UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

IMPLEMENTACIÓN DE MOTOR DE DISEÑO DE OFERTAS PERSONALIZADAS PARA EMPRESA DE VENTA DIRECTA

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL EN COMPUTACIÓN

RODOLFO ANTONIO ÁVILA PÉREZ

PROFESOR GUÍA:
NANCY HITSCHFELD KAHLER

MIEMBROS DE LA COMISIÓN:
CLAUDIO GUTIÉRREZ GALLARDO
MARISA ERNST ELIZALDE

SANTIAGO DE CHILE
2019

RESUMEN EJECUTIVO

La presente tesis se centró en la implementación de una herramienta que permita diseñar y generar ofertas personalizadas a los representantes de una empresa de venta directa. Se utilizó como base un algoritmo de recomendación empleado en empresas de retail. Se adaptaron los diferentes módulos según correspondiese.

Dentro de los desafíos iniciales estaba el comprender la diferencia entre el mundo de las empresas de venta directa y las empresas de retail. Entender cómo se lleva a cabo el proceso de compra, detectar oportunidades y restricciones propias del negocio, así como la oportunidad de incluir nuevas tecnologías, por ejemplo *Machine Learning*, para detectar patrones o comportamientos que no son detectables por un análisis regular de una persona y así mejorar la eficacia de la herramienta. Las ofertas generadas son expuestas al cliente en el portal de la empresa donde el acceso es personalizado.

En el escenario inicial ya existía un algoritmo recomendador masivo de ofertas, por lo que uno de los objetivos de éxito es superar la conversión y el valor de las ofertas ofrecidas.

La herramienta consta de 6 módulos principales que componen su esqueleto base: perfilamiento, propensión de compra, modelo de descuento, modelo de volumen, modelo de bundle y asignación. Cada uno de esto módulos opera de forma independiente, sin embargo, deben seguir un determinado orden de ejecución para el correcto funcionamiento.

Se realiza una prueba preeliminar sobre un grupo piloto comparado contra un grupo control. Se revisan los resultados y se determinan oportunidades de mejorara para el motor. Se implementan cambios a ciertas partes de algoritmo antes de una última prueba.

Al finalizar los cambios se obtuvo que en cuanto a tiempos de ejecución, para un país como Perú con alrededor de 150.000 consultoras el tiempo es de 3 horas, generando más de 3 millones de ofertas que están asociadas a cada consultora según sus características. Por el lado de resultados comerciales generales, en la prueba final se logró aumentar el ratio de conversión de las ofertas en un 15.3 % con respecto al antiguo algoritmo recomendador llegando a un 24.1 %, además se aumentó la conversión del canal de venta en un 4.7 %. Mirado desde los productos, el PPU aumentó en 1.73 soles($0,52 USD), pero el PUP disminuyó en 0.35 unidades.

Como beneficio secundario se logró bajar el tiempo generación de ofertas desde su diseño hasta la exposición en 7 días a pesar de aumentar considerablemente el volumen de ofertas generadas. Así se detectan posibles oportunidades de mejoras para futuras versiones de la herramienta.

”Excelente maestro es aquel que, enseñando poco, hace nacer en el alumno un deseo grande de aprender.”

Agradecimientos

Son muchas las personas que se me vienen a la mente a la hora de agradecer, pero hay una que toma el primero lugar. Yerko, mi hermano, es el principal artífice de este momento, su siempre incondicional apoyo me ha llevado a sortear todas las dificultades que he tenido.

Debo agradecer a mi comisión, Profesora Nancy por la eterna paciencia conmigo y mis entregas a última hora y por guiarme a completar esta última etapa universitaria. Profesora Marisa, muchas gracias por sus comentarios y correcciones, sobre todo las ortográficas, el borrador fue un desastre en ese sentido. Profesor Claudio, su visión sobre el planteamiento del desarrollo me ayudó a complementar el documento inicial. Así mismo, Agradecer a Angélica y Sandra quienes han sido un 7 a lo largo de toda la carrera y son un aporte grandísimo a todos los alumnos del DCC.

Agradecer a mis padres y mi familia, a mis amigos quienes aprovecharon cada oportunidad que tuvieron para recordarme terminar la memoria. Daniela, Felipe, Nicolás, Francisco, entre muchos otros, muchas gracias por su apoyo.

Finalmente agradecer a la empresa y mis compañeros de trabajo. Gracias a todos ellos se dio la oportunidad de realizar este bonito proyecto que finalmente terminó determinando la profesión a la que me dedicaría, al menos, por ahora.

Agradecer toda la gente maravillosa que conocí en Perú y quienes me ayudaron con consejos o me dieron un empujón cuando perdía el rumbo.

Tabla de Contenido

Capítulo 1

Introducción

Se entiende por *venta directa* la comercialización fuera de un establecimiento comercial de bienes y servicios directamente al consumidor, mediante la demostración u ofrecimiento personalizado por parte de un representante de la empresa vendedora. Este último aspecto distingue a la venta directa de las denominadas *ventas a distancia*, en las que no llega a existir un contacto personal entre la empresa vendedora y el comprador. Ejemplo de este tipo de empresas son Avon o Herbalife.

En este proceso de venta se identifican cuatro fases principales:

1. Contacto: Consiste en el contacto presencial con el potencial cliente.

2. Presentación: Consiste en mostrar y enseñar las características del producto o servicio y los beneficios. Muchas veces se utilizan muestras o revistas para enseñar los productos. Esta es la fase que más tiempo toma.

3. Cierre: Es el momento donde se completa la venta.

4. Seguimiento: Dada la inversión de tiempo realizada, se genera un seguimiento del cliente, se haya concretado o no la venta.

En este mercado, las ganancias de los representantes de las empresas de venta directa viene dado por una comisión con respecto a la venta realizada, ya sea por cantidad o por tipo de producto vendido. Ante esto, las empresas de venta directa deben incentivar a sus representantes a vender más si quieren aumentar sus ganancias. Es en este escenario que nace la necesidad de contar con una herramienta que permita generar promociones atractivas para los representantes de la empresa y que a su vez, éstos puedan ofrecer a sus clientes y obtener buenos resultados de venta. La apuesta es, entonces, ir hacia la personalización de las ofertas, detectando oportunidades para ampliar la canasta de compra de un representante tanto en amplitud(mix de productos), como en profundidad(frecuencia y cantidad).

Las ofertas generadas serán presentadas a los representantes a través del sitio web de la empresa, portal donde ingresan a realizar los pedidos de productos. Dado que se tiene un espacio limitado de la pantalla para presentar las ofertas, existen restricciones relacionada a la cantidad de productos que se pueden ofrecer, los formatos de oferta posibles y reglas asociadas al orden en que las ofertas

serán presentadas a los representantes.

Se fija, entonces, como objetivo principal de esta memoria el desarrollo de una herramienta que permita generar promociones personalizadas a los representantes de las empresa de venta directa, que cumpla con las necesidades y restricciones propias del negocio y supere en indicadores comerciales a la solución preexistente.

Para lograr el objetivo principal, se definen 3 objetivos específicos:

1. Ajustar algoritmos del mundo del retail al mercado de una empresa de venta directa.

2. Implementar restricciones propias del mundo de la venta directa.

3. Generar un proceso integrado que incluya los algoritmos ajustados, las restricciones de negocio y genere las ofertas personalizadas con el fin de obtener mejores resultados comerciales que el modelo existente.

Para la implementación de este Motor de Diseño de Ofertas Personalizadas se ajustarán algoritmos aplicados y probados en el mundo del retail para la generación de ofertas, y se llevarán al contexto de una empresa de venta directa. Se aplicarán conceptos como *clusterización, cliente espejo, canasta de compra, propensión de compra, machine learning*, entre otros.

Las reglas de negocio se irán aplicando a medida que se desarrolle el motor, siempre teniendo en mente el afectar lo menos posible la personalización de las ofertas y la calidad de éstas.

Se utilizará una herramienta de ETL[1] para *orquestar* y llevar el flujo de las diferentes partes que compondrán el motor. Así mismo, se mezclarán disferentes tecnologías con el fin de lograr los objetivos comerciales propuestos.

Al finalizar se pondrán a prueba los resultados obtenidos mediante un sistema de diseño de experimentos, con grupo control y prueba, contra el sistema de recomendación de ofertas existente que no es personalizado en la generación y diseño de ofertas, mas solo lo es en la asignación.

La metodogía que se seguirá está enfocada en generar una herramienta modular, donde cada parte constituya un proceso único y encapsulado, para así al realizar cambios no afectar las demás partes. El algoritmo de retail que se adaptará consiste en 3 partes principales, siendo éstas:

1. **Quién**: Conocer y agrupar clientes de similares características bajo alguna etiqueta común.

2. **Qué**: Determinar los productos que forman parte de la canasta de compra del cliente y su segmento.

3. **Cómo**: Define la táctica con que el producto será presentado al cliente.

El capítulo 2 presenta la situación actual de la empresa, discute acerca de diferencias entre el mundo del retail y la venta directa. Además explica el algoritmo contra el que competirá la solución propuesta y define indicadores comerciales como métricas de comparación. Finalmente expone las herramientas tanto de hardware como de software que serán utilizadas durante esta memoria y

[1] Extract-Transform-Load

expone algunos de los algoritmos de machine learning utilizados.

A continuación, el capítulo 3 explica el diseño e implementación de cada parte que componen el Motor de Diseño de Ofertas. Primero se muestra una breve descripción de las 6 etapas de la solución, luego se muestra un diagrama de flujo de la información através de los diferentes módulos. finalmente presenta la información que se dispone para el desarrollo de la memoria. Más adelante, el primer módulo, perfilamiento, muestra diferentes combinaciones de segmentación utilizando el algoritmo Kmeans y la relación de los resultados con el conocimiento del negocio. Se definen 9 perfiles de consultoras enfocados en su canasta de compra.

Siguiendo en el mismo capítulo, el Modelo de Propensión de Compra expone la problemática de falta de información relevante para la precisión de este paso. Se plantea un camino alternativo de modelamiento, se comparan 4 diferentes modelos de machine learnig y se termina mostrando una estrategia para llevar el resultado del modelo seleccionado al nivel de cliente-producto. Se continua explicando el cálculo del descuento histórico aplicado a los productos por cliente y se expresa una fórmula para asignar un descuento a los productos de las ofertas generadas. Las dos secciones siguientes presentan los dos algoritmos que determinan la táctica para generar ofertas. Primero el Modelo de Volumen muestra, de manera similar al Modelo de Propensión de Compra, una forma de modelar a un nivel de agrupación no óptimo y luego un camino para llevar la predicción al nivel deseado. Se comparan 4 modelos y se decide por la utilizacón de la mezcla de 2 de ellos. El otro formato de oferta corresponde a packs de productos. Con la ayuda de un algoritmo perteneciente al Market Basquet Analysis se generan reglas de asociación que determinan conjuntos de productos probables dentro de un perfil de cliente. Se finaliza el capítulo consolidando la información generada en las secciones anteriores y mostrando un indicador conjunto para priorizar las diferentes ofertas generadas. Además se plantea una estretegia de completación de ofertas en caso de no cumplir con la cantidad mínima exigida.

Se realiza una prueba piloto sobre un pequeño universo para ver el impacto de la solución generada, además, el capitulo 5 describe oportunidades de mejora dado el experimento. Se implementan algunos cambios y se repite la prueba sobre un universo mayor. Los resultados muestran ser exitosos en casi todas las dimensiones evaluadas.

El capítulo 4 presenta primero el proceso realizado para llevar a cabo la automatización del motor recomendador, describiendo cómo se afronta cada etapa, así mismo presenta la interfaz de usuario. Luego, dada la naturaleza de las técnica ocupadas, se exponen los mecanismos para realizar recalibración a los modelos de machine learning.

Finalmente, en el capítulo 6 se presentan las principales conclusiones, destacanto el cumplimiento de los objetivos cuando corresponda y explicando posibles cambios y ajustes para conseguir aquellos que no lograron ser cubiertos con la herramienta generada. Para cerrar se plantean posibles mejoras o futuros módulos nuevos que pueden ser de apoyo para el negocio y su operatividad.

Capítulo 2

Antecedentes

Las empresas de retail, o empresas de venta al detalle, pertenecen a un sector económico especializado en la comercialización masiva de productos o servicios uniformes a grandes cantidades de clientes. Interactúan directamente con el consumidor final a diferencia de las empresas de venta directa.

Comúnmente se asocia retail a los supermercados o tiendas por departamentos, sin embargo, también incluyen a ferreterías, farmacias y librerías entre muchas otras empresas. La complejidad del mundo del retail viene dada por la gran variedad de artículos y diferentes tipos de artículos a su vez. Así mismo, a nivel operacional se genera una gran cantidad de información transaccional que puede llegar a ser complicada de utilizar sin un conocimiento experto.

