MÁSTER IN MANAGEMENT + ANALYTICS

PROBABILIDAD DE PUBLICACIÓN EN EL MARKETPLACE

MODELADO DE MACHINE LEARNING DE CLASIFICACIÓN PARA ACCIONES DE MARKETING

TESIS

Tomás López Coppari

Mayo 2022

Tutor: Dr. Manuel Maurette

Resumen

Actualmente, como usuarios de dispositivos con conexión a internet, recibimos diariamente muchas notificaciones y por diferentes canales desde las aplicaciones que tenemos instaladas. La empresa analizada en este trabajo, una de las e-commerce más grandes de Latinoamérica, posee varios envíos automáticos y recurrentes a sus usuarios que cumplen con alguna característica particular, o bien, realizaron una determinada acción dentro del Marketplace.

El presente trabajo analiza las buenas prácticas de las acciones de marketing online, y propone una mejora al escenario actual. A través de un modelo de clasificación de machine learning, asigna a cada usuario un score de propensión a publicar dentro del Marketplace y define estrategias de envíos de pushes e emails según métricas de open rate, conversión e incrementalidad. Propone una reducción de múltiples envíos a uno solo unificado impactando en el engagement de los usuarios, conversiones y reduciendo las posibilidades de spam.

Esta tesis estudia el gran potencial en la optimización de la toma de decisiones de las grandes organizaciones que se puede generar a partir de un algoritmo de machine learning, sobre un análisis convencional de datos. Los potenciales resultados son acordes a los estándares requeridos por la empresa.

Palabras claves: Marketing online, *engagement, spam*, *machine learning*.

Abstract

Currently, as users of devices with an Internet connection, we receive many daily notifications through different channels from the apps that we have installed. The company analyzed in this work, one of the largest e-commerce in Latin America, has several automatic and recurring pushes and emails to users who have a particular characteristic, or who performed a certain action within the Marketplace.

This work analyzes the good practices of online marketing actions and proposes an improvement to the current scenario. Through a machine learning classification model, it assigns each user a propensity score to publish within the Marketplace and defines a push and email delivery strategies according to open rate, conversion, and incremental metrics. It proposes a reduction from multiple shipments to a unique sent impacting on user engagement, conversion and reducing the probability of spam.

This thesis studies the great potential in optimizing decision-making in large organizations that can be generated from a machine learning algorithm, over conventional data analysis. The potential results are in accordance with the standards required by the company.

Keywords: *Online Marketing, engagement, spam, machine learning.*

Tabla de contenido

Índice de Ilustraciones

Índice de Tablas

1. Introducción

En la actualidad, la mayoría de las personas pasan varias horas del día frente a una pantalla consumiendo todo tipo de publicidades, y, en la mayoría de los casos, esto pasa de manera involuntaria. Existen muchas teorías al respecto, y diversos estudios afirman que un individuo recibe en promedio entre 3.000 y 5.000 mensajes publicitarios diarios desde que se levanta hasta que su día termina (Arango, 2018).

Asimismo, el 60% de la población mundial es usuaria de internet y el 67% tiene un celular propio con conexión móvil. Estos porcentajes vienen en aumento año a año, impulsados estos últimos dos por la pandemia mundial de Covid-19. El promedio de horas que pasa un usuario de entre 16 y 64 años en internet, es de alrededor de 7 horas diarias (Kemp, 2021). De ese tiempo, se destina aproximadamente la mitad en búsqueda de nuevos productos y marcas. Esto explica por qué las empresas actualmente destinan grandes volúmenes de inversión en publicidad y marketing en internet.

Al analizar estos datos particularmente para el mercado brasilero, donde se implementará el modelo en primera instancia, se observa que aún existen mejores métricas porcentuales que a nivel global. El 75% de la población son usuarios de internet y el 96,3% tiene un celular propio con conexión móvil. El promedio de horas por día en internet asciende a más de 10 horas (Kemp, 2021). El sitio de e-commerce sobre el que se trabajará, se encuentra en el top diez de sitios con más tráfico, de acuerdo con las estadísticas de Semrush (Semrush, 2021).

Como usuarios activos de internet, se reciben constantemente notificaciones (*push notification*) de las aplicaciones instaladas en nuestros celulares / tablets y en nuestra casilla de mail (*email marketing*). Éstas siempre tienen como objetivo interactuar con la aplicación de alguna manera determinada. Es tal la cantidad que recibimos, que es muy relevante para las empresas poder detectar el público objetivo de cada acción de marketing y lograr segmentar a los usuarios de manera eficiente.

El envío de notificaciones de marketing online debe estar contemplado dentro de una estrategia con un fin determinado, para buscar generar confianza y fidelidad con un producto o servicio, confirmar una compra, enviar un documento informativo o comunicar novedades o acciones próximas a realizarse. Por otro lado, los envíos a cada usuario particular deben segmentarse de manera eficiente según sus acciones e intereses, evitando de esta manera el rechazo a futuras notificaciones y evitando generar *spam*.

Esta gran empresa de e-commerce de Latinoamérica analizada en el presente trabajo, está frente a un problema de bajos porcentajes de apertura de sus notificaciones por parte de sus usuarios, lo que deriva en grandes volúmenes de envíos con bajas conversiones y con posibles impactos negativos de *spam* en los individuos.

A lo largo de este trabajo se desarrollará, en el capítulo 2, los conceptos básicos de marketing digital, los principales *KPI´s* para medir el éxito de una campaña, las buenas prácticas para envíos de notificaciones, la problemática específica de esta empresa y el objetivo que viene a solucionar el modelo propuesto de *Machine Learning*.

En el capítulo 3, se desarrollará el planteo teórico del modelo de *Machine Learning* como propuesta al problema, los algoritmos más comunes y utilizados para este tipo de problemática y las principales métricas de medición de éxito de un modelo predictivo. En el capítulo 4, se

abordará el problema, los datos disponibles en la base de datos e ingeniería de atributos, finalizando con los resultados, accionables a partir de los mismos y conclusiones en los capítulos 5 y 6 del presente trabajo.

2. Marco teórico del problema

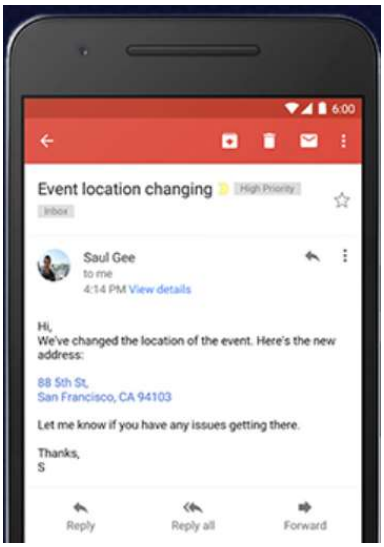
A continuación, se brindarán ciertas consideraciones teóricas necesarias para una mayor comprensión de la lectura:

2.1 Conceptos claves de marketing online

- ***Email marketing***

Se entiende como *email marketing* a aquella acción de enviar un correo electrónico desde una marca a un cliente. Es un canal de marketing que permite a particulares y empresas comunicarse en masa con sus clientes, fanáticos y suscriptores (Waldow, 2012).

Ilustración 1- Email marketing



- ***Push notification***

Se denomina *Push notification*, a aquella acción de enviar una notificación desde una marca al usuario, ya sea al centro de notificaciones de su dispositivo móvil / Tablet o al centro de notificaciones dentro de la página web en su computadora (YUJRA, 2014).

Ilustración 2 - Push notification

- ***Spam***

Con *spam* se hace referencia a los mensajes de correo electrónico y notificaciones dentro de la aplicación, habitualmente de tipo publicitario, que son enviados en grandes cantidades y que terminan perjudicando / generando molestias al receptor.

El problema del *spam* es uno de los puntos más relevantes que se desarrollará a lo largo de este trabajo. Al ser una organización tan grande y con tantas áreas, el envío de *pushes* e *emails* a los usuarios es elevado, y muchos de los usuarios bloquean las notificaciones por exceso de éstas o por no ser de su interés.

- ***Engagement***

Con *engagement* se hace referencia al compromiso, implícito, que se establece entre una marca y su audiencia en las distintas comunicaciones que producen entre sí. Está asociado tanto con el posicionamiento sustentable, como con una buena gestión de marca.

También existe un objetivo de *re-engagement* que consiste en volver a captar la atención e interés de un usuario luego de que estuvo mucho tiempo en inactividad. Por ejemplo, cuando una persona lleva mucho tiempo sin visitar un Sitio Web, sin comprar productos o sin utilizar la aplicación, se le puede enviar un email que lo motive a reingresar.

2.2 Push notification versus email marketing

El *email marketing* es una de las más poderosas herramientas de marketing como canal de comunicación a nivel mundial, pero está muy subutilizado actualmente. Ha sido el principal medio de comunicación electrónica desde la década de 1970. Desde esos entonces numerosos servicios de anuncios, servicios de mensajería instantánea, servicios de chat, redes sociales y "otros asesinos" de correo electrónico han aparecido y desaparecido. Aun así, el email continúa

siendo el principal medio de comunicación a través de internet y aún se encuentra en crecimiento (Paulson, 2019).

Entre 2018 y 2021, el número de emails existentes en el mundo creció de 6,32 billones a 7,71 billones. Esto implica un crecimiento del 22% en sólo 3 años. Actualmente más de 3,7 billones de personas, alrededor de la mitad de la población mundial, tienen acceso a un email, y más de 3 billones de emails considerados no spam son enviado y recibido a cada hora, cada día del año (Radicati, 2013).

La gran ventaja de estas herramientas es poder enviar un mensaje a la audiencia seleccionada para algún propósito específico, a cualquier hora y cualquier día del año sin ningún costo adicional por email enviado. Un estudio de *Direct Marketing Association* demuestra que las empresas ganan un promedio de U$D 34 por cada U$D 1 invertido en marketing por correo electrónico. En otras palabras, el *email marketing* genera un 4,300% de retorno de la inversión o ROI (Clark).

Cuando se crean audiencias desde *Facebook*, *YouTube* o *Instagram*, se está totalmente a merced de los caprichos de una empresa cuyos intereses no están alineados con los propios. Esto nunca será el caso del *email marketing*. Un estudio de McKinsey & Company demuestra que las empresas son 40 veces más propensas a generar un nuevo cliente desde el *email marketing* que a través de una red social (Aufreiter, 2014).

La ventaja de la *push notification* frente al *email* es que, si se está corriendo una campaña de promoción o venta, se tiene el potencial de poner esa oferta en frente de los usuarios de manera casi automática. No se necesita que abran su casilla de correo electrónico y encuentren el mensaje entre un montón de otros recibidos a diario.

De igual manera, las dos herramientas permiten enviar un mensaje a la audiencia cuando se desee compartir algo. La tecnología detrás del *email marketing* y las *pushes notifications* son diferentes, pero pueden trabajar en conjunto para ayudar a conseguir los objetivos de la compañía. Se puede pensar a las *pushes notifications* como una segunda lista de *emails*, porque básicamente consiste en otro grupo de usuario que pueden recibir una notificación, pero indefectiblemente ocurre una superposición entre ambas listas, y es posible que se envíe el mismo mensaje al mismo usuario por ambos canales de comunicación.

2.3 Usos de automatización del *Email Marketing y Push Notification*

Es recomendable mantener conversaciones directas y personalizadas con los clientes, garantizando calidad y un corto tiempo de respuesta. Llevar a cabo una estrategia de forma mucho más simple y efectiva, es posible. Implementando técnicas de automatización del *email marketing* y *pushes notifications*, se gana eficiencia y se reducen costos (Castillo, 2022).

Este tipo de envíos son muy eficientes para conseguir nuevos clientes y fidelizar a los más antiguos. Se configuran una sola vez y se envían automáticamente en los momentos que se hayan definido o cuando el usuario cumpla con un criterio determinado.

En este caso, la automatización del modelo de clasificación de *Machine Learning* consiste en correr un algoritmo para la totalidad de usuarios de cada país. Esto va a devolver una tabla con la probabilidad para cada usuario de publicar en el *Marketplace*. A su vez, de manera automática, se alimentará a una tabla de *Teradata (SQL)* que será consultada de manera automática por la herramienta de envío de *pushes* e *emails* para segmentar por orden descendente en probabilidad y definir el número de usuarios que va a recibir las notificaciones.

No sólo se tendrá en cuenta esta probabilidad a la hora del envío, sino también se segmentará y excluirá en la misma consulta los usuarios que no tengan la aplicación instalada y logeada en sus dispositivos móviles y aquellos que no abren notificaciones en un determinado tiempo fijado como punto de corte. Estos dos filtros, tener la aplicación activa en sus dispositivos y porcentaje de apertura de notificaciones, son buenas prácticas del marketing digital.

2.4 ¿Cuáles son los beneficios de la automatización?

Si tuviese que realizarse de forma manual un envío semanal, se debería correr el modelo en primer lugar, luego subir la tabla a *Teradata (SQL)*, realizar la consulta y los filtros determinados y armar el título y cuerpo de la notificación cada vez que se envía. Pero con el *push / email* automatizado sólo es cuestión de programarlo una vez y el flujo de envíos funcionará de forma automática. Ayuda a desarrollar procesos de marketing complejos que, de forma natural, serían dificultosos o no se podrían hacer.

Por otro lado, es 100% efectivo y colabora con la reducción de costos. Los empleados o encargados de comunicación no tendrán que destinar tiempo enviando cada correo de manera manual, sino que podrán ocuparse de otras tareas con mayor impacto para el negocio, que este tipo de tareas más operativas. Asimismo, proporciona un mayor control sobre las acciones de Marketing ya que podrán acceder a un análisis detallado de los resultados y evolución a través del tiempo como una constante en la segmentación de usuarios objetivo y cuerpo del mensaje. También, se logra control sobre los horarios y días de la semana en el cual se hacen los envíos. Otro beneficio es la incrementalidad de las interacciones con los usuarios.

En cuanto a *engagement*, el envío de *email marketing* y notificaciones automatizadas genera más interacciones, derivando en un posible aumento de las conversiones y *open rate*. Los hacen sentir parte de la marca.

Los beneficios de la automatización son:

- Es ideal para dar a conocer un producto o servicio.
- Mantiene vivo el interés de tu audiencia ya que brinda un acompañamiento desde primer contacto hasta transacción final.
- Se puede automatizar las notificaciones según las características del público para llegar a cada usuario con un mensaje de calidad y relevante basado en sus preferencias, intereses y comportamientos.

- Se puede configurar decenas de cadenas simultáneas sin finalización, lo cual permite mantener una relación a lo largo del tiempo con los clientes.
- Lograr que los usuarios se sientan más cercanos a la marca, mediante la automatización se le puede ofrecer beneficios o promociones exclusivas.

2.5 Buenas prácticas de marketing digital

Además de la automatización de los envíos, es recomendable seguir algunas prácticas que suelen tener impacto directo en los resultados (Paulson, 2019). Cabe destacar en este punto, que son prácticas probadas por la empresa en la práctica y confirmadas con los resultados de los experimentos realizados. Estas recomendaciones son:

- ➢ El horario óptimo de envío es cuando los usuarios están frente a su computadora o dispositivo móvil. Mediante *Google Analytics* es posible determinar el horario en que los usuarios son más activos dentro del sitio web o aplicación móvil.
- ➢ El título de un push no debe tener más de 30 caracteres y la descripción de este no debe superar los 100 caracteres.
- ➢ Enviar una prueba para previsualizar la notificación, previo a su envío. Corroborar que todos los enlaces y redireccionamientos funcionen correctamente.
- ➢ Usar botones que llamen a la acción para aumentar el ratio de clics.
- ➢ Usar iconos customizados en las notificaciones. Esto ayuda a aumentar el ratio de clics y *engagement*. Si se envía para una categoría determinada, poner una previsualización de imagen de los productos que se están promocionando.
- ➢ No enviar la misma notificación, con el mismo mensaje, dos veces. Este punto es fundamental, dado que muchas de las herramientas de automatización no proveen la opción de ir mandando aleatoriamente un título y mensaje dentro de una lista de opciones. Cuando se envía el mismo mensaje dos veces al mismo usuario, las conversiones tienden a disminuir, se genera menor impacto en el lector.
- ➢ La segmentación de audiencias es requisito necesario. Sin esto, las compañías no son capaces de segmentar a sus usuarios y terminan saturándolos de información irrelevante para ellos. Existe una positiva correlación (0,462) entre “recibo información relevante” y “mantengo una relación positiva con la marca” (Ferreira, 2016).

