 5 - 10m	0,4656	0,5356	0,5287	0,5464	0,4633	0,5500	0,5317	0,2850	0,5800	0,5475
[07] 10m - 20m	0,5060	0,2700		0,4100				0,5100		

A continuación, se busca encontrar alguna relación entre la cantidad de usuarios activos y variables. Para esto, se considera cada publicación de video como un instante de estudio (efectivo debido a que la publicación de videos durante la ventana de tiempo correspondiente a la base de datos de estudio es periódica[31]). Se ve en las **ilustraciones 4.15 y 4.16** la relación entre la cantidad de usuarios activos promedio para estos instantes de tiempo en comparación a la cantidad de concursos[32] y canjes[33] activos, respectivamente. En ambos casos se puede apreciar una posible

[31] Se publican videos repetidas veces a la semana, sin dejar pasar múltiples días sin nuevos videos (excepto en excepciones pequeñas)

[32] Un concurso es un proceso en el que se pueden cambiar puntos adquiridos en la plataforma por *tickets* que dan la posibilidad de ganar algún premio en particular. Se sortea el premio entre todos los *tickets* comprados para ese evento en particular, lo que significa una mayor probabilidad entre mayor sea la cantidad de tickets canjeados por un usuario. El costo en puntos de un concurso suele ser bajo.

[33] Un canje (o canje directo) es un intercambio de puntos adquiridos en la plataforma por un producto o servicio en particular. El cambio es inmediato y la entrega de productos dentro de una semana de tiempo. El costo de un canje es alto en comparación a la ratio de adquisición de puntos posibles en la plataforma, considerablemente más alto que el costo de participar en un concurso.

relación directa entre la cantidad de usuarios activos y las variables en cuestión, siendo aún más notoria esta posible relación para el caso de los concursos activos.

Ilustración 4.15: Usuarios activos vs concursos activos

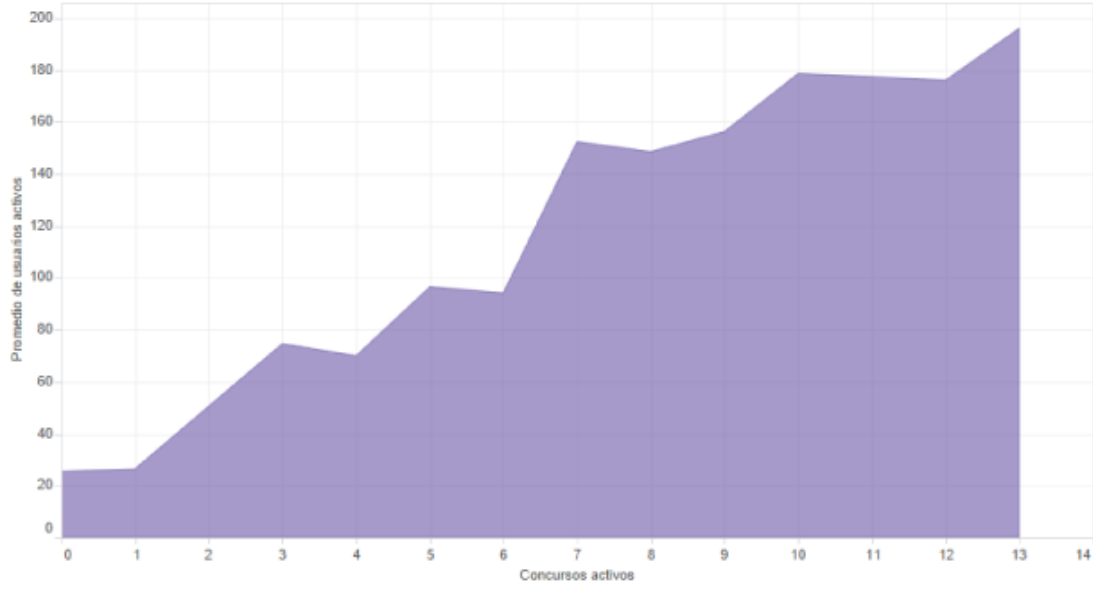
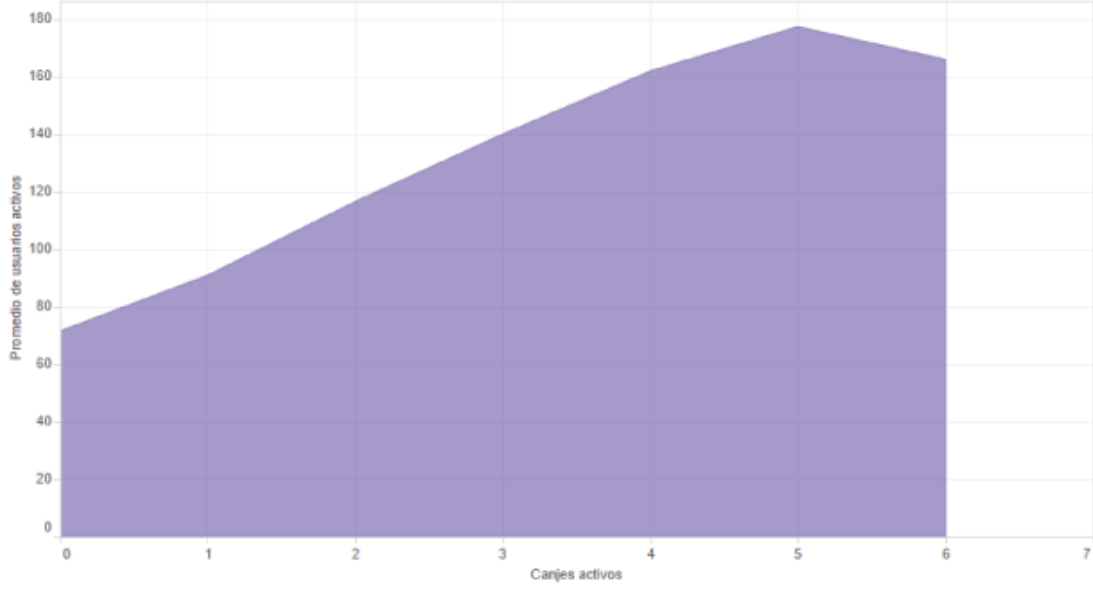


Ilustración 4.16: Usuarios activos vs canjes activos



4.2.2. Reglas de clasificación

En primer lugar, decide utilizar *Apriori*, debido a su simpleza, y a que suele dar un buen punto de partida para saber qué variables podrían estar influyendo directamente en los indicadores de interés de este estudio. Como se mencionó en capítulos anteriores, se usa *R* para este objetivo, utilizando la implementación del algoritmo del paquete *arules*.

Ya que las variables fueron discretizadas en el punto anterior, se procede a aplicar el algoritmo mencionado. Para disminuir la cantidad de resultados, se decide utilizar el parámetro *appearance*. Este parámetro limita la cantidad de variables y valores que son considerados en uno o ambos lados de las relaciones encontradas, y así reducir considerablemente las reglas resultantes de la ejecución para facilitar su análisis. Cabe destacar que esta limitante no se realiza durante la ejecución, si no únicamente al momento de mostrar los resultados. En otras palabras, se filtran los resultados automáticamente para conseguir las relaciones que tengan en la mano derecha a las variables de interés mencionadas en capítulos anteriores. La lista de *appearance* se define como:

```
apriori_users_appeareance_list = list(
  rhs = c(
    "<indicador>=<clase 1>",
    "<indicador>=<clase 2>",
    …
    "<indicador>=<clase n>",
  ),
  default = "lhs"
)
```

Esto quiere decir que se mostrarán relaciones que tengan a mano derecha (*rhs: righ hand side*) los diferentes *valores* ***clase*** para el indicador *indicador*, mientras que

a mano izquierda (*lhs: left hand side*) se mantendrán todos los resultados encontrados (funcionamiento por defecto). Al aplicar esta limitante con respecto a lo que se busca en la mano derecha de la regla, ésta pasa a ser una regla de clasificación.

Como se revisó anteriormente en la **ilustración 4.6**, sólo el 5.32% de los datos corresponden a registros de calidades usuarias positivas, por lo que ninguna regla de asociación que pudiese encontrarse tendrá mayor soporte a ese valor. Entrando en detalle en estos registros, la cantidad total de datos de cada clase positiva de calidad usuaria se aprecia en la **tabla 4.17**. De acuerdo a esta información, se sabe de antemano que no se encontrarán reglas con soporte mayor al 2.25%. Se decide finalmente descartar el uso de *apriori* para encontrar reglas de asociación para estas clases.

Tabla 4.17: Cantidad de registros de calidades usuarias positivas

Clase	Cantidad de registros (porcentaje de la muestra total)
Daily, for a month	8 registros (0.3%)
Daily, for a week	36 registros (1.35%)
Daily, constant	4 registros (0.15%)
Weekly, for a month	34 registros (1.27%)
Weekly, constant	60 registros (2.25%)

Por otro lado, la cantidad de registros que cuentan con calidades usuarias negativas es del 94.68%. En la **tabla 4.18** se presenta el total de registros de cada clase negativa.

Tabla 4.18: Cantidad de registros de calidades usuarias negativas

Clase	Cantidad de registros (porcentaje de la muestra total)
Not interested/Didn't get it	1207 registros (45.22%)
Not captured	973 registros (36.46%)
Lost	347 registros (13%)

Ya que la mayoría de las variables hace referencia a la actividad que los usuarios tienen con la plataforma, se espera poca información de las reglas cuyo *rhs* contenga el valor *"Not interested/Didn't get it"*, ya que estos usuarios, por definición, no interactuaron con la plataforma. Se decide entonces buscar reglas de asociación que contengan en su mano derecha sólo las clases *"Not captured"* y *"Lost"*; en la **tabla 4.19** se presentan las reglas encontradas para tal efecto. Cabe destacar que éstas no son las únicas reglas resultantes de la ejecución del algoritmo, pero el resto de ellas eran combinaciones de las diferentes variables dependientes de la interacción, donde todos los casos hacían referencia a los primeros intervalos de valores, lo que es de esperarse de grupos de usuarios no capturados y perdidos.

Las reglas encontradas concuerdan con la información visualizada en la primera etapa del proyecto, por lo que no se descubre nuevo conocimiento a durante este proceso.

Luego se procede a aplicar este algoritmo sobre la base de datos de videos. De la misma forma que para la de usuarios, se buscan las reglas cuyo *rhs* valores relacionadas con los indicadores de interés. Lamentablemente, no se encuentran reglas que aporten valor a los objetivos del estudio.

Tabla 4.19: Reglas de clasificación para clases *Not captured* y *Lost*, ordenadas por clase y *lift*

Clase	Mano izquierda	Mano derecha	Soporte	Confianza	Lift
Not captured	densidad_videos=Low	quality=Not captured	25.48%	63.31%	1.7
	densidad_concursos=Low	quality=Not captured	26.26%	63.44%	1.7
Lost	sistema_registro=Recruited	quality=Lost	10.29%	76.69%	5.9
	premios_canjeados=0, sistema_registro=Recruited	quality=Lost	9.98%	76.08%	5.9
	recruitments=0, sistema_registro=Recruited	quality=Lost	10.04%	76.35%	5.8
	recruitments=0, premios_canjeados=0, sistema_registro=Recruited	quality=Lost	9.7%	75.73%	5.8

4.2.3. Técnicas de clasificación

En segunda instancia, se decide utilizar técnicas de clasificación, para así poder definir un modelo que se adecue de la mejor manera posible a cada uno de los indicadores claves del estudio. Se implementan y comparan los algoritmos: árboles de clasificación[34], *random forest*[35], *naive bayes*[36] y máquinas de vectores de soporte[37]. Finalmente se compara el desempeño de cada algoritmo utilizando curvas matrices de confusión y curvas ROC.

Durante la generación de clasificadores se decide separar los datos en subgrupos de entrenamiento y prueba: 70% de los datos se utilizó para entrenamiento y un 30% para pruebas en cada uno de los casos.

[34] Paquete *rpart*
[35] Paquete *randomForest*
[36] Paquete *e1071*
[37] Paquete *e1071*

Clasificación de registros con objetivo de usuarios activos

Como el indicador de usuarios activos es numérico, se hace necesario definir previamente qué valores de éste serán considerados exitosos. Al no encontrarse referencias previas para el dominio del negocio, se decide definir sobre 150 usuarios activos como un caso de éxito. Esta métrica dependerá de la situación actual del negocio.

En la **ilustración 4.20** se muestra el árbol resultante al utilizar el algoritmo de generación de árboles de clasificación *rpart*. En cada una de las hojas del árbol, el primer número representa la probabilidad de que un registro en ella no sea exitoso, o sea, que tenga menos de 150 usuarios activos. Análogamente, el segundo valor representará la probabilidad de éxito en dicha hoja. Finalmente, el porcentaje en estos nodos hace referencia a la cantidad de datos que éste representa. En la **ilustración 4.21** se muestra gráficamente una explicación de la información presente en las hojas del árbol de decisión.

