



Universidad del Desarrollo
Facultad de Ingeniería

PERSONALIZACIÓN DE OFERTAS DE MARKETING PARA AUMENTO DE USO
DE LA TARJETA DE CRÉDITO

LUIS ESTEBAN SOTO MIRANDA

Proyecto de grado presentado a la Facultad de Ingeniería de la Universidad del
Desarrollo para optar al grado académico de Magíster en Data Science

PROFESOR GUÍA:

LORETO BRAVO

Enero 2022
SANTIAGO

Le dedico este trabajo a mi esposa Ivón, mis hijas Marisol, Francisca y a mis padres Marisol y Luis quiénes han estado apoyándome desde siempre.

También a Coopeuch que me dio el apoyo e inspiración para conseguir este importante crecimiento profesional.

TABLA DE CONTENIDO

RESUMEN	1
1. INTRODUCCIÓN.....	3
2. TRABAJO RELACIONADO	6
3. HIPÓTESIS Y OBJETIVOS.....	7
4. DATOS Y METODOLOGÍA.....	9
4.1. DATOS.....	9
4.2. METODOLOGÍA	10
5. RESULTADOS	135
6. CONCLUSIÓN Y TRABAJO FUTURO.....	27
BIBLIOGRAFÍA.....	39

Resumen

En este trabajo se propone la utilización de una solución analítica que aborda una problemática de negocio real que es aumentar el uso de un producto, en este caso la tarjeta de crédito, para alcanzar un nivel de transacciones que permita hacerlo rentable. En particular se requiere entregar ofertas personalizadas a socios de una cooperativa de ahorro y crédito de Chile que poseen el producto de tarjeta de crédito que involucren un descuento en rubros que les sean atractivos, con el objetivo de que se aumente el uso de tarjeta en clientes con baja transaccionalidad.

En una primera parte se diagnostica el problema utilizando analítica descriptiva, determinando el nivel del problema (el bajo uso), e identificando quiénes son los socios de la Cooperativa que tienen poco uso, a través de los insights que obtengamos de los datos. Para ello realizamos un EDA (Exploratory Data Analysis) con los dataset obtenidos directamente desde la Cooperativa para determinar ciertos perfiles de clientes con mayor y menor uso e implementamos un modelo RFM (Recency, Frequency y Monetary) que segmenta la cartera en base a la información transaccional.

El modelo RFM nos determina cual es el cuartil con más frecuencia y menor recencia, el que representa el 11% de la cartera, que sería el grupo más activo, mientras el cuartil menos frecuente y reciente es el 14% del grupo de inactivos.

Por último para prescribir correctamente la oferta personalizada que sea más efectiva para cada socio, utilizamos una técnica de Sistema de Recomendaciones, en particular se usó el modelo de filtrado colaborativo con feedback implícito basado en factorización matricial con algoritmo de aprendizaje ALS (Alternating Least Square).

Este modelo de sistemas de recomendaciones es ideal para las características del problema que se desea atacar, resolviendo de muy buena forma la ausencia de preferencias, maximizando las entradas obtenidas a través de las compras históricas lo que se traduce en un modelo que tiene una precisión alta en la métrica AUC (área bajo la curva ROC), Precision, Recall y MAP

(Precisión Promedio) sobre los 5 mejores recomendaciones de rubros para cada socio-cliente. Los resultados obtenidos mejoran 22 veces lo que sería una recomendación al azar de un producto.

Particularmente se aprecia el efecto de configurar apropiadamente los hiper parámetros que afectan el modelo y cómo estos influyen en obtener puntos de mejora en los resultados de la prescripción.

En suma, se implementa un método analítico que ha aumentado su popularidad en este último tiempo debido a que soluciona algunos problemas conocidos como es entregar una recomendación atractiva a clientes, de los que tenemos poco historial de uso y que tenemos la obligación de ofrecerles una recomendación asertiva. Estas técnicas innovadoras que han nacido de la mano de empresas digitales como Netflix, Amazon, Facebook, Spotify, entre otras, nos ponen de manifiesto que pueden ser aplicables a industrias totalmente distintas como es la financiera y que es la principal hipótesis que se demuestra en este proyecto.

1. Introducción

La mejor estrategia de clientes es añadirles valor, ofreciéndoles productos y servicios que realmente necesiten, en el momento correcto y de la forma apropiada. Para lograr esta promesa, sin duda que se necesita conocer a los clientes de antemano, saber lo que necesitan, saber que les gusta, sus preferencias, lo que es factible de lograr, usando la información que hay disponible para predecir su comportamiento.

Este trabajo tiene el propósito de suministrar una solución analítica de datos desarrollado específicamente para una empresa cooperativa del sector financiero, dónde los socios o clientes usan los productos financieros contratados. En particular este trabajo se enfoca en el uso de la tarjeta de crédito, la que entrega información detallada acerca de hábitos, comportamientos y estilo de vida de los clientes a través de las compras que realizan. Por ejemplo, compras en comercios, restaurant, supermercados, etc. Estas compras están categorizadas en rubros comerciales de los cuales podemos tener los insights adecuados. Con estos hallazgos en la data, es posible generar nuevas propuestas de valor y acciones de marketing más exitosas, realmente personalizada y que ofrezca a cada cliente lo que realmente necesitan.

Para lograr lo anterior, las soluciones basadas en sistemas de recomendaciones resultan ideales para prescribir ofertas personalizadas tanto para los usuarios (clientes, socios) como para los ítems (rubros de compra). En especial la técnica de filtro colaborativo que es con lo que vamos a trabajar, explota los eventos relacionados entre los usuarios y los ítems para predecir las preferencias de los usuarios que no han comprado o que les falta algún ítem deseado que comprar. Existe mucha literatura referente al filtrado colaborativo con retroalimentación explícita, es decir que los usuarios registren detalladamente sus preferencias o nos den “rating” de cada ítem utilizado. Sin embargo cada vez es más difícil que los usuarios entreguen sus preferencias y en ciertas industrias derechamente no existe la opción a que puedan expresar el nivel de preferencias frente a ciertos ítems o artículos. Es por eso que cuando no tenemos esa valoración, necesitamos inferir la preferencia de los clientes a partir de la retroalimentación implícita del usuario, esto se logra a través de sus compras o ausencia de compras, visualizaciones, reproducciones, clicks, etc. En este sentido es más fácil capturar el dato del uso, porque el usuario no tiene que expresar

explícitamente su gusto. De hecho, la retroalimentación implícita ya está disponible en casi cualquier sistema de información. Sin embargo tiene otras desventajas, porque tiene más ruido y adicionalmente se tiene que inferir lo que puede significar el que alguien haya usado, visitado o haya hecho click en algún sitio, por ejemplo. Otro problema que tenemos es la ausencia de feedback negativo por lo cual hay muchos items no utilizados por una gran mayoría de los usuarios de los cuales no tenemos información o no sabemos como interpretar. Otro desafío es determinar el nivel de intensidad de una preferencia, como por ejemplo el número de compras o cuantas veces reprodujo cierta película o la cantidad de minutos que la observó, eso también podría ayudarnos a darle algún significado distinto de la observación del dato.

En este trabajo en particular utilizamos factorización matricial que es una técnica que se viene utilizando desde comienzos del año 2000 y que tuvo un despegar en su popularidad cuando Simon Funk la utilizó en el Netflix Prize [2] usando un ensamble de varios metodos para poder sacar una predicción de ratings donde los principales fueron, una factorizacion matricial y otro de redes neuronales probabilísticas.

Este método de factorización matricial o modelo de factores latentes, provee nuevas características (features) que no son explicitas como lo son por ejemplo el género o rubro, sino lo que hace el método es representar características que son dimensiones con valores o vectores por cada item y por cada usuario, que en general no sabemos a ciencia cierta que representan, pero que son muy parecido a lo que hacen las redes neuronales dentro de sus capas.

En este caso el metodo busca un vector por cada usuario y un vector por cada item que ojalá estén en la misma dimensión o características latentes, para que después dentro de un espacio comun en el que se pueda representar tanto a los usuarios como los items, pueda recomendar a ciertos usuarios los items o productos que estén más cerca o dentro de un cierto radio. En los modelos de factores latentes o matriciales el objetivo es buscar una proyección en un espacio dónde los usuarios o items puedan mapearse en conjunto como muestra la figura 1 y la idea es aprender con un modelo de los vectores que representan a los usuarios y a los items de forma que efectivamente nos entregue la noción de que los vectores de los items más cercanos a los usuarios efectivamente son las que les van a atraer más.