2.6 Métricas para medición de éxito en campañas

A la hora de definir el éxito o fracaso de las campañas de marketing digital, se observan algunos *KPI’s (key performance indicator)* o indicadores claves de actuación que son los que definen el resultado final de una acción de push o email.

Las métricas que se usan para evaluar una campaña son:

- ***Exposición:*** porcentaje de usuarios que reciben el *push o email* del total de la base a la que fue enviada. Mientras más alto es la exposición, mejor segmentada es la base de envío.

$$exposición = \frac{recibidos}{enviados} \quad (1)$$

- ***Open rate:*** porcentaje de los usuarios que abrieron el *push o email* del total que lo recibieron. Es un buen parámetro para definir si se está enviando contenido de interés, si el *target* de usuarios está bien definido y si no se está generando *spam* a la base de usuarios. Mientras más alto es el *open rate*, mejor segmentada es la base de envío.

$$open\ rate = \frac{abiertos}{recibidos} \quad (2)$$

- ***Conversiones:*** número absoluto de publicaciones (en el caso de nuestro modelo) que se obtiene del total de usuarios que enviamos el *push o email*.

- ***Lift:*** mide los ratios de conversión (conversiones / recibidos) de cada grupo y los compara para obtener la diferencia en los ratios de conversión, el cual es ajustado por la exposición. Mientras mayor es el *lift*, mayor es el impacto del envío a nivel de conversiones.

$$lift = \frac{\left[\frac{public_{test}}{users_{test}} - \frac{public_{control}}{users_{control}}\right]}{\frac{users_{recibidos}}{users_{test}}} \quad (3)$$

- ***Conversiones incrementales:*** número incremental de publicaciones que se obtienen, atribuidas al envío de *push o email*. A veces, los envíos son a personas que, sin importar la notificación, hubieran publicado de igual manera. Las conversiones incrementales hacen referencia a las personas que no hubieran publicado sin haber recibido el push o email.

$$public_{inc} = public_{test} - \left[public_{control} * \left(\frac{users_{test}}{users_{control}}\right)\right] \quad (4)$$

2.7 Contexto

Hoy en día, las acciones de marketing que se realizan para incentivar a los usuarios a publicar un ítem x de una categoría determinada y dentro del *Marketplace* se realizan tanto de manera manual como de manera automática. Hace más de dos años, esta gran empresa de e-

commerce de Latinoamérica, viene segmentando la base de usuarios según normas de negocio, y enviando pruebas de *pushes e emails* a diferentes subconjuntos de usuarios que cumplen con determinada característica, o que realizaron determinada acción y, mediante un grupo de control y otro de testeo, se mide si el envío es estadísticamente significativo entre los dos grupos mediante un *t-test*.

Además de confirmar que las conversiones absolutas son estadísticamente significativas, también se miden métricas como el porcentaje de notificaciones abiertas (*open rate*) y la incrementalidad del envío.

Cuando se hace referencia a lógica de negocio, se hace referencia a supuestos que deberían cumplirse dada la racionalidad del comportamiento de los individuos dentro de la plataforma. A modo de ejemplo, si se busca que un usuario publique un ítem de la categoría y en el *Marketplace*, es relevante poder identificar si los usuarios tienen este bien o no. Se podría suponer que, si alguien está visitando constantemente la categoría de y, posiblemente esté pensando en comprar su primer ítem de esta categoría o quizás (estos son los usuarios que nos interesan) están pensando en un cambio de modelo a algo más moderno o de mayor confort. Cuando se identifica que ciertos usuarios visitan esta categoría, se les envía una notificación para incentivarlos a que publiquen su ítem de manera gratuita.

Luego de identificar un segmento de usuarios propensos a publicar en la categoría de relevancia del experimento, se hacen algunas pruebas de *A/B testing* para obtener la significancia estadística del envío. En caso de ser significativo, el experimento se replica, y si nuevamente se consigue un resultado significativo, se envía un push/email de manera automática a lo largo del tiempo y de manera recurrente.

Actualmente, los envíos recurrentes en Brasil (mercado bajo análisis) son las siguientes tres campañas:

- Envío a usuarios que visitaron la categoría y para que publiquen su ítem de esa misma categoría de manera gratuita.
- Envío a usuarios que visitaron categorías relacionadas a la categoría y para que publiquen su ítem de categoría y de manera gratuita.
- Envío a usuarios que visitaron la categoría de insumos de la categoría y para que publiquen su ítem de categoría y de manera gratuita.

Este trabajo, que se viene haciendo hace más de dos años, se encuentra completamente documentado con las segmentaciones y resultados de cada experimento y envío. Estas segmentaciones, fueron las bases del modelo de *machine learning* y las primeras variables explicativas que se probaron dentro del mismo por ya saber que eran relevantes de manera aislada.

Al momento de generar envíos, no solamente se probaron diferentes segmentaciones de usuarios, sino también diferentes horarios y días de la semana, diferente *wording* (o mensajes) en los envíos, tanto en el título como en el cuerpo de la notificación, como también la personalización con características particulares de cada usuario que la recibe. Con relación a este último punto, se confirmó que existe un mayor número de conversiones absolutas y *open*

rate de los usuarios si el *push* posee el nombre de pila en el título de este (la personalización genera *engagement* en los usuarios).

El total de *pushes* e *emails* que se envían de manera recurrente fueron implementados por un equipo de tecnología, fuera de la herramienta que utiliza el equipo de marketing para realizar los envíos. Esto se refleja, en que la totalidad de los envíos, aparecen bajo una sola línea o concepto en los tableros de control que se utilizan para el seguimiento de las campañas a través del tiempo.

2.8 Problemática

El problema que están experimentando muchas de las unidades de negocio que conforman esta gran multinacional de e-commerce es el bajo porcentaje de usuarios que están abriendo las notificaciones que son enviadas. Esto genera una alerta a nivel global de la empresa al no estar cumpliendo con las métricas de performance esperadas. En el afán de ser la número uno en casi todas las categorías del *Marketplace*, compitiendo con otras empresas, se crearon diversas campañas de marketing online que son enviadas de manera recurrente y automática, de manera diaria o semanal, según el caso.

Cuando estas campañas fueron creadas, la herramienta que disponía la empresa no era capaz de autoalimentarse desde la base de datos a partir de una *Query*, por lo que se crearon envíos por fuera de la herramienta actual bajo una misma campaña, generando la imposibilidad de medir los envíos de manera aislada.

No tener una estrategia sólida y bien planificada de marketing online genera ciertos problemas y dificultades que pueden impactar de manera negativa tanto en el negocio como en los clientes. Las dificultades más comunes se encuentran a la hora de medir el impacto de las acciones de marketing, así como medir la relevancia de nuevas acciones frente a las ya productivas.

Al tener el total de envíos recurrentes bajo la misma campaña para su medición y performance, se pierde la capacidad de detectar irregularidades en la performance de los envíos de manera individual. Esto puede afectar negativamente en los usuarios que reciben *pushes* e *emails*. Suponiendo que el comportamiento de los usuarios a través de los años no se mantuvo constante, y se continúa enviando notificaciones a personas que hoy ya no están interesadas en recibirlas, se estaría generando *spam*.

Asimismo, como todos los envíos recurrentes siguen una lógica de negocio, luego de dos años de pruebas y envíos, las ideas de nuevas segmentaciones ya casi no tienen fuerza y empiezan a perderse los usuarios diferenciales de estas acciones particulares y repercuten en los resultados de negocio. Esto se observa cuando grandes volúmenes de notificaciones son enviados, y se consiguen muy pocas conversiones, acompañadas de un muy bajo porcentaje de *open rate*.

2.9 Objetivo

El propósito de este trabajo de tesis es poder asignar una probabilidad a cada usuario de uno de los e-commerce más grandes del mundo, de publicar un ítem x de una categoría determinada y en el *Marketplace* de la plataforma, influenciado por acciones de *push* e *email marketing*. Con el cálculo de esa probabilidad, se podrían generar y planificar estrategias de marketing online más eficientes, con una segmentación acorde al mensaje y usuarios propensos a publicar.

La asignación de la probabilidad de publicar se realiza mediante el modelado de un algoritmo de *Machine Learning* supervisado de clasificación. Con esto, se lograría disminuir el *spam* de notificaciones que no son de interés para los usuarios y a su vez, generar *engagement* positivo con la plataforma y sus anuncios. Además, permitiría incluir atributos o segmentaciones por fuera de la lógica de negocio y obtener conversiones incrementales que no se habrían dado con experimentos manuales. También, supondrá un reemplazo de todos los envíos automáticos, recurrentes y manuales que están actualmente en producción, por un único y unificado envío, permitiendo detectar irregularidades en la performance a través del tiempo de manera más simple.

El objetivo final de este trabajo consistirá en probar la eficiencia del modelo de *machine learning* de clasificación de reemplazar la estrategia de marketing digital de hoy, de varias campañas en paralelo, por un solo y unificado envío o campaña. A su vez, el fin del algoritmo desarrollado, es poder devolver por cada usuario que tiene en el *Marketplace*, una probabilidad de publicar un ítem en una determinada categoría.

Concatenando estos dos objetivos, se podría generar una eficaz estrategia de marketing digital, con fuerte sustento en los datos, que esté alineado con las métricas objetivo de la empresa y pueda ser flexible y medible a través del tiempo, en un mercado tan volátil como son las empresas de e-commerce hoy en la actualidad.

3. Metodología

3.1Marco teórico de modelos de machine learning

La mayoría de los problemas de aprendizaje automático se pueden encasillar en una de estas tres categorías: supervisado, no supervisado y por refuerzos.

Un problema de ***aprendizaje supervisado*** implica que para cada set de variables explicativas $x_i, i = 1, \dots, n$ existe una respuesta asociada medida y_i. Se desea ajustar un modelo que relacione la variable de respuesta con los predictores, con el objetivo de predecir con precisión las futuras observaciones o encontrar una mejor comprensión de la relación entre la respuesta y los predictores.

Las dos grandes familias de algoritmos supervisados son:

- Los algoritmos de regresión cuando el resultado a predecir es un atributo numérico.
- Los algoritmos de clasificación cuando el resultado a predecir es un atributo categórico.

En contraste, existen los problemas de ***aprendizaje no supervisado***, que presentan una situación un poco más compleja donde tenemos para cada observación $i = 1, \dots, n$ un vector de mediciones x_i pero que no está asociado a ninguna variable de respuesta y_i. A priori no se conoce ningún valor objetivo o de clase, ya sea categórico o numérico. El aprendizaje no supervisado está dedicado a las tareas de agrupamiento, también llamadas *clustering* o segmentación, donde su objetivo es encontrar grupos similares en el conjunto de datos.

Existen dos grupos principales de métodos o algoritmos de agrupamiento:

- Los métodos jerárquicos, producen una organización jerárquica de las instancias que forman el conjunto de datos, posibilitando de esta forma distintos niveles de agrupación.
- Los métodos particionales o no jerárquicos generan grupos de instancias que no responden a ningún tipo de organización jerárquica.

En este caso, como se sabe el comportamiento de cada usuario dentro de la plataforma y si efectivamente publicó o no, es un problema de modelado supervisado de clasificación. Se posee el conocimiento de la variable de respuesta y.

El desafío principal es la generación de variables explicativas de relevancia entendiendo la lógica de negocio y probando algunas variables por fuera de esa lógica. Según los resultados obtenidos, se definirá un punto de corte de probabilidad para determinar los usuarios que serán *target* de notificaciones, evitando el *spam* al resto de la base.

Dentro de la gama de algoritmos de aprendizaje automático existentes se destacan actualmente dos: *Random Forest* y *XGBoost*. Ambos han adquirido gran popularidad. *Random Forest* es un algoritmo que surgió hace casi veinte años y se utiliza ampliamente por el balance que ofrece entre complejidad y resultados. Por su parte, *XGBoost* es un algoritmo que ha despertado gran interés, dado que, aunque es relativamente nuevo es considerado actualmente el estado del arte en algoritmos de aprendizaje automático por sus resultados.

Para el desarrollo de este modelo, se usa el algoritmo de *XGBoost*.

3.2 Conceptos básicos de XGBoost

Previo a hablar del modelo *XGBoost* en particular, es necesario conocer algunos conceptos previos para su mejor comprensión.

Los *árboles de decisión* son modelos predictivos formados por reglas binarias (si/no) con las que se consigue repartir las observaciones en función de sus atributos y predecir así el valor de la variable respuesta.

Muchos métodos predictivos generan modelos globales en los que una única ecuación se aplica a todo el espacio muestral. Cuando el caso de uso implica múltiples predictores, que interaccionan entre ellos de forma compleja y no lineal, es muy difícil encontrar un único modelo global que sea capaz de reflejar la relación entre las variables. Los métodos estadísticos y de *machine learning* basados en árboles engloban a un conjunto de técnicas supervisadas no paramétricas que consiguen segmentar el espacio de los predictores en regiones simples, dentro de las cuales es más sencillo manejar las interacciones.

Por otro lado, es necesario conocer la idea detrás del *boosting*. El mismo consiste en:

1. Entrenar un árbol de decisión y obtener sus predicciones.
2. Detectar las observaciones de entrenamiento que predijo mal.
3. Construir un segundo árbol de decisión que se enfoque en aquellas observaciones que el primero predice mal.
4. Tomar como predicción final alguna combinación de las predicciones de cada árbol.

Boosting hace uso de esta idea. A los modelos que combinan las predicciones de modelos más pequeños se los conoce como modelos de ensamble.

En términos más formales, el algoritmo de *boosting* sigue estos pasos:

1. Setear $\hat{f}(x) = 0$ y $r_i = y_i$ para todo i perteneciente al set de entrenamiento. Predice para todas las variables 0, entonces el error es igual a la predicción.
2. Para $b = 1,2,\ldots,B$ (cantidad de árboles) repetir los siguientes pasos:
   a. construir un árbol $\widehat{f^b}$ con d cortes ($d+1$ hojas) para los datos de entrenamiento (X, r)
   b. actualizar $\hat{f}$ añadiendo el nuevo árbol generado

$$\hat{f}(x) \leftarrow \hat{f}(x) + \lambda \widehat{f^b}(x)$$

   c. actualizar los residuales

$$r_i \leftarrow r_i - \lambda \widehat{f^b}(x)$$

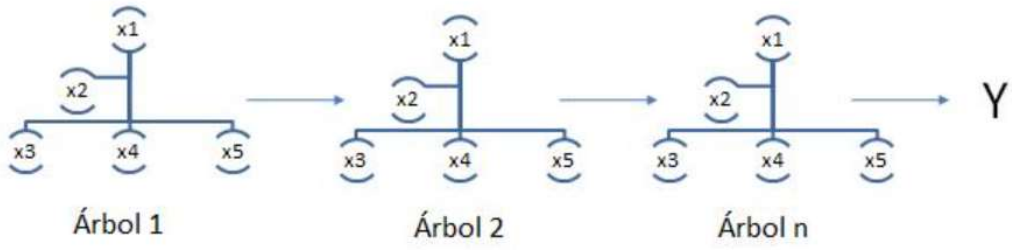
3. El resultado final del modelo de boosting:

$$\hat{f}(x) \leftarrow \sum_{b=1}^{B} \lambda \widehat{f^b}\ (x)$$

3.3 XGBoost

XGBoost es un *boosting* de árboles de decisión, por ende, lo que hace es un ensamble de *boosting* de distintos árboles de decisión, donde cada árbol siguiente intenta predecir y cambiar lo que el anterior no pudo predecir correctamente, de modo que, combinando varios árboles de decisión se logrará una buena predicción. Es un algoritmo que se utiliza tanto para problemas de regresión como de clasificación:

Ilustración 3 - Algoritmo XGBoost



Fuente: (Zúñiga, 2020)

El *algoritmo de XGBoost* funciona de la siguiente manera, usando la idea de *boosting*:

1. Se obtiene un árbol inicial F_0 para predecir la variable objetivo "y", el resultado se asocia con un residual $y - F_0$.
2. Se obtiene un nuevo árbol h_1 que se ajusta al error del paso previo.
3. Los resultados de F_0 y h_1 se combinan para obtener el árbol F_1, donde el error cuadrático medio de F_1 será menor que el de F_0:

$$F_1(x) < -F_0(x) + h_1(x)$$

4. Este proceso se sigue iterativamente hasta que el error es minimizado lo más posible de la siguiente forma:

$$F_m(x) < -F_{m-1}(x) + h_m(x)$$

Las *principales ventajas* del algoritmo XGBoost son:

a. Puede manejar grandes bases de datos con múltiples variables.
b. Puede manejar valores nulos.
c. Sus resultados son muy precisos.
d. Excelente velocidad de ejecución.