Ilustración 4.20: Árbol de clasificación para usuarios activos, *rpart*

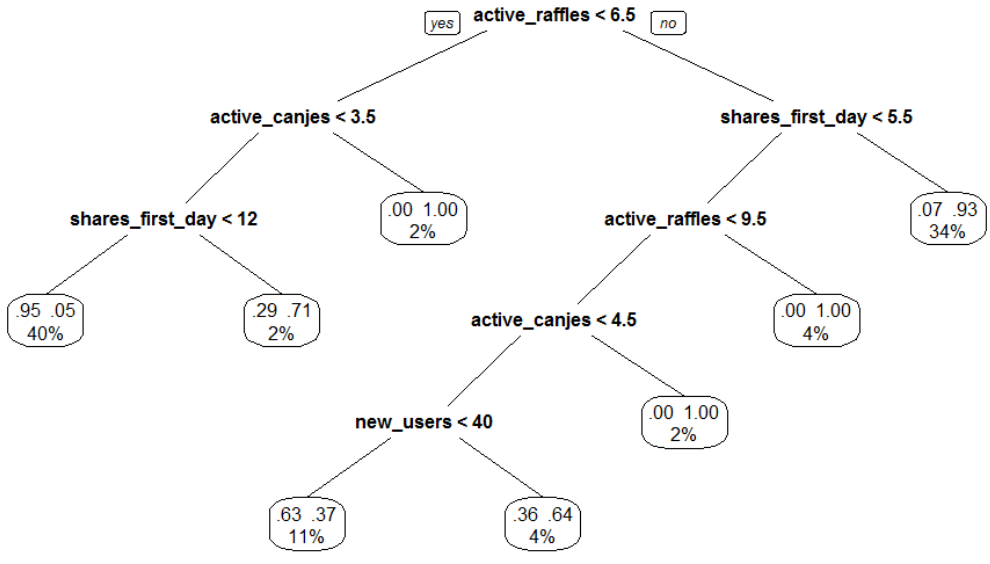
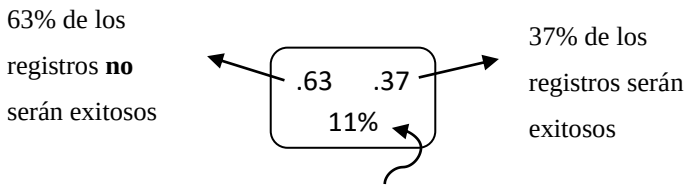


Ilustración 4.21: Explicación ejemplificada de árbol de decisión



De acuerdo al árbol encontrado, las variables *active_raffles*[38], *shares_first_day*[39], *active_canjes*[40] y *new_users*[41] parecen ser decisivas al momento de determinar la cantidad de usuarios activos de la plataforma en un momento dado. Saltan a la vista hojas que representan entre 1% y 4% de los registros, por lo que se decide podar el árbol y eliminar estas hojas debido a su poca significancia en el conjunto completo de registros. En la **ilustración 4.22** se presenta el árbol de clasificación podado para el indicador de usuarios activos. En él se aprecia que sólo dos variables permanecen luego del proceso, *active_raffles* y *active_canjes*, explicando de buena manera la totalidad de los datos con muy bajo error. Se decide dejar la hoja correspondiente al caso de *active_canjes >= 3.5*, ya que hay muy pocos registros que cumplen con esta condición, y esto hace inevitable que la cantidad de ellos representados por la condición sea poca.

En segunda instancia, se aplica el algoritmo *naiveBayes* para generar un clasificador para el indicador de usuarios activos. En la **tabla 4.23** se presenta la tabla de resultado generada por el algoritmo. En ella, se listan el promedio y desviación estándar de cada variable, para cada clase objetivo. Entonces, dada una desviación estándar pequeña en comparación al promedio, una variable cuyo promedio sea significativamente diferente para cada clase objetivo, se considerará significativa al momento de clasificar un registro. En base a esto, y acorde a lo

[38] Cantidad de concursos activos
[39] Cantidad de veces que se comparte un video en su primer día
[40] Cantidad de canjes activos
[41] Cantidad de usuarios nuevos

encontrado al analizar el árbol de este indicador generado con el algoritmo *rpart*, las variables *active_raffles*, *shares_first_day*, *active_canjes* y *new_users* parecieran ser importantes para el clasificador.

Ilustración 4.22: Árbol de clasificación para usuarios activos (*rpart*, podado)

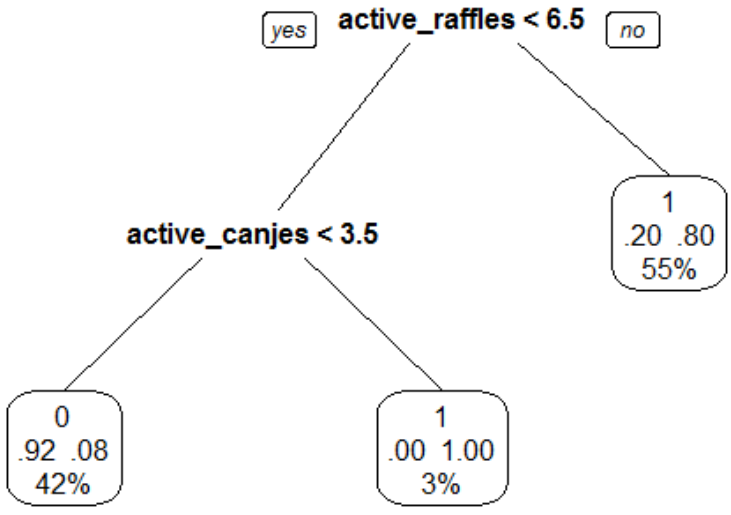


Tabla 4.23: Promedio y desviación estándar de variables en clasificador de *naiveBayes* para usuarios activos

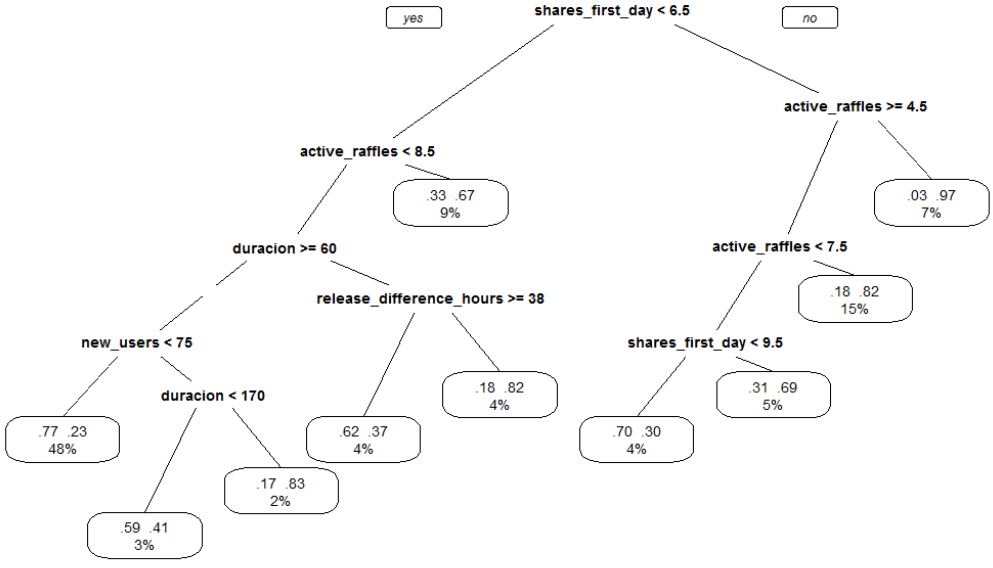
Variable	Promedio		Desviación estándar	
	0	1	0	1
duracion	160.68	162.86	115.49	120.31
shares_first_day	3.64	10.30	3.17	6.84
new_users	37.50	74.21	56.69	58.41
active_raffles	4.77	8.94	2.66	2.38
release_difference	322.88	1170.87	2158.52	3595.60
active_canjes	1.28	0.98	3.02	2.00

Luego se aplican el resto de los algoritmos mencionados, los que no generan apoyo visual del clasificador, pero cuya efectividad será comparada en el siguiente capítulo.

Clasificación de registros con objetivo de penetración

Continuando con los indicadores de interés, se repite el proceso de generación de clasificadores. Nuevamente se trata de un indicador numérico, por lo que se separa en categorías que serán consideradas provechosas o no para el negocio. En este caso, se define un caso de éxito como un video que alcanza más del 70% de penetración.

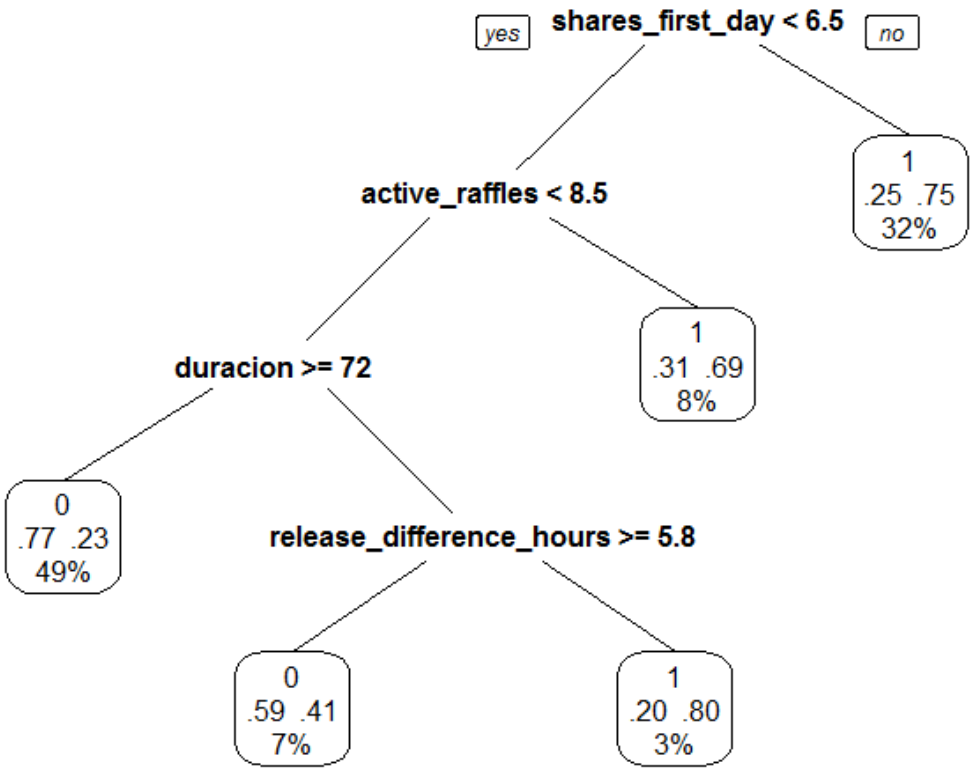
Ilustración 4.24: Árbol de clasificación para penetración (*rpart*)



En la **ilustración 4.24** se presenta el árbol de decisión resultante para el caso del indicador de penetración de video. Para este caso, las variables que influyen de forma directa con la clase objetivo son *active_raffles*, *shares_first_day*, *duración*, *new_users* y *release_difference*. Al igual que para el indicador anterior, se decide

podar el árbol con el fin de remover hojas cuya cantidad de registros es muy pequeña en comparación al total, el que se aprecia en la **ilustración 4.25**. Para este caso, se mantienen las mismas variables que en el caso anterior, con excepción de la *new_users*. A pesar de que en las hojas del último nivel la representación es de un 3% y un 7% de los datos, se decide no podarlas ya que el juntarlas significaría una representación del 10% de los datos, pero con una probabilidad cercana al 50% de cada clase, lo que sacrificaría mucho la capacidad del árbol como predictor en esos casos.

Ilustración 4.25: Árbol de clasificación para penetración (*rpart,* podado)



Para el clasificador generado por el algoritmo *naiveBayes* se presenta en la **tabla 4.26**, al igual que para el indicador anterior, la lista de promedios y

desviaciones estándar para cada variable utilizada para construir el mismo. De ella se deduce que la variable *shares_first_day* es importante para el clasificador.

Tabla 4.26: Promedio y desviación estándar de variables en clasificador de *naiveBayes* para penetración

Variable	Promedio		Desviación estándar	
	0	1	0	1
duracion	168.19	155.19	114.06	133.19
shares_first_day	3.87	8.29	3.49	6.21
new_users	40.97	58.11	58.05	69.82
active_raffles	5.34	6.59	2.69	3.54
release_difference	529.74	803.61	2471.59	3796.20
active_canjes	1.38	1.85	1.13	1.19

Al igual que para el indicador interior, el resto de los algoritmos no generan información relevante para esta sección del estudio, por lo que se revisarán más a fondo en secciones posteriores.