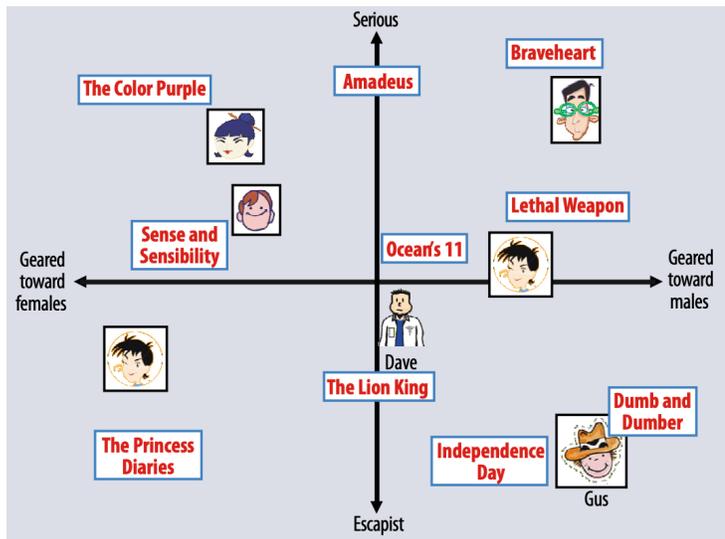


Figura 1. Ejemplo de representación espacial de usuarios e items [3]

Mínimos cuadrados alternos (ALS) es el modelo que usaremos para ajustar nuestros datos y encontrar similitudes. ALS es un proceso de optimización iterativo en el que, en cada iteración, intentamos acercarnos cada vez más a una representación factorizada de nuestros datos originales.

Se ha construido un modelo de datos para los clientes que tienen tarjeta de crédito con información interna de la entidad financiera, demográfica, tenencia de productos, información externa enriquecida como la conformación del grupo familiar, tenencia de vehículos y bienes raíces.

Actualmente se cuenta con 83.053 clientes que poseen el producto de tarjeta de crédito y de ellos hay 12 meses de historia de transacciones con un total de 1,35 MM registros.

En una primera etapa, el análisis está enfocado en la caracterización de todos los clientes a través de un EDA (Análisis exploratorio de los datos). En tanto para segmentar a los clientes se trabajó con un análisis RFM que generará una clasificación en base a la frecuencia de uso, recencia y monetarización de la transacción.

2. Trabajo Relacionado

El trabajo presente en este documento está basado en filtros colaborativos utilizando feedback implícito. Uno de los primeros sistemas de recomendaciones basados en esta técnica de filtros colaborativos se remonta a 1992 con el sistema llamado “Tapestry” realizado por D. Goldberg, D. Nichols, B. M. Oki and D. Terry. D. Goldberg [12] el cual estaba desarrollado para Xerox y tenía el propósito de proveer listas de mails favoritas para los usuarios. En el año 2000 se empezó a usar la Factorización Matricial, ahí encontramos algunos trabajos de referencia de Badrul Sarwar con las primeras aproximaciones que tuvieron algún grado de éxito. En el 2000 publicó “Application of Dimensionality Reduction in Recommender System – A Case Study” [4] y en el 2002 “Incremental singular value decomposition algorithms for highly scalable recommender systems” [5]. Sin embargo la factorización matricial sumó popularidad cuando un miembro del equipo de Netflix Prize, Simon Funk detalló en un blogpost la solución presentada en un cálculo incremental de SVD (singular value decomposition) usando regularización [2]. Para encontrar los valores de los vectores para los usuarios e items podemos utilizar la técnica SVD [6] o usar alguna función de pérdida, como un problema de optimización con regularización que es a lo que vamos a dar énfasis en este trabajo. Para mejorar la representación de estos factores latentes y además mejorar la optimización del costo computacional es que se utilizan estas otras técnicas que son detalladas en el trabajo de Yehuda Koren, Yahoo Research, Robert Bell and Chris Volinsky, tanto en el paper de 2008 “Collaborative filtering for implicit feedback datasets” [3] y en el artículo de la revista Computer del año 2009 Matrix factorization techniques for recommender systems. [1]. En ese artículo se da una explicación acabada de como entender los sistemas de recomendaciones relacionados con factorización matricial tomando como base el caso de éxito Netflix Prize. Se detallan los diferentes algoritmos de aprendizaje que se pueden utilizar, en este caso la optimización SGD (Descenso de la Gradiente Estocástico) popularizado por Simón Funk como a su vez el algoritmo de ALS (Alternating Least Square) que si bien puede que al parecer cumplen la misma función, este último tiene gran ventaja por su forma de trabajar ya que paraleliza su proceso haciéndolo más eficiente y manejando de mejor manera el que la gran mayoría de los datos vengan vacíos y sin información y aun así lograr entregar una recomendación acertada. Además de mejorar ostensiblemente el costo de cómputo al procesar de mejor manera la

factorización matricial. Adicionalmente se incorporan propuestas de como agregar otros atributos al modelo que permitan enriquecerlo y a su vez generar recomendaciones. Ya en el desarrollo del modelo utilicé recomendaciones del blog <https://restanalytics.com> [11] para ajustar los hiperparámetros, también conceptos de Ben Frederickson [10] dónde detalla optimización del computo del modelo y de Jesse Steinweg-Woods [14] donde despliega conceptos en la implementación del modelo.

3. Hipótesis y Objetivos

La hipótesis que se quiere demostrar es si es aplicable utilizar un sistema de recomendación a una industria tan diferente como es la industria Financiera. Si bien es cierto la mayoría de los trabajos que hay acerca de los sistemas de recomendaciones están relacionados en promocionar libros, musica, videos, o articulos de retail en ecommerce, se desea demostrar que es posible construir un modelo para el sector financiero, que sea eficiente en proponer recomendaciones altamente atractivas que apoyen la estrategia de marketing de la Cooperativa, que busca aumentar el uso del producto de tarjeta de crédito en sus socios.

El objetivo principal es construir una solución analítica que apoye esta estrategia comercial y en especial diseñar un modelo que sea eficiente en recomendar rubros de interés a los socios/clientes para que puedan aumentar el uso del producto.

Para lograr el objetivo general debemos conseguir los siguientes objetivos secundarios:

- a) en primer lugar realizar un análisis exploratorio de los datos para entender la problemática del negocio
- b) como segundo aspecto a lograr es necesario identificar quiénes son los socios/clientes que debemos activar y cómo.
- c) en tercer lugar diseñar un modelo de recomendación que entregue una sugerencia a cada socio con el producto de tarjeta de crédito. Esta sugerencia es una campaña personalizada para cada usuario con un descuento en el rubro prescrito por el modelo, esto con el objetivo que el cliente aumente el uso.

- d) Como último objetivo validar el rendimiento del modelo de acuerdo a las métricas apropiadas que se detallan más adelante en este documento.

En particular este trabajo se va a enfocar en socios con bajo uso a los cuales se buscará aumentar su frecuencia de compra con el modelo de recomendación, dejando para los próximos pasos la optimización del modelo, para que incluya adicionalmente a los usuarios sin utilización de la tarjeta de crédito.

4. Datos y Metodología

4.1. Datos

El Dataset construido se compone de 83.053 clientes que poseen el producto de tarjeta de crédito vigente a abril 2019, como también su historial de transacciones desde abril 2018 hasta marzo 2019.

Entrando de manera más detallada en la información contenida en el Dataset de Clientes, se pueden identificar 4 grupos de variables muy claras:

- Información demográfica que permitirá perfilar diferentes clúster de clientes.
- Información asociada a la tenencia de productos que tiene cada cliente (que van desde tarjetas de crédito, cuentas vista, créditos hipotecarios, depósitos a plazo, cuentas de ahorro y seguros asociados a dichos créditos). Así mismo, también se incluyen los niveles de uso de estos productos (saldos asociados a cada uno).
- Información externa: deuda que cada persona tiene en el sistema financiero.
- Enriquecimiento Externo: Datos de conformación familiar, tenencia de bienes raíces, tenencia de vehículos.

El Dataset transaccional consta de 12 meses de historia de transacciones con un total de 1,35 MM registros e incluye compras nacionales, compras internacionales, pagos y avance. El dataset contiene como dato relevante el comercio, rubro asociado, cliente, la fecha de la transacción y monto asociado.

4.2. Metodología

4.2.1 Análisis Exploratorio de Datos (EDA)

Explorando el dataset se visualiza clientes con edades sobre los 40 años, mayoritariamente mujeres que son los que poseen este producto, como se aprecia en la figura 2 y 3. Podríamos decir que es una cartera con edad adulta concentrada entre 50 y 60 años.

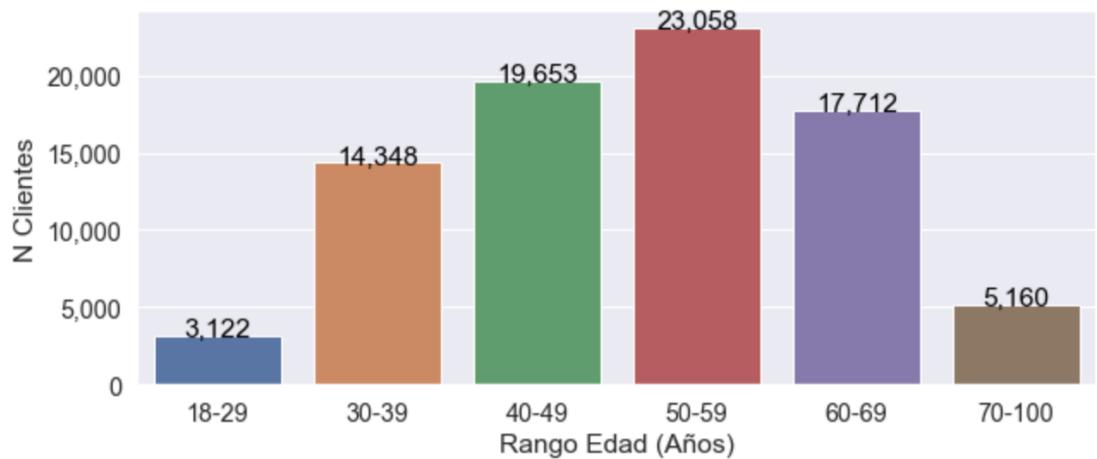


Figura 2. Número de clientes según rango de edad

Se analizó otros atributos demográficos como nivel educacional de la cartera de clientes, el cual está orientado a niveles de educación universitaria y también técnico profesional.