Por otra parte, sus *principales desventajas* son:

a. Puede consumir muchos recursos computacionales en grandes bases de datos, por lo que se recomienda antes de aplicar esta técnica, determinar cuáles son las variables que aportarán más información a fin de considerar sólo dichas variables en la obtención del modelo.

b. Se deben ajustar correctamente los parámetros del algoritmo a fin de minimizar el error de precisión y evitar sobreajuste del modelo (lo que puede darse si se maneja un número muy grande de árboles - *overfitting*).

c. Solo trabaja con vectores numéricos, por lo que se requiere convertir previamente los tipos de datos no numéricos a numéricos.

3.4 Métricas de éxito de un modelo de *Machine Learning*

En esta sección, se describen varias métricas usadas en la industria actualmente, que definen la calidad de un modelo. En este trabajo se desarrollan varias, dado que cada una aporta un punto de vista diferente y complementario a las demás. A continuación, se presentarán las principales.

Matriz de confusión

En el campo de la inteligencia artificial y el aprendizaje automático una matriz de confusión es una herramienta que permite visualizar el desempeño de un algoritmo de aprendizaje supervisado. Cada columna de la matriz representa el número de predicciones de cada clase, mientras que cada fila representa a las instancias en la clase real, o sea en términos prácticos nos permite ver qué tipos de aciertos y errores está teniendo nuestro modelo a la hora de pasar por el proceso de aprendizaje con los datos.

Ilustración 4 - Componentes de la matriz de confusión

		predicción	
		0	**1**
realidad	**0**	TN	FP
	1	FN	TP

Fuente: (Heras, 2020)

Se presentan dos variables distintas, las predicciones por si publica o no publica y lo que realmente se observa en la realidad, si el usuario efectivamente publicó o no. Esto genera un cuadro de doble entrada con cuatro opciones posibles escenarios, que son los siguientes:

- **Verdadero positivo (TP):** *Cuadrante de abajo a la derecha.* El valor real es positivo y la prueba predijo también que era positivo. Son los usuarios que se predijo que, si publicaron, y efectivamente publicaron.

- **Verdadero negativo (TN):** *Cuadrante de arriba a la izquierda.* El valor real es negativo y la prueba predijo también que el resultado era negativo. Son los usuarios que se predijo no publicaron, y efectivamente no publicaron.

- **Falso negativo (FN):** *Cuadrante de abajo a la izquierda.* El valor real es positivo, y la prueba predijo que el resultado es negativo. Son usuarios que se predijo que no publicaron, pero en la realidad sí publicaron.

- **Falso positivos (FP):** *Cuadrante de arriba a la derecha.* El valor real es negativo, y la prueba predijo que el resultado es positivo. Son usuarios que se predijo que publicaron, pero en la realidad no publicaron.

La matriz puede expresarse en valores absolutos, como en porcentajes de aciertos y errores de los verdaderos valores observados. La ventaja que tiene representar la matriz con valores absolutos es que permite obtener algunas otras métricas de performance muy interesantes que se desprenden de este resultado.

Accuracy

La exactitud o *accuracy* (en inglés) del modelo se refiere a lo cerca que está el resultado de las predicciones del verdadero valor observado en la realidad. En términos estadísticos, la exactitud está relacionada con el sesgo de una estimación. Se calcula de la siguiente manera:

$$Accuracy = \frac{VP + VN}{VP + FP + FN + VN}$$

Precision

Con la métrica de precisión se puede medir la **calidad** del modelo de *machine learning* en tareas de clasificación. En el modelo, se refiere a que la precisión es la respuesta a la pregunta ¿qué porcentaje de los clientes que se contacten vía *push* o *email* estarán interesados en efectivamente publicar en el *Marketplace*? Se calcula de la siguiente manera:

$$Precisión = \frac{VP}{VP + FP}$$

Recall

La métrica de exhaustividad o *recall* (en inglés) nos va a informar sobre la **cantidad** que el modelo de *machine learning* es capaz de identificar. En el modelo, se refiere a que la exhaustividad es la respuesta a la pregunta ¿qué porcentaje de los clientes que están interesados en publicar son posibles de identificar? Se calcula de la siguiente manera:

$$Recall = \frac{VP}{VP + FN}$$

F1 Score

El valor F1 se utiliza para combinar las medidas de *precision* y *recall* en un sólo valor. Esto es práctico porque hace más fácil el poder comparar el rendimiento combinado de la precisión y la exhaustividad entre varias soluciones.

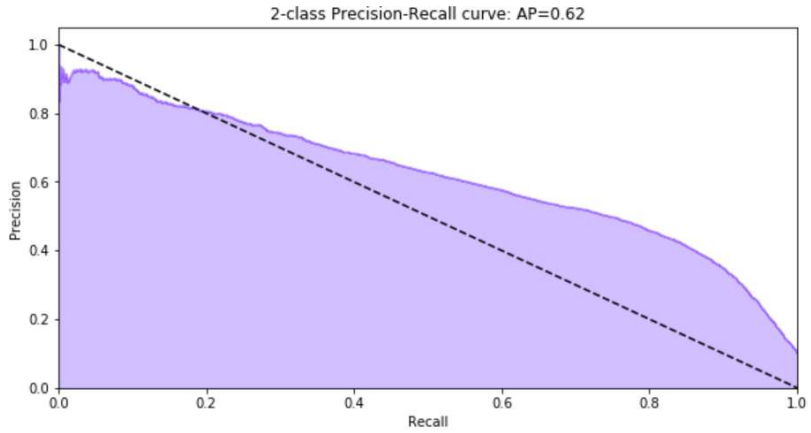
F1 se calcula haciendo la media armónica entre la precisión y la exhaustividad:

$$F1 = 2 * \frac{precisión * recall}{precisión + recall}$$

Precision – Recall Curve (PR Curve)

Las métricas de *precision* y *recall* están relacionadas de manera que, si se entrena al clasificador para aumentar la *precision*, disminuirá el *recall* y viceversa (Ramírez, 2018). La curva PR es el resultado de dibujar la gráfica entre el *precision* y el *recall*. Esta gráfica permite ver a partir de qué *recall* tenemos una degradación de la precisión y viceversa. Lo ideal sería una curva que se acerque lo máximo posible a la esquina superior derecha (alta *precision* y alto *recall*).

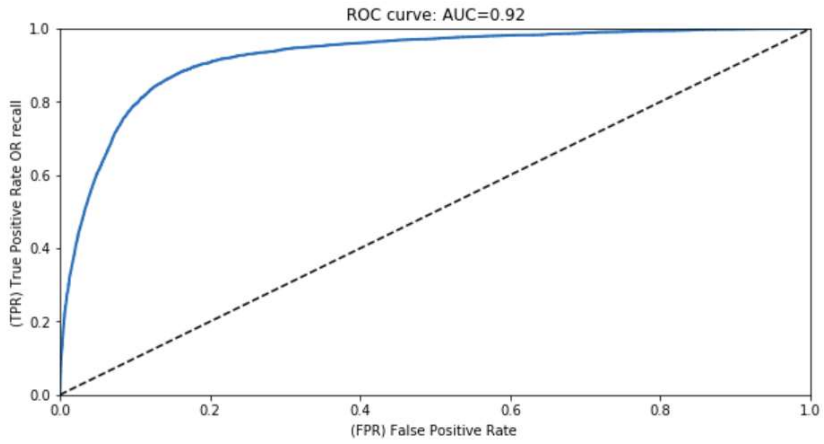
Ilustración 5 - PR Curve



ROC curve

La curva ROC es similar a la curva PR del punto anterior, pero cambian algunos valores. Relaciona el *recall* con el ratio de los falsos positivos, lo que implica, que relaciona la sensibilidad de nuestro modelo con clasificar los negativos como positivos. Esto tiene sentido, pensando en que generalmente, si aumenta el *recall*, nuestro modelo tenderá a ser más optimista e introducirá más falsos positivos en la clasificación.

Ilustración 6 - Curva ROC - Receiver Operating Characteristic



En las curvas ROC, interesa que la curva se acerque lo máximo posible a la esquina superior izquierda de la gráfica, de manera que al aumentar la sensibilidad (*recall*) no haga que aparezcan más falsos positivos en las predicciones.

Feature importance

Otro de los resultados más importantes que brinda el modelo de *XGBoost*, es la importancia de atributos o "*feature importance*". Este resultado es de gran utilidad para el modelo, y posiblemente sea el más potente. Devuelve el orden de relevancia de las variables explicativas, en función de su poder predictivo sobre la variable a predecir.

Permutation Feature importance

Una métrica que se desprende de la importancia de los atributos es la importancia de la permutación de los atributos. Mide la importancia de un atributo calculando el aumento en el error de predicción del modelo después de permutar ese atributo en particular. Un atributo es "importante" si cambiar sus valores aumenta el error del modelo, porque en este caso el modelo se basó en el atributo para la predicción. Un atributo es "no importante" si cambiar sus valores deja el error del modelo sin cambios, porque en este caso el modelo ignoró el *feature* para la predicción.

4. Abordado del problema

En esta sección se exponen los datos disponibles de los usuarios para abordar el problema planteado a través de una solución de *Machine Learning*.

4.1 Datos

Para solucionar esta problemática particular y lograr los objetivos propuestos, se cuenta con una base de datos extensa de todos los usuarios que forman parte de la plataforma, sus datos geográficos, demográficos, nivel socio económico, y comportamiento dentro de la *app* y *web*.

Para el armado del modelo de *machine learning* de clasificación se utilizaron los siguientes datos de un período de 12 meses con ventana de corte móvil, para estos grupos de usuarios en particular:

- ***Grupo de usuarios que efectivamente publicaron*** un ítem x de la categoría de interés y. Se tomaron usuarios únicos, sin diferenciar por cantidad de publicaciones, pero con la condición de ser usuarios particulares y no comercios especializados.

- ***Grupo de usuarios que no publicaron*** un ítem x de la categoría de interés y. Se tomó como único requisito que hayan realizado al menos una compra en el transcurso del año 2021, para confirmar que efectivamente son usuarios que utilizan el *Marketplace* y pueden ser considerados usuarios activos dentro del mismo.

Los datos que se utilizaron de ambos grupos se encuentran representados en la **¡Error! No se encuentra el origen de la referencia.**. Es un extracto del set de datos original, con algunas modificaciones en los valores y datos con el fin de ser anonimizada. Las variables representadas son las siguientes:

- ✓ **X1:** Género. Información declarada por el usuario → variable categórica.
- ✓ **X2:** Ingresos. Información declarada por el usuario → variable categórica.
- ✓ **X3:** Edad. Información calculada a partir de la fecha de nacimiento → variable categórica.
- ✓ **X4:** Geolocalización. Dirección declarada por el usuario para su último envío en los últimos 2 años → variable categórica.
- ✓ **X5 y X7:** Cantidad de interacciones con categorías afines a y → variable numérica.
- ✓ **X6:** Cantidad de interacciones con la categoría y → variable numérica.
- ✓ **X8:** Diferenciar usuarios de los productos complementarios que ofrece el *Marketplace* → variable numérica.
- ✓ **X9:** Diferenciar vendedores a través de los productos complementarios que ofrece el *Marketplace* → variable numérica.
- ✓ **X10:** Nivel del programa de fidelidad → variable numérica.
- ✓ **X 11:** Monto de las compras de los últimos 12 meses → variable numérica.

- ✓ **X12:** Cantidad de ítems que compró en los últimos 12 meses → variable numérica.
- ✓ **X13:** *Open rate* histórico de notificaciones en general → variable numérica.
- ✓ **X14:** Año en que el usuario visitó la categoría y por primera vez.
- ✓ **X16:** Mes en que el usuario visitó la categoría y a analizar por primera vez.
- ✓ **X18:** Cantidad de meses desde que ingresó por primera vez a la categoría y. Esto nos devuelve, básicamente, el tiempo que pasó desde que el usuario sabe que puede vender y comprar ese tipo de productos dentro de la plataforma → variable numérica.
- ✓ **X15:** Año en que el usuario visitó la categoría y por última vez.
- ✓ **X17:** Mes en que el usuario visitó la categoría y a analizar por última vez.
- ✓ **X19:** Cantidad de meses desde que ingresó por última vez a la categoría y. Característica también conocida como recencia → variable numérica.

Cabe destacar en este punto, que dada la gran potencialidad del motor de consulta de la base de datos relacionales utilizado (SQL), muchas de las construcciones de variables y limpieza de los datos fueron realizadas directamente en la extracción de las bases.

En cuanto a la selección de variables, se tomaron la totalidad de los datos demográficos y geográficos disponibles de los usuarios. Por el lado de las variables de comportamiento, se basó su elección en las pruebas de *A/B Testing* realizadas anteriormente en los experimentos manuales, y se sumaron algunas otras variables bien genéricas utilizadas en la construcción de otros modelos de similar índole dentro de la compañía.

Tabla 1 - Dataset anonimizado y modificado de los atributos del modelo

ID	Y	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	X16	X17	X18	X19
1	0	36	no declara	no declara	SP	1	3	116	0	0	4	4081,94	19	1,5	2018	2021	1	12	47	0
2	1	32	hasta 5,000	M	SP	1	4763	1177	0	0	3	906,36	9	12	2018	2021	1	12	47	0
3	1	44	hasta 10,000	M	No declara	1	2679	1124	1	0	4	804,9	7	7,1	2018	2021	1	12	47	0
4	1	48	hasta 2,000	M	Noreste	1	15	22	0	0	1	29,55	1	2,4	2017	2021	2	11	58	1
5	1	44	hasta 2,000	F	No declara	0	2	5	0	0	1	0	0	3,5	2018	2021	1	5	47	7
6	1	49	hasta 10,000	M	Sudeste	1	7	864	1	0	3	2334,37	18	7,6	2017	2021	1	12	59	0
7	1	52	hasta 7,000	M	Noreste	1	100	2820	1	0	4	2839,11	26	47,1	2017	2021	1	12	59	0
8	1	39	hasta 10,000	M	SP	0	355	162	0	0	2	288,87	5	6,4	2017	2021	1	12	59	0
9	1	48	hasta 10,000	M	SP	1	60	110	0	0	4	5619,63	26	13,1	2017	2021	1	12	59	0
10	1	34	hasta 10,000	M	Noreste	1	37	589	1	0	3	937,28	9	10,3	2017	2021	1	12	59	0
11	1	48	hasta 7,000	no declara	Noreste	1	11	54	1	1	6	4370,05	8	13,3	2020	2021	1	12	23	0
12	1	44	hasta 10,000	M	No declara	1	63	468	1	0	6	42412,4	183	1,1	2017	2021	1	12	59	0
13	1	40	hasta 2,000	F	No declara	1	124	237	1	0	3	2349,55	37	3,6	2020	2021	1	12	23	0
14	1	34	hasta 5,000	M	SP	1	11	365	1	0	4	3835,55	29	43,9	2019	2021	1	12	35	0
15	1	50	hasta 10,000	M	SP	1	178	241	0	0	3	3223,4	25	42,2	2017	2021	1	12	59	0
16	1	42	hasta 3,000	M	SP	1	134	616	1	1	6	14653,4	102	5,1	2017	2021	1	12	59	0
17	1	40	hasta 5,000	M	SP	1	20	99	1	0	3	1567,11	14	5,6	2020	2021	2	12	22	0
18	1	29	hasta 7,000	M	No declara	1	197	713	0	0	3	1587,28	28	2,1	2017	2021	1	12	59	0
19	1	50	hasta 10,000	M	SP	1	1311	175	1	0	2	145,59	5	3,9	2017	2021	1	12	59	0
20	1	75	hasta 10,000	M	SP	1	494	1220	1	1	5	6418,5	32	11,7	2017	2021	1	12	59	0
21	1	43	hasta 7,000	M	Sudeste	1	60	92	1	0	6	456,54	4	2,5	2017	2021	1	12	59	0
22	1	68	hasta 3,000	M	Sudeste	1	1	155	1	0	4	4034,56	15	10,5	2017	2021	1	12	59	0
23	1	48	hasta 1,000	M	Sudeste	0	2	8	0	0	4	6572,47	37	0,9	2021	2021	6	6	6	6
24	1	41	hasta 10,000	F	SP	0	3	0	0	0	2	0	0	2,5	2021	2021	4	5	8	7
25	1	45	hasta 10,000	M	Sudeste	1	23	405	0	0	4	3758,09	21	10,2	2017	2021	1	12	59	0
26	1	33	hasta 10,000	F	SP	0	63	4	1	0	3	381,56	7	3,9	2020	2021	2	12	22	0
27	0	31	no declara	no declara	Sudeste	1	552	784	1	0	6	62993,3	184	16,5	2020	2021	1	12	23	0
28	1	44	hasta 10,000	M	SP	1	4	7	0	0	4	3513,32	21	7,5	2017	2021	1	12	59	0
29	0	37	hasta 10,000	F	Sur	0	0	0	1	0	5	9367,9	93	2,3	2017	2017	5	5	55	55
30	1	41	hasta 7,000	M	No declara	1	37	142	1	0	4	4652,42	16	3,8	2017	2021	1	12	59	0
31	1	41	hasta 2,000	M	SP	1	779	304	0	0	4	2437,94	16	2	2019	2021	1	12	35	0
32	0	47	hasta 10,000	F	Sudeste	0	0	12	1	0	4	4348,13	34	23,3	2017	2017	1	1	59	59
33	1	42	hasta 10,000	M	SP	1	32	86	0	0	4	6865,44	25	1,6	2019	2021	1	12	35	0
34	0	50	no declara	no declara	SP	0	0	45	0	0	1	0	0	1,3	2018	2020	2	8	46	16
35	1	31	hasta 7,000	M	No declara	1	869	799	1	0	3	774,79	6	28,7	2017	2021	1	12	59	0
36	1	47	hasta 7,000	M	SP	1	182	377	0	0	3	1621,73	12	6,4	2017	2021	1	11	59	1
37	1	41	hasta 2,000	M	SP	1	66	22	1	1	4	4240,07	30	5,6	2017	2021	1	12	59	0
38	0	46	hasta 10,000	M	Sur	0	0	1	0	0	2	99,9	1	13,6	2018	2018	3	3	45	45
39	1	40	hasta 1,000	F	No declara	0	0	16	0	0	1	0	0	1,4	2018	2020	9	11	39	13
40	0	no declara	no declara	no declara	Noreste	0	0	0	0	0	2	159,9	1	0	2017	2017	1	1	59	59

Los usuarios que efectivamente publicaron un ítem en la categoría y en 2021 en Brasil, son aproximadamente 330 mil. Se toma el total de estos usuarios para el grupo que cumple con el objetivo de publicar, y una muestra aleatoria del mismo tamaño de aquellos usuarios que no publicaron en el *Marketplace*, con el objetivo de asegurar una muestra balanceada y evitar complicaciones en el modelado.