Hasta este punto, el mundo del retail se asemeja bastante al de una empresa de venta directa, diferenciándose solo en la forma en que interactúan con el consumidor final. Mientras que en el retail por lo general se utiliza el canal de venta compuesto por tiendas o locales comerciales, las empresas de venta directa requieren de muchos representantes para lograr ofrecer sus productos a una gran cantidad de público. De ahora en adelante, al *cliente* o representante de la empresa se le llamará *consultora*, dado que es el nombre que reciben en el rubro, por ser en su totalidad mujeres de diferente rango etario.

La empresa de venta directa en cuestión cuenta con presencia en varios paises de Latinoamérica, sin embargo el desarrollo de esta solución se llevará a cabo utilizando la información asociada a un solo país: Perú. A nivel operativo, un año calendario está compuesto por 18 campañas, siendo ésta la unidad de medida de tiempo que se utilizará en adelante.

La empresa ya cuenta con un algoritmo llamado ARP(Algoritmo de Recomendación Personalizado). Las ofertas que recibe el ARP son generadas a criterio experto por un conjunto de personal dedicado a esta tarea. La cantidad de ofertas generadas son del orden de 500 por campaña. Las ofertas son el dato de entrada para el ARP, que se encarga de asignar a cada consultora las mejores 8 ofertas según criterios de RFM (Recencia[1], Frecuencia, Monto)[10].

Si bien es cierto que dentro de las 18 campañas existen algunas enfocadas a ciertas festividades,

[1] cantidad de días transcurridos desde la última compra realizada.

por lo que deben ser tratadas con ofertas especiales según corresponda, en su mayoría las campañas son neutras, por lo que una mejora en los tiempos de generación de ofertas afectaría positivamente la operatividad de generación de las mismas.

El desarrollo del Motor de Diseño de Ofertas (MDO) tiene como objetivo comercial superar en KPIs[2] comerciales claves, como son el PPU (Precio Promedio por Unidad), PUP (Promedio Unidades por Pedido) o Ratio de Conversión, al modelo existente actualmente.

Se utiliza un algoritmo base centrado, precisamente, en RFM para la construcción del MDO. Está probado que a pesar de ser un modelo relativamente simple, el enfoque RFM resulta eficiente a la hora de categorizar a los clientes, así como también sus resultados son de fácil interpretación.[9]. Analizar la canasta de compra de las consultoras permitirá encontrar grupos de ellas con compras similares y propiedades parecidas, entonces se utilizará el principio de cliente espejo para generar una oportunidad cuando existan productos que son comprados por algunas consultoras y no por otras, siendo éstas del mismo segmento.

2.1. Apartado Técnico

Con respecto al hardware y software a utilizar, se dispone de lo siguiente:

Hardware:

- Servidor Ubuntu 16.04, 32 GB RAM, 4 Procesadores.
- Notebook HP Envy 14 Core i5, Windows 7, 8 GB RAM.

Software:

- R version 3.3.2
- Rstudio version 1.1
- Motor SQL Infobright.
- Pentaho Kettle versión 8.1

Por el lado de las herramientas de machine learning y minería de datos, se utilizarán los siguientes métodos:

- **Kmeans**: Tiene como objetivo la partición de un conjunto de n observaciones en k grupos en el que cada observación pertenece al grupo cuyo valor medio es más cercano.
- **Árbol de Decisión**: A partir de un conjunto de datos se fabrican diagramas de construcciones lógicas, muy similares a los sistemas de predicción basados en reglas, que sirven para representar y categorizar una serie de condiciones que ocurren de forma sucesiva, para llegar a la resolución de un problema.
- **Árbol de Regresión**: Son Árboles de Decisión donde la variable de respuesta puede tomar valores reales.

[2]Indicadores principales de desempeño.

- **Regresión Lineal**: Es un modelo matemático usado para aproximar la relación de dependencia entre una variable dependiente, las variables independientes (predictores), agregando un término aleatorio.
- **Regresión Logística**: Es un tipo de análisis de regresión utilizado para predecir el resultado de una variable categórica en función de las variables independientes o predictoras.
- **Regresión de Poisson**: Es un tipo de modelo lineal generalizado en el que la variable de respuesta tiene una distribución de Poisson y el logaritmo de su valor esperado puede ser modelado por una combinación lineal de parámetros desconocidos.
- **Red Neuronal Perceptrón Multicapa**: Es una red neuronal artificial formada por múltiples capas, de tal manera que tiene capacidad para resolver problemas que no son linealmente separables.
- **Tobit**:es un modelo estadístico propuesto para describir la relación entre una variable dependiente no negativa y una variable independiente (o vector de variables).
- **Distancia de Mahalanobis**: Es una forma de determinar la similitud entre dos variables aleatorias multidimensionales. Se diferencia de la distancia euclídeana en que tiene en cuenta la correlación entre las variables aleatorias.

Capítulo 3

Diseño e implementación del MDO

A lo largo del tiempo se han desarrollado diferentes algoritmos en el mundo del retail para generar ofertas, ya sean personalizadas o masivas. En los últimos años, con la masificación de la utilización de herramientas de Minería de Datos y Machine Learning en diferentes industrias, se han ajustado muchos de estos algoritmos logrando resultados que impactan fuertemente el mercado y la forma en cómo son percibidas las ofertas por parte de los clientes las empresas que utilizan estas técnicas.

La segmentación de clientes que antes se realizaba según juicio experto y reglas de negocio, hoy es realizada mediante algoritmos estadísticos que permiten identificar grupos de clientes con similares comportamientos con respecto a las variables de interés para el negocio.

El problema de saber qué producto es más afín a un cliente o tipo de clientes era abordado mediante la captura del conocimiento obtenido por los vendedores de tienda. En la actualidad, Modelos de Propensión de Compra son generados utilizando técnicas de Machine Learning para obtener la probabilidad que un cliente compre un producto dadas ciertas condiciones.

Determinar la cantidad óptima de unidades que debo ofrecer a un cliente, dadas las restricciones tecnológicas, se traducía en ofrecer el promedio de compra en un plazo de tiempo determinado, o mediante series de tiempo sobre conjunto de clientes. Algoritmos tanto estadísticos como de Machine Learning permiten estimar, cliente a cliente, las unidades que estaría dispuesto a comprar.

Dado todo lo anterior se plantea la construcción del motor de diseño de ofertas en 6 etapas principales, a las cuales se les agrega donde correspondan las restricciones del negocio dadas (éstas restricciones serán explicadas cuando sean aplicadas con el objetivo de dar contexto). Las etapas son:

- **Perfilamiento**: en esta etapa se deben determinar conjuntos de cliente que cumplan similares características relevantes para el problema a resolver. Esto servire para *suavizar* el efecto de los modelos aplicados más tarde, así como para mantener cierto grado de agrupación del volumen total de cliente para realizar análisis. Se utilizan algoritmos de segmentación estadística.
- **Determinar Productos**: para cada cliente se determina un mix de productos que son más afi-

nes para él. Estos son generados mediante un Modelo de Propensión de Compra. Se utilizan modelos de Machine Learning del tipo clasificadores.

- **Determinar Formato**: además de tener los productos afines a un cliente, es necesario determinar con qué estrategia se le mostrarán. Para esto se cuenta con tres formatos de oferta posibles. Individual (una unidad), Volumen (más de una unidad del mismo producto), Bundle (set de distintos productos en una misma oferta).
- **Determinar Unidades**: es necesario saber cuánto ofrecer a cada cliente, para esto se utilizarán modelos que permitan estimar una cantidad óptima a ofrecer por producto.
- **Determinar Descuento**: no basta solo con saber qué y cuánto ofrecer, además se necesita saber a cuánto ofrecerlo. Para esto se buscará obtener un precio óptimo para cada oferta dependiendo de los productos, unidades y formato que la componen.
- **Priorizar**: ya con las diferentes ofertas (productos, unidades, formato, descuento) determinadas para cada cliente, es necesario ordenarlas según un criterio que logre incentivar la compra de éstas.

Las primeras dos etapas resultan cruciales a la hora de proyectar posibles resultados del desarrollo, dado que es allí donde se concentra gran parte del trabajo de conversión del mundo del retail hacia el de venta directa.

Figura 3.1: Diagrama de módulos del MDO.

La Figura 3.1 muestra un díagrama de los diferentes módulos que componen el MDO. Si bien todas las etapas contienen desarrollo en SQL, las etapas que consideren modelamiento incluyen la utilización del software R para este fin. La integración entra ambas herramientas se logra gracias a la utilización del paquete *RMySQL*[5] de R, que permite extraer datos desde el motor de base de datos y, también, crear tablas e insertar datos. Para llevar el flujo del proceso se utiliza la herramienta Kettle, de Pentaho[7], que permite la integración de diferentes tecnologías, en particular, la utilización de código SQL y R(mediante la interfaz de código shell).

Cuando se habla de la utilización del leguanje SQL se refiere a la generación de múltiples tablas de datos que van generando, ya sea, cálculos intermedios, bases analíticas para ser consumidas en scripts de R o bien resultados finales. Dado el gran volumen de datos que se procesa, el motor de

base da datos Infobright ofrece una gran ventaja al ser del tipo columnar, lo que optimiza de forma nativa la manera de realizar cruces de información y el almacenamiento.

La información llega desde la empresa al repositorio de información en Infobright por un proceso ETL encargado de ir a buscar la información en un servidor SFTP periódicamente. La información no es transformada y es almacenada en el mismo formato en que es recibida, esto porque se planea migrar la solución generada al ambiente productivo de la empresa luego que se hayan realizado todas las pruebas y ajustes necesarios para el correcto funcionamiento de la herramienta.

Las tablas con las que se cuenta para el desarrollo del motor fueron llenadas en un principio con 2 años de historia y a medida que se avanzó en el desarrollo del proyecto se agregó la información que se fue generando campaña tras campaña.

Listado de tablas disponibles para el desarrollo:

- Tabla transaccional de ventas: contiene el detalles de los pedidos realizados por las consultoras campaña a campaña.
- Tabla transaccional de estados: detalla el estado de las consultoras campaña a campaña. Variables como segmento de valor, región y otras pueden cambiar de una campaña a otra.
- Tabla transaccional de ofertas: aquí se muestran los productos ofertados en una campaña, el descuento según el tipo de canal, el precio sin descuento, precio de catálogo, entre otros. Se tiene la información de dos campañas futuras.
- Maestro de consultoras: contiene la información relacionada a una consultora y que no varía, en su mayoría, en el tiempo. Por ejemplo, el código del país al que pertenece o la fecha de ingreso a la campañía pertencen a esta tabla.
- Maestro de productos: tiene la información relacionada a los productos, la marca a la que pertenecen, la categoría y el tipo.
- Maestro de tipo de oferta: la información de los diferentes tipos de ofertas existentes están en esta tabla. Permite determinar qué poductos fueron *regalados* como parte de campañas promocionales.

Con esta información en mente se enfrenta, en primer lugar, el desafío de entender al *"cliente"* como un conjunto de clientes finales, es decir, su comportamiento de compra no refleja necesariamente sus intereses.

3.1. Perfilamiento

Como base para la construcción de las ofertas y pensando en la personalización, se realiza un perfilamiento de las consultoras. Este es un paso clave en el mundo del retail, dado que permite identificar grupos de clientes con similares características y por ende, encontrar *grupos potenciales* o *clientes espejos*, donde lo que se busca es ampliar la cantidad de productos que el cliente compra, dado que existe otro cliente de similares características (mismo perfil) que compra otros productos o más unidades de alguno.

Regularmente, en retail, las variables más usadas son sociodemográficas y relacionadas con el

comportamiento de compra (RFM) del cliente. Existía previamente dentro de los datos asociados a los clientes un perfilamiento de este tipo, que divide a los clientes en siete diferentes grupos. A pesar de lo anterior, para poder ajustar los algoritmos a este mundo de venta directa, es necesario cambiar la forma de perfilar para poder captar la variable principal que rige al motor recomendador: la canasta de compra.

Se llama canasta de compra al mix de productos que un cliente compra o consume. Esta *variable* en el fondo agrupa un conjunto de características propias de los productos comprados por el cliente, algunas características son formato, tamaño, marca, olor, sabor, color, etc.

Dado el mix diferente de productos que existe en el catálogo, variables físicas como olor, sabor o color quedan descartadas. Por otro lado, es necesario, por regla de negocio, que la cantidad de perfiles no exceda de los 10, dado que este perfilamiento podría ser un input para otros proyectos comerciales paralelos.

El algoritmo a utilizar para la segmentación es k-Means[8], algoritmo que aplica sobre espacios de *n-dimensiones*, determinando una cantidad dada de puntos centrales o centroides, para luego asignar a cada elemento de la muestra un cluster según el centroide más cercano. Los centroides son tales que minimizan la suma de los cuadrados dentro de cada grupo o cluster.