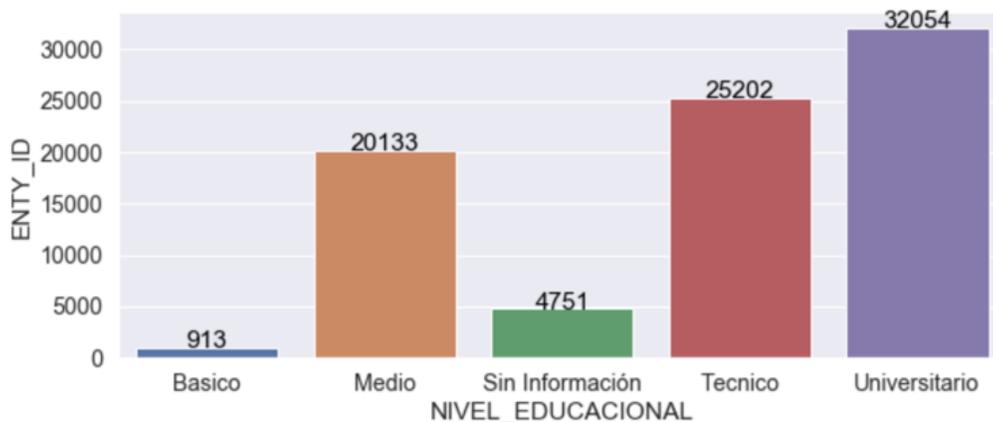


Figura 3. Número de Clientes según Nivel Educativo

En la figura 4 vemos que la cartera de clientes está concentrada en mujeres profesionales en una edad madura. En especial del nivel universitario.

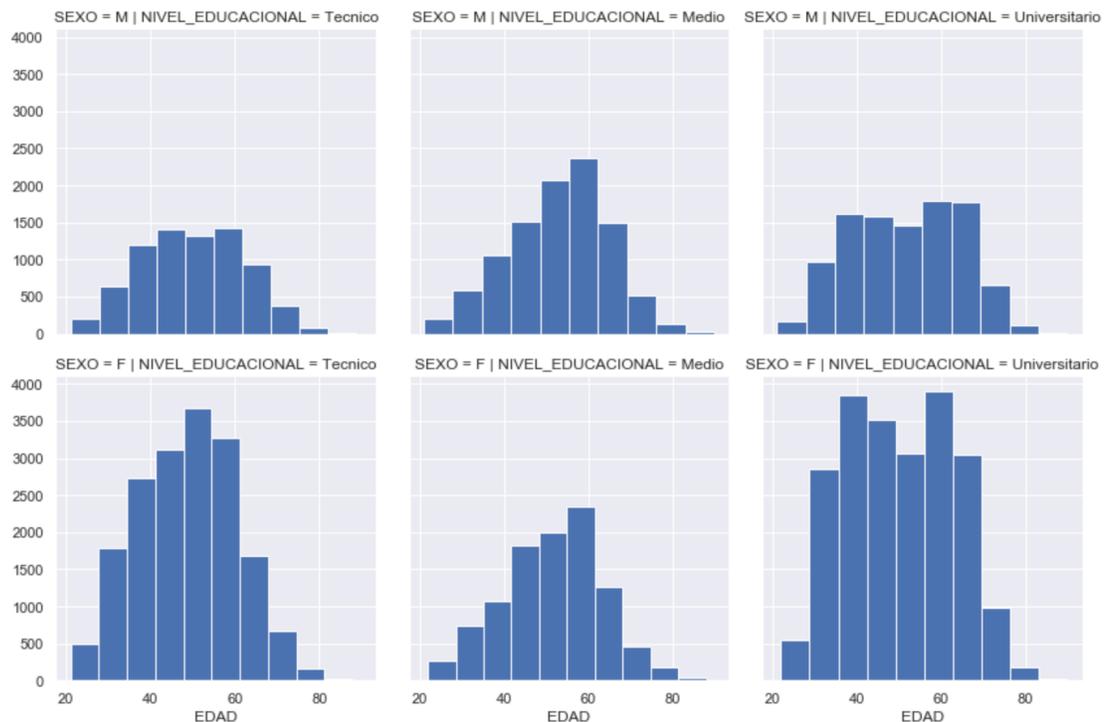


Figura 4. Número de clientes según género, nivel educacional y edad.

Por otra parte si nos concentramos en las transacciones que queremos monitorear tenemos cerca de 1,35 MM de transacciones en 12 meses, en particular solo el 61% realizó algún tipo de transacción. Esto quiere decir que en promedio los clientes realizaron 26 transacciones en un año y 2 cada mes. Dentro del análisis que se requiere profundizar realicé una segmentación para ver clúster de clientes con mayor ocupación y otros con menos o casi nada.

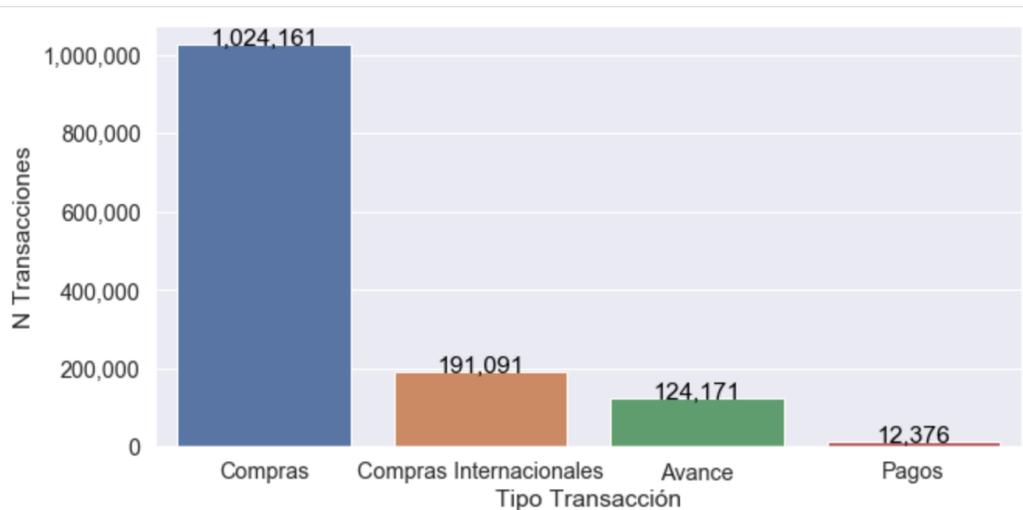


Figura 5. Número de transacciones por tipo

Por otra parte vemos en la figura 6 en qué tipo de rubros se realizan las transacciones para ver de que forma podemos perfilar hábitos y estilos de vida que nos permitan crear propuesta de valor más enriquecidas y porque no decirlo crear productos o promociones que puedan ser más exitosas.