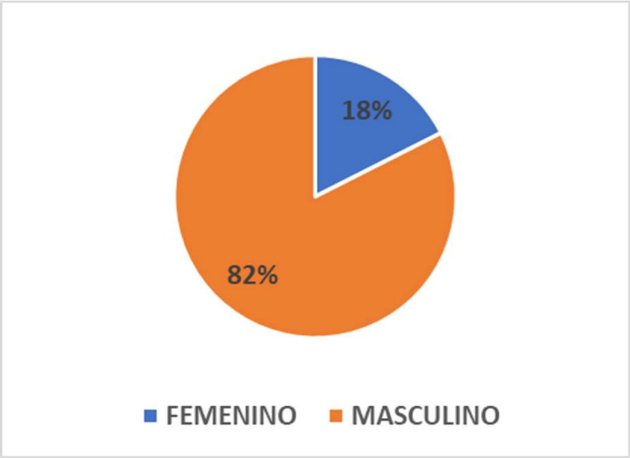
A partir de los ID de los usuarios de ambos grupos, se generó una tabla única y unificada con todos los atributos mencionados anteriormente. Esto se puede realizar de manera simple y sencilla, dado que la empresa posee grandes bases de datos muy estructuradas que pueden ser consultadas vía *SQL* sin mayores complicaciones.

La información, que es de carácter personal, es optativa a la hora de registrarse como usuario dentro de la plataforma, lo que conlleva a que muchos usuarios prefieran no declarar su género, su edad, su dirección o sus ingresos mensuales. En cuanto al comportamiento dentro del *Marketplace*, los usuarios no poseen la decisión de que sus datos no sean rastreados. La existencia de los datos nulos o no declarados tendrá una gran relevancia dentro del desarrollo del modelo y modelado de los datos.

A continuación, se expondrán los primeros gráficos de los usuarios *target* (que publicaron en la categoría de relevancia) como un primer análisis descriptivo de los datos. Es importante entender y procurar detectar algunos *insights* importantes de este primer análisis, para luego confirmarlos dentro del modelado y ver su verdadera relevancia de variable explicativa en comparación con el resto de los atributos.

En primera instancia, se analizan los datos demográficos declarados por los usuarios al momento de registrarse.

Ilustración 7 - Género de los usuarios que publicaron en la categoría y en 2021

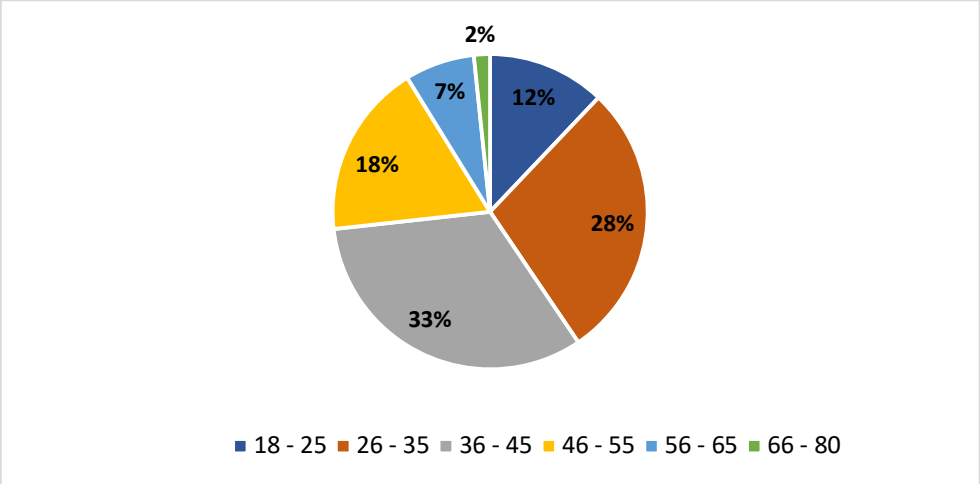


A simple vista se detecta una predominancia del género masculino a la hora de determinar el perfil de los usuarios que publican en la categoría y dentro del *Marketplace*, con

un 82% de los casos. Podría tomarse como hipótesis que el género, será una variable relevante que explique el comportamiento de los usuarios.

El mismo caso puede verse con los rangos etarios de los clientes:

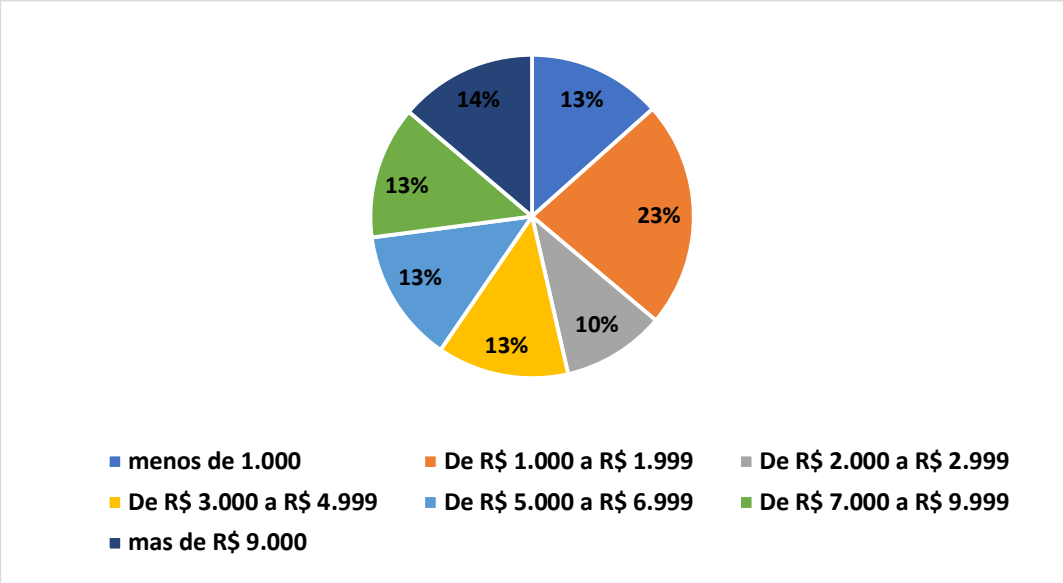
Ilustración 8 - Rango etario de los usuarios que publicaron en la categoría y en 2021



Se observa una clara predominancia del rango entre 26 y 45 años, con un 55% del total de las publicaciones asignadas a estas edades. Se asigna a esta variable un límite inferior de 18 y un límite superior de 99 años, por lo que los usuarios que declararon un valor fuera de ese rango se trabajaron como nulos.

Sin embargo, no siempre los gráficos del análisis descriptivo muestran alguna tendencia o *insight* en los datos. A continuación, se presenta el caso del nivel socio económico de los usuarios.

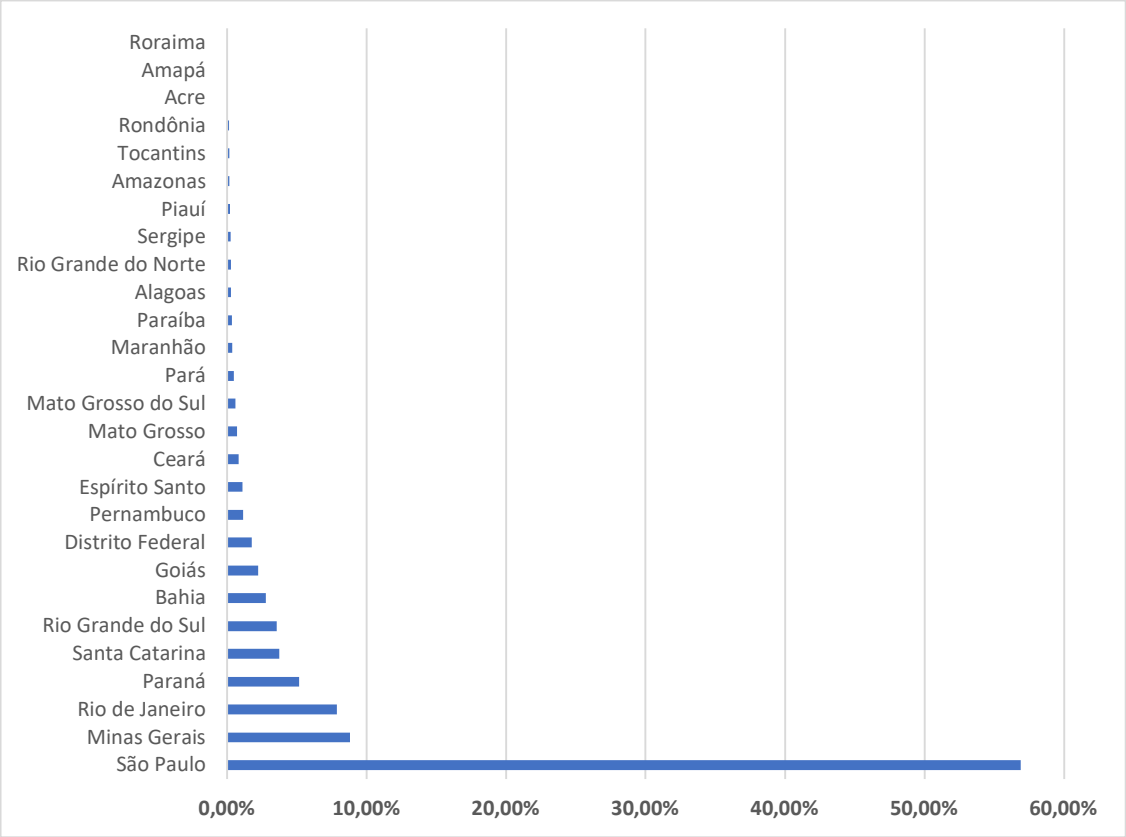
Ilustración 9 - Ingresos mensuales declarados de los usuarios que publicaron en la categoría y en 2021 (en reales)



Se presenta una distribución símil uniforme entre los diferentes rangos de ingresos. Hay una pequeña diferencia en el rango de 1.000 a 1.999 reales, pero no pareciera ser suficiente para generar una hipótesis clara a probar con el algoritmo predictivo, aunque de igual manera será una de las variables que se pondrán a prueba.

En cuanto a la geolocalización de los usuarios que publicaron en la categoría y, se encuentran concentrados en tres estados principalmente. En São Paulo reside el 57% de los usuarios que publicaron en la categoría y, 9% en Minas Gerais y un 8% en Río de Janeiro. De los 27 estados que posee Brasil, el 74% de las publicaciones se encuentran en estos tres estados.

Ilustración 10 - Share por estado de los usuarios que publicaron en la categoría y en 2021.



Otro atributo importante que se puede obtener de los usuarios corresponde a las categorías que más compran dentro del *Marketplace*:

Tabla 2 - Top 10 categorías de compras de los usuarios que publicaron en la categoría y en 2021

Ranking	Categoría
1	Insumos para categoría "*y*"
2	Casas, muebles y decoración
3	Calzado, ropa y bolsos
4	Deportes y fitness
5	Celulares y teléfonos
6	Electrodomésticos
7	Belleza y cuidado personal
8	Herramientas
9	Electrónica, Audio y Vídeo
10	Informática

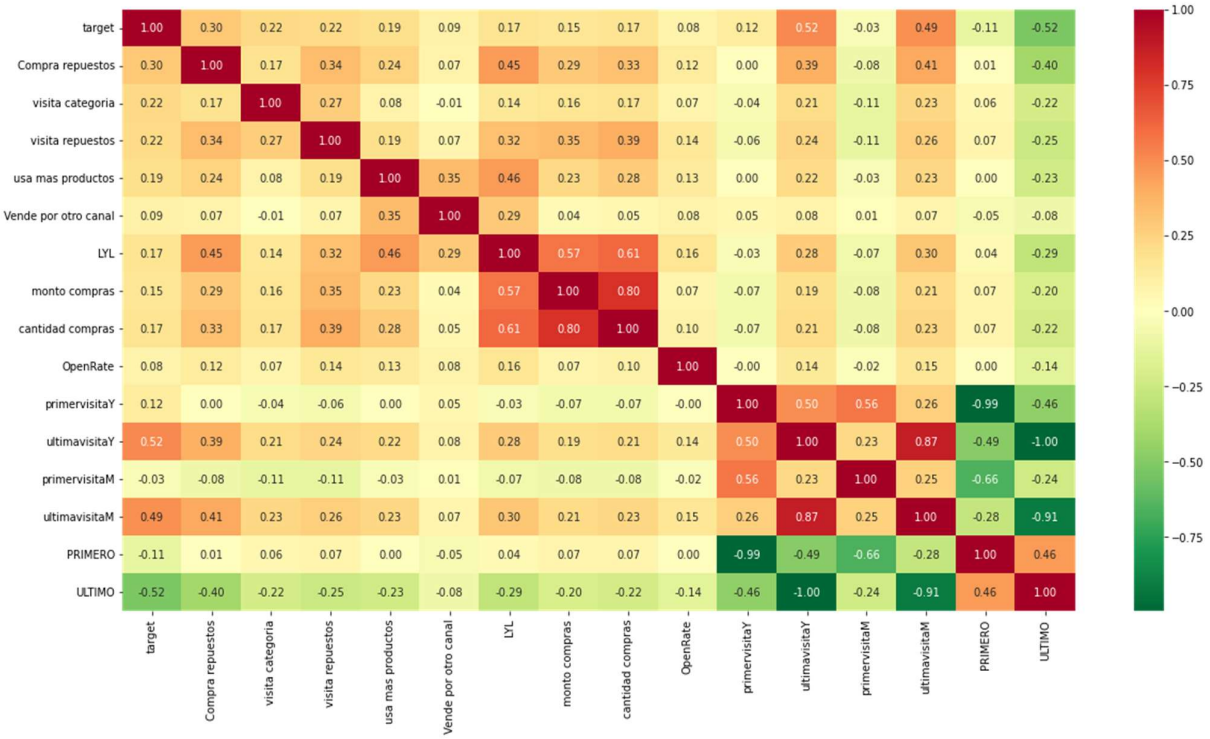
Que los usuarios hayan comprado insumos para los productos de la categoría y, parece ser determinante a la hora de predecir que están en búsqueda de arreglar su producto, para publicarlo y venderlo en mejores condiciones que las actuales, y de esta manera obtener un mayor rédito económico. A su vez, se supone que es más fácil vender un producto en perfectas condiciones, que venderlo con un listado de arreglos a futuro para el nuevo dueño.