Las características representativas de la canasta de compra de los clientes se enfocan en las dos principales, que son transversales a todo el catálogo de productos, estas son marca y categoría. Dentro del catálogo existen 3 marcas y 5 diferentes categorías. Dentro de la jerarquía de productos está presente también un subnivel a la categoría, llamado *tipo*, sin embargo, según la categoría, pueden haber más de 50 tipos distintos, por lo que no se llega a ese nivel de granularidad. La regla de negocio de no superar los diez perfiles apoya la decisión de incluir variables que agrupen a mayor nivel los productos.

De las marcas podemos decir que son suplementarias entre sí y, dentro del público objetivo, representan diferentes niveles económicos y sociales. Las categorías son transversales a las tres marcas y son complementarias entre sí. Según los datos, se puede apreciar que en el país donde se desarrolla el motor recomendador, una marca prepondera por sobre las otras (alrededor de un 55 % del total de la venta, contra un 35 % de otra y un 10 % de la última). Las categorías por su parte se encuentran concentradas en 3 de las 5, con un total de 97 % de la venta total en monto de dinero, los segmentos encontrados utilizando el algoritmo k-Means reflejan esta realidad. El dataset que se utiliza considera la historia de las últimas 6 campañas dado que luego de un periodo de 6 campañas sin pasar pedido una consultora pasa a ser inactiva.

Con el conocimiento del negocio antes expuesto, se lleva a cabo un perfilamiento en dos niveles: primero aplicar el algoritmo k-means sobre el grupo de clientes discriminando sobre el *share de venta*[1] en unidades con respecto a las diferentes marcas; y en un segundo paso aplicar, sobre cada uno de los conjuntos antes encontrados, nuevamente el algoritmo k-means, pero ahora discriminando con respecto al *share de venta* en unidades de las categorías. Share de venta refiere al porcentaje de compras, en unidades, que representa una cierta marca/categoría con respecto al total comprado por la consultora.

Como contraste, se probó aplicar el mismo algoritmo aplicado sobre las variables juntas, es de-

[1] Distribución en porcentaje.

cir, el share de venta por marca y categoría, como una combinación, teniendo entonces 15 variables sobre las cuales segmentar.

Según se muestra en la tabla 3.1, se probó una segmentación en cinco grupos, en los cuales se marca con negrita los grupos más representativos de cada conjunto. La última columna representa la cantidad de consultoras que son asignadas a cada cluster. En el tercer cluster no se ve una clara tendencia por alguna de las combinaciones de marca-categoría, por lo que se categoriza como una mezcla de las dos marcas dominantes sin especificar una categoría.

En la Tabla 3.2 se muestra una segmentación de marca-categoría con una separación en seis grupos. En este nuevo escenario se vuelve aún más complejo dilucidar una preferencia marcada en cada unos de los grupos encontrados. Siguen existiendo conjuntos enfocados en solo una marca-categoría, así como otros donde la tendencia está sobre las marcas y no muy diferenciado por categoría.

De acuerdo a lo encontrado, se procede a cambiar la forma de segmentar, dejando de lado marca-categoría, por un enfoque de marca y luego por categoría. Se hace de esta forma, y no a la inversa, dado que según se vio en las segmentaciones anteriores, existen varios casos donde la preferencia es por marca más que por una categoría en específico.

Lo primero es realizar el *gráfico de codo* para intentar estimar el número óptimo de clusters para la segmentación por marca. En este gráfico lo que se busca es encontrar un punto de inflexión en la curva que representa un número ideal de clusters o grupos.

clus	lb_cp	lb_f	lb_maq	lb_tc	lb_tf	es_cp	es_f	es_maq	es_tc	es_tf	cz_cp	cz_f	cz_maq	cz_tc	cz_tf	k
1	1%	1%	1%	0%	1%	**43%**	16%	11%	4%	2%	3%	8%	8%	0%	0%	39.381
2	1%	2%	2%	0%	1%	18%	**41%**	11%	3%	2%	3%	8%	7%	0%	0%	21.184
3	1%	3%	2%	1%	1%	**22%**	**17%**	**13%**	4%	2%	4%	**16%**	**13%**	0%	0%	55.863
4	1%	2%	4%	1%	2%	15%	14%	**33%**	3%	2%	3%	7%	12%	0%	0%	31.749
5	1%	2%	2%	0%	1%	13%	11%	15%	3%	2%	4%	11%	**33%**	0%	1%	26.465

Tabla 3.1: Segmentación Marca-Categoría, 5 clusters

clus	lb_cp	lb_f	lb_maq	lb_tc	lb_tf	es_cp	es_f	es_maq	es_tc	es_tf	cz_cp	cz_f	cz_maq	cz_tc	cz_tf	k
1	1%	3%	3%	1%	1%	**15%**	**16%**	**11%**	3%	2%	4%	**27%**	12%	0%	0%	19.548
2	1%	1%	1%	0%	1%	18%	**42%**	11%	3%	2%	3%	8%	7%	0%	0%	25.740
3	1%	2%	2%	0%	1%	18%	**42%**	11%	3%	2%	3%	8%	7%	0%	0%	19.328
4	1%	2%	3%	0%	1%	**27%**	**18%**	**15%**	4%	2%	4%	**10%**	**13%**	0%	0%	56.248
5	2%	2%	4%	1%	2%	13%	14%	**34%**	3%	2%	3%	7%	12%	0%	0%	27.898
6	1%	2%	3%	0%	1%	13%	12%	15%	3%	2%	4%	10%	**33%**	0%	1%	25.880

Tabla 3.2: Segmentación Marca-Categoría, 6 clusters

cluster	Share LB	Share ES	Share CZ	k
1	**28 %**	**49 %**	**23 %**	16.484
2	4 %	**50 %**	**46 %**	66.709
3	4 %	**73 %**	23 %	91.510

Tabla 3.3: Distribución de venta según Segmentación Marca

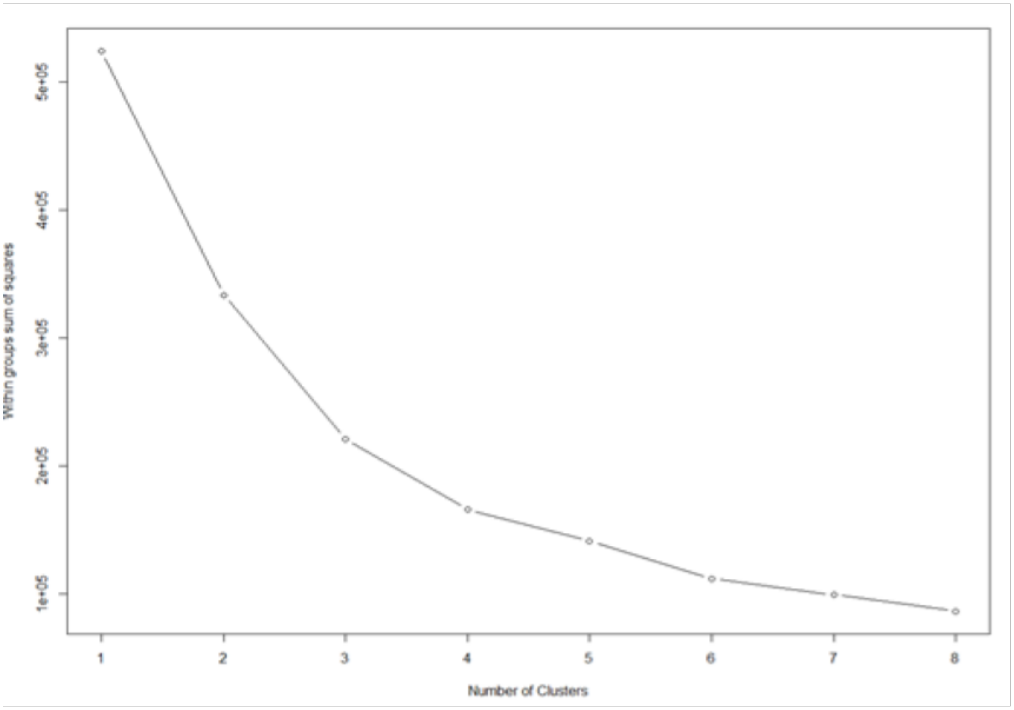

Figura 3.2: Gráfico de Codo de Segmentación por marca

Como se puede ver en la Figura 3.2, no existe un punto donde sea completamente evidente que se produce la inflexión buscada, sin embargo, la cohesión de los diferentes conjuntos parece disminuir de forma lineal a partir de punto 3, por lo que este es el número de cluster por los que se comienza la segmentación de marca.

En la Tabla 3.3 se destacan las marcas dominantes de cada conjunto. Se ve que el grupo mayor, el tercero, con más de la mitad del total de consultoras, tiene una notoria preferencia por la marca ES, mientras que el segundo presenta una distribución bastante pareja entre las marcas ES y CZ. El conjunto uno por su parte, que representa cerca del 10 % del total de consultoras, en su distribución se nota que existe una preferencia significativa hacia la marca LB. Como conocimiento del negocio, se valida que existe alrededor de un 10 % del total de consultoras que consume la marca LB, por lo que el cluster uno viene a representar en cierta medida a ese conjunto de consultoras.

La Figura 3.3 muestra en dos de las dimensiones como se distribuyen las consultoras en los diferentes clusters asignados.

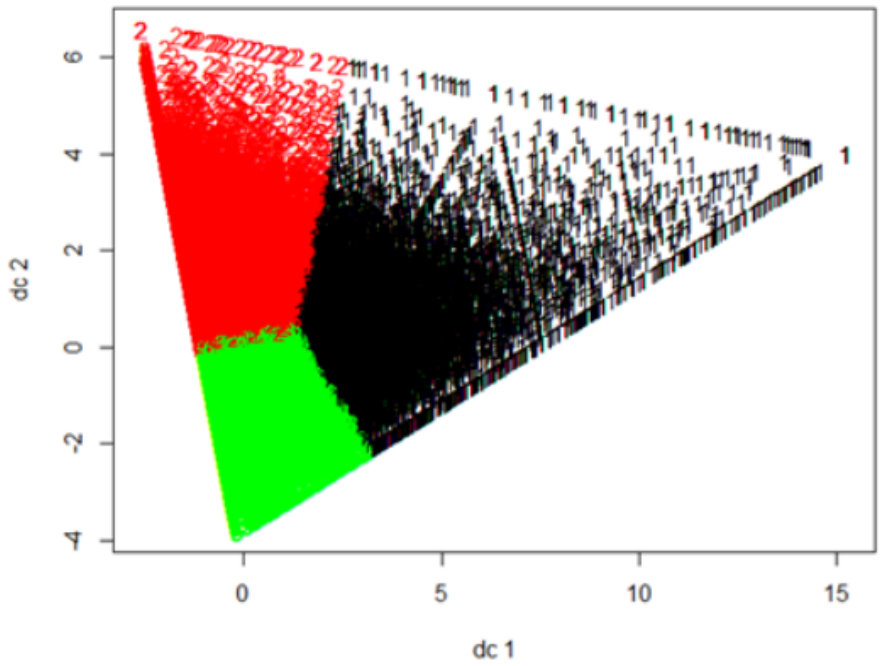

Figura 3.3: Representación Gráfica de los Clusters de Segmento Marca

cluster	Share LB	Share ES	Share CZ	k	Nombre
1	**28 %**	**49 %**	**23 %**	16.484	MULTI
2	4 %	**50 %**	**46 %**	66.709	ESCZ
3	4 %	**73 %**	23 %	91.510	ES

Tabla 3.4: Nombre de Segmentos Marca.

Luego, se validan los resultados con la contraparte comercial de la empresa y se procede con la siguiente etapa del perfilamiento. Cada uno de los segmentos de marca encontrados son divididos según una nueva aplicación del algoritmo K-Means. Primero, para facilitar la lectura, se nombran los 3 segmentos encontrados (Tabla 3.4).

Como ya se mencionó, de las 5 categorías en consideración sólo 3 concentran casi la totalidad de las compras. Al igual que en el caso anterior se procede a segmentar con respecto al share de venta en unidades de las diferentes categorías usando la misma metodología pero sobre cada uno de los clusters antes encontrados.

En la Figura 3.4 se muestra el gráfico de codo correspondiente al segmento Multi. En él no se puede apreciar claramente un punto de inflexión, por lo que se prueban diferentes combinaciones entre 2 y 6 clusters. Siempre el objetivo es encontrar segmentos que se diferencian notoriamente en

cluster	CP	FR	MQ	TC	TF	k	Nombre
1	16 %	**49 %**	23 %	5 %	7 %	3.776	MULTI-FR
2	**39 %**	22 %	24 %	6 %	9 %	5.695	MULTI-CP
3	15 %	19 %	**51 %**	5 %	10 %	7.013	MULTI-MQ

Tabla 3.5: Segmentación Marca-Categoría Perfil MULTI.