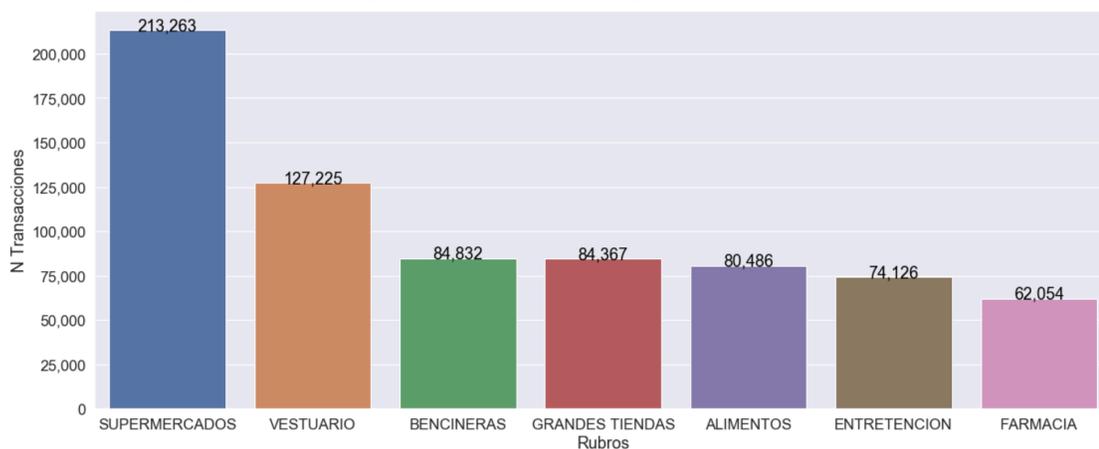


Figura 6. Número de transacciones de compra por rubro

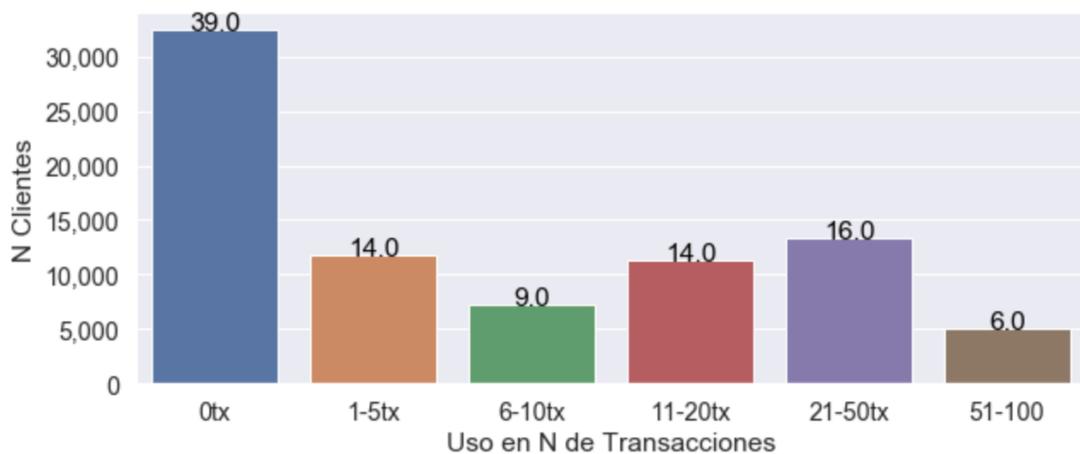


Figura 7. Distribución porcentual de clientes según rango de transacción (últimos doce meses)

En la figura 7 vemos las transacciones de los últimos 12 meses dónde se observa que el 39% no posee transacciones, el 23% tiene una transacción baja y el resto es decir el 36% de la cartera tiene una transaccionalidad adecuada. Este dato reafirma la necesidad de accionar comercialmente este segmento de clientes “dormidos”.

Si analizamos la relación que tiene el uso de la tarjeta de crédito con la tenencia de otro producto de la entidad financiera nos damos cuenta de que el menor uso es cuando el cliente no tiene otro producto adicional. Tiene sentido debido a que el cliente eventualmente se encuentra con menor vinculación. Sin embargo tiene una relación virtuosa cuando está asociada a productos de crédito o la cuenta vista, por lo que en ese mix de productos el cliente aumenta su transaccionalidad como se aprecia en la figura 8. Sin embargo cuando son productos de pasivos como cuentas de ahorro y depósito a plazo la tendencia es a un menor uso. Específicamente este último producto tiene una relación inversa con el uso de tarjeta de crédito siendo el que tiene mayor porcentaje de clientes sin transacciones (45% sin uso) comparado con los clientes que poseen otros productos en conjunto con la tarjeta de crédito.

Cross Producto	S/Trx	Trx Baja		Trx Media		Trx Alta	
	0tx	1-5tx	6-10tx	11-20tx	21-50tx	51-100	> 100tx
CONSUMO	26%	16%	11%	17%	20%	7%	2%
HIPOTECA	32%	10%	9%	21%	17%	8%	4%
AHORRO	37%	14%	9%	14%	17%	7%	3%
DAP	45%	13%	7%	11%	13%	6%	4%
CUENTA VISTA	21%	15%	11%	17%	23%	10%	4%
SIN OTRO PRODUCTO	56%	12%	6%	9%	10%	4%	1%

Figura 8. Distribución porcentual de clientes con tenencia de otros productos según rango de transacción (últimos doce meses)

Si analizamos el ciclo de vida de los clientes encontramos que clientes que tienen niños pequeños y bebés tienen un mayor grado de utilización que los que no tienen hijos o que tienen hijos adultos. De hecho las familia que no tienen hijos el 44% no generó transacciones en todo un año. El 11% de los clientes que tienen bebés mantienen una transaccionalidad adecuada.

Ciclo Vida	S/Trx	Trx Baja		Trx Media		Trx Alta	
	0tx	1-5tx	6-10tx	11-20tx	21-50tx	51-100	> 100tx
Bebes	39%	11%	7%	13%	18%	9%	2%
Hijos Pequeños	36%	13%	9%	15%	18%	7%	2%
Adolescentes	35%	14%	9%	15%	19%	6%	2%
Hijos Adultos	41%	15%	9%	13%	15%	5%	2%
Sin Hijos	44%	13%	7%	12%	15%	6%	3%

Figura 9. Distribución porcentual de clientes según su ciclo de vida familiar y rango de transacción (últimos doce meses)

Al comparar clientes que poseen bienes como casas o vehículos, el análisis nos indica que quienes no tienen bienes raíces y/o vehículos son los que tienen mayor nivel de transaccionalidad que los que si poseen, de hecho el 45% de quienes tienen bienes raíces y además vehículos no tienen uso de la tarjeta durante los últimos 12 meses.

	S/Trx	Trx Baja		Trx Media		Trx Alta	
	0tx	1-5tx	6-10tx	11-20tx	21-50tx	51-100	> 100tx
Bienes							
Bien Raiz	41%	15%	9%	13%	14%	5%	2%
Bien Raiz + Vehiculos	45%	12%	7%	13%	14%	6%	3%
Vehiculos	35%	14%	9%	14%	18%	7%	2%
Sin Bienes	33%	16%	10%	15%	18%	6%	2%

Figura 10. Distribución porcentual de clientes según su tenencia de bienes y rango de transacción (últimos doce meses)

Cuando analizamos si tienen productos financieros en otras instituciones, podemos notar que quienes tienen productos hipotecarios tienen mayor concentración de clientes con alta transaccionalidad. En el lado inverso los que poseen mayor concentración de clientes sin transacción es el grupo de quienes poseen consumo más hipoteca en otras instituciones que alcanza el 45%.

	S/Trx	Trx Baja		Trx Media		Trx Alta	
	0tx	1-5tx	6-10tx	11-20tx	21-50tx	51-100	> 100tx
Deuda Otras Inst							
Consumo	37%	15%	9%	14%	17%	6%	2%
Consumo+Hipoteca	45%	11%	7%	12%	15%	6%	3%
Hipoteca	43%	11%	8%	12%	15%	8%	4%
Sin Deuda	40%	14%	9%	13%	15%	6%	3%

Figura 11. Distribución porcentual de clientes según tenencia de productos en otras instituciones y rango de transacción (últimos doce meses)

Considerando la siguiente tabla se puede apreciar que los segmentos más jóvenes tienen un mayor nivel de uso de la tarjeta que los segmentos más adultos. En especial menores a 50 se ve una mayor utilización.

	S/Trx	Trx Baja		Trx Media		Trx Alta	
	0tx	1-5tx	6-10tx	11-20tx	21-50tx	51-100	> 100tx
Rango Edad							
18-29 Años	34%	18%	10%	14%	17%	6%	1%
30-39 Años	36%	14%	9%	15%	19%	6%	2%
40-49 Años	37%	13%	10%	15%	18%	6%	1%
50-59 Años	39%	15%	9%	14%	17%	5%	1%
60-69 Años	46%	15%	8%	12%	14%	4%	1%
70-79 Años	47%	15%	7%	12%	13%	5%	1%

Figura 12. Distribución porcentual de clientes según rango de edad y rango de transacción (últimos doce meses)

Si quisiéramos resumir el perfil de quienes tienen mayor transaccionalidad diríamos que son clientes más jóvenes, que tienen hijos pequeños, sin bienes raíces ni vehículos, con productos de créditos y/o cuenta vista en la entidad financiera y que además en otras instituciones poseen productos hipotecarios.

4.2.2 Feature Engineering

Con el análisis descriptivo logrado en el EDA se pudo armar un perfil del cliente que más utiliza y perfilar también al que tiene bajo uso. Sin embargo para determinar a quiénes activar y cuándo, se hace necesario crear nuevas variables que faciliten esta tarea.

Es por ello que se crearon variables en base a la información transaccional con el fin de realizar una segmentación comportamental. En concordancia también con el alcance del proyecto, se va a trabajar solamente con socios que hayan hecho transacciones, es decir que hayan utilizado el producto, quedando el 39% de la cartera afuera. Para el grupo de uso se realizó un análisis RFM que genera una segmentación de cuartiles para los tres atributos relevantes del comportamiento de uso del cliente con el producto, en este caso la tarjeta de crédito. A continuación las tres variables adicionales que se incorporan al modelo son:

Recency: Días desde la última compra.

Frequency: nivel de frecuencia de utilización del producto.

Monetary que indica el valor del ticket promedio que realiza el cliente.

	Cuartil	Recencia (N días)		Frecuencia (q Compras)		Monetización (\$ compras acum)	
		Desde	Hasta	Desde	Hasta	Desde	Hasta
Mejor	1	0	3	57	1.157	2.148.856	82.700.715
	2	4	12	30	56	1.096.278	2.148.698
	3	13	25	15	29	505.071	1.096.256
Peor	4	26	363	1	14	0	505.054

Figura 13. Tabla con umbrales obtenidos en la segmentación RFM para generar los cuartiles.