Cabe destacar que, de todas las variables que se presentan en el modelo, sólo cuatro son variables categóricas. Estas son: género, estado / cuidad, ingresos y edad. Para las variables ingresos y edad, a pesar de ser variables numéricas, se crearon diferentes niveles para poder trabajarlas como categóricas, y de esta manera poder incluir como uno de los niveles el "no declara" cuando el usuario hace omisión de esa información en la creación del perfil.

A continuación, se presentarán algunos gráficos del análisis exploratorio previo al armado del modelo. Las variables que se muestran a continuación son las seleccionadas para trabajar el modelo final luego de algunas pruebas anteriores de otros modelos y variables.

En la Ilustración 10, se expone la una matriz de correlación entre las variables explicativas y la variable *target*.

Ilustración 11 - Matriz de correlación de todas las variables explicativas vs. target



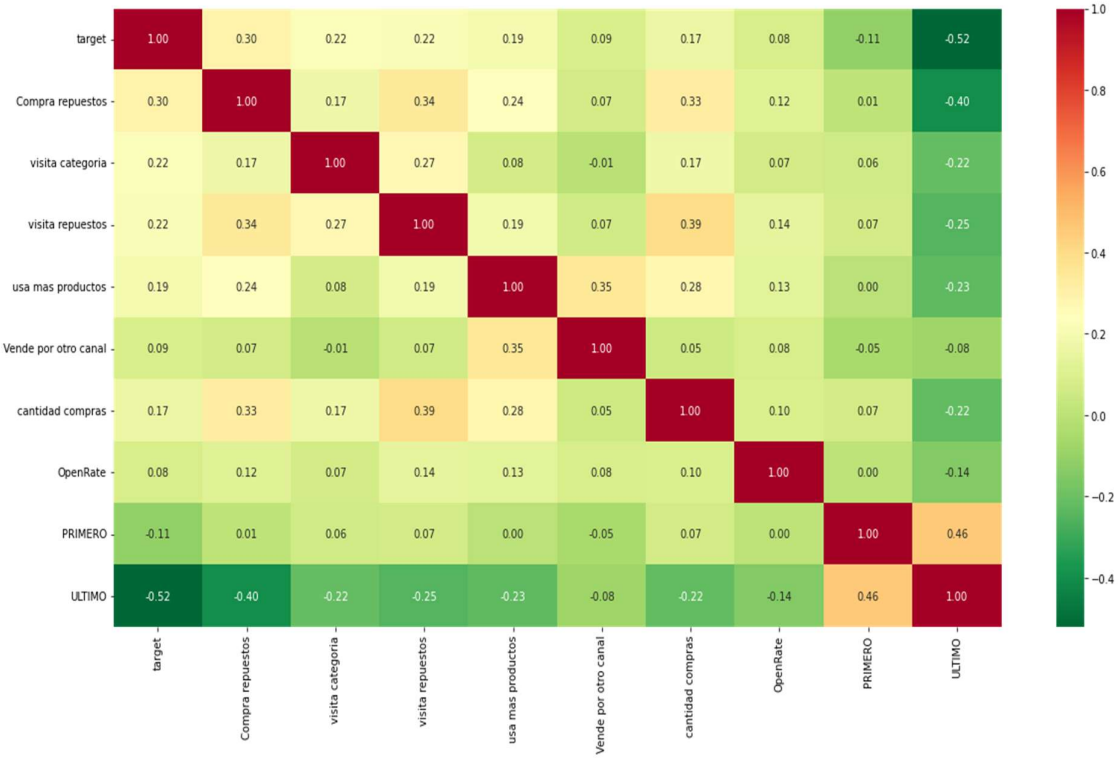
Puede observarse que existe una alta correlación entre varias variables explicativas sobre las cuales trabajar previo al desarrollo del modelo.

Las variables que incluyen el año y mes de la primera y última visita a la categoría y parecen estar muy correlacionadas. Dado que sólo importa la cantidad de meses desde que estos dos eventos sucedieron, se eliminan las variables año y mes.

También, se aprecia que la variable LYL, cantidad de compras, y monto de compras están muy correlacionadas y de manera positiva, lógicamente. Se optó por trabajar con la variable cantidad de compras, siguiendo las recomendaciones de otros modelos de *machine learning* ya creados y utilizados como experiencia previa para el desarrollo del actual modelo.

Con estas modificaciones, la matriz de correlación de las variables que finalmente se utiliza para el modelo queda reducida a la siguiente gráfica:

Ilustración 12 - Matriz de correlación reducida de todas las variables explicativas vs. target



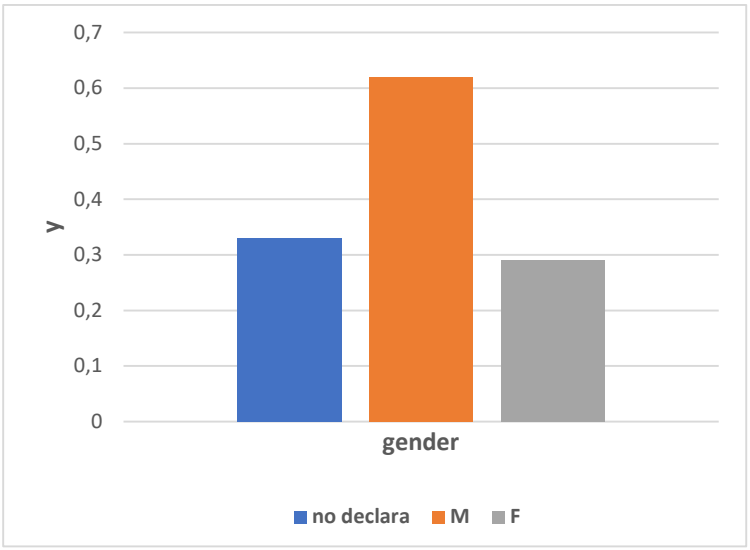
Se observa que con la supresión de variables muy correlacionadas se soluciona el problema de correlaciones fuertes entre variables explicativas.

Además, como parte del análisis exploratorio, se realizó el análisis de las variables categóricas contra la variable *target*, dado que este tipo de variables no son incluidas en la tabla de correlaciones expuesta anteriormente. De igual manera, este análisis determinado de variable por variable también puede realizarse para las variables numéricas ya analizadas.

Al trabajar con datos que involucran variables categóricas, las mejores herramientas para visualizar y comparar diferentes características de los datos son los gráficos categóricos.

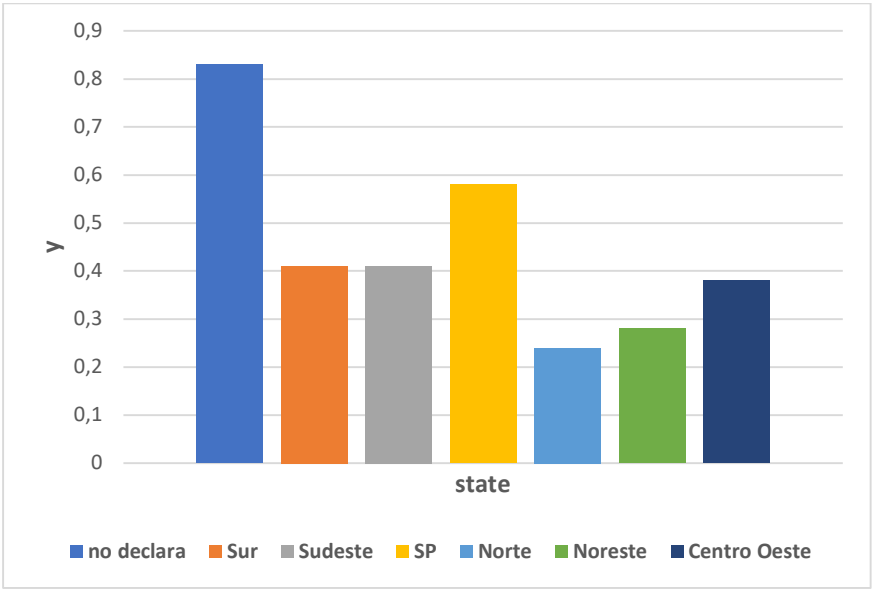
Trazar parcelas categóricas es muy fácil en *seaborn*[1], una librería de *Python* muy potente para estos análisis.

Ilustración 13 - Variable categórica género vs. target



A simple vista, como se había analizado en la **¡Error! No se encuentra el origen de la referencia.** que el género masculino es mejor predictor de que un usuario va a publicar un ítem de la categoría *y* que los usuarios de género femenino o que decidieron no declarar.

Ilustración 14 - Variable categórica estado / ciudad vs. target

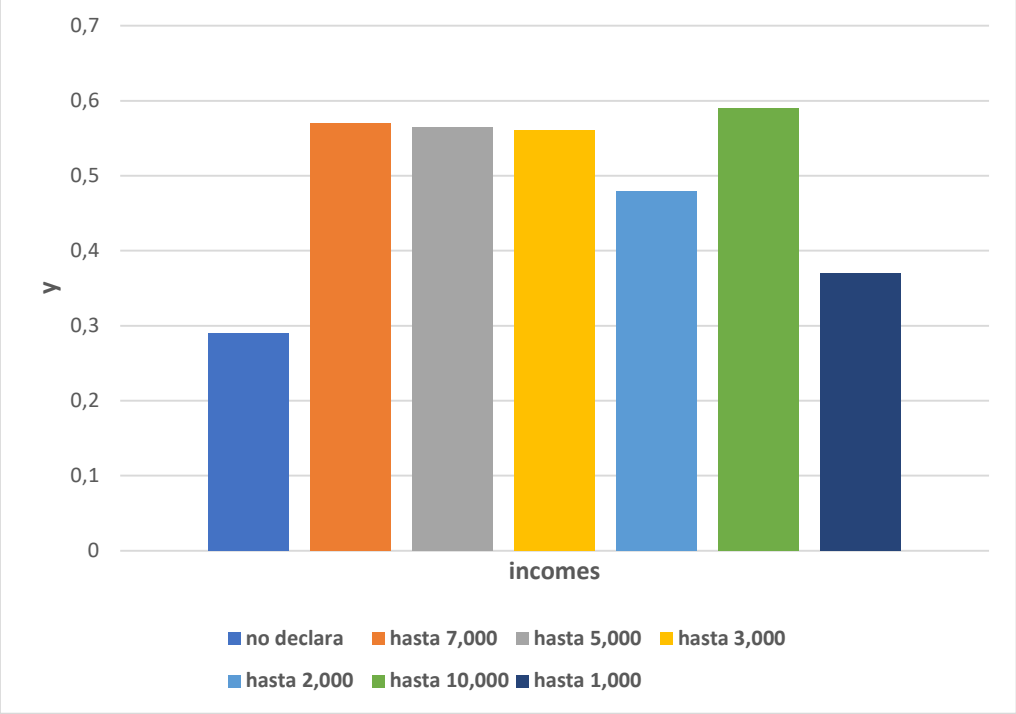


[1] https://seaborn.pydata.org/

En el caso particular de esta variable, la categoría 'no declara' es la mejor predictora. En el caso de los usuarios que no publicaron en la categoría bajo análisis, el requisito de la base fue una compra en período bajo análisis. Esto implica que, necesariamente, el usuario que realizó una compra tuvo que ingresar una dirección de entrega de esta.

Para los usuarios que publicaron un ítem de categoría y no era requisito una compra en ese año. Por lo que no necesariamente ingresaron una dirección de entrega, y muchos optaron por no declarar.

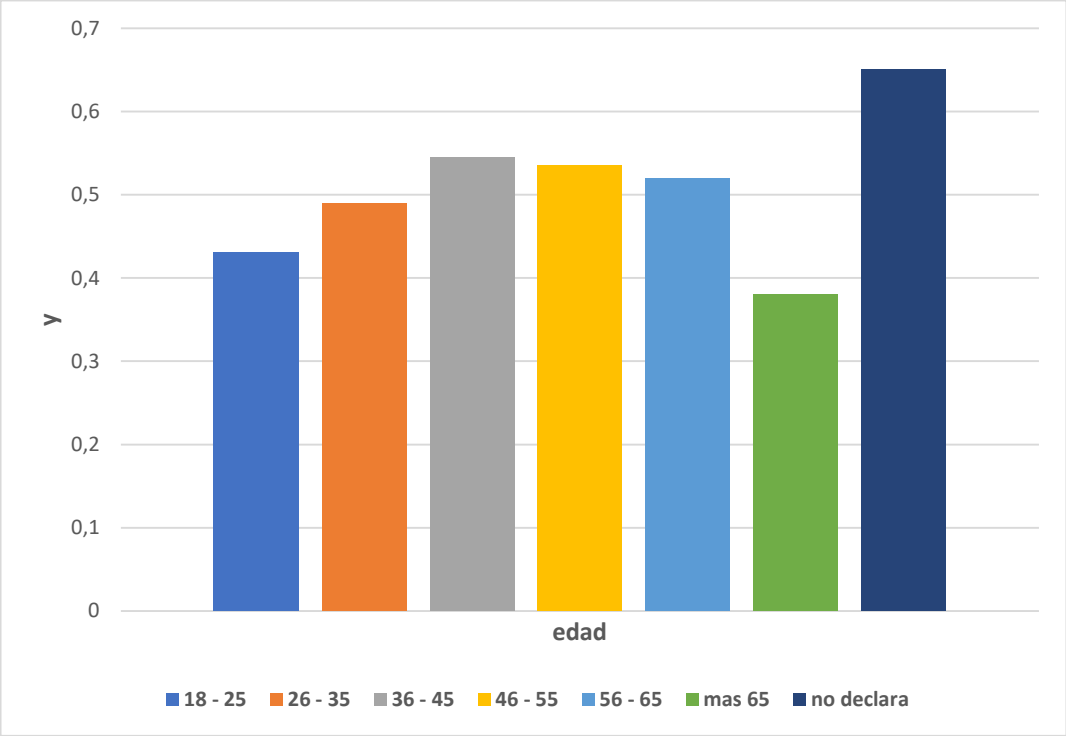
Ilustración 15 - Variable categórica ingreso vs. target



En esta categoría no se encuentra ningún nivel que predomine sobre los demás de manera significativa. Seguramente a la hora de evaluar la importancia de atributos, sea un atributo que no tenga gran influencia en la predicción del modelo.

Finalmente, la última variable categórica por analizar es la edad de los usuarios que publicaron un ítem dentro de una categoría determinada y.

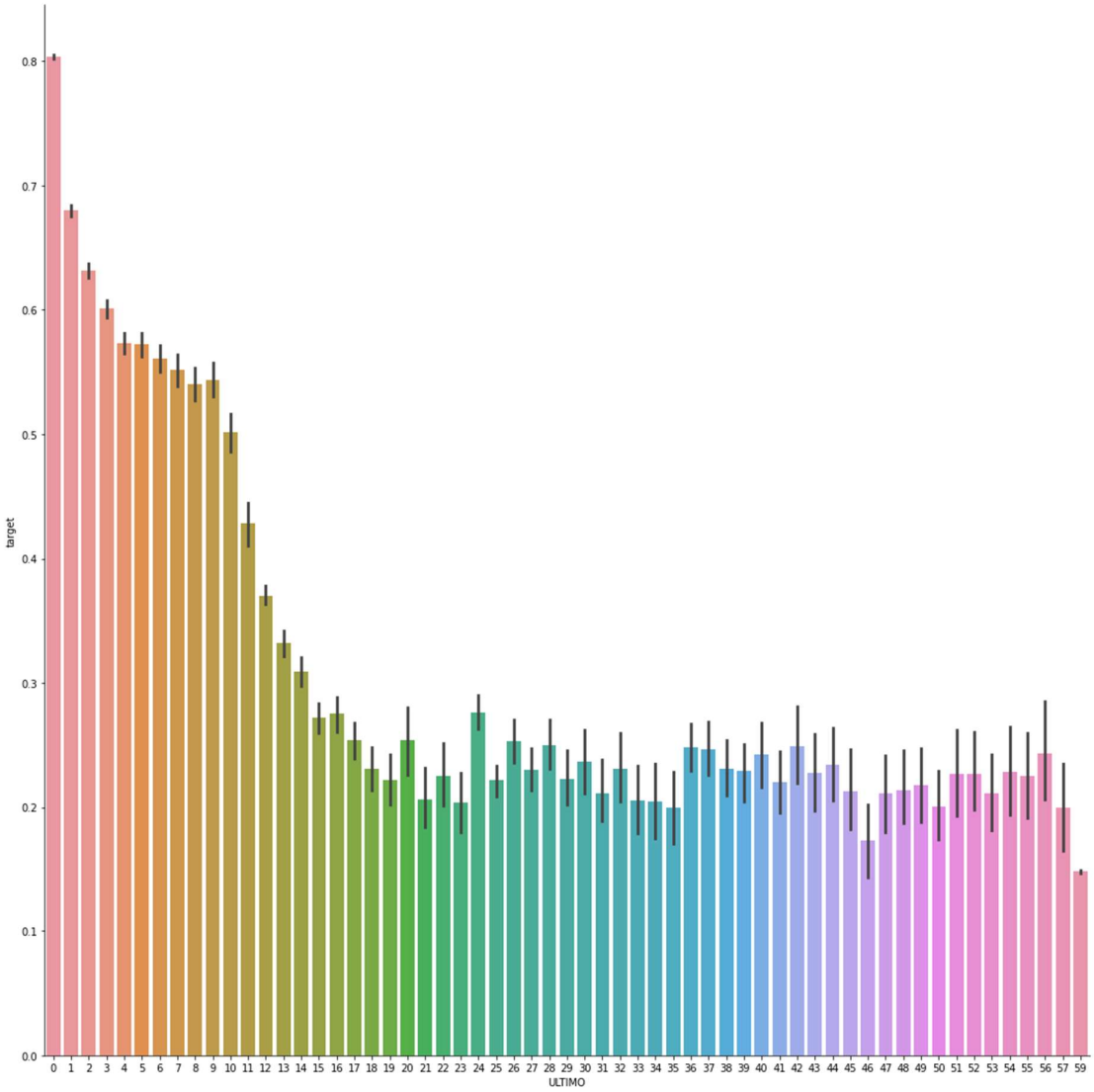
Ilustración 16 - Variable categórica edad vs. target



Nuevamente, no se visualiza ninguna categoría con predominancia predictiva sobre el resto de las categorías, a pesar de haber visto en el análisis descriptivo de los usuarios *target*, una gran diferencia de los usuarios entre 26 y 45 años sobre el resto.