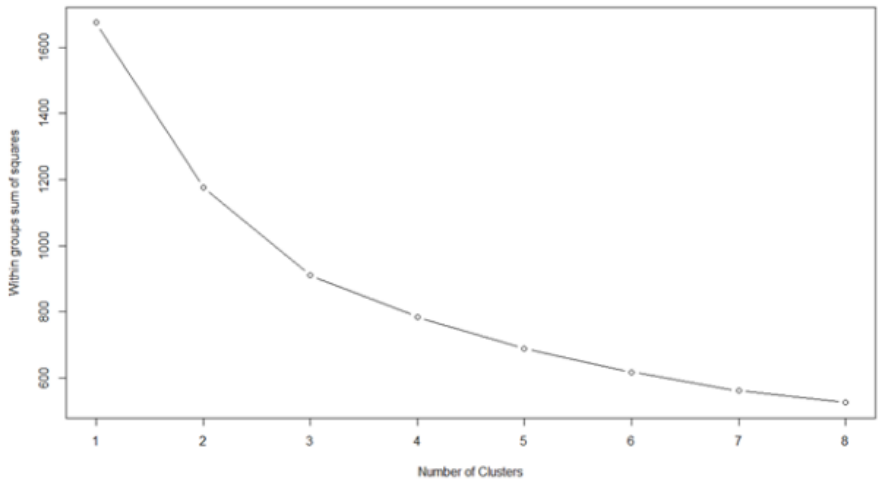

Figura 3.4: Gráfico de Codo Segmento Multi en subsegmentos

su interés por alguna o algunas categorías. El resultado que mostró más diferenciación correspondió a la elección de 3 cluster como se puede apreciar en la Tabla 3.5.

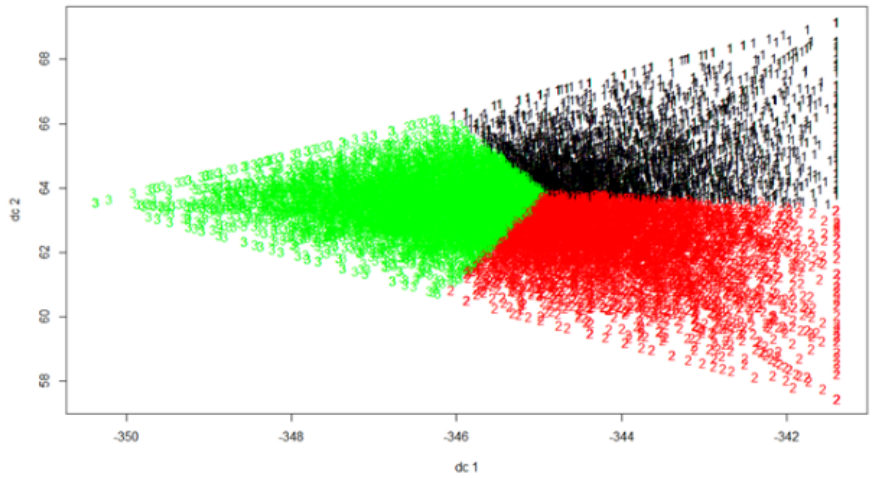

Figura 3.5: Distribución Clusters Perfil MULTI.

En la Figura 3.5 se muestra la distribución en dos dimensiones de los clusters seleccionados. Mismo procedimiento se realiza para los otros clusters: ESCZ y ES; los resultados se pueden observar en la Figura 3.6 y Tabla 3.6 para el caso de ESCZ, y Figura 3.8 y Tabla 3.7 para ES.

En resumen, se generan 3 clusters por marca y luego cada uno de ellos fue segmentado de acuerdo a las categorías y se determinaron 3 divisiones por cada grupo, completando un total de 9 segmentos o perfiles, estos se muestran en la Tabla 3.8 con sus respectivos nombres y total de consultoras que integran el perfil.

cluster	CP	FR	MQ	TC	TF	k	Nombre
1	16%	23%	**54 %**	4%	4%	26.793	ESCZ-MQ
2	**35 %**	**27 %**	**30 %**	5%	3%	23.818	ESCZ-MULTI
3	18%	**49 %**	26%	4%	3%	16.058	ESCZ-FR

Tabla 3.6: Segmentación Perfil ESCZ por Categorías.

cluster	CP	FR	MQ	TC	TF	k	Nombre
1	**47 %**	26%	19%	5%	3%	33.780	ES-CP
2	22%	25%	**44 %**	5%	4%	33.822	ES-MQ
3	24%	**49 %**	20%	4%	3%	23.883	ES-FR

Tabla 3.7: Segmentación perfil ES por Categorías.

Nombre Perfil	Cantidad Consultoras
MULTI-FR	3.776
MULTI-CP	5.695
MULTI-MQ	7.013
ESCZ-MQ	26.793
ESCZ-MULTI	23.818
ESCZ-FR	16.058
ES-CP	33.780
ES-MQ	33.822
ES-FR	23.883

Tabla 3.8: Perfiles Finales.

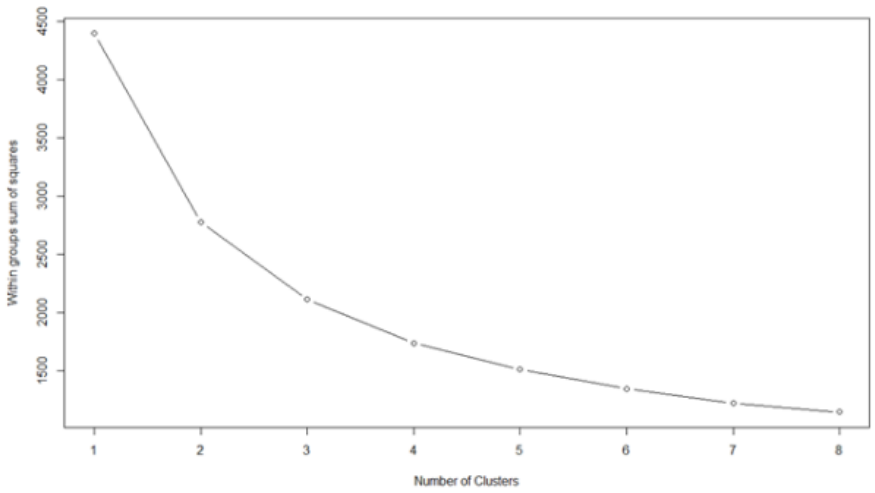

Figura 3.6: Gráfico de Codo Segmento ESCZ en subsegmentos.

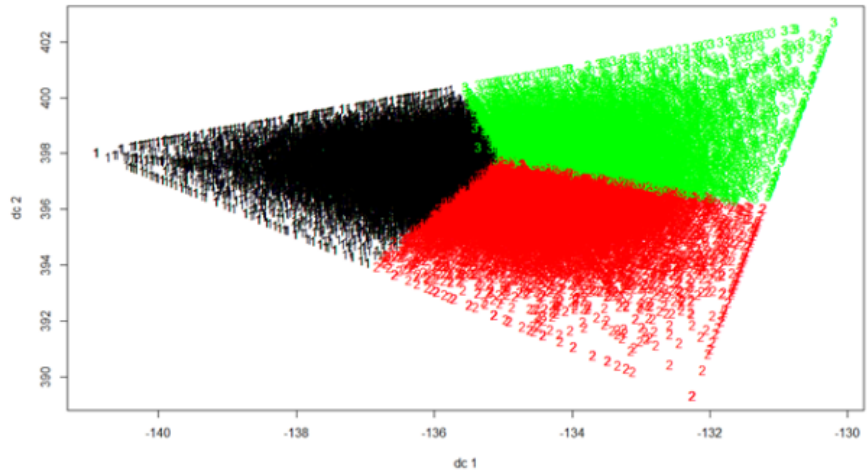

Figura 3.7: Distribución Clusters Segmento ESCZ.

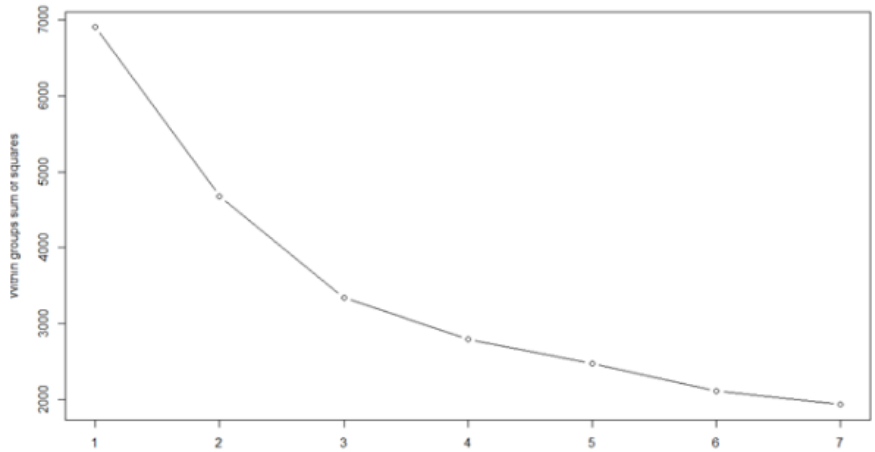

Figura 3.8: Gráfico de Codo Segmento ES en subsegmentos.

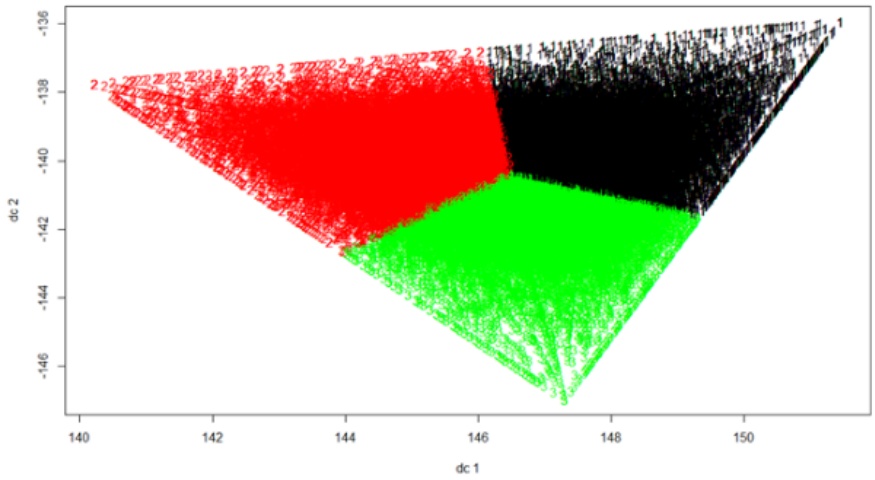

Figura 3.9: Distribución Clusters Segmento ES.

3.2. Propensión de Compra

Teniendo una segmentación de clientes, el siguiente paso en el algoritmo recomendador es determinar los productos más afines a cada cliente. Para realizar este proceso generalmente en retail lo que se hace es utilizar la recencia y la frecuencia para así tener una idea de cuándo el cliente quiere volver a comprar el producto, y por tanto, debe ser ofrecido con algún tipo de oferta. En este caso el tema de cómo ofrecer el producto, consistente en formato, unidades y precio, será tema de las secciones posteriores, éste capítulo se centrará en determinar la afinidad de cada consultora hacia el portafolio de productos, de modo que luego, utilizando esta priorización o preferencia por ciertos productos, generar una oferta que sea atractiva y logre también cumplir con los objetivos de negocio que se requieren.

Para calcular la propensión de compra de productos lo principal es utilizar las características de ellos, dado que al ser productos con una duración por temporadas, estos son reemplazados por otros de similares características pero con otro nombre o formato; esto ocurre frecuentemente en los retails de ropa, donde sus productos son solo por temporada, sin embargo siempre venden los mismos tipos de artículos. En los retails que venden alimentos, al contrario, pueden utilizar agrupaciones por marcas, categorías, tipos y otras sub agrupaciones jerárquicas que se estimen convenientes para calcular los productos preferidos por sus clientes.

En el caso en el que se trabaja lo idóneo es realizar el modelo de propensión de compra considerando los atributos de los productos, sin embargo el maestro de productos no se encuentra con este detalle al momento de realizar esta parte de la memoria, por lo que quedará propuesto realizar un nuevo modelo de propensión cuando esté la información de los productos. Se continua entonces con un enfoque menos óptimo utilizando características de las jerarquías que agrupan a los productos y luego utilizar un método para llegar a una priorización a nivel de productos.

Para el modelado se probaron diferentes tipo de modelos de machine learning, entrenando todos con el mismo dataset y luego siendo testeados contra el mismo set de prueba, éste set incluye la información de las últimas 18 campañas para considerar productos de todas las campañas realizadas en un año y evitar problemas de estacionalidad. La métrica de comparación fue la precisión

Modelo	Entrenamiento	Test
Regresión logística	87.6 %	85.2 %
Árbol de Regresión	92.5 %	91.4 %
Árbol de Inferencia	91.9 %	90.2 %
Regresión Lineal	82.3 %	79.7 %

Tabla 3.9: Resumen Precisión Modelos Propensión de Compra.

de estos con respecto a la variable objetivo. Dado que los modelos serán calculados a nivel de marca-categoría-tipo, es necesario de alguna forma reducir la cantidad total de modelos que serán calibrados. Existen alrededor de 150 combinaciones de marca-categoría-tipo(de ahora en adelante se abreviará como MCT). Para reducir la cantidad de MCT se utiliza un concepto bastante común en el mundo del retail: la concentración. Se determina la regla de pareto *ochenta-veinte*, es decir, las principales MCT(las que más aportan) que hacen el 80 % del total de la venta en dinero. Con este criterio se calcula que 27 MCT concentran el 80 % de la venta del país, por lo que solo se calibrarán 27 modelos.