En la figura 13 apreciamos como el mejor cuartil (1) ha hecho compras hace solo 3 días o menos, realizó más de 57 compras en el año y tiene un monto de compra acumulado superior a 2 millones de pesos.

Con estos datos se crearon variables categóricas para segmentar en cuartiles los clientes de acuerdo a cada una de las tres variables. Esto nos permite crear un atributo nuevo “ClassRFM” que identifica a un cluster al cual pertenece cada cliente de acuerdo al comportamiento de uso que tiene con el producto

y permite identificar dentro de una dimensión espacial en que lugar se encuentra ubicado.

	recency	frequency	monetary_value	R_Quartile	F_Quartile	M_Quartile	RFMClass
customer							
680261	4	577	46154402.00	2	1	1	211
462253	5	132	44729009.00	2	1	1	211
336775	4	180	30963536.00	2	1	1	211
219939	4	341	30586479.71	2	1	1	211
1028033	4	76	27336278.81	2	1	1	211

Figura 14. Tabla con muestra de los atributos creados para el analisis RFM

En base a estas nuevas características creadas se define una estrategia de acción que se visualiza en la figura 15. Para simplificar la acción y visualización utilizamos los atributos de frecuencia y recencia, dejando afuera la monetización. El cuadrante verde corresponde a socios o clientes ideales, es decir quienes tienen menos recencia y una alta frecuencia de compra, el segmento que debemos “fidelizar”, es decir mantener su nivel de transaccionalidad. El cuadrante naranja que son clientes con un buen nivel de frecuencia pero que últimamente no han realizado compras con su tarjeta es el nicho que posiblemente se esté “fugando” y que es imperioso reactivar rápidamente. Tenemos también el cuadrante amarillo que son los que han comprado recientemente pero que tenemos que mejorar su frecuencia, en este caso debemos “asegurar” que compre regularmente. Por último tenemos el cuadrante rojo que son los que están menos vinculado ya que no han comprado recientemente ni tampoco lo hacen frecuentemente. Este nicho es el que hay que “revivir”. Con este análisis RFM determina cual estrategia comercial a seguir y responde las preguntas de a quiénes y cuando activarlos comercialmente.

		+ Frecuency Quartil -			
		1	2	3	4
Recency Quartil + -	1	11%	7%	5%	3%
	2	7%	8%	7%	4%
	3	5%	7%	7%	5%
	4	1%	3%	6%	14%

Figura 15. Distribución porcentual de clientes según rango de transacción (últimos doce meses)

4.2.3 Modelo

Después de conocer a través de los datos cómo es el perfil de la cartera de clientes, diagnosticar en qué situación se encuentra el problema de poca utilización del producto que estamos abordando, determinar a quiénes debemos “accionar” y qué estrategia comercial seguir, nos enfrentamos al “cómo”, con cual oferta podemos lograr “activar a un cliente”. En esta línea de trabajo podemos decir que la experiencia que se ofrece a los clientes es quizá el factor clave a la hora de que un cliente se incline a tomar o utilizar un producto o servicio frente al mar de ofertas a los que se ve expuesto todo el tiempo. Algo que contribuye enormemente a mejorar esta experiencia son los sistemas de recomendaciones y que son utilizados ampliamente en cada interacción que hacen los clientes en la web y que también son parte fundamental en las campañas de marketing digital.

Llevado al problema que nos hemos propuesto resolver y de acuerdo a la naturaleza de los datos con los que se cuenta y que fue observado en el análisis exploratorio, nos lleva a escoger un algoritmo de sistema de recomendación de **filtrado colaborativo con Feedback Implícito**. Si bien es cierto que no es la única manera de resolver esta problemática, si podríamos decir que es una de las más populares y con mejor eficiencia de estos tiempos. Debido a que las compras realizadas por los clientes con tarjeta de crédito no contienen “rating” explícito por parte de ellos, se asume que el comprar o la frecuencia de comprar en ciertos rubros y comercios implica inherentemente una preferencia y un “rating” artificial. Desde luego el hecho de contar con que 39% de los clientes no poseen transacciones (estos no son parte del alcance de este proyecto) y que un porcentaje no menor posee baja transaccionalidad atenta sobre un buen rendimiento del modelo, sin embargo voy a explorar una técnica que ayudan a sobrellevar este problema de mejor manera. También es esa una de las razones de porqué estamos utilizando este algoritmo.

El modelo que vamos a utilizar se basa principalmente en el trabajo realizado por Yehuda Koren, Robert Bell and Chris Volinsky, en especial en el paper de 2008 “Collaborative filtering for implicit feedback datasets”[3] y en el artículo de la revista Computer del año 2009 Matrix factorization techniques for recommender systems.[1]. En ambos documentos nos presentan una manera de abordar datasets donde solo tenemos feedback implícito, tal como es el proyecto que estamos abordando de clientes con tarjeta de crédito de la cooperativa financiera, donde el número de compras son el rating que tenemos de los clientes sobre un rubro de comercio.

El modelo que vamos a utilizar es la **factorización matricial** que nos ayuda a reducir una matriz grande de interacciones usuario, elemento y descubrir las características latentes (u ocultas) que las relacionan entre sí en una matriz mucho más pequeña de características de usuario y características de elementos.

En la figura 16 vemos una matriz original R de tamaño $M \times N$, donde M es el número de usuarios y N es el número de elementos. Los “Rating” en este caso son las compras que realizan los clientes (usuarios) en los distintos rubros (items). Esta matriz es bastante escasa, sobretodo en el problema que estamos abordando donde el 99,97% de los datos de la matriz son escasos ya que la mayoría de los usuarios compran poco y solo en algunos rubros. Aquí podemos factorizar esta matriz en dos matrices más pequeñas separadas: una con dimensiones $M \times K$ que serán nuestros vectores de características latentes de usuario para cada usuario(cliente)

(U) y una segunda con dimensiones $K \times N$, que tendrá vectores de características de ítems latentes para cada ítem-rubro (V).

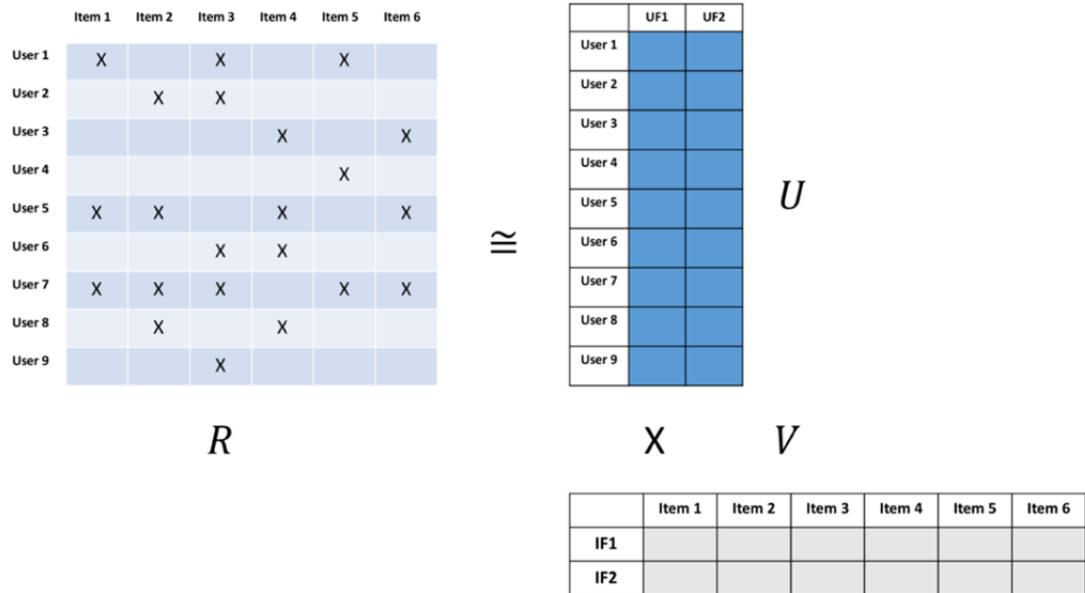


Figura 16. Representación gráfica de Factorización matricial

En nuestro proyecto tenemos mil rubros distintos (items) y 50 mil clientes con uso (users), sin embargo como revisamos en el análisis descriptivo la mayoría de los clientes tienen bajo uso y por ende pocos rubros utilizados por cada usuario lo que hace que la matriz original sea muy dispersa o poco densa, ya que la mayoría de los elementos no se usan, como se visualiza en la figura 17, tiene un 99,97% de escasez.

99.97865788501605

Figura 17. Resultado del cálculo de densidad de la matriz.

Se puede ver a cada usuario (cliente) como un vector de sus valores de lo que prefiere, y al mismo tiempo expresar cada ítem (rubro en el que compra) como un vector de los gustos que representan. Esto permite hacer

una recomendación con bastante facilidad y nos da la posibilidad de encontrar conexiones entre clientes que no tienen rubros específicos en común, pero comparten gustos comunes. Estos métodos han gozado de popularidad en este último tiempo por su escalabilidad y exactitud, lo que permite modelar diversas situaciones de la vida real.