Clasificación de registros con objetivo de calidad usuaria

El último de los indicadores a abordar con técnicas de clasificación es el de calidad usuaria. En este caso, la variable objetivo cuenta con 7 valores posibles, por lo que el árbol resultante no es tan claro como en los casos anteriores. En las hojas se lista cada una de las posibles clases, junto con la probabilidad de que un registro pertenezca a cada una de ellas, además del porcentaje total de observaciones que corresponden a dicha hoja.

En la **ilustración 4.27** se muestra el árbol resultante para el indicador de calidad usuaria. A primera vista, las variables que influyen en él son *calidad_videos*, *recruitments*, *concursos_participados*, *premios_canjeados*, *edad*, *sistema_registro* y *densidad_videos*.

Para simplificar el análisis, se decide agrupar las calidades usuarias en calidades positivas y negativas. Como se ha referenciado en secciones anteriores, el agrupamiento se realiza de la siguiente manera:

- Clases positivas de calidad usuaria: diario constante, diario semanal, diario mensual, semanal mensual y semanal constante.

- Clases negativas de calidad usuaria: perdido, no capturado, no interesado/no comprendió.

En base a esta nueva forma de clasificación, se genera nuevamente el árbol de clasificación para el indicador. En la **ilustración 4.28** se presenta el nuevo árbol obtenido, esta vez sólo con dos clases posibles en el indicador objetivo. En este caso, se ve una fuerte dependencia de las variables *recruitmets*, *concursos_participados*, y *calidad_videos*. La calidad usuaria como indicador no se trata de una variable que haya estado definida en el grupo de datos original, si no que fue calculada en base al resto de ellas. No todas fueron utilizadas en el cálculo, pero como las mismas que sí lo fueron ahora se están considerando al momento de generar el clasificador, se espera una altísima, y sesgada, eficacia del mismo. Se decide entonces realizar un segundo grupo de clasificadores, para todos los algoritmos, dejando de lado las variables: *concursos_participados, premios_canjeados* y *recruitments*, ya que ellas fueron utilizadas el momento de definir el indicador. Por otro lado, hay hojas de este árbol que cuentan con una muy baja representatividad en comparación al total de los datos. Se intenta podar el árbol, pero los resultados obtenidos son muy inferiores, dejando únicamente dos niveles de clasificación. Se decide no podar el árbol y mantener la granularidad que ya posee.

Ilustración 4.27: Árbol de clasificación para calidad usuaria

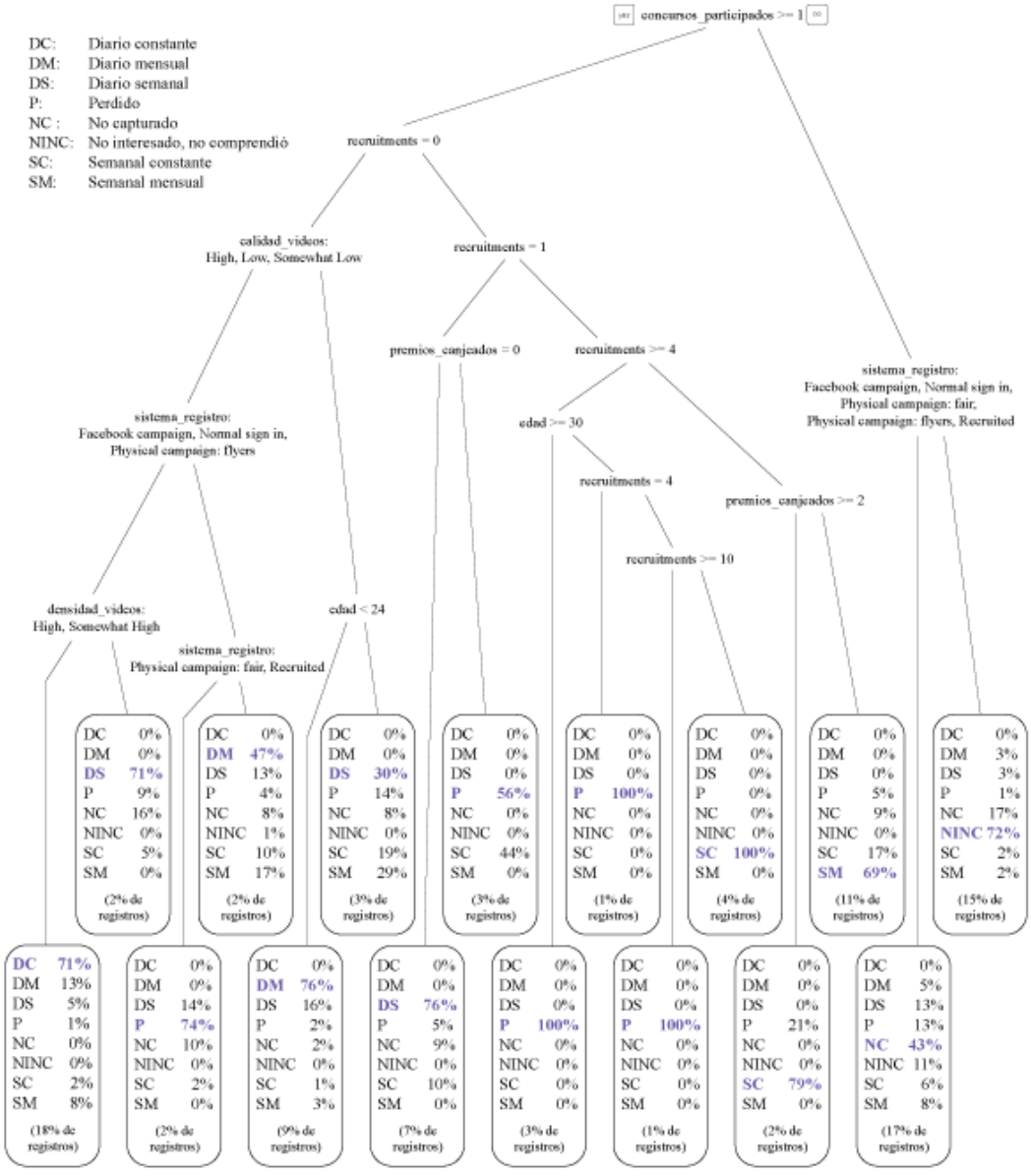
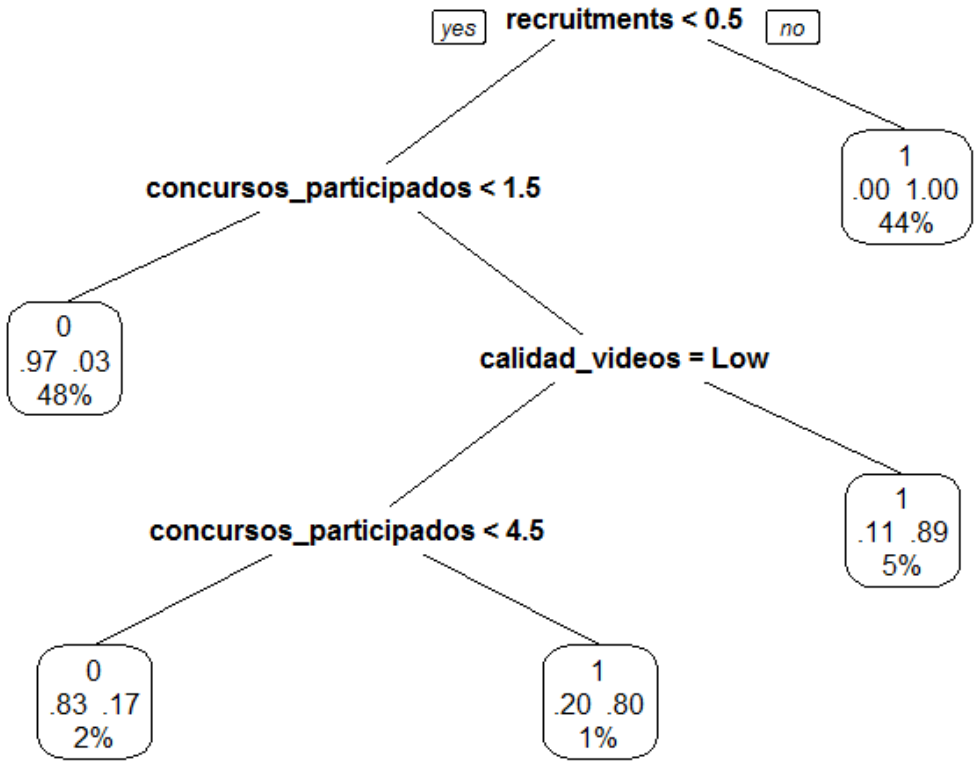


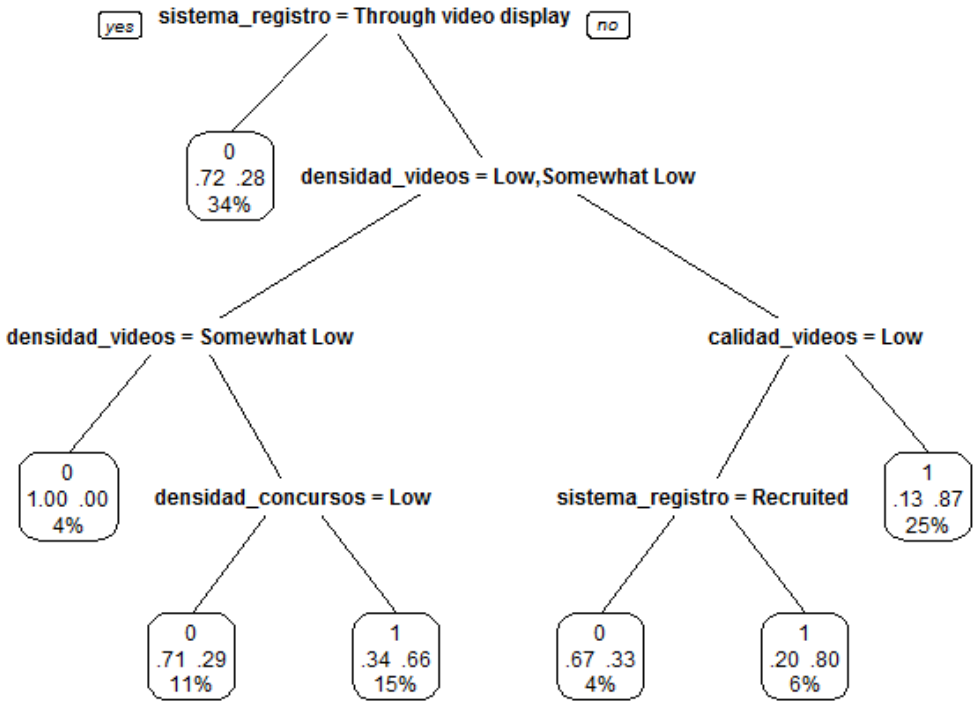
Ilustración 4.28: Árbol de decisión (*rpart*) para calidad usuaria simplificada



En la **ilustración 4.29** se presenta el árbol de clasificación generado por la ejecución del algoritmo *rpart* para el indicador de calidad usuaria, dejando de lado las variables que se sabe fueron utilizadas en el cálculo del mismo. En este nuevo árbol se aprecian *sistema_registro*, *densidad_videos*, *calidad_videos* y *densidad_concursos*. Al analizar este nuevo árbol generado se puede deducir que un sistema de registro a través de vitrina[42] es probable que genere un usuario de calidad negativa. Además, que una densidad de videos regular o mejor tiene como consecuencia una probable calidad usuaria positiva.

[42] *Through video display*

Ilustración 4.29: Árbol de decisión (*rpart*) para calidad usuaria simplificada y filtrada



Para los clasificadores generado por el algoritmo *naiveBayes* se presentan en las **tablas 4.30(a,b,c,d,e,f)**, los resultados generados por el algoritmo para el caso que considera todas las variables y el que se encuentra filtrada, para ambos casos se mantienen los mismos valores, aunque algunas variables no se encuentran en el segundo caso. En ellas se lista el promedio y desviación estándar para el caso de variables numéricas, y la probabilidad condicional dada la clase objetivo en el caso de las variables categóricas.

Se puede rescatar de estas tablas que la edad o el género no son significativos en la construcción de este clasificador. Por otro lado, un registro *Through video display* pareciera influir negativamente en la calidad usuaria, junto con los niveles inferiores de calidades de video y densidad de concursos.