La variable objetivo se define con valores binarios como 1 si en el siguiente periodo la consultora compra la MCT o 0 si no compra la MCT.

En la Tabla 3.9 se muestra la comparación de precisión promedio de las 27 MCT obtenida por los diferentes modelos aplicados sobre los set de entrenamiento y test.

Se opta entonces por el modelo de árbol de regresión. Si bien la precisión parece bastante prometedora, es necesario utilizar otra métrica para ver la calidad del modelo, es por esto que en la Tabla 3.10 se muestran el área bajo la curva ROC, conocido este indicador como AUC.

Se espera que el indicador AUC sea cercano a 100 %(siendo visto como porcentaje), sin embargo como se aprecia en la Tabla 3.10 los valores de los modelos están un poco alejados del mejor valor, esto puede deberse en gran parte a la calidad de las variables predictoras, como se comentó en un principio, se utilizan valores agrupados por MCT, lo cual quita detalle a la información provista por las variables.

Finalmente con estos modelos es necesario lograr una priorización a nivel de producto-consultora. Ya tenemos la relación consultora-MCT, por lo que es necesario construir una forma de llevar esta probabilidad a un nivel inferior de agregación. Se utiliza de guía el enfoque RFM del retail para construir una pseudo probabilidad asociada a cada producto de una MCT. Esta pseudo probabilidad se define como:

$$\mathbb{P}(ComprarProducto) = \mathbb{P}(MCT) * ShareVenta * (\frac{R}{F})^{-1}$$

Esta fórmula está compuesta por la probabilidad de las MCT asociada al producto, el share de venta del producto con respecto a su MCT en los últimos seis períodos y la división de la recencia en la frecuencia de compra del producto. En retail se utiliza la división de la recencia en la frecuencia de compra para denotar el periodo de recompra de un producto, cuando este valor es más cercano

MCT	AUC_TRAIN	AUC_TEST
MCT1	68,45 %	72,27 %
MCT2	69,43 %	70,45 %
MCT3	67,77 %	71,18 %
MCT4	68,64 %	68,21 %
MCT5	69,40 %	72,90 %
MCT6	68,65 %	67,45 %
MCT7	70,01 %	71,96 %
MCT8	67,21 %	67,45 %
MCT9	66,34 %	65,35 %
MCT10	70,23 %	69,02 %
MCT11	66,19 %	66,16 %
MCT12	67,46 %	67,11 %
MCT13	66,88 %	67,51 %
MCT14	70,13 %	69,91 %
MCT15	67,79 %	68,87 %
MCT16	68,72 %	67,63 %
MCT17	68,49 %	67,94 %
MCT18	68,27 %	65,96 %
MCT19	67,49 %	65,65 %
MCT20	66,74 %	65,41 %
MCT21	67,02 %	65,26 %
MCT22	65,83 %	66,84 %
MCT23	76,02 %	74,70 %
MCT24	66,62 %	65,63 %
MCT25	67,94 %	64,91 %
MCT26	65,02 %	65,77 %
MCT27	69,31 %	67,09 %

Tabla 3.10: AUC para los Modelos de Propensión de Compra.

a 1 significa que el producto está pronto a cumplir su ciclo de recompra, mientras que mayores a uno significa que pasó el periodo de recompra y no fue comprado, por lo que debe ser ofrecido al cliente, sin embargo, en nuestro caso, por conocimiento de negocio se sugiere utilizar el indicador de manera inversa, aludiendo a que es un producto con menor recencia es más probable que esté en la memoria de la consultora y por lo tanto ésta podría ofrecer el producto a sus cliente. Esto se basa netamente en el tipo de interacción que existe entre la consultora y el consumidor final del producto. Queda propuesto utilizar el indicador sin invertir la división.

Al finalizar esta etapa ya se cuenta con una relación consultora-producto y una segmentación de consultoras que nos va a permitir empezar a generar las ofertas. Falta entonces determinar el formato, el descuento y las unidades en que serán ofrecidos los productos a las consultoras. Todo esto se presenta en las secciones siguientes.

3.3. Descuento Personalizado

El objetivo de este etapa es encontrar un descuento personalizado por consultora-producto para luego ser utilizado en la generación de ofertas según el formato que se determine.

Habiendo afrontado el problema de la falta de información con respecto a los productos en la etapa de propensión de compra 3.2, se opta por un enfoque más práctico y menos técnico, donde lo que se busca es dar a cada consultora el mínimo descuento moda para cada uno de sus productos. Ya que no se tiene acceso a la información de costo de los productos, el realizar un modelo de pricing[2] no es factible. Se asume, entonces, el supuesto que los precios ofrecidos históricamente en general debieran estar por sobre el precio del costo.

El periodo sobre el que se calcula el descuento moda corresponde a 18 campañas, es decir, un año. Obtener el descuento dado a una consultora no es directo. Se tiene la información del precio que la consultora paga por un producto, el precio de oferta del producto(no necesariamente el mismo que el precio que una consultora termina pagando) y el precio normal o precio base de un producto.

Las consultoras reciben un descuento sobre el total del pedido, llamado *comisión*, el que se calcula como:

$$Comision = 1 - \frac{VentaFacturaCatalogo}{VentaPorCatalogo}$$

Si el valor obtenido es menor a cero se reemplaza por cero. Luego, con el valor de la comisión se procede a calcular el precio de venta unitario del catálogo (PrecioVentaCatalogo):

$$PrecioVentaCatalogo = \frac{VentaPorCatalogo}{Unidades}$$

[2]Modelo que permite estimar el precio óptimo dado un set de atributos.

Con estos datos el descuento dado a una consultora por un producto se calcula con la siguiente fórmula:

$$Descuento = \begin{cases} 1 - \frac{PrecioVentaCatalogo*(1-Comision)}{PrecioNormal}, & \text{Si VentaPorCatalogo} > 0 \\ 1 - \frac{PrecioVentaCatalogo}{PrecioNormal}, & \text{Si VentaPorCatalogo} \leq 0 \end{cases}$$

Este valor de calcula con respecto a cada consultora y cada producto, por lo que se llega al problema de no tener un descuento para productos que la consultora no ha comprado en las últimas 18 campañas. Para solucionar esta situación es necesario ir un nivel más arriba en 2 posibles dimensiones: consultora y producto. En el primer caso el nivel de agrupación superior corresponde al perfil calculado en la primera sección(véase 3.1), en el segundo, marca-categoría-tipo es el nivel superior.

La estrategia es calcular los descuentos a nivel de perfil-producto, consultora-mct y perfil-mct adicionalmente al consultora-producto. En todos los casos el descuento se calcula como la moda del valor encontrado utilizando el mismo cálculo que para el descuento consultora-producto.

Los descuentos deben cumplir con ciertas reglas de negocio dadas que dependen del tipo de formato de oferta que se vaya a ofrecer a una consultora y los productos asociados. Esto se verá en las 2 secciones siguientes.

3.4. Modelos de Formato

3.4.1. Modelo de Volumen

El Modelo de Volumen constituye la primera de las dos estrategias de generación de formatos de ofertas. En esta etapa se determinará si un producto se ofrecerá de manera individual o en 2 o más unidades. Aquí aplica una regla de negocio asociada a la forma en que las ofertas son ofrecidas a las consultoras: las ofertas generadas se le comunican a las consultoras mediante el sitio web de la empresa con el uso de una imagen asociada a la descripción de la oferta, por lo que no es posible colocar más de cinco productos en la imagen, sino los productos no llegan a ser distinguibles entre sí.

Con lo anterior como margen de desarrollo, en este modelo se procede de forma similar a lo realizado en el Modelo de Propensión, es decir, se realizan un modelo de machine learning para cada mct relevante y, además, se genera un modelo adicional que agrupa a todas las mct restantes. Finalmente, dada la misma problemáticas de no contar con datos de atributos de productos, teniendo una estimación por mct, se procede a generar una estrategia para llevar la estimación a nivel de producto.

Para modelar se consideran un dataset de las últimas 3 campañas y de las siguientes características descritas en la imagen 3.10. Las primera sección de variables encerradas en color verde

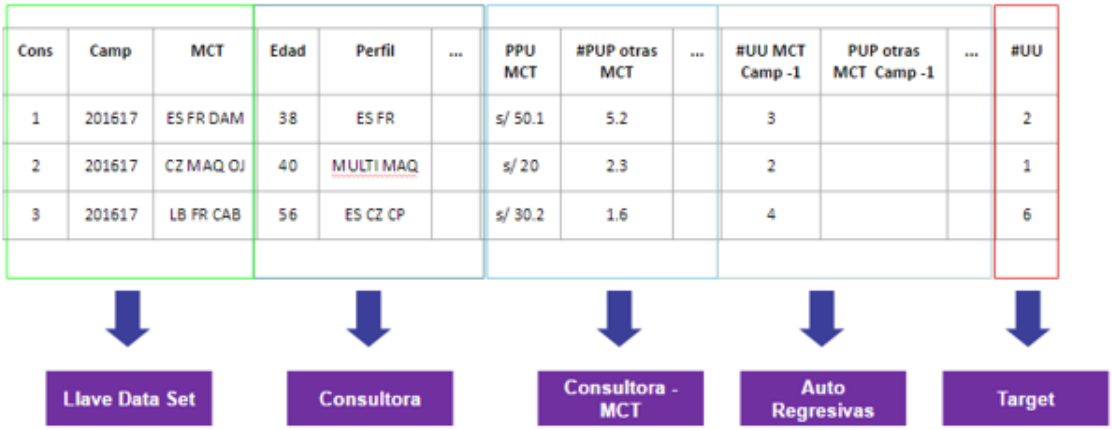

Figura 3.10: Imagen compacta del dataset de entrenamieto del Modelo de Volumen.

corresponden a las llaves del dataset: Código de consultora, el número de la capaña a la que pertenecen los datos y la MCT asociada. El siguiente bloque hace referencia a variables sociodemográficas de la consultora. El tercer bloque corresponde a variables transaccionales asociadas a la MCT y la consultora, además se incluyen variables autorregresivas. La última columna, marcada en rojo, indica el valor de unidades compradas de la MCT en la campaña asociada. Esta última columna es la variable a predecir. Cabe notar que la variables es siempre mayor o igual a cero y de valor entero.

Para el modelado se consideran 2 modelos lineales generalizados. Poisson y Tobit censurado, y 2 modelos computacionales, Árbol de Regresión y Redes Neuronales (Perceptrón multicapa). Dado que el modelado se hace en el lenguaje R, para el modelo Tobit se usa la función tobit() del paquete *AER*[2], para el Árbol de Regresión se utiliza la función rpart() del parquete *RPART*[6], para la red neuronal, nnet() del paquete *NNET*[3] y par el Modelo de Poisson se usa la función glm() del paquete *STATS*[1].

Se construyen los 4 modelos para cada una de las 27 MCT relevantes y para el dataset de las MCT restantes. Para comparar el resultado de los modelos se utiliza la metodología **K folds Cross Validation**, con k=5. Lo que propone esta metodología es generar k repeticiones, 5 en nuestro caso, para cada uno de los 4 tipos de modelos sobre los 28 datasets. Luego, es necesario utilizar una métrica de comparación, dado el tipo de variable de respuesta en los modelos, se utiliza el índicador **RMSE**[3]. Éste valor es graficado en cada una de las k iteraciones.

De la Figura 3.11 se obtiene que los modelos Árbol de Regresion y Tobit son los que mejor describen el comportamiento del número de unidades a comprar por una consultora, sin embargo, luego de realizar unas pruebas, se obtiene un mejor resultado al combinar ambos modelos mediante el promedio de ambas estimaciones.

Teniendo la predicción a nivel de MCT es necesario crear un método para repartir estas unidades en los diferentes productos asociados a la MCT respectiva. Se define el siguiente algoritmo para llevar la predicción a nivel producto:

1. Calcular número de distintos productos comprados por cada consultora con respecto a cada una de las MCT en las últimas 3 campañas.