Con datos implícitos (a diferencia de los modelos explícitos en los cuales se cuentan con valoraciones, ratings, etc.), la diferencia radica en cómo se maneja todos los datos faltantes en la matriz muy escasa (con muchísimas celdas vacías). Básicamente, se necesita alguna forma de aprender de los datos que faltan. Al tener tan pocos datos la matriz (densidad menor a 2%), el procedimiento corre el riesgo de sobre-ajustarse (over-fitting). En el caso de la matriz de nuestro proyecto que se está analizando es menor del 1% por lo que será necesario utilizar un regularizador para penalizar valores muy altos de los coeficientes de los vectores latentes.

Asignando aleatoriamente los valores en U y V y usando mínimos cuadrados iterativamente podemos llegar a qué pesos producen la mejor aproximación o representación de R (la matriz original). El enfoque de mínimos cuadrados en sus formas básicas significa ajustar alguna línea a los datos, midiendo la suma de las distancias al cuadrado de todos los puntos a la línea y tratando de obtener un ajuste óptimo minimizando este valor. Con el enfoque de **mínimos cuadrados alternos (ALS)** se trabaja con la misma idea, pero se alterna iterativamente entre optimizar U y fijar V y viceversa. Se hace esto para cada iteración para llegar más cerca de $R = U \times V$.

El primer desafío es hacer esta factorización de manera eficiente: al asumir que las incógnitas (sin valores, nulos, etc.) son negativas, la implementación lo toma como una entrada de la matriz, elevando el número de entradas totales a considerar, en comparación con el número infinitamente inferior de entradas distintas de cero.

El segundo problema es que no se puede estar completamente seguros de que un cliente que no compre en un rubro realmente signifique que no le gusta.

El objetivo entonces es aprender los factores del usuario X_u y los factores del rubro Y_i minimizando una función de pérdida de la suma de errores cuadrados ponderada de confianza:

$$loss = \sum_u \sum_i C_{ui} (P_{ui} - X_u Y_i)^2 + \lambda (\|X_u\|^2 + \|Y_i\|^2)$$

C_{ui} es la confianza que tenemos de que al usuario le gusta el ítem (rubro), P_{ui} es un valor binario que indica si el usuario compró en el rubro o no (preferencia), y λ es un regularizador $L2$ básico para reducir el sobreajuste.

Para minimizar los factores del usuario, fijamos los factores del elemento constantes y luego tomamos la derivada de la función de pérdida para calcular X_u directamente:

$$X_u = (Y^T C_u Y + \lambda I)^{-1} (Y^T C_u P_u)$$

Los factores de los elementos se calculan de manera similar, y todo se minimiza alternando hacia adelante y hacia atrás hasta que converge.

La parte interesante es cómo aprende sobre todos los datos, aunque solo tiene que trabajar en los elementos distintos de cero, lo que hace que tenga una eficiencia computacional importante comparado con SVD que trabaja con toda la matriz.

Para la programación del modelo se utilizó la librería Implicit que desarrolló Ben Frederickson [10] que usa Cython y OpenMP para paralelizar el cálculo. Esto hace que los cálculos de la matriz así como el entrenamiento del modelo sea mucho más rápido. Realizamos la configuración de los siguientes parámetros:

- Número de factores latentes (características latentes definidas)
- Número de iteraciones (cuantas veces vamos a iterar para optimizar)
- Regularización (constante que será usada en la función de costo)
- Alpha (valor correspondiente al factor que le vamos a aplicar a la métrica de confianza).

Mínimos cuadrados alternos (ALS)

ALS es un proceso de optimización iterativo en el que, en cada iteración, tratamos de acercarnos cada vez más a una representación factorizada de la matriz original.

Debido a que tanto X_u y como Y_i son desconocidas, la ecuación a resolver no es convexa. Sin embargo, si fijamos una de las desconocidas, el problema de optimización se convierte en cuadrático y se puede resolver de manera óptima, por lo tanto, las técnicas de ALS rotan entre fijar las X y fijar las Y . Cuando todas las X son fijas, el sistema vuelve a calcular las Y_i resolviendo un problema de mínimos cuadrados y viceversa. Esto asegura que cada paso disminuya la ecuación hasta la convergencia.

ALS tiene dos ventajas importantes, la primera es que el sistema puede usar la paralelización. en ALS, el sistema calcula cada Y_i independientemente del otro item, factoriza y calcula cada X independientemente de los otros factores del usuario. Esto da lugar a una paralelización potencialmente masiva del

algoritmo lo que lo hace ser muy eficiente en la potencia de cálculo. El segundo caso es que es ideal para sistemas centrados en datos implícitos. Debido a que el conjunto de entrenamiento no puede considerarse escaso, no sería práctico hacer un bucle sobre cada caso de entrenamiento individual, como lo hace por ejemplo el descenso de gradiente. ALS puede manejar estos casos de manera eficiente.

Para el entrenamiento se trabajó con el 20% del dataset y se dejó el 80% restante como prueba.

5. Resultados

Calcular las métricas de calidad de las recomendaciones para los sistemas de recomendación de retroalimentación implícita no es una tarea sencilla. A diferencia de los modelos de clasificación de machine learning en los que utilizamos una matriz de confusión y podemos sacar métricas de precisión y recall en base a las predicciones y las clases reales, en el caso de los sistemas de recomendación necesitamos evaluar un conjunto de recomendaciones de items para cada usuario.

Por otra parte para los modelos de retroalimentación explícita tenemos los “ratings” dados por los usuarios para cada Item y en base a eso se han creado una gama de métricas que se han popularizado pero que en el caso de modelos de retroalimentación implícita no aplican porque no tenemos esos ratings.

Para los modelos de filtrado colaborativos con retroalimentación implícita las métricas se calculan por usuario, generando una clasificación de los items según las predicciones del modelo (en orden descendente), ignorando los elementos que se encuentran en los datos de entrenamiento para cada usuario. Los elementos que no fueron consumidos por el usuario (no presentes en X_{train} y no presentes en X_{test}) se consideran entradas "negativas", mientras que los elementos en X_{test} se consideran entradas "positivas", y los elementos presentes en X_{train} se ignoran para estos cálculos.

Para este proyecto en particular vamos a utilizar la librería `recommetrics` que se diseñó especialmente para simplificar la medición del rendimiento de estos modelos de filtros colaborativos con retroalimentación implícita. [15]

Las métricas que utilizamos en el proyecto se van a basar en las 5 primeras recomendaciones para cada usuario ordenadas de forma descendente de acuerdo al score. Las métricas se detallan a continuación:

P@K (“precision-at-k”)

Indica la proporción de elementos entre los k(5) principales recomendados (después de excluir aquellos que ya estaban en el dataset de entrenamiento) que se pueden encontrar en el conjunto de prueba para ese usuario:

Total consumidos (10)

$$\text{Precision} = \frac{|\text{Recomendados} \cap \text{Consumidos}|}{|\text{Recomendados}|}$$



$$\text{Precision (0.6)} = \frac{|\text{Recomendados} \cap \text{Consumidos}(3)|}{|\text{Recomendados}(5)|}$$

Figura 18. Ejemplo de como obtener la métrica de Precision.

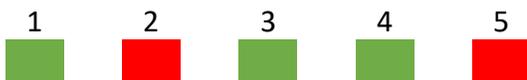
Esta es quizás la métrica más intuitiva y sencilla, pero puede presentar mucha variación entre los usuarios y no tiene en cuenta aspectos como la cantidad de elementos de prueba disponibles o los rangos específicos en los que se muestran. Para el caso dónde tenemos una cantidad importante de items para recomendar el valor puede ser especialmente bajo.

R@K ("Recall-at-k")

Proporción de los elementos de prueba que se recuperan en la lista recomendada de los K principales. El cálculo es lo mismo que la precisión, pero la división es por el número de elementos de prueba en lugar de k:

Total consumidos (10)

$$\text{Recall} = \frac{|\text{Recomendados} \cap \text{Consumidos}|}{|\text{Consumidos}|}$$



$$\text{Recall (0.3)} = \frac{|\text{Recomendados} \cap \text{Consumidos (3)}|}{|\text{Consumidos}(10)|}$$

Figura 19. Ejemplo de como obtener la métrica de Recall.

MAP@K (“average precision-at-k”)

La precisión y el Recall observan todos los elementos en el top-K por igual, mientras que uno podría querer tener en cuenta también la clasificación dentro de esta lista top-K, para lo cual esta métrica es útil. “Precisión promedio” intenta reflejar las precisiones que se obtendrían en diferentes puntos del Recall:

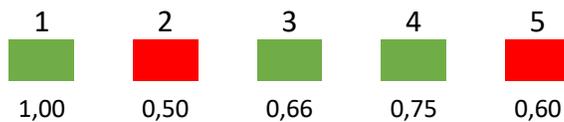


Figura 20. Ejemplo de como obtener la métrica de MAP.