Tabla 4.30a: Resultado de *naiveBayes* para calidad usuaria simplificada, variables numéricas

Variable	Promedio		Desviación estándar	
	0	1	0	1
recruitments[43]	0.01	3.37	0.31	2.82
concursos_participados[43]	0.19	6.89	0.76	4.73
premios_canjeados[43]	0.01	1.68	0.10	1.65
edad	25.08	25.90	6.19	4.06

Tabla 4.30b, 4.30c, 4.30d, 4.30e, 4.31f: Resultado de *naiveBayes* para calidad usuaria, variables categóricas

Variable: genero Valor	Calidad negativa (0)	Calidad positiva (1)
F	0.48	0.52
M	0.51	0.49

Variable: sistema_registro Valor	Calidad negativa (0)	Calidad positiva (1)
Facebook capaign	0.09	0.12
Normal sign in	0.15	0.18
Physical campaign: fair	0.08	0.10
Physical campaign: flyers	0.05	0.21
Recruited	0.14	0.20
Through video display	0.50	0.19

Variable: densidad_videos Valor	Calidad negativa (0)	Calidad positiva (1)
Low	0.43	0.33
Somewhat Low	0.19	0.00
Regular	0.09	0.15
Somewhat High	0.14	0.33
High	0.16	0.20

[43] Esta variable no se considera para el caso filtrado

Variable: densidad_concursos Valor	Calidad negativa (0)	Calidad positiva (1)
Low	0.42	0.26
Somewhat Low	0.28	0.25
Regular	0.07	0.24
Somewhat High	0.11	0.13
High	0.13	0.13

Variable: calidad_videos Valor	Calidad negativa (0)	Calidad positiva (1)
Low	0.30	0.23
Somewhat Low	0.39	0.24
Regular	0.17	0.15
Somewhat High	0.10	0.26
High	0.05	0.12

4.3. Evaluación

En esta etapa de *CRISP-DM* se procede a evaluar los modelos obtenidos en la etapa anterior, compararlos entre sí y concluir en base al modelo que sea más adecuado para cada indicador.

4.3.1. Reglas de clasificación

En la **tabla 4.19**, se presentaron las reglas de clasificación encontradas para las calidades usuarias *No capturado* y *perdido*. De ellas se extrae que, para la primera de estas clases, malos valores de densidad de videos y concursos serán una característica común, con un soporte de 25% y confianza de 63%, aproximadamente. Por otro lado, para el caso de los usuarios perdidos (clase *Perdido*), los soportes son preocupantemente bajos para concluir decisivamente, (alrededor del 10% para cada

una de las cuatro reglas encontradas), aunque, por otro lado, cada una de estas reglas se encuentra relacionada con una o más de las mismas combinaciones variable=valor (*sistema_registro=Recruited*, *premios_canjeados=0*, *recruitments=0*), y todas ellas tienen una confianza superior al 75%.

4.3.2. Técnicas de clasificación

En esta sección se evaluar individualmente cada uno de los clasificadores generados en la etapa anterior. Se analizarán los resultados y comparaciones por cada indicador objetivo, ya que la efectividad de cada algoritmo depende del escenario en el que se esté aplicando.

Clasificación de indicador de usuarios activos

Para la ejecución del algoritmo ***rpart*** es importante, antes de concluir en torno a un árbol de decisión, saber qué tan bueno es prediciendo resultados de la clase objetivo. En este marco, se presenta en la **tabla 4.31** la matriz de confusión generada al evaluar el árbol relacionado con usuarios activos con el grupo de pruebas para este caso.

Tabla 4.3: Matriz de confusión para árbol de decisión (*rpart*) de usuarios activos

	No exitoso, predicción	Exitoso, predicción
No exitoso, original	215	43
Exitoso, original	13	191

En base a estos valores, se calculan métricas importantes para el caso de los clasificadores, que hacen referencia a su calidad: precisión, sensibilidad y especificidad. La precisión se define como la probabilidad de que una predicción sea

correcta; la sensibilidad, como la probabilidad de detectar un registro de clase exitosa en un universo de registros exitosos y la especificidad como la habilidad de detectar un registro no exitoso en un universo de datos no exitosos. Cada una de estas métricas se calcula de acuerdo a las siguientes formulas *[6]*:

$$Precisión = \frac{(Verdaderos\ positivo + Verdaderos\ falsos)}{(Universo\ positivo + Universo\ negativo)}$$

$$Sensibilidad = \frac{Verdaderos\ positivos}{Universo\ positivo}$$

$$Especificidad = \frac{Verdaderos\ negativos}{Universo\ negativo}$$

Se concluye que el árbol generado es un método predictivo suficientemente bueno, teniendo aproximadamente un 83% de especificidad, un 94% de sensibilidad, y una precisión del 88%.

Teniendo en consideración las variables que, para el árbol dispuesto en la **ilustración 4.21**, definen al indicador de usuarios activos, es interesante definir en qué magnitud influyen en el resultado final de dicho indicador. Para este objetivo se utiliza una herramienta presente en el paquete utilizado, llamada *variable.importance*, que da un acercamiento a qué tan importante es cada variable en comparación al resto. El resultado de esta función se aprecia en la **tabla 4.33**, donde a mayor valor, mayor la importancia al momento aplicar el predictor.

Como era de esperarse por visualizaciones anteriores, la cantidad de concursos activos (*active_raffles*) juega un papel importante al momento de definir la cantidad de usuarios activos en un momento dado en la plataforma.

Tabla 4.32: Ranking de importancia de variables en árbol (*rpart*) de usuarios activos

Variable	Importancia relativa
active_raffles	259.13
active_canjes	146.42
shares_first_day	113.45
new_users	68.75
release_difference	36.88
duracion	3.79

En la **tabla 4.33** se presenta la nueva matriz de confusión para el clasificador generado por el algoritmo *rpart*, luego de ser podado. De ella se desprende que este tendrá aproximadamente un 96% de especificidad, un 82% de sensibilidad, y una precisión del 88%. Se concluye que el árbol podado será mejor que su contraparte sin poda al momento de predecir un registro no exitoso, pero no así en el caso de uno que si lo es.

Tabla 4.33: Matriz de confusión para árbol de decisión podado (*rpart*) de usuarios activos

	No exitoso, predicción	Exitoso, predicción
No exitoso, original	178	7
Exitoso, original	50	227

Luego, se presenta en la **tabla 4.34** el ranking de importancia de variables para el árbol podado de usuarios activos. Se aprecia que, a pesar de que los valores de la importancia relativa cambian, el orden del ranking sigue siendo el mismo.

Tabla 4.34: Ranking de importancia de variables en árbol podado (*rpart*) de usuarios activos

Variable	Importancia relativa
active_raffles	224.80
active_canjes	127.83
shares_first_day	62.16
new_users	53.39
release_difference	31.38
duracion	2.34

Siguiendo con la evaluación de los algoritmos aplicados, se presenta en la **tabla 4.35** la matriz de confusión para la ejecución de ***randomForest***. Para este caso, se aprecian una especificidad del 87%, una sensibilidad del 89% y una precisión del 88%, aproximadamente. De la misma forma que para el clasificador anterior, se presenta en la **tabla 4.36** la lista de importancia de cada variable en este escenario. La columna "Clase: no exitoso", hace referencia a qué tan importante es esta variable al momento de predecir un registro de clase "no exitoso", o sea que, a mayor valor, más importante es la variable; análogamente, la columna "Clase: exitoso" hará referencia a la clase "exitoso". *Mean Decrease Accuracy* hará referencia a qué tanto debe considerarse esta variable en el predictor con el fin de disminuir errores de clasificación. Finalmente, *Mean Descrease Gini* hace referencia a cómo disminuye la inequidad en la clasificación, valores más altos son mejores, ya que una inequidad baja significa una variable que en particular juega un papel más importante al dividir los datos en clases definidas. A modo de resumen, para la tabla de importancia de variables de *randomForest*, mayores valores se traducen en una variable más

importante al momento de clasificar. Nuevamente *active_raffles* destaca como la variable más significativa.

Tabla 4.35: Matriz de confusión para clasificador de *randomForest* de usuarios activos

	No exitoso, predicción	Exitoso, predicción
No exitoso, original	202	30
Exitoso, original	26	204

Tabla 4.36: Ranking importancia de variables en clasificador de *randomForest* de usuarios activos

Variable	Clase: no exitoso	Clase: exitoso	*Mean Decrease Accuracy*	*Mean Decrease Gini*
active_raffles	48.37	23.86	46.87	180.3
active_canjes	26.61	14.19	26.59	86.71
shares_first_day	22.84	11.27	23.89	105.32
new_users	14.19	0.62	13.19	69.53
release_difference	7.53	4.26	7.96	51.09
duracion	2.3	2.68	3.74	39.08

El siguiente clasificador evaluado es el generado por la ejecución del algoritmo *naiveBayes*. En la **tabla 4.37** se presenta la matriz de confusión relacionado con este clasificador, se desprende de ella que el clasificador tendrá una especificidad del 83%, una sensibilidad del 91% y una precisión del 87%, aproximadamente. El algoritmo utilizado para la generación del clasificador de *naiveBayes* no cuenta con herramientas incorporadas para definir la importancia de las variables en cada caso.

Tabla 4.37: Matriz de confusión para clasificador de *naiveBayes* de usuarios activos

	No exitoso, predicción	Exitoso, predicción
No exitoso, original	209	42
Exitoso, original	19	192

Para el caso de máquinas de vectores de soporte (*SVM* por su sigla en inglés[44]), se generaron dos clasificadores diferentes: uno utilizando como núcleo separadores lineales, y otro usando separadores radiales. Este parámetro tiene influencia directa en el clasificador final, variando de caso a caso qué núcleo representa de mejor manera al conjunto de datos en cuestión.

Para el caso de *SVM lineal*, se presenta su matriz de confusión en la **tabla 4.38**. De ésta se desprende que este clasificador tendrá una especificidad del 85%, una sensibilidad del 92% y una precisión del 88%, aproximadamente. Luego, en la **tabla 4.39**, se presenta la matriz de confusión para el caso de *SVM radial*, de donde se calcula un 85% de especificidad, un 93% de sensibilidad y un 89% de precisión, aproximadamente. La ejecución de este algoritmo no genera elementos de importancia de variables.

Tabla 4.38: Matriz de confusión para clasificador de *SVM lineal* de usuarios activos

	No exitoso, predicción	Exitoso, predicción
No exitoso, original	211	38
Exitoso, original	17	196

[44] Support Vector Machines

Tabla 4.39: Matriz de confusión para clasificador de *SVM radial* de usuarios activos

	No exitoso, predicción	Exitoso, predicción
No exitoso, original	214	37
Exitoso, original	14	197

Finalmente se presenta en la **tabla 4.40** la **precisión**, **sensibilidad** y **especificidad** para el indicador de usuarios activos para cada uno de los clasificadores diferentes generados.

Tabla 4.40: Métricas de clasificadores para caso de usuarios activos

Clasificador	Precisión	Sensibilidad	Especificidad
Árbol (rpart)	87.88%	93.62%	83.33%
Árbol podado (rpart)	87.66%	81.95%	96.22%
Random Forest	87.88%	88.7%	87.07%
Naive Bayes	86.8%	91%	83.27%
SVM lineal	88.1%	92.02%	84.74%
SVM radial	88.96%	93.36%	85.26%

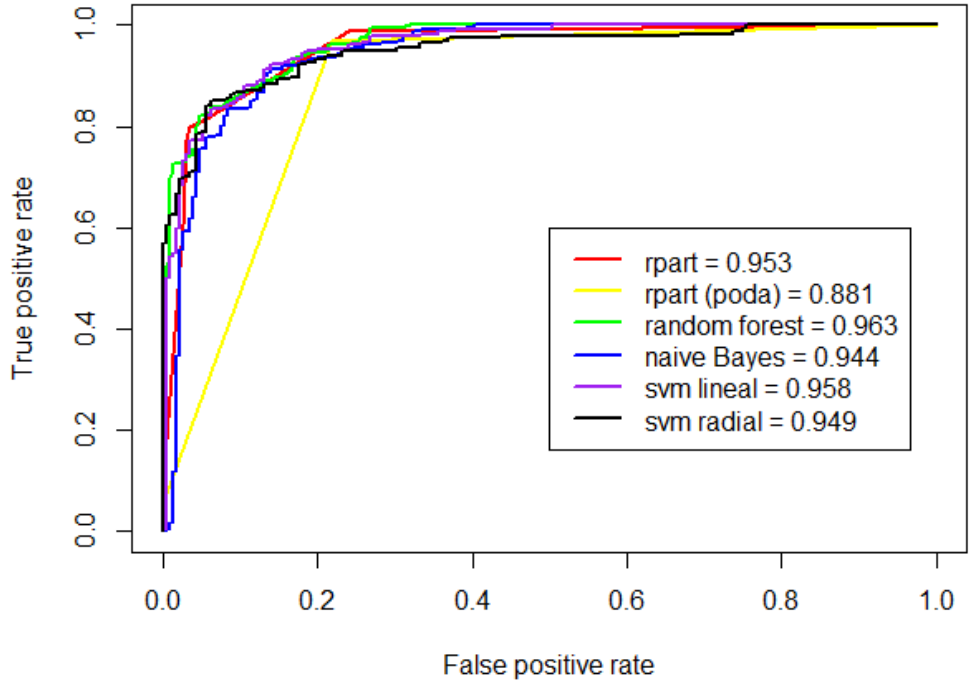
Estas métricas dan un buen primer acercamiento para poder comparar los clasificadores generados, pero sólo cubren algunos de los casos. Como es propuesto por *[7]*, las curvas *ROC*[45] son una buena forma de modelar cómo se comportan la sensibilidad y especificidad (cada una en relación a la otra).