[3]Root-Mean-Square Error

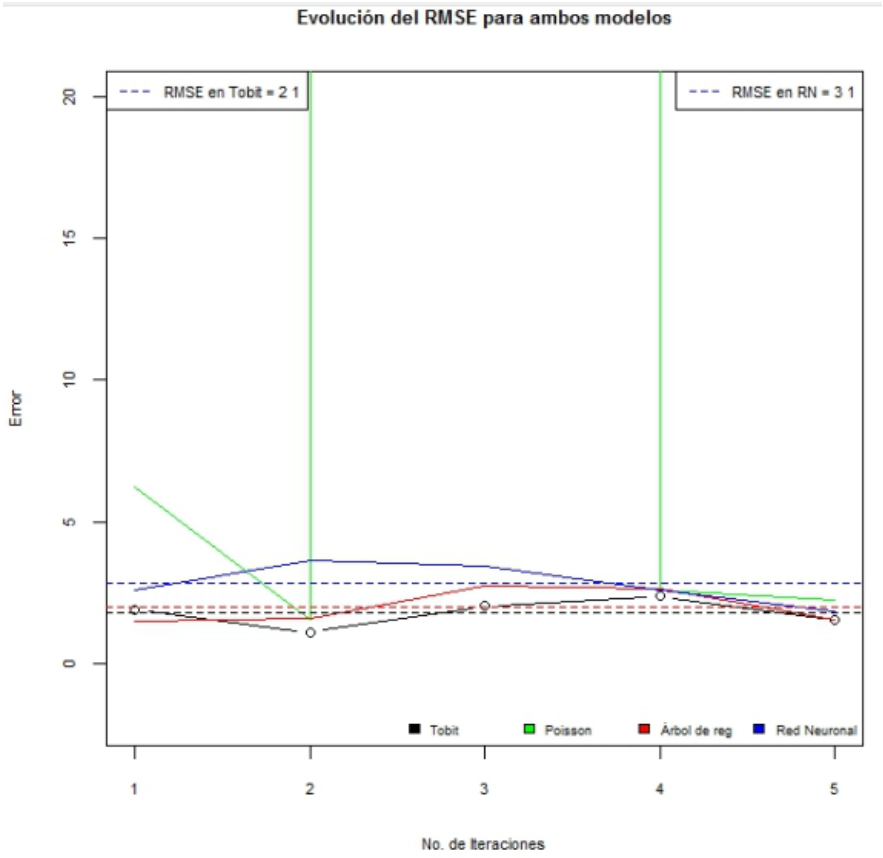

Figura 3.11: Error Modelos de Volumen con respecto a las Iteraciones.

2. Obtener el promedio de los 3 valores obtenidos por cada consultora y MCT.

3. Dividir el número de unidades predecidas por los modelos en el valor obtenido en el paso anterior.

4. Aplicar reglas de negocio:

    - Sumar una unidad al valor obtenido, esto para aumentar el número de unidades ofrecidas pensando generar venta incremental.
    - El valor de las unidades a ofrecer no puede ser mayor a 5, dado que las ofertas deben ser expuestas mediante una imagen donde más de 5 unidades hacen poco distinguible los productos.

Luego de que se tiene las unidades por cada producto de una MCT es necesario asociar este valor a los productos. Es aquí donde se juntan los valores obtenidos en la salida del modelo de propensión.

Cabe destacar que aquí también se generan ofertas del tipo individual, ya que si el modelo determina cero unidades, por regla de negocio se añade una unidad.

Finalmente lo que se obtiene de esta etapa es una tabla que contiene por cada fila el código de una consultora, un código de producto y un valor de unidades estimadas, además de su pseudo

probabilidad asignada en el modelo de propensión y valores asociados a la consultora y el producto como el perfil o la MCT. Por temas de compatibilidad con el Modelo de Bundle(véase siguiente sección) se agrega un valor indicando que el producto es el *trigger* de la oferta.

3.4.2. Modelo de Bundle

El Modelo de Bundle consiste en generar *packs* de productos que componen una única oferta. Éste set de productos debe ser personalizados y debe cumplir también con ciertas reglas de negocio asocidas.

Se utiliza un algoritmo fundamental en el mundo del retail para la generación de ofertas que mezclen productos: apriori[4]. Éste algoritmo es crucial en la realización de un **Market Basket Analysis**.

El algoritmo apriori lo que permite es descubir asociaciones ocultas entre productos a través del estudio de combinaciones de productos que frecuentemente ocurren en las transacciones realizadas por las consultoras.

Es necesario manejar las siguientes definiciones:

- **Productos**: Objetos a los cuales se trata de encontrar relaciones entre ellos.
- **Itemset**: Conjunto de productos.
- **Transacción**: Instancia de grupos de productos que coexiten juntos.
- **Regla**: Una regla es una notación que representa qué producto es frecuentemente comprado con algún otro producto o productos. Posee un *lado izquierdo*(LHS o antecedente) y un *lado derecho*(RHS o consecuente), que se denotan de la siguiente manera:

$$X \Rightarrow Y$$

Además, cada regla tiene un conjunto de métricas asociadas que permiten determinar la *fuerza* de ésta:

- **Soporte**: Supp(X), indica la proporción de transacciones en la base de datos que contienen el producto o cojunto de productos X.
- **Confianza**: Conf(X=>Y), es obtenida por una regla e indica qué tan probable es que el producto Y sea comprado cuando el producto X es adquirido. X e Y pueden ser un conjuntos de productos.
- **Lift**: lift(x=>Y), indica la probabilidad de que ocurran todos los ítems en una regla dividido por el producto de las probabilidades de ocurrencia de los ítems del lado izquierdo.

El algoritmo apriori trabaja en dos pasos:

1. Identifica sistemáticamente itemsets que ocurren frecuentemente en los datos con un soporte mayor a un límite pre-especificado.

2. Calcula la confianza de todas las posibles reglas dados los itemsets frecuentes y se queda sólo con lo que tienen una confianza mayor a un valor pre-especificado.

Entonces para poder utilizar el algoritmo es necesario definir un soporte mínimo y una confianza mínima que deben cumplir las reglas. Además, el dataset sobre el que trabaja el algoritmo es algo particular, ya que cada fila de él debe representar una transacción, es decir, cada fila debe contener una lista con los códigos de los distintos productos llevados en la transacción que se representa en fila. Con el objetivo de tener un gran se de reglas donde poder escojer, se determinan los siguientes valores de para el algoritmo:

- Soporte = 0.01, esto implica que los productos de la regla ocurren por lo menos en el 1 % de las transacciones.
- Confianza = 0.5, significa que por lo menos para el 50 % de las transacciones la regla es correcta.

Para aplicar el algoritmo la idea es seguir otro concepto de *cliente espejo*, el cual nos dice que bajo ciertas condiciones dos clientes debiesen ser similares, entonces, lo que se busca es generar packs de productos sobre un conjuntos de transacciones realizadas por clientes similares, es decir, a clientes que pertenecen a un mismo perfil. Como ya se tiene el perfil de cada cliente determinado según la sección 3.1, el problema se reduce entonces a generar un dataset de transacciones de la última campaña según cada uno de los perfiles determinados.

Una llamada al algoritmo apriori, del paquete *ARULES* de R, se ve de la siquiete manera:

```
rules = apriori(transacciones,
                parameter = list(supp=0.01, conf=0.5))
```

Luego de ordenar las reglas obtenidas según su confianza de forma decreciente, una inspección a los primeros 6 valores se muentra en la Figura 3.12.

```
    lhs                          rhs         support confidence lift
[1] {200056636,200087691}    => {200017521} 0.013   1.00        6.4
[2] {200040880,200056636}    => {200017521} 0.013   0.97        6.2
[3] {200083898}              => {200084078} 0.011   0.96       70.2
[4] {200076632,P0210136000}  => {200064178} 0.010   0.95       35.6
[5] {200084783,200084784}    => {200084815} 0.026   0.95       19.3
[6] {200084783,200084815}    => {200084784} 0.026   0.93       19.8
```

Figura 3.12: Primeros 6 valores obtenidos del algoritmo apriori.

Para que el resultado del modelo apriori sea utilizable y se conecte con las demás partes del MDO es necesario formatear la salida dada. Es necesario contar con los diferentes productos que componen una regla en diferentes columnas.

Por construcción del algoritmo al lado derecho de la regla quedan solamente conjuntos de un único elemento, esto ocurre por la confianza mínima determinada. Luego, es necesario separar el lado derecho de la regla en una columna por cada uno de los productos. Para realizar la separación se utiliza un proceso almacenado de SQL que itera obtiene el máximo número de elementos en todos los lados derechos de las reglas y por va agregando una columna nueva y separando los productos del conjunto hasta que solo existan columnas con un elemento. Los grupos con menos miembros en el lado derecho tendrán columnas con valores vacíos donde corresponda.

Lo siguiente es normalizar el formato según lo que se obtuvo en el Modelo de Volumen, se procede entonces a enumerar las diferentes combinaciones de reglas obtenidas para luego proceder

producto	trigger	support	confidence	lift	num_bundle
200056636	1	0.013	1	6.4	1
200087691	1	0.013	1	6.4	1
200017521	0	0.013	1	6.4	1
200040880	1	0.013	0.97	6.2	2
200056636	1	0.013	0.97	6.2	2
200017521	0	0.013	0.97	6.2	2

Tabla 3.11: Ejemplo de las 2 primeras reglas obtenidas en formato fila.

a llevar los productos de cada columna a un formato de filas donde se identifican dos posibles tipos de productos: los triggers(aquellos del lado derecho de la regla) y los recomendados(los del lado izquierdo). Se llaman productos trigger ya que son estos los que dan pie a que la regla ocurra y, por consiguiente, son quienes gatillan la compra de los productos recomendados. La tabla 3.11 muestra como se ve regla 1 y 2 de la Figura 3.12 en el formato de filas.

Con el objetivo de ampliar el mix de la canasta de compra de cada consultora, las reglas obtenidas con el formato adecuado son cruzadas contra la tabla de pseudo probabilidades de los productos y consultoras. Según el perfil de cada consultora se alisgnarán todas las reglas de asociación encontradas que cumplan con las siguientes 2 reglas:

1. La consultora compró los productos del lado izquierdo de la regla durante la última campaña.

2. La consultora **no** compró los productos del lado derecho de la regla durante la última campaña.

Con esto lo que se obtiene son sets de productos que en un perfil se llevan juntos, pero algunas consultoras no están comprando todos los productos del set.

Al finalizar esta etapa lo que se tiene es una tabla donde cada fila contiene el código de consultora, un código de producto, indicadores de la regla asociada(soporte, confianza, lift y número de bundle), el tipo de producto(trigger o no), la pseudo probabilidad del producto calculada en el capítulo de propensión de compra3.2 y el perfil de la consultora.

3.5. Asignación de Ofertas

Esta es la etapa final del MDO. Aquí es donde se formarán las ofertas finales que serán presentadas a cada consultora en el portal de la empresa.

De las secciones anteriores ya se cuenta con la información de formatos de oferta, descuentos y productos probables para cada consultora, por lo que los siguientes pasos consisten en los siguientes puntos:

1. Consolidación: unir formatos con descuentos y dar formato único de oferta.

2. Priorización: Determinar el orden en que serán presentadas las ofertas.

3. Relleno: Generación de ofertas por perfil.

En ésta sección se verá la aplicación de varias reglas de negocio basadas en conocimiento comercial del mercado de la empresa.

3.5.1. Consolidación de ofertas

De las secciones 3.4.1 y 3.4.2 tenemos la relación consultora-producto-unidades-formato, por lo que es necesario unir los resultados con lo obtenido en la sección 3.3, los descuentos.

Descuentos Formato Volumen

La primera parte será unir los resultados del modelo de volumen con los descuentos, dado que la relación está a nivel de consultora producto en el modelo de volumen, es directo el cruce contra la tabla de los descuentos. En los casos en que no esté el descuento se procede en el siguiente orden asignando el descuento correspondiente:

1. Descuento consultora MCT del producto en el modelo de volumen.

2. Descuento Perfil de la consultora con respecto al producto del modelo de volumen.

3. Descuendo Perfil de la consultora y MCT del producto.

El descuento por regla de negocio no puede superar al 60 %, por lo que en caso de haber obtenido un descuento mayor a este valor, se cambia por 60. Además se aplica una regla de negocio asociada a otro canal de venta: el medio impreso. Dado que existe una revista donde se ofrecen los diferentes productos, sólo las ofertas cuyo descuento seá más atractivo(mayor descuento) que el medio impreso son aceptadas como válidas, las que no cumplan esta restricción son descartadas.

La tabla final tiene entonces la relación consultora-producto-formato-unidades-descuento.

Descuento Formato Bundle

El formato bundle presenta un mayor desafío a la hora de determinar el descuento asociado al set. Con los descuentos calculados en diferentes niveles de agrupación, se procede igual que en 3.5.1, producto por producto de cada set.

El descuento asociado a este formato de oferta debe ser único para todo el conjunto de productos, por lo que se consolidan cada uno de los descuentos que forman el set de la siguiente manera:

1. Calcular el precio final de cada producto del set aplicando el descuento asignado.

2. Calcular el valor del set sin aplicar descuento alguno a los productos que lo componen, esto es, sumar el precio de cada producto.