Sólo se analiza de manera particular una sola de las variables numéricas del modelo. La variable “ULTIMO”, que es la recencia (medida del tiempo que ha pasado desde la última visita que realizó un cliente) del usuario a la categoría en la cual se está buscando que publique. Es importante este análisis por ser el *push / email* recurrente que tuvo mejor performance si es analizado en comparación al resto de los productivos y recurrentes.

Ilustración 17 - Variable ULTIMO / recencia vs. target



En este caso, puede verse claramente que mientras menos tiempo haya pasado desde la última visita a la categoría y, el usuario es más propenso a realizar una publicación de esa misma categoría. Esto es uno de los resultados que se espera que el modelo pueda validar dentro de la importancia de atributos. En los *A/B testing* que se realizan de manera manual, esta variable es siempre estadísticamente significativa.

4.2 Modelo propuesto XGBoost

Analizando la base de datos obtenida y el punto "c" de las desventajas (Solo trabaja con vectores numéricos, por lo que se requiere convertir previamente los tipos de datos no

numéricos a numéricos., el modelo *XGBoost* requiere que las variables no numéricas se conviertan en numéricas. Cabe recordar que la base sólo presenta cuatro atributos que son no numéricos, y son variables categóricas: género, localidad, ingresos, edad (un nivel era no declara).

Para contrarrestar esta dificultad, se generan variables *dummies* a partir de las variables categóricas. Esto hace que se generen nuevas columnas en nuestra base de datos con valores 0 o 1 según cumpla o no con el nivel de la columna. Afortunadamente, la librería de *Python Pandas*[2] tiene un comando que realiza esta transformación de manera automática y sin mayores complicaciones.

Siguiendo con las transformaciones de los datos, los mismos fueron estandarizados. Este proceso consiste en re-escalar la distribución de los valores para que la media observada sea de cero y el desvío estándar sea uno. Esto es fundamental a la hora de preparar la base de datos previo al modelo, dado que, si evitamos este paso, estaríamos dando indefectiblemente mayor relevancia a las variables con valores más altos de manera errónea (Zheng, 2018).

Además, en este tipo de modelos muchas veces el éxito depende en gran parte de la ingeniería de atributos. La misma consiste en generar nuevas variables explicativas, en función de las ya existentes. Alguna de las variables que se generaron y se probaron dentro del modelo, pero sin éxito fueron:

- *score* por cliente en función de la cantidad de diferentes categorías que compra dentro del *Marketplace*. Mientras más categorías compra, mayor *score* asignado. Muy correlacionada con compras totales, no es relevante.
- *score* del cliente por cantidad de productos adicionales que consume, además de la plataforma. Muy correlacionado con uno de los productos en particular, no es relevante.
- como vimos en la sección de datos, gran parte de los individuos que publican son menores a 45 años, se probó creando una variable categórica sólo con dos niveles, siendo mayor o menor a esta edad. Tampoco es relevante para el modelo.
- *score* en función de la primera vez que el usuario visita una categoría determinada, y la última vez que lo hizo. Muy correlacionada con la variable recencia y sin relevancia.

Otra definición previa a correr el modelo es definir el porcentaje de la base que será utilizado para entrenar, testear y validar. En nuestro caso se usa el 75% de los datos para el entrenamiento del modelo, y la mitad del restante (12,5% por lado) tanto para validación como testeo.

Por otra parte, haciendo un análisis más exhaustivo sobre la desventaja mencionada en el punto "b" (Se deben ajustar correctamente los parámetros del algoritmo a fin de minimizar el error de precisión y evitar sobreajuste del modelo (lo que puede darse si se maneja un número muy grande de árboles - *overfitting*). , es un algoritmo que tiene muchos hiperparámetros que pueden repercutir en los resultados del modelo y que por eso hay que ajustarlos eficientemente.

[2] https://pandas.pydata.org/

Los hiperparámetros que se suelen modificar a la hora de construir un modelo con *XGBoost* son los siguientes:

- *nrounds:* número de árboles total del algoritmo.
- *max_depth:* máxima profundidad de los árboles.
- *eta:* disminución del aprendizaje de cada árbol que se va generando, para evitar el sobreajuste u *overfitting* del modelo. Importancia que el nuevo árbol entrenado le asigna a los errores del anterior.
- *gamma:* mínima reducción del error para generar un corte nuevo.
- *colsample_bytree:* porcentaje de variables aleatorias a muestrear y considerar en cada árbol.
- *min_child_weight:* mínima cantidad de observaciones en los hijos para considerar un corte.
- *subsample:* muestro de observaciones a considerar en cada árbol.
- *n_estimators:* cantidad de árboles del boosting que permitiremos.
- *alpha:* término utilizado para la regularización del modelo. Aumentando este valor el modelo se convierte en más conservador.

Tradicionalmente, los hiperparámetros se ajustaban manualmente por ensayo y error. Esto todavía se hace comúnmente, y los ingenieros de datos experimentados pueden probar los valores que ofrecerán una alta precisión del modelo. Sin embargo, hay una búsqueda continua de métodos mejores, más rápido y más automáticos para la optimización de hiperparámetros.

Dado que son muchos hiperparámetros a definir, existe una librería muy potente de *Python* denominada *scikit-learn* que nos va a simplificar esta ardua tarea. La misma posee la capacidad de optimizar la selección de los hiper parámetros a través de su función *RandomizedSearchCV*[3] utilizando búsqueda aleatoria. Básicamente, consiste en un proceso de simulación de Montecarlo en el cual se van probando valores para los hiperparámetros, hasta encontrar su valor óptimo.

En nuestro modelo, se utilizan rangos para la búsqueda de los hiperparámetros óptimos para la mejor performance y eficiencia del modelo. El número de iteraciones de búsqueda se establece en función del tiempo o los recursos disponibles (Chauhan, 2020). Se corrieron 300 iteraciones con diferentes combinaciones aleatorias y resultaron los siguientes valores óptimos:

- ★ *n_estimators* → 73
- ★ *max_depth* → 11
- ★ *eta o learning_rate* → 0.025599452033620268

[3] https://scikit-learn.org/stable/

- ★ *colsample_bytree* →0.9252155845351104
- ★ subsample → 0.8783523142867211
- ★ gamma → 0.7800932022121826
- ★ *min_child_weight* → 7.674172222780436
- ★ *reg_alpha → 3.304442364744264*

Con estos hiperparámetros ya optimizados, se corre el modelo *XGBoost* y se obtienen los resultados expuestos a continuación.

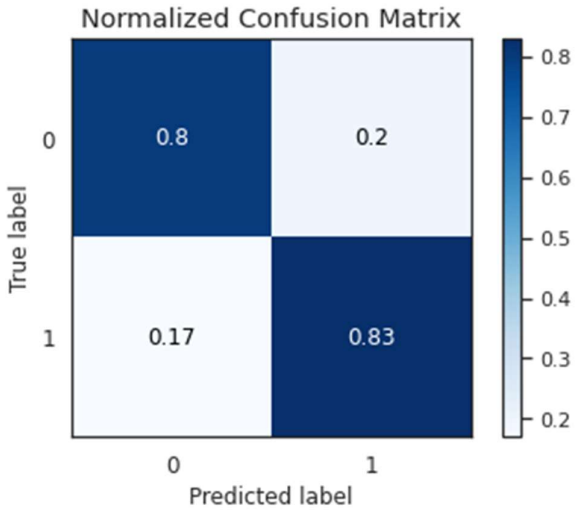
5. Resultados

En esta sección, se enumeran los resultados del modelo y las métricas de performance detalladas en la Sección 3.4 Métricas de éxito de un modelo de *Machine Learning* para medir la calidad del modelo. Además, usaremos como *benchmark* las métricas obtenidas por el modelo de regresión logística, el cual será utilizado para realizar comparaciones con el modelo XGBoost final.

A continuación, los principales resultados obtenidos:

Matriz de confusión del modelo final

Ilustración 18 - Matriz de confusión en valores porcentuales con datos de validación

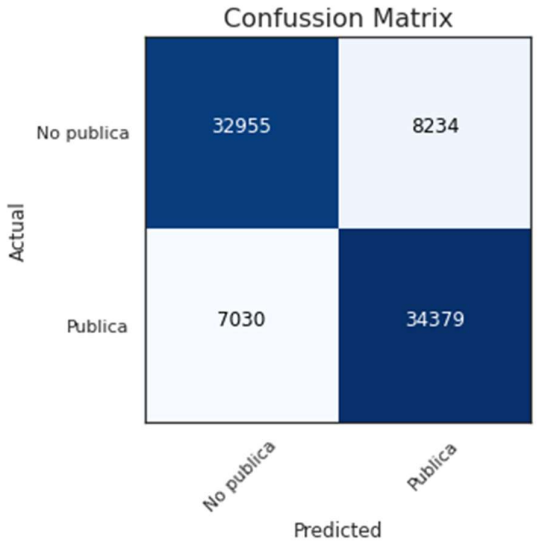


- **Verdadero positivo (VP):** El modelo acierta en el 83% de los casos verdaderos.
- **Verdadero negativo (VN):** El modelo acierta el 80% de los casos falsos.
- **Falso negativo (FN):** El modelo falla en el 17% de los casos verdaderos, también conocido como error tipo II.
- **Falso positivos (FP):** El modelo falla en el 20% de los casos falsos, también conocido como error tipo I.

Por su parte, el modelo de regresión logística obtuvo un porcentaje de verdaderos positivos del 75%, verdaderos negativos del 84%, falsos negativos del 16% y falsos positivos del 25%. Comparando con el modelo XGBoost, este modelo predice mucho peor los casos donde el usuario no publicó el ítem, generando un error de tipo I significativamente superior.

También puede representarse la matriz de confusión con los valores absolutos en los datos de validación del modelo. Cabe recordar que para la validación se usó el 12,5% del total de los datos que se tenían disponibles.

Ilustración 19 - Matriz de confusión en valores absolutos con datos de validación



Las métricas que se desprenden de la matriz de confusión son:

Accuracy del modelo final

El modelo propuesto acierta el 81,5% de las predicciones que realiza.

$$Accuracy = \frac{34.379 + 32.955}{32.955 + 8.234 + 7.030 + 34.379} = 81{,}52\ \%$$

Precisión del modelo final

Del total de *push e emails* que propone enviar el modelo, el 80,7% de los usuarios que los reciban estarán interesados en publicar un ítem en el *Marketplace*.

$$Precisión = \frac{34.379}{34.379 + 8.234} = 80{,}68\ \%$$

Recall del modelo final

Del total de clientes que efectivamente está considerando publicar un ítem en el *Marketplace* el modelo detecta con éxito el 83,0% de los clientes.

$$Recall = \frac{34.379}{34.379 + 7.030} = 83{,}02\ \%$$

F1 Score del modelo final

Para el cálculo del F1 score, se asume que el modelo propuesto asigna igual relevancia a precision y recall.

$$F1 = 2 * \frac{80{,}68\% * 83{,}02\%}{80{,}68\% + 83{,}02\%} = 81{,}8\ \%$$

Al comparar estas últimas cuatro métricas con el modelo *benchmark* de regresión logística se observa una mejoría en:

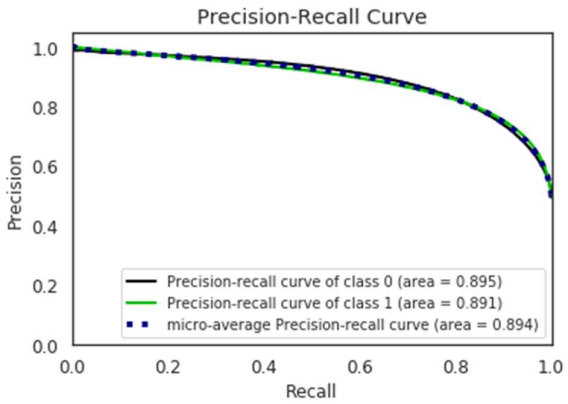
- *Accuracy*: más de dos puntos porcentuales frente al 79,43% del modelo base.
- *Precision*: más de tres puntos porcentuales frente al 77,03% del modelo base.
- *F1 score*: más de dos puntos porcentuales frente al 80,36% del modelo base.

La única métrica donde se observa una leve desmejora es en *Recall* y es menor al punto porcentual sobre el modelo de regresión logística que obtuvo un 83,99%. El modelo base tiene más probabilidad de acierto en detectar los usuarios que efectivamente publicaron un ítem.

PR Curve del modelo final

El modelo asume que importa de igual manera *precision* y *recall*. *Precision* hace referencia a evitar el spam de enviar notificaciones a usuarios que no estén interesados, y *recall* hace referencia a no perder de notificar a ningún cliente que efectivamente esté interesado en publicar. Ambas métricas, pueden ser representadas gráficamente mediante la curva *precisión-recall*:

Ilustración 20 - Curva precision – recall (PR curve)

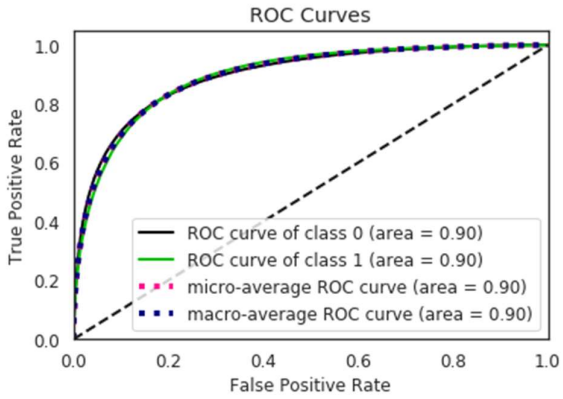


En el gráfico, se puede ver en línea punteada azul, el *average precision* que es de 0,894. Es una manera de calcular el área bajo la curva PR o PR AUC, o lo que es lo mismo, el resultado de integrar la curva. Esta métrica sirve para evaluar y comparar el rendimiento de modelos entre

sí. Mientras más se acerca a uno, mejor será el modelo. En este caso es un muy buen modelo dado esta métrica. Comparando con el modelo de regresión logística, el *average precision* es ampliamente superior, de casi 5 puntos porcentuales (0,856).