El área bajo la curva del gráfico de sensibilidad (verdaderos positivos) contra 1-especificidad (falsos positivos) será un buen indicador de qué tan bueno es un

[45] Acrónimo in inglés de *Receiver-operating characteristic.*

clasificador. En el caso óptimo, esta curva tendrá un área de 1 (considerando los porcentajes de 0 a 1), por lo tanto, mientras mayor sea el área bajo una curva *ROC*, mejor será el predictor. En la **ilustración 4.41** se presenta la curva de cada uno de los clasificadores generados, y en la leyenda, el área bajo la curva respectiva. A pesar de que visualmente no se aprecia mayor diferencia, numéricamente se concluye finalmente que el mejor clasificador para este indicador en particular, por muy poca diferencia, es el generado por el algoritmo *randomForest*.

Ilustración 4.41: Curvas *ROC* y áreas bajo estas para usuarios activos



Clasificación de indicador de penetración

Para el clasificador del algoritmo *rpart* se presenta en la **tabla 4.42** la matriz de confusión del árbol generado del indicador de penetración. Se desprende que el clasificador tiene una especificidad del 71%, una sensibilidad del 78% y una precisión del 74% aproximadamente.

Tabla 4.42: Matriz de confusión para clasificador de *rpart* de penetración

	No exitoso, predicción	Exitoso, predicción
No exitoso, original	182	76
Exitoso, original	44	152

Siguiendo con el análisis, se presenta en la **tabla 4.43** el ranking de importancia de las variables relacionadas con el árbol de clasificación de penetración. En ella se aprecia que *shares_first_day*, o sea, la cantidad de veces que se comparte un video en su primer día de publicación, es más importante al momento de clasificar un registro bajo este indicador. Además, se reitera la importancia de *active_raffles*, o sea, la cantidad de concursos activos al momento de la publicación del video.

Tabla 4.43: Ranking de importancia de variables en árbol (*rpart*) de penetración

Variable	Importancia relativa
shares_first_day	85.49
active_raffles	43.48
duracion	18.9
release_difference	13.83
new_users	9.11
active_canjes	5.1

Para el caso del árbol de penetración podado, se presenta en la **tabla 4.44** su matriz de confusión. De ésta se desprende que el clasificador tendrá una especificidad de 70%, una sensibilidad del 82% y una precisión del 76% aproximadamente, siendo mejor que su predecesor sin poda en términos de sensibilidad y precisión, pero no así al momento de clasificar un registro negativo.

Tabla 4.44: Matriz de confusión para clasificador de *rpart* de penetración (podado)

	No exitoso, predicción	Exitoso, predicción
No exitoso, original	175	74
Exitoso, original	51	154

En la **tabla 4.45** se aprecia el ranking de importancia de variables del caso del árbol de clasificación podado para el indicador de penetración. Se mantiene el mismo orden que en el caso del árbol sin poda, aunque con los valores de la importancia relativa levemente menores.

Tabla 4.45: Ranking de importancia de variables en árbol (*rpart*) de penetración (podado)

Variable	Importancia relativa
shares_first_day	75.19
active_raffles	35.33
duracion	17.29
release_difference	8.81
new_users	8.22
active_canjes	0.88

El siguiente algoritmo aplicado para este indicador fue el del *randomForest*. En la **tabla 4.46** se presenta la matriz de confusión para este caso. El clasificador tendrá una especificidad del 80%, una sensibilidad del 83% y una precisión de 82%, aproximadamente.

Tabla 4.46: Matriz de confusión para clasificador de *randomForest* de penetración

	No exitoso, predicción	Exitoso, predicción
No exitoso, original	189	47
Exitoso, original	37	181

Siguiendo con el orden de este estudio, en la **tabla 4.47** se aprecia el ranking de importancia de cada variable para el caso del clasificador generado con *randomForest* para indicador de penetración. De ella se puede concluir que no todas las variables son importantes en iguales escenarios; por ejemplo, al tratar de predecir un video exitoso a nivel de penetración, la cantidad de usuarios nuevos (*new_users*), la cantidad de usuarios activos (*active_users*) y la cantidad de canjes activos (*active_canjes*) no son importantes, pero sí lo son al momento de clasificar un video de clase no exitosa.

En relación al clasificador generado con el algoritmo *naiveBayes*, se presenta en la **tabla 4.48** su matriz de confusión para el indicador de penetración. Una primera mirada revela que no parece ser de los mejor clasificadores de este estudio, cuenta con una especificidad del 61%, una sensibilidad del 76% y una precisión de sólo el 66%, aproximadamente. Como se mencionó anteriormente, el algoritmo de *naiveBayes* utilizado en este estudio no brinda información con respecto a la importancia de las variables.

Tabla 4.47: Ranking de importancia de variables en clasificador de *randomForest* de penetración

Variable	Clase: no exitoso	Clase: exitoso	*Mean Decrease Accuracy*	*Mean Decrease Gini*
shares_first_day	24.63	7.98	25.39	92.54
active_raffles	24.87	2.35	25.63	80.69
new_users	19.24	-3.19	14.21	69.49
active_users	18.6	-4.19	16.95	71.23
duracion	10.38	8.31	13.03	83.98
active_canjes	10.62	-1.57	9.5	31.46
release_difference	9.22	3.87	9.6	71.42

Tabla 4.48: Matriz de confusión para clasificador de *naiveBayes* de penetración

	No exitoso, predicción	Exitoso, predicción
No exitoso, original	193	122
Exitoso, original	33	106

Los últimos clasificadores a evaluar son los generados por el algoritmo *SVM*, con núcleos lineal y radial. En las **tablas 4.49 y 4.50** se presentan las matrices de confusión para las ejecuciones de los algoritmos *NVM lineal* y *radial* respectivamente; de ellas se concluye que el clasificador en su versión *radial* es más efectivo al momento de clasificar registros ya que cuenta con una especificidad del 69%, contra 62%; una sensibilidad del 82%, contra 78%; y una precisión del 74% contra 67%, aproximadamente.

Tabla 4.49: Matriz de confusión para clasificador de *SVM lineal* de penetración

	No exitoso, predicción	Exitoso, predicción
No exitoso, original	196	120
Exitoso, original	30	108

Tabla 4.50: Matriz de confusión para clasificador de *SVM radial* de penetración

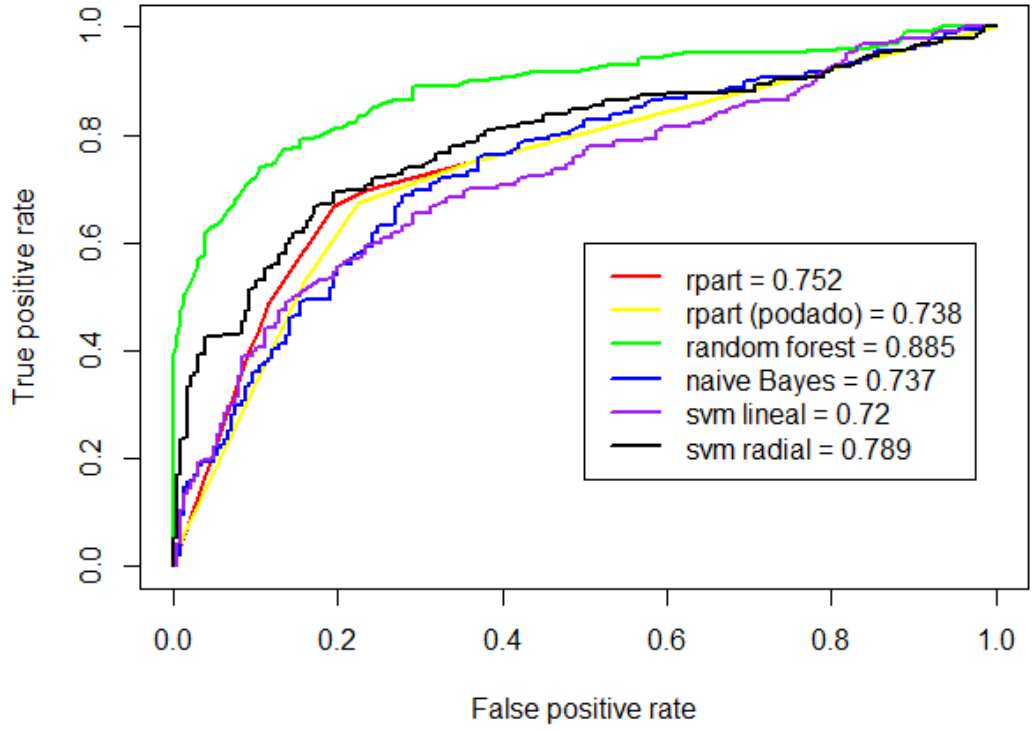
	No exitoso, predicción	Exitoso, predicción
No exitoso, original	192	87
Exitoso, original	32	141

A modo de resumen, se comparan en la **tabla 4.51** las métricas de precisión, especificidad y sensibilidad para cada uno de los clasificadores utilizados para el indicador de penetración. En ella se aprecia que el mejor clasificador para todos los casos es el generado por *randomForest*, y el peor el generado po *naiveBayes*.

Tabla 4.51: Métricas de clasificadores para caso de penetración

Clasificador	Precisión	Sensibilidad	Especificidad
Árbol (rpart)	73.57%	77.55%	70.54%
Árbol podado (rpart)	72.47%	75.12%	70.28%
Random Forest	81.5%	83.03%	80.08%
Naive Bayes	65.86%	76.26%	61.27%
SVM lineal	66.96%	78.26%	62.03%
SVM radial	73.67%	81.5%	68.82%

Ilustración 4.52: Curvas *ROC* y áreas bajo estas para penetración



Finalmente, en la **ilustración 4.52** se presenta la curva de cada uno de los clasificadores generados, y en la leyenda, el área bajo la curva respectiva. Se concluye que el mejor clasificador para este indicador en particular (penetración), por una diferencia considerable, es el generado por el algoritmo *randomForest*.

Clasificación de indicador de calidad usuaria (simplificada)

Por simplicidad, posibilidad de evaluación y comparación, y necesidades del negocio, se decide no utilizar el modelo de calidades usuarias generado para los ocho valores posibles, si no el simplificado. Para el objetivo de este estudio, no es necesario ahondar en una calidad usuaria positiva, ya que cualquiera de éstas es satisfactoria para el objetivo del negocio.

En la **tabla 4.53** se presenta la matriz de confusión relacionada con el árbol de clasificación generado con *rpart* para para el indicador de calidades usuarias (simplificadas). De ella se desprende que el tendrá una especificidad del 97%, una sensibilidad del 98% y una precisión 98%, aproximadamente; por lo que, a primera vista, pareciera ser un muy buen modelo para estos escenarios.

Tabla 4.53: Matriz de confusión para clasificador de *rpart* de calidades usuarias simplificadas

	Calidad mala, predicción	Calidad buena, predicción
Calidad mala, original	739	20
Calidad buena, original	16	695

Luego se procede a evaluar las variables incluidas en el árbol, cuyo ranking de importancia se ve representada en la **tabla 4.54**. Como era de esperarse al ver el árbol relacionado, la cantidad de reclutamientos tiene una importancia resaltante en el modelo, junto con concursos participados.

Del segundo modelo aplicado al indicador de calidad usuaria simplificada, *randomForest*, se muestra la matriz de confusión en la **tabla 4.55**. De ella se obtiene una especificidad del 99%, una sensibilidad del 99% y una precisión del 99%, aproximadamente. Luego, en la **tabla 4.56**, se presenta el ranking de importancia de variables de este modelo.

El siguiente algoritmo aplicado, al igual que en los indicadores anteriores, es *naiveBayes*. Del modelo generado de su ejecución se desprende la matriz de confusión presentada en la **tabla 4.57**. Se obtiene de ella que el modelo cuenta con un 96% de especificidad, un 98% de sensibilidad y un 97% de precisión, aproximadamente.