3. Determinar el cuociente entre la suma de los valores de los productos luego de aplicar el descuento a cada uno, sobre el valor calculado al sumar los precios sin descuento.

4. El descuento es de 1 menos el valor calculado en el paso anterior.

La interpretación del algoritmo descrito anteriormente es: *el descuento de la oferta esta determinado por el monto descontado del total de la oferta al aplicar un descuento personalizado sobre cada uno de los elementos que la componen*.

Al igual que en el caso de las ofertas de volumen, si el descuento determinado es mayor 60 %, se cambia por 60.

Con esto se tiene una tabla con una información similar a la obtenida en el caso del volumen. Lo siguiente es determiar qué ofertas serán ofrecidas finalmente a cada consultora.

3.5.2. Priorización de ofertas

La priorización de ofertas se realiza según un indicador conjunto construido a partir de las características de la oferta. Se define el indicador de priorización según la siguiente fórmula:

$$I_{cons,of} = Dscto_{cons,of} + Prop_{cons,of} + Form_{cons,of}$$

Donde $Dscto_{cons,of}$ corresponde al descuento de la oferta, $Prop_{cons,of}$ corresponde a la *"probabilidad"* de la oferta y $Form_{cons,of}$ está determinada por el formato de la oferta.

El valor de $Dscto_{cons,of}$ ya está determinado para cada uno de los formatos. Determinar el valor de $Prop_{cons,of}$ se afronta de manera similar al descuento. En el caso de las ofertas de volumen, este valor está dado por la pseudo probabilidad asociada al producto de la oferta. Para los bundles es necesario consolidar el valor de esta pseudo probabilidad de cada uno de los productos trigger que componen la oferta. Se consideran sólo los productos triggers ya que por construcción de la oferta, el recomendado no es comprado por la consultora, por lo que son los triggers los que se llevan el peso del *"valor"* de la oferta. Luego, el valor de $Prop_{cons,of}$ se define como la media de las pseudo probabilidades de los productos triggers.

Determinar $Form_{cons,of}$ para el caso de los set de productos sigue la lógica de la *fortaleza* del conjunto generado. Se calcula la confianza de los productos que componen la oferta con respecto a la MCT del producto recomendado. En el caso de las ofertas de volumen el valor de $Form_{cons,of}$ corresponde a la cantidad de pedidos en las últimas 6 campañas donde la consultora haya pedido un número de unidades igual o superior a lo que se le está ofreciendo dividido en la ventana de tiempo, es decir, 6.

Con los 3 valores calculados se procede a calcular el indicador conjunto para cada una de las ofertas asociadas a cada consultora. Lo siguiente es ordenar las ofertas de cada una con respecto al valor de este indicador de manera decreciente para dejar primero las mejores ofertas según este indicador.

Posición	Objetivo Comercial	Posible Formato Oferta 1	Posible Formato Oferta 2
1°	Frecuente	Individual	Volumen
2°	Perfil	Bundle	Individual
3°	No Compra	Bundle	Individual
4°	Frecuente	Individual	Volumen

Tabla 3.12: Orden que deben cumplir las ofertas a mostrar a las consultoras.

A pesar de ya tener una priorización de ofertas determinada por el indicador recién calculado, como regla de negocio se debe cumplir con un orden específico que considere el valor ya obtenido. Dado que las ofertas son ofrecidas por el sitio web, se estudió el mapa de calor del sitio para determinar cuáles son las ofertas que primero ven las consultoras(donde se realizan los mayores clicks). Según lo anterior se define la Tabla 3.12 que determina el orden que deben cumplir las ofertas y el tipo de oferta que puede ser ofrecido en cada caso con el objetivo de presentar las ofertas más relevantes en la posición que primero mira la consultora. Dado que el ordenamiento propuesto considera 4 posiciones, a partir de la posición 4 se repetirá el mismo ordenamiento en proporción de la cantidad de ofertas generadas, es decir, si son 10 ofertas, por ejemplo, este ordenamiento se repite 3 veces.

Las ofertas entonces se van ordenando según el posible tipo de oferta según la posición que se debe completar eligiendo siempre la con mayor indicador conjunto de ordenamiento. Además, es necesario aplicar otra regla de negocio, la cual dice que **no se puede ofrecer un producto en más de un formato**. Teniendo esto en consideración, a medida que se van ordenando las ofertas según el orden comercial establecido es necesario ir descartando ofertas que contienen productos ya ofertados en algún otro formato anteriormente.

Al terminar de ordenar las ofertas ocurre que el número de ofertas por consultora es variado dependiendo del tamaño de la canasta de compra que habitualmente consume cada consultora. Se tiene la regla de negocio que exige tener como **mínimo 8 ofertas** y un máximo de 40 ofertas por consultoras. Esto obliga a generar un método para completar el mínimo necesario para cumplir la regla.

3.5.3. Relleno

Existen consultoras cuyo comportamiento de compra es bastante inconstante y cuyo mix de compras es reducido. En estos casos, el MDO como está hasta éste punto no logra satisfacer el mínimo de ofertas impuesto por regla de negocio. Se define entonces una estrategia de relleno que debe ser capaz de cumplir con todas las reglas de negocio ya impuestas y, además, generar el suficiente número de ofertas para completar todos los casos donde no existan al menos 8 ofertas generadas.

Siguiendo la idea de la personalización y pensando en ampliar el mix de compra se decide ir a buscar las ofertas de bundle generadas para cada perfil de las consultoras, pero solo aquellas donde el número de triggers sea 1, dado que se espera ofrecer packs pequeños esperando que sean

aceptados y luego, si son comprados, el MDO naturalmente irá generando packs de mayor tamaño en siguientes ejecuciones.

Se genera un apartado con todas las consultoras que no tienen 8 ofertas y según su perfíl, se les añaden todas las ofertas de bundle que cumplan con lo antes descrito, luego, se van priorizando según su confianza y se van descartando, primero los bundles que incluyan productos ya ofertados en alguna otra oferta no de relleno y luego, las ofertas que tengan productos ya ofertados en alguna otra oferta de relleno con mejor confianza. Dado que la idea es atraer la compra, se adjunta el mayor descuento posible al set, 60 %.

Aquí se observa que los valores de confianza y soporte determinados en el modelo de bundle(3.4.2) juegan un papel crucial en determinar el tamaño de ofertas posibles a ser seleccionadas en la estrategia de relleno.

Finalmente se une el grupo de las consultoras *rellenadas* con el grupos de las que no utilizaron relleno, completando así el total de ofertas para todas las consultoras que han generado transacciones en las últimas 6 campañas. Queda aun cubrir el caso de las consultoras que no poseen historia transaccional, ya sea por ser nuevas en la empresa o por estar retornando a ella. Este tipo de consultora se le denomina *consultora dummy* y sus ofertas serán determinadas de la siguiente forma: a partir de las compras realizadas por las consultoras nuevas en la última campaña, se toman los productos más frecuentes y con mayor cantidad de unidades vendidas. Se determina el descuento promedio y el promedio de unidades compradas. Luego se ofrecen los productos más frecuentes con el descuento promedio y las unidades promedio. Estas ofertas también deben cumplir todas las reglas de negocio aplicadas a las ofertas antes generadas.

Capítulo 4

Automatización y recalibraciones

4.1. Automatización MDO

Con los algoritmos ya definidos y construidos, es necesario lograr la integración de estos en un proceso tal que el usuario final sólo deba ingresar parámetros iniciales y esperar obtener las ofertas generadas por el MDO.

4.1.1. Perfilamiento

Para afrontar la etapa de perfilamiento debemos entender cómo funciona el algoritmo de kmeans. La distancia euclidiana es la métrica de distancia implementada en el paquete *stats* de *R*. La salida de este algoritmo son *n* puntos en el espacio correspondientes a los centroides determinados, donde luego a cada punto de entrada utilizado en el algoritmo se le asigna el centroide más cercano según distancia euclidiana. Para automatizar la asignación de un perfil a cada consultora, se almacenan 3 tablas que contienen la información de los centroides de marca, marca-categoría y una última con la etiqueta del perfil. En la Tabla 4.1 se muestran los centroides almacenados que son válidos desde la campana 201807 en adelante. En la sección siguiente se explicará el significado de ese valor. La Tabla 4.2 muestra el valor de los centroides de la clusterización por segmento marca-categoría válidos a partir de la campaña 201807 para cada uno de los segmentos de Marca previamente encontrados. Finalmente, la Tabla 4.3 muestra la conversión del par segmento Marca y segmento Marca-Categoría hacia una etiqueta final. Ésta es la etiqueda asignada para fines de entendimiento con el negocio. Las tablas antes mencionadas son completadas en el apartado de calibración que se presentará más adelante.

Cluster	Share LB	Share ES	Share CZ	Periodo
1	0.28	0.49	0.23	201807
2	0.04	0.5	0.46	201807
3	0.04	0.73	0.23	201807

Tabla 4.1: Centroides Marca válidos desde la campaña 201807.

Cluster Marca	Cluster Categoría	CP	FR	MQ	TC	TF	Periodo
1	1	0.16	0.49	0.23	0.05	0.07	201807
1	2	0.39	0.22	0.24	0.06	0.09	201807
1	3	0.15	0.19	0.51	0.05	0.01	201807
2	1	0.16	0.23	0.54	0.04	0.04	201807
2	2	0.35	0.27	0.3	0.05	0.03	201807
2	3	0.18	0.49	0.26	0.04	0.03	201807
3	1	0.47	0.26	0.19	0.05	0.03	201807
3	2	0.22	0.25	0.44	0.05	0.04	201807
3	3	0.24	0.49	0.2	0.04	0.03	201807

Tabla 4.2: Centroides Marca-Categorías válidos desde la campaña 201807 para cada uno de los segmentos de Marca previamente determinados.

Cluster Marca	Cluster Marca-Categoría	Etiqueta Perfil	Periodo
1	1	MULTI-FR	201807
1	2	MULTI-CP	201807
1	3	MULTI-MQ	201807
2	1	ESCZ-MQ	201807
2	2	ESCZ-MULTI	201807
2	3	ESCZ-FR	201807
3	1	ES-CP	201807
3	2	ES-MQ	201807
3	3	ES-FR	201807

Tabla 4.3: Perfiles con respecto a los segmentos Marca y Marca-Categoría válidos desde la campaña 201807.

Luego, con la misma base con la que se utiliza el algoritmo kmeans, se calcula la distancia euclidiana a cada uno de los centroides de la tabla de Marca(4.1. Se obtiene la mínima distancia entre las 3 calculadas y ese valor se le asigna como segmento Marca a la consultora. Con este valor se procede a calcular la distancia al centroide de Marca-Categoría del segmento Marca antes determinado(*Cluster Marca* en 4.2). Nuevamente se obtiene el mínimo valor de las distancias euclidianas para cada consultora a los centroides de Marca-Categoría que le corresponden y se obtiene así el par de valores Cluster Marca y Cluster Marca-Categoría. Finalmente, utilizando la 4.3 se determina la etiqueda de perfil que corresponde a cada consultora.

La interfaz de usuario es bastante intuitiva y cuenta con un módulo que muestra la descripción de cada uno de los parámetros necesarios para la ejecución. En la imagen 4.1 se puede ver que además la herramienta permite ofrecer valores por defecto para los usuarios, esto es de gran utilidad, ya que sirven de un ayuda memoria para saber qué valor debe colocar en cada parámetro.

Em la figura 4.2 se muestra el panel donde se pueden configurar los valores por defecto, así como también asignar una descripción a cada uno de los parámetros para ayudar al usuario final. Cabe notar que la herramienta tiene un módulo que permite su ejecución por línea de comando y también permite recibir parámetros de ejecución por este medio, lo que hace fácil su automatización mediante la utilización de un elemento como Crontab, de linux.

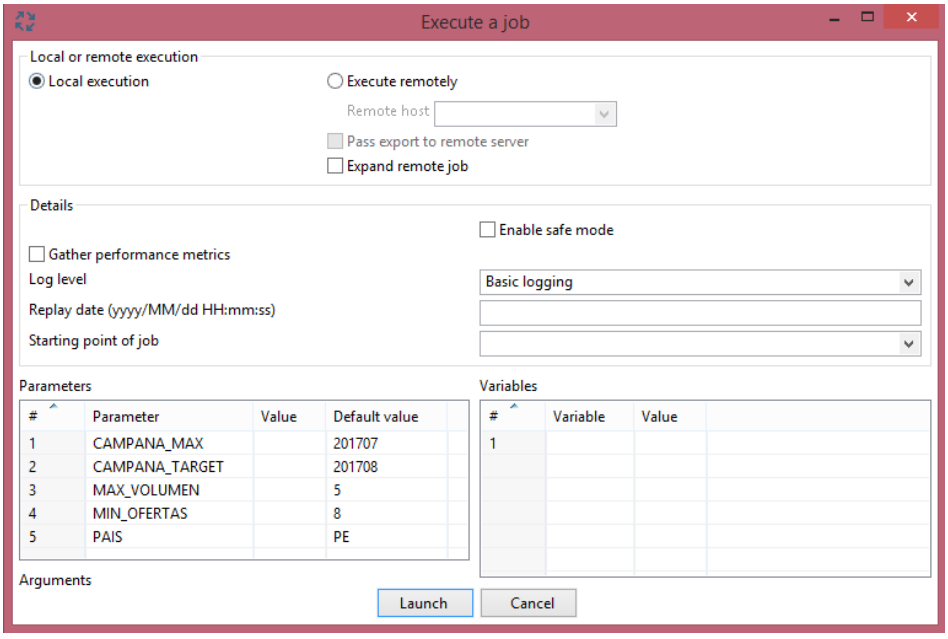

Figura 4.1: Panel para ingresar los parámetros de ejecución e iniciar el proceso de generación de ofertas.