ROC curve

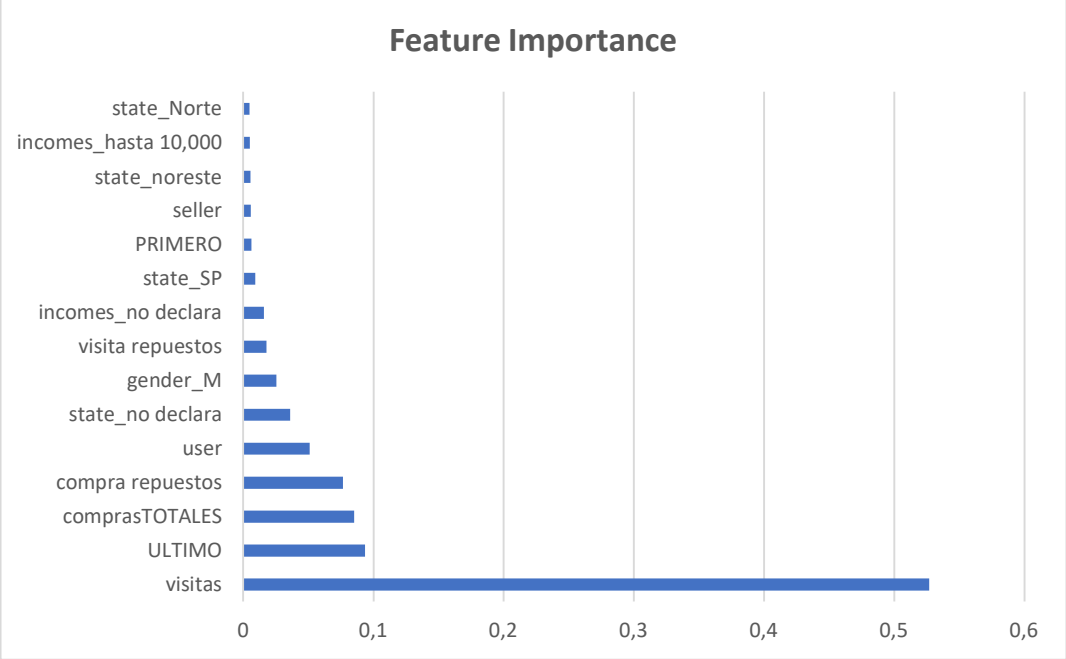
Ilustración 21 - Curva ROC – Receiver Operating Characteristic



En el modelo propuesto, se puede también calcular el ROC AUC, o el área bajo la curva, que también sirve como métrica para resumir la curva y poder comparar modelos. De manera similar al *Average Precision*, nos interesa que su valor se acerque lo máximo posible a 1. El ROC AUC de nuestro modelo es de 0,9, muy cercano al valor óptimo de 1.

Feature importance

Ilustración 22 - Gráfico de feature importance



El atributo con más relevancia a la hora de predecir que un usuario va a publica un ítem de una categoría determinada y, se explica en un 52,6% porque ese usuario haya visitado esa categoría previamente. El segundo atributo más importante, con un 9,36%, es la recencia, que implica que cuanto menor tiempo haya pasado desde que el usuario visitó la categoría de interés, más propenso a publicar será.

Entre estos dos atributos, se explica casi el 62% de nuestro modelo. Cabe recordar que uno de los envíos de *push e email* que están prendidos a la fecha es para los usuarios que visitaron la categoría en los últimos días, lo que implica que, se están considerando los dos mejores atributos predictivos en la actualidad.

No muy por detrás, la cantidad de compras que realiza un usuario explica el 8,51% del modelo. En cuarto lugar, se encuentra el atributo que indica si el usuario compro o no productos usados como repuestos para la categoría y con un 7,65% y en quinto lugar que el usuario sea también usuario de alguno de los productos alternativos que ofrece el *Marketplace* con un 5,12%.

El antecedente que el usuario haya visitado previamente la categoría de interés es uno de los resultados más esperados e intuitivos. Posiblemente, el individuo que realiza la visita esté en búsqueda de un relevamiento del mercado en donde querrá vender su producto para entender el precio posible de venta, y opciones de comprar algo mejor y más acorde a sus necesidades.

La recencia también es un resultado esperado. Mientras menos tiempo pasó desde que el usuario visitó la categoría, se demuestra que está interesado en la misma en la actualidad. No es un comportamiento muy lógico ver precios de los productos de determinada categoría si se está pensando en comprar o vender en varios meses y no hoy.

El atributo de compras totales demuestra que mientras más compra el usuario en la plataforma, es más propenso a publicar dentro de ella. Esto demuestra la confianza del individuo en sus interacciones con la plataforma, y que lo llevan a confiar y vender a través de la misma.

Esto último, está muy relacionado a que el usuario es también consumidor de alguno de los otros productos que ofrece la empresa. Están relacionados de manera positiva, la confianza y uso de la plataforma y sus extras.

Se puede concluir, que el modelo se explica un 83,1% con sólo cinco variables de comportamiento de los individuos. Las características demográficas, geográficas y socioeconómicas no tienen relevancia a la hora de predecir el comportamiento de un usuario dado el modelo propuesto.

Permutation feature importance

Tabla 3 - Permutation feature importance

Feature	Weight
visita categoría	0,1035 + / - 0,0013
comprasTOTALES	0,0534 + / - 0,0010
ULTIMO	0,0108 + / - 0,0006
state_no declara	0,0079 + / - 0,0005
user	0,0070 + / - 0,0007
visita repuestos	0,0041 + / - 0,0007
compra repuestos	0,0033 + / - 0,0008
state_SP	0,0031 + / - 0,0003
PRIMERO	0,0026 + / - 0,0008
OpenRate	0,0016 + / - 0,0004
incomes_hasta 10,000	0,0014 + / - 0,0007
seller	0,0012 + / - 0,002
incomes_no declara	0,0010 + / - 0,0002
State Noreste	0,0006 + / - 0,0003
Incomes_hasta 7,000	0,0006 + / - 0,0003

Cuatro de los primeros cinco atributos de la tabla de importancia de atributos se repiten en el top cinco de la tabla anterior. Esto refuerza que los atributos que devuelve el modelo como los más relevantes, son los correctos bajo diferentes pruebas.

Este modelo final *XGBoost* mostrado hasta el momento, es el tercer modelo desarrollado y el final para esta primera vuelta de resolución del problema de generar nuevas publicaciones evitando grandes impactos en spam. El primer modelo fue un modelo de regresión logística (utilizado como *benchmark*), que obtuvo un 77,03% de aciertos, seguido de un primer modelo *XGBoost* sin mucha ingeniería de atributos con un 79,3% de aciertos y finalmente el modelo final con un 80,7% de aciertos.

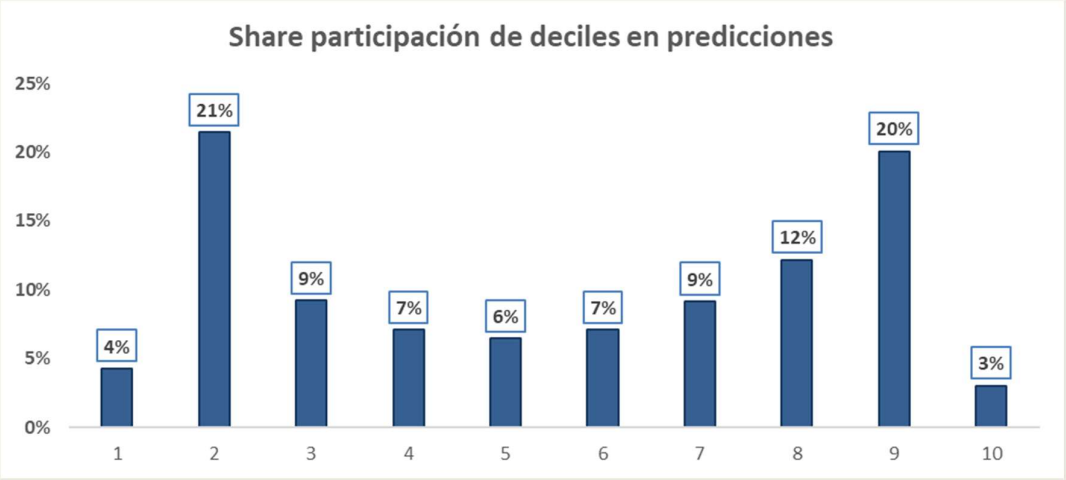
Seguramente, documentando la experiencia de los primeros resultados, se generen algunos otros modelos alternativos a futuro, usando los aprendizajes de estos primeros modelos en ejecución. Cómo una primera solución, más de un 80% de accuracy para el modelo es de gran impacto para el negocio.

5.1 Accionables a partir de los resultados

Una vez finalizado el modelo, con buenas métricas de performance e importancia de atributos, se procede a analizar los resultados de manera exhaustiva para definir la futura estrategia de marketing digital, con los aprendizajes obtenidos.

Como punto de partida, se puede analizar la distribución de los resultados devueltos por el modelo en el siguiente gráfico:

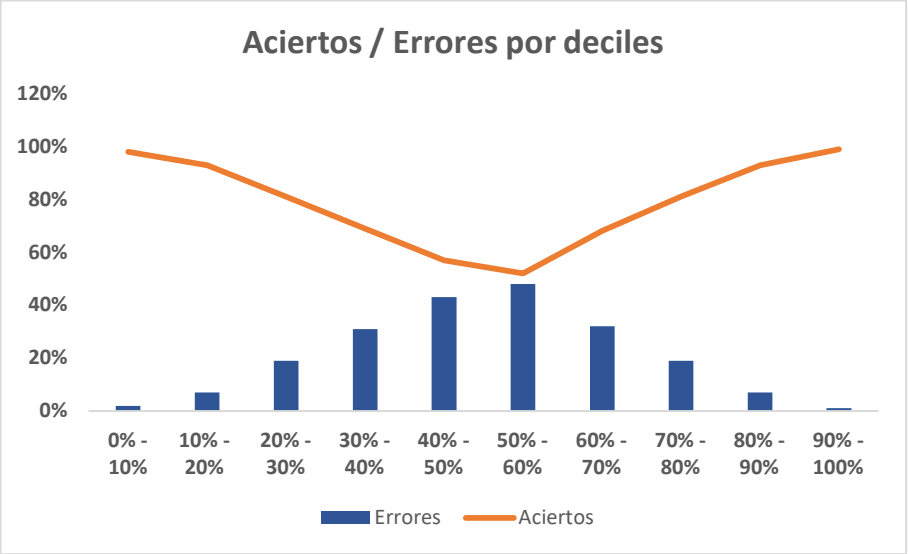
Ilustración 23 - Share de participación (por deciles) en las predicciones



Se observa que la distribución de las predicciones (en probabilidad de publicar), se distribuye de manera inversa a una curva de distribución normal. Esto hace que grandes volúmenes de resultados se acumulen en los extremos, y se disponga de pocas cantidades de predicciones en los valores medios. Sin duda, esto es un gran indicio de que el modelo está funcionando de manera eficiente.

En los extremos, en modelo está asignando una probabilidad muy baja o alta a que el usuario no publique o publique un ítem de una categoría determinada. A medida que se produce un acercamiento desde ambos extremos, hacia los valores cercanos al 50% de probabilidad, el modelo empieza a "dudar" más en la predicción. Siguiendo el análisis de errores por deciles, se observa el siguiente gráfico:

Ilustración 24 - Porcentaje de aciertos / errores por deciles

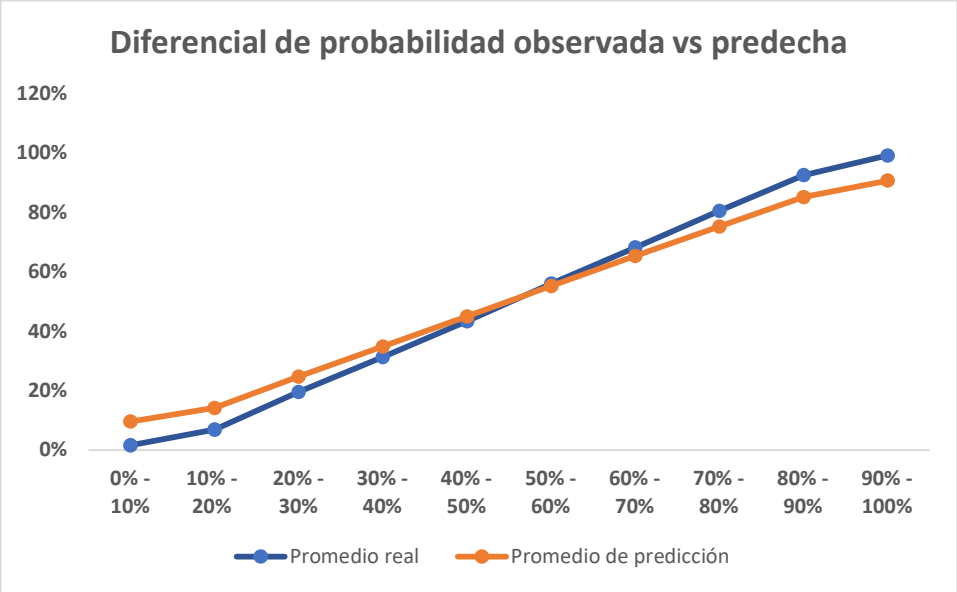


Como era esperable a medida que nos acercamos a los valores medios de las predicciones, el porcentaje de error crece, devolviendo una curva con distribución normal de los errores por deciles.

Si se superpone la distribución inversa a la normal de las predicciones, contra los errores distribuidos de manera normal, se puede confirmar que la performance del modelo es eficiente, asegurando muchas predicciones con bajos porcentajes de error.

Para poner a prueba el modelo con datos externos reales, en primera instancia, se armaron campañas por deciles. Para definir los deciles puestos a prueba, se puede ver a continuación el gráfico que compara las probabilidades promedio reales por deciles contra las probabilidades pronosticadas por nuestro modelo. De esta manera, podrían detectarse deciles con oportunidades de incrementalidad.

Ilustración 25 - Diferencial de probabilidad observada vs. predicha por el modelo



Se observa que a partir del decil 5 en adelante, las observaciones reales tienen mayor probabilidad de la que predice el modelo para esos deciles (posible oportunidad). En otros modelos desarrollados dentro de la empresa, se tomó como política, medir hasta una distancia de 0,4 puntos del *target* (100%). Esto implica que, las pruebas sobre el modelo son para los deciles desde el 6 al 10.

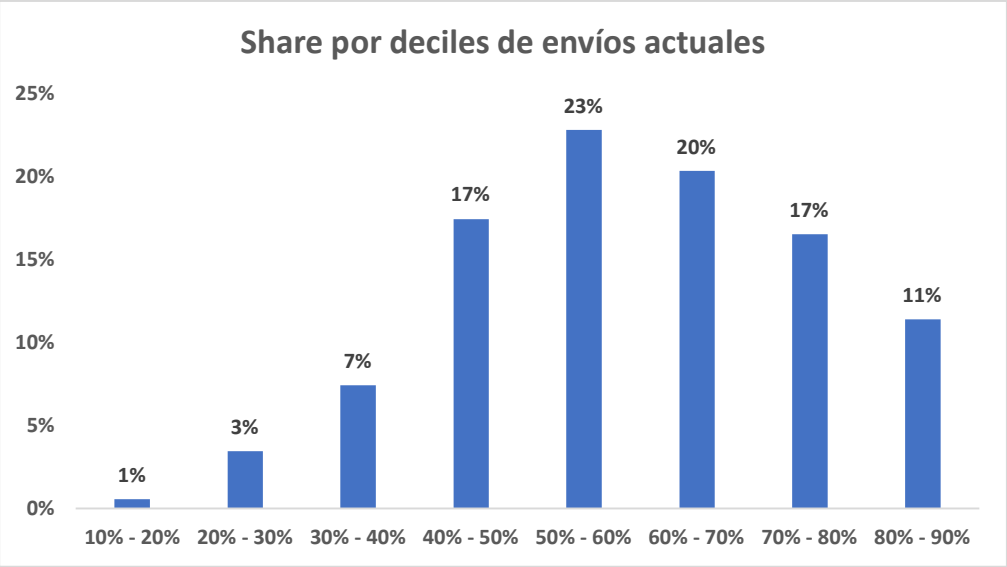
Cabe aclarar para este caso particular que, cuando un usuario recibe un *push* o *email*, inmediatamente queda bloqueado por el resto del día para recibir otra notificación de la misma unidad de negocio, evitando de cierta manera el *spam* de manera automática, a través de la herramienta de envíos. Esto hace que muchos usuarios que visitaron la categoría bajo análisis y, o compraron repuestos de la categoría y, o hicieron compras afines, ya reciban de manera automática y recurrente *push* o *emails* de las campañas automatizadas. Esto genera un sesgo importante en la medición del modelo con datos externos, dado que se tiene un grupo muy propenso de usuarios a publicar en la categoría y que no son "afectados" por los experimentos *Ad Hoc* de medición del modelo por ya ser *target* de otras campañas.

Esta situación lleva indefectiblemente a buscar otra alternativa de medición del modelo con datos externos a los de entrenamiento, testeo y validación ya usados. Dado que hay una gran variedad de usuarios que actualmente reciben notificaciones similares a las que se generarían con el modelo sobre invitar a los usuarios a publicar en una determinada categoría, se podría usar esta base y aplicar la medición sobre *pushes* o *emails* ya enviados.