Tabla 4.54: Ranking de importancia de variables en árbol (*rpart*) de calidades usuarias simplificadas

Variable	Importancia relativa
recruitmets	1329.17
concursos_participados	1152.92
premios_canjeados	843.66
calidad_videos	252.68
densidad_videos	233.11
edad	229.97
sistema_registro	14.21
densidad_concursos	1.04

Tabla 4.55: Matriz de confusión para clasificador de *randomForest* de calidades usuarias simplificadas

	Calidad mala, predicción	Calidad buena, predicción
Calidad mala, original	749	9
Calidad buena, original	6	706

Tabla 4.56: Ranking de importancia de variables en clasificador de *randomForest* de calidades usuarias simplificadas

Variable	Clase: mala	Clase: buena	*Mean Decrease Accuracy*	*Mean Decrease Gini*
recruitments	48.98	27.35	50.38	719.09
concursos_participados	46.25	17.44	47.67	618.28
premios_canjeados	27.48	16.13	28.26	266.21
sistema_registro	20.81	14.12	25.68	71.49
densidad_videos	11.29	9.11	11.44	51.33
edad	6.84	7.42	9.7	32.25
calidad_videos	12.84	5.38	13.92	33.53
densidad_concursos	7.99	0.72	6.58	18.21
genero	-1.74	-0.76	-1.9	4.35

Tabla 4.57: Matriz de confusión para clasificador de *naiveBayes* de calidades usuarias simplificadas

	Calidad mala, predicción	Calidad buena, predicción
Calidad mala, original	744	34
Calidad buena, original	11	681

El último de los algoritmos para modelar clasificadores para el indicado de calidad usuaria es *SVM*. Al igual que en casos anteriores, se aplica con núcleos *lineal* y *radial*, y sus matrices de confusión resultantes se aprecian en las **tablas 4.58 y**

4.59. De ellas se desprende que ambos modelos son buenos para clasificar tanto un usuario malo como bueno, contando con, aproximadamente, una especificidad del 98%, una sensibilidad del 99% y una precisión del 99% en ambos casos, respectivamente.

Tabla 4.58: Matriz de confusión para clasificador de *SVM lineal* de calidades usuarias simplificadas

	Calidad mala, predicción	Calidad buena, predicción
Calidad mala, original	751	14
Calidad buena, original	4	701

Tabla 4.59: Matriz de confusión para clasificador de *SVM radial* de calidades usuarias simplificadas

	Calidad mala, predicción	Calidad buena, predicción
Calidad mala, original	751	18
Calidad buena, original	4	697

Como resumen se presentan en la **tabla 4.60** las diferentes métricas de calidad de los modelos construidos para el indicador de calidad usuaria simplificada, y en la **ilustración 4.61**, sus curvas *ROC* junto con los valores de área bajo la curva en cada caso.

Una mirada a los valores y formas de las curvas *ROC* generadas hace ver que los valores son extrañamente altos. Esto se puede deber a que, como se mencionó en el capítulo anterior, algunas variables[46] utilizadas para el modelamiento de los clasificadores fueron además utilizadas para el cálculo inicial del indicador en cuestión. Se decide modelar también clasificadores con todos los algoritmos con las

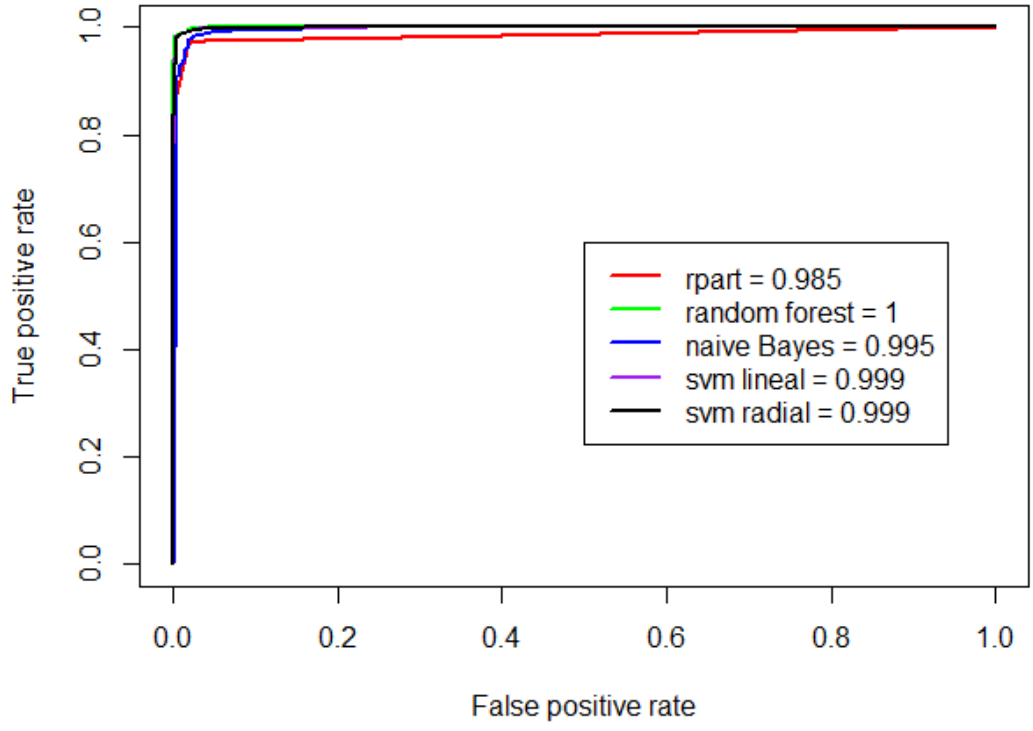
[46] *Recruitments, concursos_participados* y *premios_canjeados*

variables de entrada filtradas, es decir, sin aquellas que fueron usadas para definir en primera instancia el indicador objetivo. A continuación, se evaluará cada uno de esos modelos.

Tabla 4.60: Métricas de clasificadores para caso de calidades usuarias simplificadas

Clasificador	Precisión	Sensibilidad	Especificidad
Árbol (rpart)	97.55%	97.75%	97.36%
Random Forest	98.98%	99.16%	98.81%
Naive Bayes	96.94%	98.41%	95.63%
SVM lineal	98.78%	99.43%	98.17%
SVM radial	98.5%	99.43%	97.66%

Ilustración 4.61: Curvas *ROC* y áreas bajo estas para calidades usuarias simplificadas



En la **tabla 4.62** se presenta la matriz de confusión para el caso del árbol generado con el algoritmo *rpart* para el indicador de calidad usuaria simplificada y filtrada. De ella se desprende que el modelo tendrá una especificidad del 74%, una sensibilidad del 79% y una precisión del 76%, aproximadamente.

Tabla 4.62: Matriz de confusión para clasificador de *rpart* de calidades usuarias simplificadas y filtradas

	Calidad mala, predicción	Calidad buena, predicción
Calidad mala, original	625	217
Calidad buena, original	130	498

Tabla 4.63: Ranking de importancia de variables en árbol (*rpart*) de calidades usuarias simplificadas y filtradas

Variable	Importancia relativa
sistema_registro	245.43
densidad_videos	229.11
densidad_concursos	78.96
caldiad_videos	47.80
edad	12.94

En la **tabla 4.63** se lista el ranking de importancia de las variables para este caso. Dejando de lado la evidencia de que no se encuentran las variables que eran más importantes por órdenes de magnitud en el caso del árbol no filtrado, ya que fueron removidas para estas ejecuciones, cambia el orden de la importancia de

variables, tomando importancia el sistema de registro y la densidad de concursos, que solían estar al fondo del ranking para el caso anterior.

El siguiente algoritmo aplicado para el caso del indicador de calidad usuaria simplificada y filtrada es, al igual que en los indicadores anteriores, *randomForest*. En relación al modelo obtenido de esta aplicación se presenta en la **tabla 4.64** su matriz de confusión. De ella se obtiene una especificidad del 79%, una sensibilidad del 83% y una precisión del 81%. Luego, en la **tabla 4.65** se lista el ranking de importancia de las variables relacionadas con el modelo. De ella se desprenden dos puntos importantes de mencionar: por un lado, la variable que pareciera ser más significativa al momento de definir un usuario de calidad negativa es su sistema de registro, lo que calza con análisis anteriores en otros modelos; por otro lado, para el caso de los usuarios de calidades positivas, la importancia no es la misma, sino que la densidad de videos es la variable que destaca. Al igual que en otros modelos, el sexo del usuario no tiene importancia al momento de definir su calidad, estando por una gran diferencia, último en el ranking.

Tabla 4.64: Matriz de confusión para clasificador de *randomForest* de calidades usuarias simplificadas y filtradas

	Calidad mala, predicción	Calidad buena, predicción
Calidad mala, original	648	175
Calidad buena, original	107	540

Tabla 4.65: Ranking de importancia de variables en clasificador de *randomForest* de calidades usuarias simplificadas y filtradas

Variable	Clase: mala	Clase: buena	*Mean Decrease Accuracy*	*Mean Decrease Gini*
densidad_videos	35.83	40.08	49.40	284.94
sistema_registro	40.72	30.83	45.05	285.09
edad	28.40	27.29	32.86	374.46
calidad_videos	29.59	15.19	29.77	178.58
densidad_concursos	31.83	10.04	28.19	170.49
genero	3.59	1.87	3.75	38.43

El siguiente algoritmo aplicado es *naiveBayes*, cuya matriz de confusión se presenta en la **tabla 4.66**. De ella se obtiene que el modelo cuenta con una especificidad del 80%, una sensibilidad del 78% y una precisión del 79%.

Tabla 4.66: Matriz de confusión para clasificador de *naiveBayes* de calidades usuarias simplificadas y filtradas

	Calidad mala, predicción	Calidad buena, predicción
Calidad mala, original	591	149
Calidad buena, original	164	566

Finalmente se evalúan los modelos generados por las aplicaciones de los algoritmos *SVM lineal* y *radial*. En las **tablas 4.67 y 4.68** se presentan las matrices de confusión para el caso de *SVM lineal* y *radial* respectivamente. De ellas se obtiene que el modelo generado por la *SVM radial* es mejor en todos los casos que el

generado por su versión *lineal*, teniendo 81% de especificidad contra 78%, 81% de sensibilidad contra 77%, y 81% de precisión contra 77%, aproximadamente.

Tabla 4.67: Matriz de confusión para clasificador de *SVM lineal* de calidades usuarias simplificadas y filtradas

	Calidad mala, predicción	Calidad buena, predicción
Calidad mala, original	594	172
Calidad buena, original	161	543

Tabla 4.68: Matriz de confusión para clasificador de *SVM radial* de calidades usuarias simplificadas y filtradas

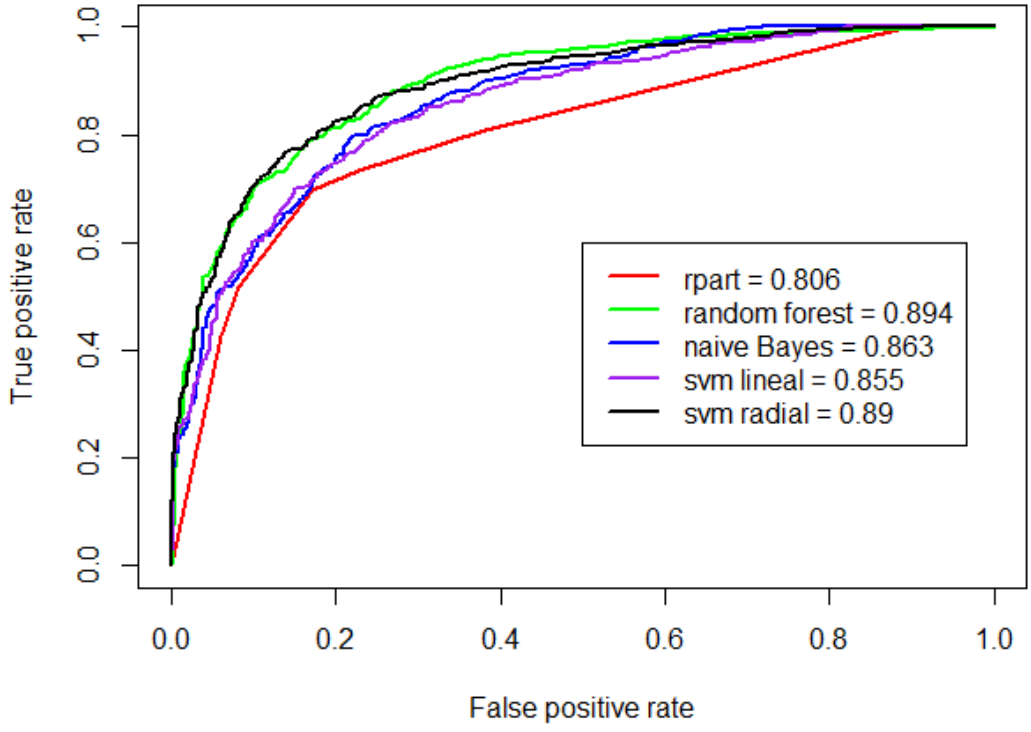
	Calidad mala, predicción	Calidad buena, predicción
Calidad mala, original	625	149
Calidad buena, original	130	566

Finalmente, se presentan en la **tabla 4.69**, de forma resumida, la especificidad, sensibilidad y precisión de todos los modelos generados para el caso del indicador de calidad usuaria simplificado y filtrado. Esta tabla cuenta con valores menores a la conseguida en el caso de los modelos de calidad usuaria sin filtrar. En este caso, el algoritmo que pareciera generar un mejor modelo es *SVM radial*, superado sólo por *randomForest* en el caso de la sensibilidad. Para continuar con el análisis, se muestran en la **ilustración 4.70** las diferentes curvas *ROC* de estos modelos, junto con sus valores de área bajo la curva. Confirmando lo observado en la tabla de métricas de estos clasificadores, las áreas bajo sus curvas *ROC* dan a ver que el modelo más efectivo es, al igual que para los indicadores anteriores, *randomForest*, aunque por muy poca diferencia de *SVM lineal*.