En el ejemplo de la figura 20, el valor del promedio de las precisiones en cada punto sería 0,7.

Esta es una métrica que, hasta cierto punto, considera la precisión, el recall y la clasificación dentro de top-K.

El promedio de esta métrica entre los usuarios se conoce como "Precisión promedio media" o "MAP@K". Es quizás la métrica más representativa a la hora de evaluar el modelo.

ROC-AUC (área bajo la curva Roc)

Esta métrica considera la clasificación completa de los elementos en lugar de solo el top-K. Está acotado entre cero y uno, con un valor de 0,5 correspondiente a un orden aleatorio y un valor de 1 correspondiente a un orden perfecto (es decir, cada elemento positivo tiene una puntuación predicha más alta que cada elemento negativo).

Hiperparámetros a ajustar

Factores: Número de Factores Latentes, o grados de la dimensionalidad del modelo para los usuarios y elementos o items.

Regularización (L2): Escala de Regularización de Usuarios (Clientes) y Elementos (Rubros). Parámetro de penalización para evitar el sobreajuste.

Alpha: Escala de Confianza.

Iterations: Número de Iteraciones a correr hasta optimización ALS

Vamos a realizar ajustes en los hiper parámetros hasta converger en valores de Precision, Recall, MAP y AUC máximos que permitan mejorar las prescripciones del modelo.

factores latentes	Tunning Modelo			Precision@K	Recall@K	MAP@K	ROC_AUC
	regularizacion	confianza	interacciones				
	Random			0,0122	0,0200	0,0098	0,5024
25	0,10	25	15	0,1299	0,2288	0,1289	0,7632
20	0,10	25	15	0,1633	0,3027	0,1787	0,7959
15	0,10	25	15	0,1540	0,2876	0,1680	0,8162
10	0,10	25	15	0,1614	0,3029	0,1788	0,8543
5	0,10	25	15	0,1665	0,3062	0,1886	0,8854
5	0,05	25	15	0,1636	0,3050	0,1887	0,8831
5	0,01	25	15	0,1657	0,3068	0,1885	0,8842
5	0,10	15	15	0,1785	0,3228	0,2003	0,8878
5	0,10	10	15	0,1846	0,3280	0,2075	0,8757
5	0,10	5	15	0,1929	0,3399	0,2198	0,8782
5	0,10	3	15	0,1951	0,3415	0,2242	0,8704
5	0,10	1	15	0,1910	0,3332	0,2171	0,8493
5	0,10	3	10	0,1947	0,3413	0,2223	0,8724
5	0,10	3	20	0,1938	0,3383	0,2209	0,8686
5	0,10	3	30	0,1947	0,3417	0,2239	0,8673

Figura 21. Ajuste de hiper parámetros del modelo

En el ejercicio de ajustar los hiper parámetros pasamos de un rendimiento aleatorio muy bajo en todas las métricas, en especial MAP@K(5) de 0,0098 y ROC AUC de 0,5 que es entregar una recomendación al azar, a optimizarlo casi 22 veces con los ajustes de parámetros que se muestran en la figura 21. Este hiper parámetro Alpha (Confianza) nos permite ajustar el modelo de acuerdo a la preferencia inferida en el uso del producto (inferencia binaria) considerando la cantidad utilizada de forma óptima para reflejar la intensidad de una preferencia. Por ende se partió probando con un valor alto (25), iterando hasta lograr un mejor performance con un valor 3. Luego se probó con el hiper parámetro de número de factores latentes con el valor 25 y fui bajando hasta llegar al óptimo de 5 el cual incrementó la performance de forma notable en las cuatro métricas que estamos observando. Este fue el hiper parámetro más influyente en la mejora del rendimiento del modelo. Lo sorprendente es que bajar el numero de factores latentes hace que la eficiencia del modelo aumente. Probablemente el utilizar un número alto de factores latentes aumenta la dimensionalidad y sobre representa el vector de los clientes (usuarios) y rubros (items) lo que provoca overfitting. También es importante destacar que a diferencia de otros dataset que se utilizan en sistema de recomendaciones (industrias del entretenimiento que tienen

cientos de miles de items que ofrecer), en nuestro caso el número de items (rubros) es limitado llegando a 1.000 el potencial de recomendaciones. En general bajar la regularización no mejora el indicador porque disminuir considerablemente la penalización provoca sobreajuste del modelo y afecta considerablemente la calidad de las recomendaciones. En cuanto al hiper parámetro del número de iteraciones no mejoró significativamente la performance del modelo por lo que se mantuvo el parámetro inicial de 15 iteraciones. En este sentido conocer a que se refiere cada hiper parámetro, conocer la naturaleza de los datos y entender como se ajustan, resulta ser una herramienta poderosa para aumentar el rendimiento del modelo.

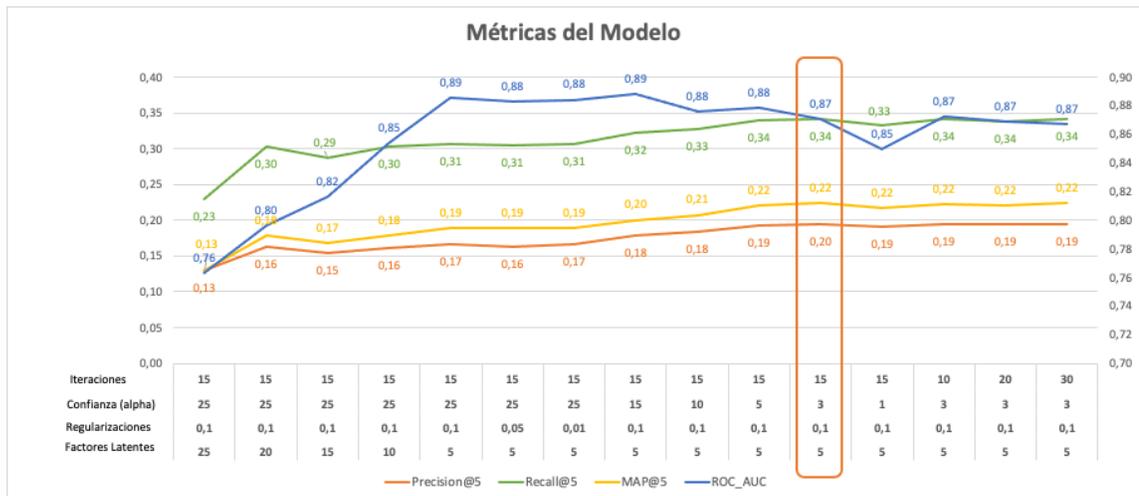


Figura 22. Evaluación de ajustes de hiper parámetros del modelo con las cuatro métricas

Ejemplos de recomendación con el modelo Calibrado

Ejercicio de búsqueda de rubros similares dado un rubro en particular

Se genera una búsqueda de rubros similares para el Rubro **Supermercados**

	rubro	score
0	SUPERMERCADOS	0.211696
1	SUPERMERCADOS CASO A CASO	0.211537
2	FARMACIAS	0.210939
3	RESTAURANTES	0.205373
4	COMBUSTIBLES COPEC	0.173842
5	SERVICIOS DE REMOLQUE	0.061941
6	INSTALACIONES SANITARIAS	0.053692
7	AGENCIA DE EMPLEO	0.048301
8	CASAS DE CAMBIO	0.046399
9	FUMIGACION Y DESINFECCION	0.046082

Figura 23. Búsqueda de rubros similares ordenado por el puntaje que entrega el modelo

En general se encontró rubros similares y tiene correspondencia con el score. Sin embargo desde el rubro 5 en adelante baja considerablemente el score y también la calidad de la recomendación, proponiendo rubros que distan mucho de la naturaleza del rubro de “Supermercados”.

Ejercicio de búsqueda de recomendación de rubros para un cliente en particular

Se genera una búsqueda de recomendaciones de rubros para el cliente **1030995**, este realiza las siguientes compras que se detallan en la figura 24, se ordena de acuerdo al número de compras totales y se rankea de acuerdo a lo mismo.

	user	rubro_id	rubro	compras	rank
246320	1030995	117	SUPERMERCADOS CASO A CASO	15	1.0
246319	1030995	68	SEGUROS GENERALES	12	2.0
246323	1030995	703	GRANDES TIENDAS	8	3.0
246322	1030995	702	TIENDAS DE CALZADO PARA LA FAMILIA	6	4.0
246324	1030995	704	TIENDAS POR DEPARTAMENTO	2	5.0
246321	1030995	118	SUPERMERCADOS	1	6.0

Figura 24. Compras reales realizadas por el cliente 1030995

El sistema de recomendación propone ofrecer descuentos en estos rubros para aumentar la transaccionalidad, todos poseen un score alto (puntaje que entrega el modelo sobre una recomendación) y son afines a las compras históricas del cliente:

	rubro	score
0	GRANDES TIENDAS	1.048270
1	SEGUROS GENERALES	1.002402
2	SUPERMERCADOS CASO A CASO	1.000743
3	TIENDAS DE ROPA PARA LA FAMILIA	0.996791
4	FARMACIAS	0.984775
5	TIENDAS POR DEPARTAMENTO	0.970645
6	SUPERMERCADOS	0.968131
7	TIENDAS DE CALZADO PARA LA FAMILIA	0.936915
8	RESTAURANTES	0.811414
9	PERFUMERIAS	0.729235