Con la base de usuarios que recibieron en los últimos días una notificación recurrente (las que serían reemplazadas por una única unificada del modelo), se construye una base con exactamente los mismos atributos que la base original utilizada para entrenar el modelo. Esta base, se procesa por el mismo algoritmo de *XGBoost* y se asigna a cada usuario una probabilidad de publicar.

Al analizar el share por deciles de los envíos actuales de la probabilidad de publicar, se observa podemos observar la siguiente distribución de observaciones:

Ilustración 26 - Share por deciles de envíos actuales



Se observa una tendencia ascendente desde el decil 1 al 5, a partir del cual presenta un descenso suave a través del resto de los niveles.

Además, se puede saber si los usuarios que fueron impactados por un envío recurrente son tomados en cuenta para el grupo de testeo o para el grupo de control y si publicaron efectivamente o no dentro del *Marketplace*. Si un usuario fue contemplado para grupo control, esto se mantiene en el tiempo.

De esta manera, los datos disponibles son:

- Usuarios identificados que fueron impactados por un *push / email* recurrente de interés por reemplazar.
- Su probabilidad de publicar según el modelo.
- Deciles armados según probabilidad asignada.
- Clasificación de grupo de control o testeo de los usuarios, dentro de cada decil.
- Cantidad de publicaciones (conversiones) dentro de cada grupo por decil.

Con esta información, se arman los *KPI´s* de interés que fueron definidos en la sección de metodología. Se compara el escenario actual vs. el escenario proyectado dado las mediciones obtenidas con datos externos al modelo. Los *KPI´s* que se observan para la toma de decisiones son: *open rate*, publicaciones (conversiones absolutas) e incrementalidad.

Con toda esta información, para el armado del escenario proyectado, se supone que se enviará el mismo número de *pushes* que se envió el año pasado. Para ellos, se distribuye este

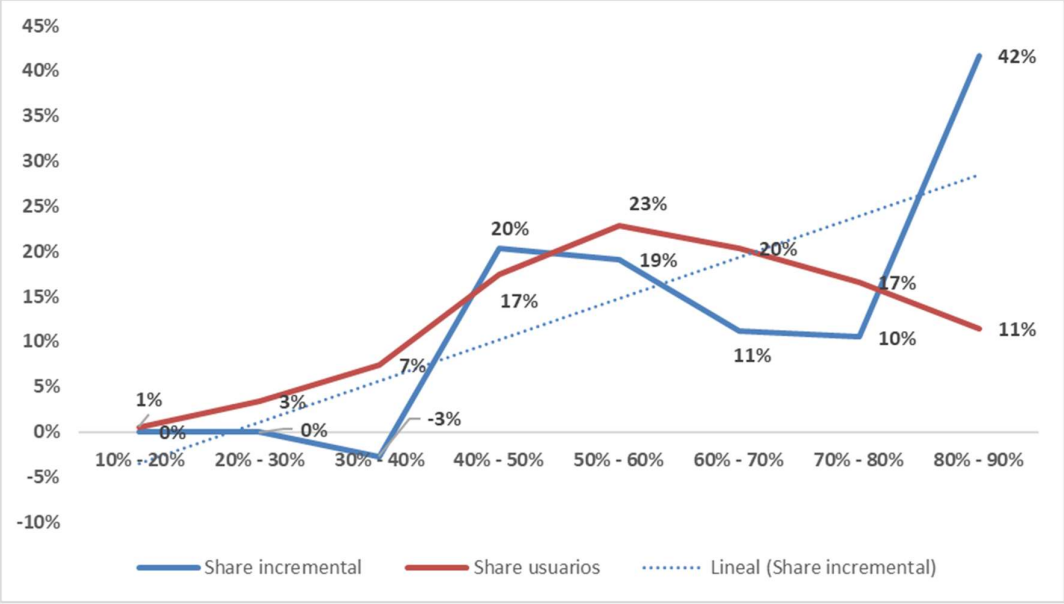
número total de posibles envíos por decil, según el share de participación por decil que nos devolvió el modelo en primera instancia.

Analizando decil por decil, se obtiene que el máximo diferencial de ratio de conversiones (conversiones / enviados) entre grupo de testeo y control se da en el decil del 80% al 90%, mientras que el resto de los deciles desde el cuarto, tienen un diferencial también positivo, pero al menos cinco veces menor al máximo.

De esta manera, si se optara por enviar las notificaciones, solamente a los usuarios con una probabilidad de publicar de entre el 60% y el 90%, se reducirían en un 59% el total de push e emails enviados actualmente, obteniendo aun así un 33% de aumento del total de publicaciones absolutas que hoy se obtiene mediante estos envíos, pero un incremental de publicaciones atribuido a la acción de marketing de un 38% inferior que el actual. Esto ocurre principalmente, porque los usuarios de los grupos de control de estos deciles tienen buena probabilidad de publicar, aún sin recibir una notificación.

La métrica de incrementalidad es una de las más analizadas a la hora de pensar en la puesta en producción de una campaña de marketing. Al demostrar la cantidad de nuevas publicaciones que estamos generando con los envíos de *pushes e emails* por sobre las que ya hubieran ocurrido de manera orgánica, es de gran relevancia a la hora de la toma de decisiones estratégicas. Aun así, es la métrica más desafiante, por presentar un constante *trade-off* entre eficiencia de los envíos e incrementalidad. Al analizar el share de incrementalidad versus el share de población se obtiene:

Ilustración 27 - Share por deciles de usuarios vs incrementalidad



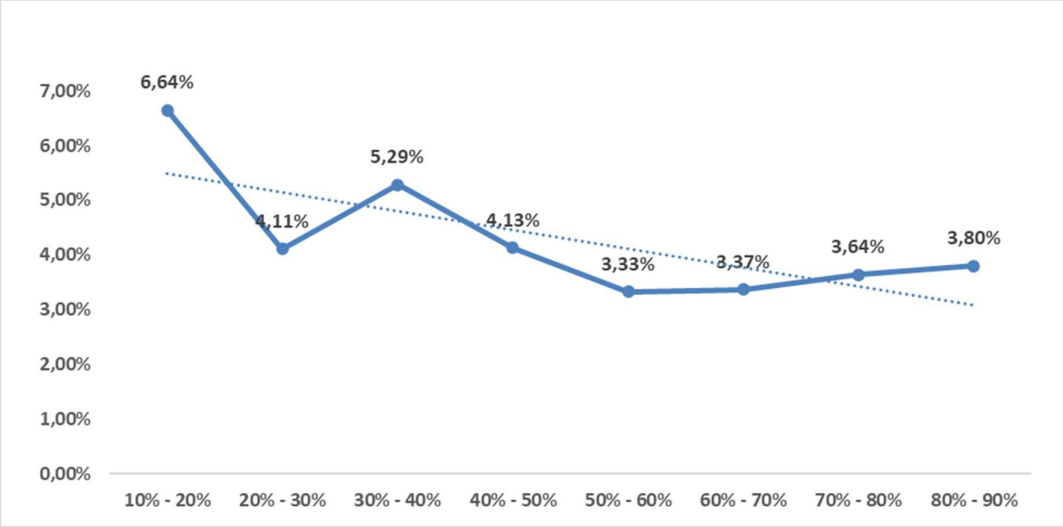
El decil 80% al 90% es el óptimo, por generar en un 42% de las publicaciones incrementales, solamente con el 11% de los envíos totales. Aun así, si solo se enviarán notificaciones a este decil, el incremental estaría muy por debajo del actual como en el caso del

primer escenario proyectado. Dada esta situación, se optó por el armado de otro escenario proyectado, donde los usuarios de los deciles 60% al 90% recibirán dos pushes o emails en el transcurso del año.

Cabe destacar, que existe actualmente un segundo envío a usuarios que recibieron un primer envío de las campañas recurrentes y no publicaron, que tiene resultados muy similares al primer envío. Tomando este resultado como base para el armado del segundo escenario productivo, se reducirían en un 17% el total de *pushes e emails* enviados actualmente, obteniendo un 165% de aumento del total de publicaciones absolutas que hoy se obtiene mediante estos envíos, y un incremental de publicaciones atribuido a la acción de marketing de un 24% superior que el actual. Con este escenario, no solo se generaría una reducción del *spam*, sino una mayor eficiencia de la campaña en todas las métricas de interés.

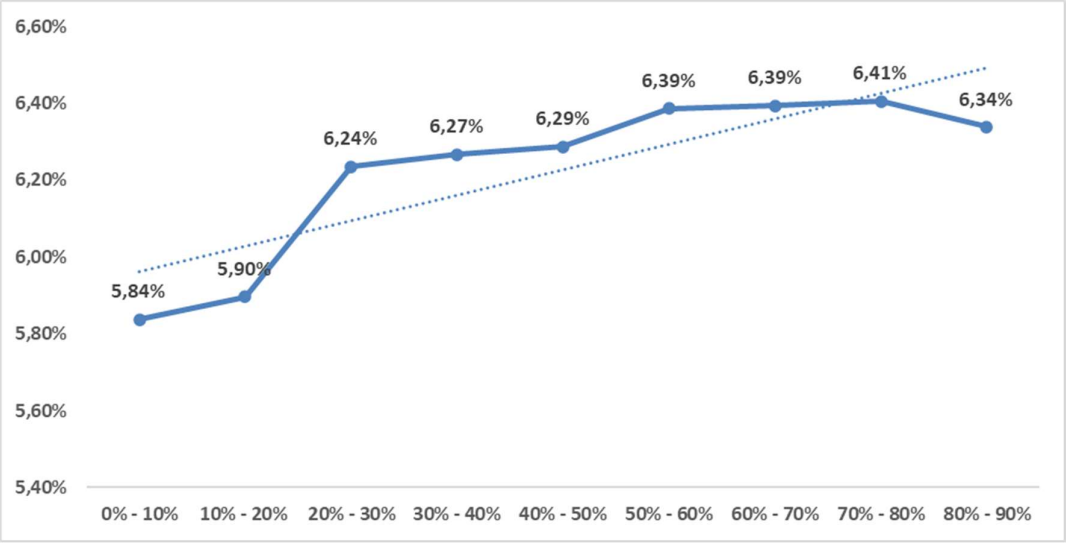
Por el lado del *open rate*, métrica también muy relacionada con *spam* y *engagement* logrado por las campañas, al analizar los datos de los usuarios que hoy reciben *pushes*, se puede ver que tiene una tendencia negativa, a medida que mayor es la probabilidad de publicar asignada por el modelo:

Ilustración 28 - Share por deciles de open rate de envíos actuales



En promedio el *open rate* de los segmentos que se propone atacar a futuro es menor al 4%, métrica que a priori preocupa. Sin embargo, al calcular el *open rate* del total de los usuarios a nivel Brasil, y no solamente de los que reciben *pushes* actualmente, los deciles del 60% al 90% tienen actualmente un open rate de arriba del 6% para el último año.

Ilustración 29 - Share por deciles de open rate de futuros receptores



Al combinar los dos gráficos, se puede pensar que se están enviando notificaciones a usuarios no tan propensos a abrir las notificaciones como el promedio real por decil. Si se supone que el promedio del 2021 se mantendrá constante, en el escenario proyectado donde se mandará al total de usuarios de los deciles seleccionados, no sólo se enviarán *menos pushes e emails* para generar mayor cantidad de publicaciones, sino que se mejoraría la métrica de *open rate* en más de un 50%.

Este escenario proyectado está armado con datos del pasado, con usuarios que ya recibieron notificaciones y se está evaluando su performance y comportamiento. Cuando finalmente se efectúe el reemplazo de las campañas actuales por la generada por el modelo propuesto, se generará un nuevo escenario proyectado, pero con datos futuros, lo cual dará una mayor seguridad a la hora de no solo pausar de manera temporal las campañas actuales, sino de proponer su apagado indefinido.

6. Conclusiones

Hoy en día la estrategia de marketing digital empleada por la compañía no es muy robusta y eficiente. Existen muchos envíos recurrentes que se envían de manera automática, pero sin un seguimiento estricto de las métricas de interés para medir su eficiencia a lo largo del tiempo.

La empresa está en una situación donde, varias de las campañas prendidas como recurrentes ya no están teniendo los resultados que se obtuvieron con las primeras pruebas manuales de hace dos años. El comportamiento de los usuarios a través del tiempo, y con una pandemia de por medio, se ven afectados y no siempre es estable.

El modelo de *machine learning* propuesto, podría mejorar este grave problema de *trackeo* de *performance* de las campañas actuales, trayendo como propuesta concatenar todas las segmentaciones de bases que hoy se aplican en cada envío aisladamente, en una única y unificada campaña de marketing digital.

Esto no solo permite comparar la importancia de los atributos que son fundamentales a la hora de predecir que un usuario va a publicar en una categoría determinada del *Marketplace*, sino que además permite de manera muy fácil y sencilla, agregar y probar cualquier otra característica del usuario que se quiera poner a prueba. Lo importante del modelo es que puede medir el "impacto" de una nueva segmentación, dado la importancia de atributos que nos devuelve el modelo contra las ya utilizadas.

El apagado de los envíos recurrentes actuales y su reemplazo por una campaña acorde a los resultados del modelo, reducirá el envío a sólo un 83% de los que se envían hoy. Esta disminución del número de envíos impacta en el *open rate* de manera directa, generando un aumento de más del 50% de apertura por envío. Implícitamente, el *target* que se está "atacando" es más acertado, evitando el *spam* a muchos usuarios que no están interesados en publicar.

La reducción del total de envíos no genera repercusión negativa en la cantidad de conversiones absolutas, al contrario, se estiman en un 165% más sobre el volumen actual. Por otro lado, la incrementalidad de la campaña pasa a ser apenas superior. Se obtendría un aumento en la incrementalidad del 22% sobre el escenario actual.

Con los resultados obtenidos, podría priorizarse la automatización del modelo de *Machine Learning* que asigna a cada uno de los usuarios del *Marketplace* la probabilidad de publicar un ítem en las categorías de interés. Con este resultado, se podría considerar el reemplazo de las campañas actuales por una nueva, única, unificada y eficiente campaña de marketing online que se alimente de los datos de salida generados por el modelo.

Bibliografía

Arango, C. F. (24 de Mayo de 2018). Presidente de Sancho Bbdo. (J. C. Nonsoque, Entrevistador)

Aufreiter, N. B. (2014). *Why marketers should keep sending you emails.* McKinsey & Company.

Castillo, E. (21 de Enero de 2022). *Doppler*. Obtenido de https://blog.fromdoppler.com/

Chauhan, N. S. (27 de Agosto de 2020). *Data Source*. Obtenido de https://www.datasource.ai/es/data-science-articles/optimizacion-de-hiper-parametros-para-modelos-de-aprendizaje-automatico

Clark, B. (s.f.). *Email Marketing Essentials.* Copyblogger. Rainmaker Digital, LLC.

Ferreira, C. P. (2016). *How can efficiently and effectively align its e-mail marketing efforts to create engagement and avoid spam? (Doctoral dissertation).*

Heras, J. M. (09 de 10 de 2020). *IArtificial*. Obtenido de https://www.iartificial.net/precision-recall-f1-accuracy-en-clasificacion/

Kemp, S. (2021). *Data Reportal*. Obtenido de https://datareportal.com/reports/digital-2021-brazil

Kemp, S. (2021). *DataReportal*. Obtenido de https://datareportal.com/global-digital-overview

Paulson, M. (2019). *Email Marketing Demystified: Build a Massive Mailing List, Write Copy that Converts and Generate More Sales. American Consumer News.*

Radicati, S. &. (2013). *Email statistics report, 2013–2017.* The Radicati Group, Inc., Tech. Rep. .

Ramírez, J. (19 de Julio de 2018). *Medium*. Obtenido de https://medium.com/

Semrush. (2021). Obtenido de https://es.semrush.com/

Waldow, D. &. (2012). *The Rebel´s Guide to Email Marketing: Grow Your List, Break the Rules, and Win.*

YUJRA, H. A. (2014). *WEB SCRAPING PARA LA OBTENCIÓN DE INFORMACIÓN ACTUALIZADA EN INTERNET CON PUSH NOTIFICATIONS PARA SMARTPHONE.* La paz - Bolivia.

Zheng, A. &. (2018). *Feature engineering for machine learning: principles and techniques for data scientists.* O'Reilly Media, Inc.

Zúñiga, J. J. (2020). *Aplicación de algoritmos Random Forest y XGBoost en una base de solicitudes de tarjetas de crédito. Ingeniería, investigación y tecnología.*