Tabla 4.69: Métricas de clasificadores para caso de calidades usuarias simplificadas y filtradas

Clasificador	Precisión	Sensibilidad	Especificidad
Árbol (rpart)	76.39%	79.30%	74.23%
Random Forest	80.95%	83.93%	78.65%
Naive Bayes	78.71%	77.53%	79.86%
SVM lineal	77.35%	77.13%	77.55%
SVM radial	81.02%	81.32%	80.75%

Ilustración 4.70: Curvas *ROC* y áreas bajo estas para calidades usuarias simplificadas y filtradas



Finalmente pareciera que, a pesar de que todos los algoritmos utilizados generan modelos satisfactorios para cada indicador, en todos los casos el que pareciera ser mejor es *randomForest*.

4.4. Despliegue

En esta, la última etapa del proceso *CRISP-DM*, se despliega de forma resumida toda la información relevante obtenida en las fases anteriores. Es dentro de este objetivo que se dividirá esta sección en tres partes, abordando en cada una la información rescatable referente a los indicadores propuestos para este estudio.

4.4.1. Despliegue para usuarios activos

Es importante para el negocio saber en qué condiciones la plataforma ha tenido cúspides de actividad usuaria, para así poder replicar dichas condiciones, ver cómo influyen, e incluso mejorarlas. En base a este principio, los puntos más importantes revelados por el estudio para potenciar este indicador son:

- La cantidad de **concursos activos** juega un papel importantísimo al momento de definir la cantidad de usuarios activos. Esto se observó tanto durante la visualización de datos con *Tablau* como al analizar y evaluar los modelos de clasificación generados. Una mayor cantidad de concursos activos tendrá como consecuencia una mayor cantidad de usuarios activos.

- Los **canjes activos**, de la misma manera que los concursos, influenciarán de forma positiva a la cantidad de usuarios activos, aunque en una menor magnitud[47]. Esto se deduce del proceso de visualización con herramienta *OLAP* y del análisis de los modelos de clasificación generados.

[47] Aproximadamente la mitad de importancia de acuerdo al modelo generado con *randomForest*.

- Otra variable, que afecta en menor medida a este indicador, es **la cantidad de usuarios nuevos**, lo que es esperable debido a que un usuario nuevo tenderá a interactuar con la plataforma, transformándose en un usuario activo. Esto se concluye de los modelos de clasificación generados.

- Si bien el modelo generado por *randomForest* para este indicador hace ver que la **cantidad de veces que un video se comparte en su primer día** influye positivamente en el indicador de usuarios activos, por conocimiento del negocio se sabe que esta relación, si bien existe, no tiene la dirección necesaria para ser explotada. Es a mayor cantidad de usuarios activos que un video se comparte más en su primer día, justamente porque esta es una de las condiciones que puede definir a uno de ellos como activo.

4.4.2. Despliegue para penetración

Un video con buena penetración tiene como consecuencia una mayor propagación del contenido del cliente, tanto en la plataforma como en las redes sociales de sus usuarios, además de una ayuda evidente a la plataforma como marca, al compartirse su imagen y dominio junto con el video. En este contexto, para mejorar la penetración de un video, se deben considerar los siguientes puntos:

- Durante el proceso de visualización con *Tableu* se encuentra una relación muy interesante para el estudio. La **diferencia de lanzamiento**, en conjunto con la **duración** de un video, influirán directamente sobre la penetración que este consiga dentro de la plataforma: entre menor duración y diferencia de lanzamiento, mayor penetración. A pesar de que la lógica dicta que un video de menor duración es más probable de ser compartido, esto selimita exclusivamente a esta variable, sino que sólo cumple la tendencia cuando este video es suficientemente nuevo para el grupo usuario. En particular, los

mejores resultados de penetración se observan para los casos en que el contenido es publicado entre cero y seis horas después de su lanzamiento, y tiene una duración menor a un minuto. Esta relación además se ve reflejada en los modelos de clasificación generados, donde ambas variables son importantes al momento de evaluar la importancia de las mismas para el clasificador[48]. En otras palabras, **los videos más cortos, y más nuevos, tienen mejor aceptación**. Un video de menor duración tiene mejor aceptación por la comunidad de la plataforma. Además, es fundamental que el lanzamiento de dicho video sea rápido en comparación a otras fuentes de publicación. Se recomienda traspasar esta información al cliente para que lo tenga en consideración al momento de generar nuevos contenidos. Además, se descubre que la cantidad de veces que un video se comparte en el primer día de su publicación influye directamente en la penetración que conseguirá finalmente, por lo que se recomienda potenciar los videos en sus inicios, y así generar esta relación.

- De acuerdo a los modelos de clasificación generados, y en particular al más efectivo, resultado de la ejecución de *randomForest*, la **cantidad de veces que se comparte un video en su primer día** de publicación juega un papel importante para que el mismo alcance una buena penetración[49]. A pesar de que esta relación parece evidente, no se contaba con respaldo para afirmarlo.

- Al igual que para el indicador de usuarios activos, la cantidad de **concursos activos** influye directa y positivamente sobre la penetración. Aunque en menor medida que las variables mencionadas anteriormente. Esto se puede deber que, al aumentar los usuarios activos, se produce por consecuencia un aumento en

[48] A pesar de que se repite esta situación en múltiples modelos, se hace referencia al generado con *randomForest*
[49] Del 70% o más.

la cantidad de veces que se comparte en la plataforma, potenciando a su vez la penetración que alcanza un video.

4.4.3. Despliegue para calidad usuaria

La calidad usuaria es un factor fundamental para entender a la comunidad de la plataforma, y poder potenciar aquellos factores que atraen, o potencian, las calidades provechosas para conseguir los objetivos del negocio. De acuerdo a lo estudiado, cabe destacar los siguientes puntos:

- En relación a las características propias de un usuario, su sexo no afectará de ninguna manera a su calidad usuaria. En relación a su edad, la mayoría de ellos se posiciona entre los 20 y 26 años, aunque siendo los más activos los que se encuentran entre los 14 y 28. Esto se concluye en base a la visualización de los datos con *Tableau*.

- Hay una serie de variables de entorno que influenciarán directamente en la calidad usuaria, como la **calidad de videos**, la **densidad de concursos**, y la **densidad de videos**. Sorprendentemente, esta última es la más importante de acuerdo al mejor modelo de clasificación generado. En otras palabras, el usuario interactuará a mayor medida con la plataforma cuando sus opciones de videos a ver y compartir sean más numerosas. Por otro lado, si bien la **densidad de concursos** no va a definir a un usuario de buena calidad en gran medida, de acuerdo a las reglas de clasificación generadas, si lo hará para un usuario de calidad negativa. Para el caso de la calidad usuaria *No capturado*, análisis visuales revelan que hay una gran influencia por parte de una baja densidad de concursos. Esto quiere decir, que al momento en que se registró un usuario y la semana siguiente a este registro, la densidad de concursos fue baja. En otras palabras, en la primera semana de la mayoría de los usuarios *No*

capturados, la densidad de concursos fue baja. En muy pocos casos, fue Regular o mejor.

- De la misma forma que un buen valor de **densidad de videos** influye de forma positiva sobre la calidad usuaria, lo hará de forma negativa en los escenarios en que su valor se malo. De acuerdo a lo observado en Tableau, Para la gran mayoría de los casos de usuarios No capturados, la densidad de videos al momento de registro y su semana siguiente es baja, y en una segunda mayoría, relativamente baja. Esta relación se confirma luego con la generación de reglas de clasificación.

- El **sistema de registro** más común, por una amplia mayoría, para el caso de usuarios *No interesado/No comprendió* es a través de la vitrina de videos[50]. Este sistema de registro también destaca para la clase usuaria *No capturado.* Por otro lado, se aprecia una relación entre el sistema de registro de *reclutamiento* y la calidad usuaria *Perdido*. La que se confirma luego por la generación de reglas de clasifiación. En referencia a las calidades usuarias positivas, la mayoría de los registros se realiza a través de *campañas de Facebook*, y *campañas de flyers*. En otras palabras, **el sistema de registro influye directamente en la calidad que tendrá un usuario a futuro**. Se descubrió que las formas más efectivas de conseguir usuarios de calidad son a través de campañas en Facebook (a través del *fanpage*) y campañas de *flyers*. Por otro lado, la forma menos efectiva de conseguir usuarios es a través del registro en vitrina. Esto se puede deber a que las personas que se registran a través de este medio no tienen información del sitio ni de la plataforma, y al tratarse de un sistema bloqueante[51], el usuario no sabe en qué se está registrando. Se cree que además esto puede producir roce para usuarios

[50] Through video display
[51] Actualmente, sólo se puede ver la vitrina de un video y la página principal (sin contenido atractivo) de Kikvi siendo un usuario no registrado.

potencialmente buenos, y que no da tiempo de mostrar lo atractivo de la plataforma. Se recomienda liberar las secciones principales de la plataforma, y motivar al usuario a registrarse cuando este ya tenga un entendimiento mayor de la misma, y se haya familiarizado con el contenido y la comunidad.

- La **cantidad de usuarios que recluta** un usuario dado tiene influencia directa sobre su calidad. Esto se ve reflejado tanto en reglas de asociación como en árboles de decisión. Esto quiere decir que los usuarios de buena calidad suelen reclutar a otros usuarios, pero no se trata de una variable que pueda ser manejada por quien administra la plataforma. Cabe destacar que se descubrió anteriormente que los usuarios reclutados suelen terminar siendo de la clase *Perdido*. En otras palabras, **el sistema actual de reclutamiento no genera valor para el negocio.** Si bien los usuarios de buenas calidades reclutan a otros usuarios, estos nuevos reclutas tienen una altísima probabilidad de convertirse en usuarios perdidos. Se recomienda la eliminación del sistema de reclutamiento, o realizar estudios siguientes para idear uno que capture usuarios de buena calidad.

- **Es fundamental mantener un buen ambiente en la plataforma**. Se ha descubierto que el ambiente al momento de registro tiene influencia en la calidad que tendrá un usuario a futuro, además de su permanencia en el sistema. Las consideraciones más importantes en torno a este punto se relacionan por un lado con la cantidad de concursos y canjes, y por otro lado a la densidad de videos. A mayor cantidad de concursos, más usuarios activos habrá en la plataforma, y los usuarios que se registren en este período tendrán mayor probabilidad de pertenecer a las calidades usuarias positivas. En relación a la densidad de videos, a mayor cantidad de videos (nuevos) al momento de registro de un nuevo usuario, es más probable que este se mantenga interactuando con la plataforma. La calidad de los videos también

juega un papel en el ambiente al momento de registro, aunque su importancia es menor a las de las variables ya mencionadas.

Conclusiones

Luego de este estudio se tiene un panorama más completo del funcionamiento de Kikvi y de sus usuarios. Cabe destacar que todas estas conclusiones aplican para el contexto del estudio realizado, y que deben ser comprobadas caso a caso, aunque se recomienda tomarlas como referencia o punto de partida en servicios similares.

Una primera mirada a los datos reveló que la mejor categoría de videos para ser compartida por la comunidad es la de **videojuegos**. Sobre esto, se plantean la siguiente consideración: los videos de videojuegos en la plataforma comprenden principalmente *teasers* y *trailers* de nuevos lanzamientos, y no videos de *streaming*[52] ni *gameplays*[53]. Esta notoria ventaja de los videos de videojuegos no se ve traducida en mayor cantidad de vistas. De la misma manera, no se encuentra una relación visible entre la cantidad de veces que se comparte un video y la cantidad de vistas que consigue.

El estudio realizado se espera que beneficie de forma directa al crecimiento de la plataforma y su comunidad. A pesar de que su desarrollo no fue sencillo, se consigue obtener conclusiones muy significativas en torno a qué pasos seguir y hacia dónde enfocar esfuerzo y recursos para lograr que Kikvi tenga más valor para sus usuarios, y como consecuente, para sus clientes. Se espera que el conocimiento generado sea aplicado, aunque este signifique un fuerte cambio de paradigma para el producto en sí.

Una de las medidas sobre las que más se debería hacer hincapié es en replantearse la forma en que se desarrolla hoy la interacción de usuario y plataforma, ya que se considera que las interfaces actuales son demasiado bloqueantes, des

[52] Video generalmente en vivo de una persona jugando algún videojuego en particular.
[53] Video que muestra fragmentos de una persona jugando un videojuego en particular.

motivantes, y poco claras; lo que genera usuarios que no comprenden o no sienten interés por la plataforma una vez que se registran en ella. Es importante enfocar esfuerzos y recursos en **conseguir y mantener** usuarios de calidad, y no únicamente conseguir registros y considerar el número de usuarios nuevos como un indicador importante.