Figura 25. Recomendaciones realizadas por el modelo para el cliente 1030995

Se genera una búsqueda de recomendaciones de rubros para el cliente **593880**, este realiza las siguientes compras:

	user	rubro_id	rubro	compras	rank
136257	593880	119	MINIMARKET	5	1.5
136259	593880	320	RESTAURANTES	5	1.5
136258	593880	224	FARMACIAS	2	4.0
136255	593880	109	FAST FOOD	2	4.0
136254	593880	102	ALIMENTOS EN GENERAL	2	4.0
136256	593880	118	SUPERMERCADOS	1	6.0

Figura 26. Compras reales realizadas por el cliente 593880

El sistema de recomendación propone ofrecer descuentos en estos rubros para aumentar la transaccionalidad, todas las recomendaciones poseen un puntaje alto lo que garantiza una recomendación adhoc a lo que el cliente ha comprado en el pasado:

	rubro	score
0	SUPERMERCADOS CASO A CASO	1.024253
1	SUPERMERCADOS	1.008550
2	MINIMARKET	0.995018
3	RESTAURANTES	0.991683
4	ALIMENTOS EN GENERAL	0.967221
5	FARMACIAS	0.965296
6	COMIDAS PREPARADAS	0.899298
7	FAST FOOD	0.881923
8	PANADERIAS	0.816650
9	COMBUSTIBLES COPEC	0.769214

Figura 27. Recomendaciones realizadas por el modelo para el cliente 593880

Se genera una búsqueda de recomendaciones de rubros para el cliente **643107**, este cliente solo posee una compra en el historial que tenemos del dataset:

	user	rubro_id	rubro	compras	rank
154824	643107	80	MUNICIPALIDADES	1	1.0

Figura 28. Compra real realizada por el cliente 643107

El sistema de recomendación propone ofrecer descuentos en estos rubros para aumentar la transaccionalidad, no todas las recomendaciones poseen un score alto, solamente el rubro ya utilizado y encuentra otros rubros no tan afines a la compras histórica del cliente:

	rubro	score
0	MUNICIPALIDADES	0.791731
1	REPUESTOS Y ACCESORIOS	0.528822
2	CLINICAS DENTALES	0.354401
3	COMBUSTIBLES SHELL	0.326586
4	PERFUMERIAS	0.284463
5	COMBUSTIBLES ESSO	0.277301
6	CORREDORES SEGUROS	0.260799
7	COMBUSTIBLES COPEC	0.246287
8	VESTUARIO PARA MUJER	0.245880
9	NEUMATICOS Y BATERIAS	0.242475

Figura 29. Recomendaciones realizadas por el modelo para el cliente 643107

Podemos ver en el caso del cliente 643107, que influye cuando hay poca información en el dataset acerca del cliente, porque la recomendación no es tan afin como para otros clientes que tienen más compras en el historial. Sin embargo el sistema de igual forma ofrece recomendaciones con un score

menor pero que potencialmente van a tener un grado de éxito en las campañas.

6. Conclusiones y próximos pasos

Este trabajo se concentró en estudiar y aplicar los filtros colaborativos en un dataset de la industria financiera. Pudimos observar que los dataset con feedback implícito están en todas las industrias ocultando a simple vista los patrones y relaciones que tienen los clientes con los productos. Sin lugar a dudas estas técnicas ocupadas en este trabajo proporcionan las herramientas adecuadas para extraer estos insights que dan una ventaja competitiva a la hora de proporcionar una experiencia a los usuarios cuando interactúan con los diversos canales de cada organización.

En particular puedo decir que es muy necesario entender a cabalidad la problemática del negocio y luego profundizar en el conocimiento de los datos (realizar un buen EDA) que nos permita comprender y explicar las distintas problemáticas del negocio. En especial se pudo aplicar un mix de técnicas, algoritmos, modelos y análisis para resolver y solucionar lo que se había puesto como desafío. En este sentido existen innumerables modelos y algoritmos de sistemas de recomendaciones, sobre los cuales debemos elegir en base a lo que el negocio indica, lo que la estrategia comercial también defina. Adicionalmente los mismos datos, indicadores, KPI's, deben proporcionar guía para esta misma estrategia, llegando incluso a modificarla si los insight que obtenemos de los datos nos dicen otra cosa. También la data con la que se cuenta condiciona el modelo o técnica que se va a emplear.

Para las recomendaciones personalizadas acertadas para los clientes, sin duda hay algunas tareas clave que debemos realizar para mejorar los resultados, como lo es el preprocesamiento de los datos, la optimización de los hiper parámetros, la Regularización o penalización para evitar el sobreajuste del modelo, Alpha (confianza), los factores, las iteraciones, etc.

El desafío siguiente será como abordar los clientes que no transaccionan y que lamentablemente carecen de data para tener algún feedback de lo que le es de interés. Este problema llamado “partida en frío” (Cold Start) es algo que llama la atención para abordar en una siguiente etapa y es donde más se puede obtener logros significativos que aporten a la rentabilidad del negocio. El 39% de la cartera de clientes está en una situación de desvinculación absoluta con el producto y es el motivo que nos impulsa a lograr seducirlos a que usen por primera vez el producto de tarjeta de crédito con un gancho comercial atractivo, con una comunicación personalizada y principalmente

con una recomendación que cautive el interés del cliente y que permita sacarlo de su letargo comercial.

En este sentido a nuestro favor tenemos excelentes datos del perfil de los clientes para superponerlos a las recomendaciones, para crear un resultado conjunto o un modelo híbrido que entregue una mejor solución a casi la mitad de la cartera que no transacciona. En particular se cuenta con varios atributos interesante que pueden ayudar a la segmentación y generar clones que mitiguen el problema de los nuevos usuarios o antiguos de los cuales no hay mucha data transaccional. También será interesante explorar modelos de deep learning que han aparecido ultimamente en el mundo de los sistemas de recomendaciones.

Finalmente es interesante seguir incorporando nuevas métricas que permitan mejorar la performance del modelo de recomendación. Si bien es cierto los indicadores relacionados con la curva ROC infieren la calidad del modelo, no son tan certeros a la hora de definir si la recomendación es precisa o no, por lo mismo se agregó nuevas métricas especialmente diseñadas para medir el rendimiento de este tipo de modelos con feedback implícito. No obstante hay otras nuevas métricas que tal vez nos proporcionen elementos que nos ayuden a evaluar los distintos aspectos la calidad del modelo y sus recomendaciones, que nos guiarán en los puntos de mejora que requieren atención.

Bibliografia

[1] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, (8):30–37, 2009.

[2] Funk, S. (2006). Try this at home.
Blog post: <http://sifter.org/~simon/journal/20061211.html>. December 11.

[3] Y. Hu, Y. Koren, and C. Volinsky. Collaborative filtering for implicit feedback datasets. In *IEEE International Conference on Data Mining (ICDM 2008)*, pages 263–272, 2008.

[4] Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2000). Application of Dimensionality Reduction in Recommender System -- A Case Study

[5] Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2002). Incremental singular value decomposition algorithms for highly scalable recommender systems. In *Fifth International Conference on Computer and Information Science* (pp. 27-28).

[6] Hofmann, T. (2004) Latent Semantic Models for Collaborative Filtering. *ACM Transactions on Information Systems (TOIS)*, 22, 89-115

[7] Rong Pan, Yunhong Zhou, Bin Cao, Nathan N. Liu, Rajan Lukose, Martin Scholz, Qiang Yang HP Laboratories, (2008) One-Class Collaborative Filtering

[8] D. Goldberg, D. Nichols, B. M. Oki and D. Terry, “Using Collaborative Filtering to Weave an Information Tapestry”, *Communications of the ACM* 35 (1992).

[9] ALS Implicit Collaborative Filtering, <https://medium.com/radon-dev/als-implicit-collaborative-filtering-5ed653ba39fe>

[10] Ben Frederickson, Finding Similar Music using Matrix Factorization
<http://www.benfrederickson.com/matrix-factorization/>

[11] Nana Boateng, Hyperparameter Tuning The Alternating Least-Squares Algorithm for A Recommender System. <https://restanalytics.com/2019-02-27-Hyperparameter-Tuning-Alternating-Least-Squares-Recommender-System/>

[12] Zhou – Wilkinson, Scale Parallel Colaborative Filtering for the Netflix Price.

- [13] R. Bell, Y. Koren, and C. Volinsky. The bellkor solution to the netflix prize. Netflix Prize Progress Award, October 2007.
[http://www.netflixprize.com/assets/ProgressPrize2007 KorBell.pdf](http://www.netflixprize.com/assets/ProgressPrize2007%20KorBell.pdf).
- [14] A Gentle Introduction to Recommender Systems with Implicit Feedback
<https://jessesw.com/Rec-System/>
- [15] Documentación librería RecoMetrics <https://recometrics.readthedocs.io/en/latest/>
- [16] Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, Introduction to Information Retrieval, Cambridge University Press, cap 8. 2008