Ya que una cantidad considerable de registros se genera desde la vitrina, y se recomienda eliminar dicho medio de registro, se hace necesario mejorar esas interfaces para mantener el interés del usuario en la plataforma. Como primer acercamiento, un sistema de videos similares o recomendados.

Algo importante de mencionar es que el estudio será de gran ayuda al negocio ya las variables que impactan en mayor manera a los indicadores objetivo, como son la cantidad de concursos activos, la duración de los videos o la diferencia de lanzamiento; son manejables directamente por quien administra el sistema. Esto quiere decir que será posible generar un alto impacto sobre el negocio si se enfocan correctamente los esfuerzos de quienes administran la plataforma.

En relación a la metodología usada, *CRIPS-DM* es muy recomendada por su facilidad de seguimiento, debido a su estructura en fases. Además, al considerar aspectos del negocio, y tener fases previas de estudio, ha sido más fácil enfocar las etapas posteriores al objetivo buscado. Claro que, por esta misma razón, en un escenario en que el conocimiento del negocio sea poco o inexistente, se recomienda recurrir a otras metodologías de estudio.

En relación a la herramienta utilizada para los procesos de minería de datos, *R*, cabe destacar que, a pesar de que se trata de una buena solución, puede tener complicaciones durante el proceso. Una de las razones principales por las que se decide utilizar *R* es debido a la gran cantidad de implementaciones de algoritmos con las que cuenta, distribuidas en gran número de paquetes y hasta en diferentes

repositorios. Esto puede generar una complicación al momento de buscar alguna implementación en particular, debido al gran volumen de opciones. Por otro lado, algo que significó una gran ventaja para el estudio fue la capacidad de personalización de la herramienta. Al tratarse principalmente de una serie de paquetes e implementaciones traducidas en funciones, es posible buscar, y encontrar, exactamente lo que se está buscando, pudiendo en el peor de los casos acomodar lo encontrado a requerimientos especiales.

Una de las cosas que *R* como herramienta de minería de datos no resuelve de buena manera es todo el apoyo visual relacionado con los procesos. Se requiere de muchos pasos y consideraciones para poder generar apoyo visual de los modelos que se pudiesen generar, tomando una gran cantidad de tiempo en comparación a otras herramientas más enfocadas en esto como puede ser *weka*.

En relación al aporte que significó lo aprendido durante el curso de la carrera de ingeniería civil informática, lo más importante, y a la vez más subjetivo, es la lógica. De forma inconsciente se ha aprendido a lo largo de los años de estudio a enfrentar los problemas y situaciones con un pensamiento lógico y secuencial, pudiendo así enfrentar problemas encontrados desde diferentes aristas hasta conseguir soluciones y conclusiones satisfactorias. Por otro lado, y dejando lo intangible más de lado, fue muy significativo para este estudio lo aprendido durante los siguientes ramos:

- **Bases de datos**: el manejo efectivo de consultas considerablemente complejas fue significativo al momento de rescatar los datos almacenados en diferentes bases de datos *MySQL*.

- ***Business intelligence***: el marco de conocimiento adquirido durante este curso no sólo ayudó durante el desarrollo del estudio, si no que fueron los responsables de la toma de decisión de enfocar al mismo desde esta arista.

Además, gracias a este curso se tenía una idea previa de las herramientas y procesos existentes para alcanzar los objetivos buscados.

- **Diseño de interfaces usuarias**: responsable de generar consciencia de la importancia de una interfaz clara y de fácil uso, este curso fue el responsable del análisis inicial del estudio.

Por otro lado, los cursos de desarrollo y programación fueron indispensables para el desarrollo práctico del proyecto, ya que fue necesario manejar una gran cantidad de lenguajes diferentes para concretarlo, y la versatilidad es un pilar muy bien abordado durante la carrera en la universidad.

Finalmente, se recomiendan como extensiones a este estudio:

- **Adquirir una mayor cantidad de datos personales del usuario**. El estudio se vio ampliamente limitado por el hecho de que se contaba con muy poca información sobre el usuario en sí. Podría haberse perfilado de mucha mejor manera a cada grupo o calidad usuaria teniendo información de residencia, origen, nivel de estudios, nivel socioeconómico, etc. Se recomienda entonces recopilar información de esta naturaleza y analizar su impacto en las conclusiones y relaciones encontradas.

- **Utilización de otras herramientas para abordad los mismos casos**. Si bien se utilizaron las herramientas con las que quien realiza el estudio se sentía más cómodo, estas no podrían ser las más óptimas para conseguir los objetivos propuestos.

- **Evaluación de resultados y realización de estudios futuros**. Se recomienda una segunda evaluación y estudio de los resultados una vez aplicadas las

medidas propuestas por este estudio, con el fin de definir si ellas se acomodan específicamente al momento específico en el que se encuentra la plataforma en este momento o son transversales para todos los niveles de crecimiento. Nuevos estudios y evaluaciones podrían significar nuevo conocimiento para el negocio.

Referencias bibliográficas

[1] **KDD, SEMMA AND CRISP-DM: A PARALLEL** OVERVIEW - Azevedo & Santos - 2008

[2] **Identification of Outliers** -- Douglas M. Hawkins, 1980

[3] **Refinement of approximate domain theories by knowledge-based neural networks**, G. Towell, J. Shavlik, and M. Noordewier, 1990.

[4] **Active Users: Measure the Success of Your Business**, Claudiu Murariu, 2014 - https://blog.innertrends.com/active-users-2/748 - última vista 14 junio 2016

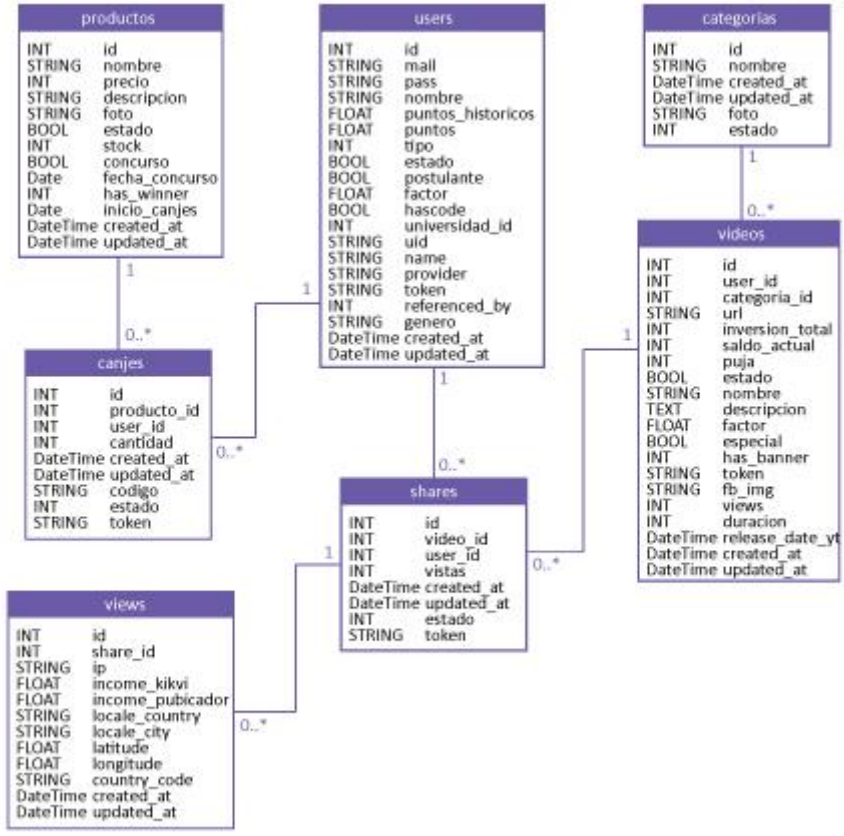
[5] **Algorithms and Applications for Spatial Data Mining -** Martin Ester, Hans-Peter Kriegel, Jörg Sander (University of Munich), 2001.

[6] **An Introduction to Error Analysis**, John Robert Taylor, 1997.

[7] **The meaning and use of the area under a receiver operating characteristic (ROC) curve.** -Hanley, J. A. and B. J. McNeil, 1982.

Anexo 1: Modelo de datos relacional

Tablas relevantes al estudio.



Anexo 2: Script API Facebook PHP

```
<?php
   require(“fbSDK/autoload.php”);
   use Facebook\FacebookSession;
   use Facebook\FacebookRequest;
   use Facebook\GraphUser;
   use Facebook\FacebookRequestException;
   use Facebook\FacebookRedirectLoginHelper;
   $facebook = FacebookSession::setDefaultApplication(<app_id>,
<app_token>);
   $session = FacebookSession::newAppSession();
   $sql = “SELECT
           users.*
         FROM
           users
         WHERE
           users.id != 0
           AND users.id != 8
           AND users.tipo = 1
           AND users.estado = 1
           AND users.uid IS NOT NULL”;
   $db = mysql_connect(“localhost”,<db_user>,<db_pass>);
   $selected = mysql_select_db(<db_name>);
   $rs = mysql_query($sql);
   $getFromFacebook = array();
   while($row = mysql_fetch_assoc($rs)){
      if(!empty($row[“uid”])){
         $tmp = new StdClass();
         $tmp->id = $row[“id”];
         $tmp->fbid = $row[“uid”];
         array_push($getFromFacebook, $tmp);
      }
   }
   mysql_close();
   $total = sizeof($getFromFacebook);
   $current = 1;
   $updateQuery = ““;
   foreach($getFromFacebook as $user){
      print_r(“Fetching... “.$current.”/”.$total.”\n”);
      print_r(“/”.$user->fbid.”\n”);
      $request = new FacebookRequest($session, 'GET', '/'.$user->fbid);
```

```
     try{
        $response = $request->execute();
        $graphObject = $response->getGraphObject();
        $gender = $graphObject->getProperty('gender');
        $gender = strtoupper(substr($gender,0,1));
        $updateQuery .= “UPDATE users SET genero = '$gender' WHERE id =
'$user->id';”;
     }
     catch (Exception $e){
        print_r(“Error: perfil borrado”.”\n”);
     }
     $current++;
  }
  $file = “sqlQuery.sql”;
  file_put_contents($file, $updateQuery);
?>
```

Anexo 3: Script API Youtube (RoR)

```
 def get_duracion_videos_from_yt
  if not params[:videos].nil?
     concatenated_yt_ids = Video.where(:id =>
params[:videos]).map(&:url).join(“,”)
        url = <GOOGLE API URL WITH PRIVATE TOKEN>
        begin
           video_info = open(url)
           video_info = JSON.parse video_info.read
           video_info[“items”].each do |v|
              duration = v[“contentDetails”][“duration”]
              seconds = 0
              duration = duration.sub! “PT”, ““
              if duration.include? “H” and duration.include? “M” and
duration.include? “S”
                 duration_hours = duration.split(“H”)[0].to_i
                 duration_minutes = duration.split(“M”)[0].split(“H”)[1].to_i
                 duration_seconds = duration.split(“M”)[1].gsub! “S”,””
                 duration_seconds = duration_seconds.to_i
                 seconds = duration_hours*3600 + duration_minutes*60 +
duration_seconds
              elsif duration.include? “M” and duration.include? “S”
                 duration_minutes = duration.split(“M”)[0].to_i
                 duration_seconds = duration.split(“M”)[1].gsub! “S”,””
                 duration_seconds = duration_seconds.to_i
                 seconds = duration_minutes*60 + duration_seconds
              elsif duration.include? “H” and duration.include? “M”
                 duration_hours = duration.split(“H”)[0].to_i
                 duration_minutes = duration.split(“H”)[1].gsub! “M”,””
                 duration_minutes = duration_minutes.to_i
                 seconds = duration_hours*3600 + duration_minutes*60
              elsif duration.include? “H” and duration.include? “S”
                 duration_hours = duration.split(“H”)[0].to_i
                 duration_seconds = duration.split(“H”)[1].gsub! “S”,””
                 duration_seconds = duration_seconds.to_i
                 seconds = duration_hours*3600 + duration_seconds
              elsif duration.include? “H”
                 duration_hours = duration.split(“H”)[0].to_i
                 seconds = duration_hours*3600
              elsif duration.include? “M”
                 duration_minutes = duration.split(“M”)[0].to_i
```

```
            seconds = duration_minutes*60
          elsif duration.include? “S”
            duration_seconds = duration.split(“S”)[0].to_i
            seconds = duration_seconds
          end
          Video.where(:url => v[“id”]).update_all(:duracion => seconds)
        end
      rescue StandardError=>e
        puts “Error”
      end
    end
    render :nothing => true
end
```