4.1.2. Modelos de Machine Learning

Para los modelos de Propensión de Compra y Volumen, donde son utilizados algoritmmos de machine learning, se utilizará el mismo enfoque ya que ambos tienen una lógica de diseño y construcción similar, por lo que se explicará sólo el modelo de Propensión de Compra.

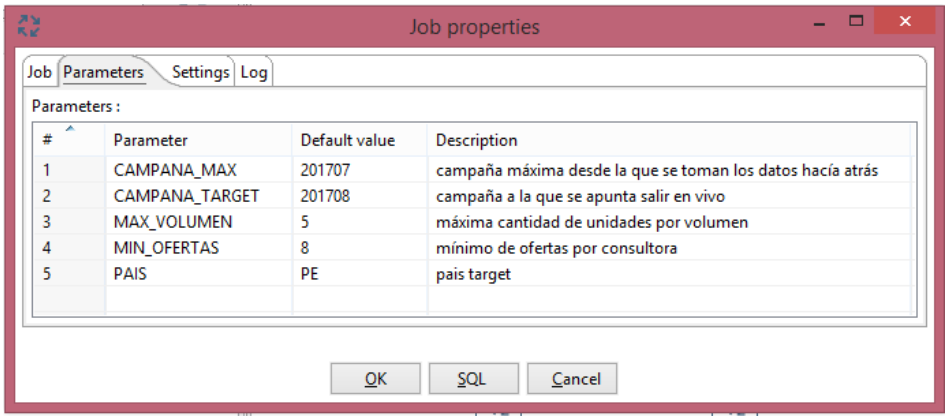

Figura 4.2: Descripción de los parámetros necesarios para la ejecución del MDO

Lo primero es generar la base analítica o tablón sobre la cual se aplicará el modelo, este se realiza con un script SQL. Luego, el siguiente paso es utilizar el modelo ya calibrado sobre esta base para predecir y obtener el resultado. Para esto el script R se conecta a la base de datos y extrae el tablón generado, luego da formato a las variables según fue calibrado el modelo(variables continuas, categóricas, enteras, etc.). Entonces, para cada una de las combinaciones de marca-categoría relevantes va, dentro de una ruta específica, a buscar el modelo de machine learning ya calibrado, lo lee y luego predice utilizándolo. Finalmente, consolida todos los resultados de todas las marcas-categorías en un solo dataframe[1] y, a continuación, lo inserta en una tabla de la base de datos, donde luego esto es el input del script que realiza la bajada a nivel de producto.

El algoritmo apriori del Modelo de Bundle utiliza sólo una tabla con los parámetros de corte de las reglas de asociación generadas, es decir, confianza, lift y soporte mínimo que deben tener las reglas generadas.

4.2. Recalibraciones

Los modelos de machine learning permiten detectar patrones no visibles son análisis realizados por personas en tiempos muy reducidos; éstos van *aprendiendo* de la información con la que se los entrena, sin enbargo, a medida que transcurre el tiempo van quedando obsoletos ante nuevos comportamientos ya sean naturales de los factores o por la implementación de medidas a partir de los resultados de un primer modelamiento.

Los periodo de recalibración de algoritmos en el mundo de retail dependen del negocio en cuestión. Donde un retail de ropa, por ejemplo, utiliza 2 periodos anuales diferentes(invierno-verano), pero otros, como los supermercados, utilizan periodos más cortos dado que sus productos son consumibles todo el año. Pensando en esto, en el mercado de la empresa de venta directa para la cual se desarrolla el motor los productos tienen una estacionalidad menos marcada, por lo que se sigue un enfoque inicial de dividir el año en 3 periodos. Un año cuenta con 18 campañas o periodos de ventas, por lo que se realizará una recalibración de los modelos cada 6 campañas.

[1]Tipo de dato de R, similar a las matrices.

Para el caso del Perfilamiento, se generará la tabla con los datos y se repetirá el proceso descrito en el capítulo 3.1, aplicando el algoritmo kmeans y determinando los segmentos de Marca y de Marca-Categoría que correspondan. Los valores de los centroides encontrados se insertarán en las tablas correspondientes con el valor de la máxima campaña utilizada en la base más 1, dado que los perfiles encontrados serán válidos a partir de la siguiente campaña. No necesariamente se conserva la cantidad de perfiles determinados. El ver cómo evolucionan los perfiles podría permitir el diseño de otro tipo de ofertas a futuro.

Siendo el perfilamiento el una de las bases para los demás modelos, se sigue la misma configuracion para los demás modelos. Se desarrollan módulos externos al MDO para poder realizar estas calibraciones de manera más rápida y evitar cualquier tipo de inconsistencia que alterase el correcto funcionamiento del motor en el futuro luego de un proceso de recalibración.

Los modelos de machine learning utilizados corresponden a modelos supervisados y no supervisados, en ambos casos es necesario la revisión por parte de una persona que mida los resultados y haga los ajustes que sean necesarios para lograr la precisión esperada. Los módulos de recalibración constan de unos script que requieren parámetros de entrada como la campaña objetivo, de la que se calcula la variable de respuesta, y la máxima campaña a partir de la cual se contarán 5 campañas más hacia atrás para generar la base del modelo correspondiente.

Una vez que los modelos estén listos, los objetos generados anteriormente(archivos .RData) son llevados a otra carpeta solo para fines de tener un histórico y, luego, los nuevos modelos generados son colocados en la carpeta correspondiente que es la carpeta que el MDO va a leer según la etapa que esté ejecutando.

Para el caso algoritmo apriori es necesario ejecutarlo y luego revisar la cantidad de reglas de asociación generadas por perfil. Acá es importante que el número de reglas permita al MDO generar un set de ofertas suficientes para que pueda optar por las mejores para cada consultora. Un número muy pequeño puede gatillar en solo ofertas de relleno, mientras que un número muy grande puede terminar en una sobrecarga de procesamiento y con ofertas que no soy seleccionadas para ninguna consultora.

Capítulo 5

Validación y ajustes del MDO

5.1. Prueba Piloto del Motor

Para probar el Motor de Diseño de Ofertas se utilizó el enfoque de *Diseño de Experimentos*[1], donde se determina un grupo de control y un grupo piloto, ambos de similares características. Luego, al grupo piloto se le aplica el estímulo (el MDO en este caso), mientras que al grupo control se le aisla del fenómeno(se le ofrecen las ofertas generadas por el algoritmo anterior).

Considerando el impacto comercial que puede tener la implementación de un cambio en la forma de generar ofertas, se define un grupo de control de 653 consultoras y un grupo piloto de 654 consultoras, escogidas aleatoriamente sobre una zona geográfica determinada, pero cuyas características transacionales y de comportamiento de compra son similares. Las consultoras seleccionadas pertenecen en su mayoría a los segmentos más bajos de compra, principalmente del tipo inconstante(que compra en promedio cada 3 campañas) y nuevas, siendo un total del 64 % del experimento.

Se aplica entonces el MDO durante una campaña *neutra*, es decir, no asociada a ninguna festividad para no influenciar el comportamiento de compra. Se espera que los resultados den una muestra de las debilidades del motor con respecto a los segmentos donde es fundamental impulsar la venta.

Al evaluar los resultados del piloto se observa lo siguiente:

- El ratio de conversión aumentó en un 5.8 % con respecto al otro algoritmo (14.6 % vs 8.8 %)
- PPU [2] aumentó en 1.68 soles llegando a los 6.85 soles.
- El PUP[3] disminuyó en 0.42 unidades.
- Como consecuencia directa de los puntos anteriores, el P$P[4] del canal de oferta aumento en 0.72 soles.

El tiempo de ejecución no fue del todo satisfactorio, ya que para generar ofertas para unas

[1] Es considerado el estándar más preciso e inequívoco para probar una hipótesis.
[2] Precio Promedio por Unidad
[3] Promedio de Unidades por Pedido
[4] Precio Promedio por Pedido

150.000 consultoras tomó un tiempo de poco más de 5 horas. Si bien es un tiempo aceptable para la empresa, se detectaron las partes de la implementación que generaron los tiempo más extensos y se vieron oportunidades de mejora en ellas.

5.1.1. Oportunidades de Mejora

Se analizaron los resultados desde varios ángulos con el fin de obtener la mayor cantidad de posibles alternativas de mejora. Por el lado de la operación del motor, existen partes cuyo cálculo toma un tiempo excesivo para lo esperado, principalmente cruce de tablas no óptimos o rutinas SQL que pudiesen ser cáculadas de forma alternativa para mejorar el performance del algoritmo y aprovechar las ventajas del motor SQL sobre el que está construido (Infobright). El módulo de Modelo de Bundle es el que presenta el mayor tiempo de ejecucion con un promedio de casi 2 horas, debido a los cruces por campos de texto.

En el sentido comercial la idea es apuntar a optimizar el ratio de conversión, que si bien aumentó, aún no cumple con las expectativas, por lo que se plantea realizar mejoras al indicador conjunto de priorización de ofertas, se estudió que no todas las variables son medidas en la misma escala, por lo que es necesario un cambio en el indicador. En la misma línea, se revisa la implementación del cálculo de la pseudo probabilidad de los producto, ya que existen casos con valores muy superiores a 1.

En cuanto a las ventas por formato de ofertas, los resultados relacionados a los sets (bu dles) fueron deficientes (un 32 % del total de las ventas), por lo que se revisan las restricciones de negocio asociadas a este tipo de formato.

5.2. Ajustes y Resultados finales

Recorriendo el MDO desde principio a fin se van implementando mejoras apuntando a incrementar los indicadores comerciales y las métricas de desempeño operativas.

Dado que el motor de base de datos no permite la implementación de índices, se opta por generar tablas intermedias similares a las tablas de dimensiones en un modelo estrella de un data warehouse. La idea es generar un índice (valor único correlativo) por cada cierta combinación de valores, por ejemplo por cada MCT. Además, se detectaron las *queries* que representaban el mayor tiempo de ejecución y, de ser posible, se separaron en sub rutinas de *queries* tal que realicen cálculos más atómicos y que el total del tiempo de ejecución fuese menor que el de la *query* original.

Persiguiendo mejoras en los indicadores comerciales, el primer cambio se realiza en el cálculo de la pseudo probabilidad que asocia un producto con una consultora. Para esto se reescalaron los valores de $\frac{F}{R}$, ya que se encontraban muy dispersos dentro de los valores asociados a cada consultora. El reescalamiento se hizo con respecto a los extremos, es decir con respecto al máximo y mínimo valor(ver 5.1). Además para los casos donde la recencia fuese 0, se determinó que el valor de $\frac{F}{R}$ fuese 1.

$$Factor = \frac{R}{F}^{-1}(prod) \tag{5.1}$$

$$FactorReescalado = \frac{Factor - min(Factor)}{max(Factor) - min(Factor)} \tag{5.2}$$

Siguiendo las etapas del MDO, el próximo paso es revisar las restricciones de negocio aplicadas al modelo de bundle. Se observó que dado el tipo de consultora sobre el que se aplicó el piloto, en su mayoría terminó con ofertar bundles de relleno. Para aumentar la conversión de este tipo de ofertas se decidió eliminar la regla de negocio que obligaba a considerar solo packs de productos donde la consultora haya comprado todos los triggers y no haya comprado el producto recomendado. Con esto se permiten sets donde la consultora haya comprado todos los productos del mismo. La estrategia detrás de esto es fidelizar a aquellas consultoras de baja frecuencia de compra, permitiéndole obtener sus productos en un formato más atractivo.

Si bien la venta del formato volumen e individual fue buena, se observó que en los productos ofrecidos no estaban varios productos *populares* por tener un descuento inferior al ofrecido en el medio impreso. Se decide relajar la regla de negocio y ahora si el descuento en el medio impreso es mejor que el calculado por el MDO, entonces se mantendrá el descuento del medio impreso. Esto sólo si el descuento no supera el límite del 60 %.

El indicador de asignación fue revisado en su concepción, y si bien es cierto hace sentido de negocio, es necesario llevar este indicador a una forma donde sus componentes sean comparables. Se utiliza una técnica que permite considerar todos los elementos que componen el indicador conjuntos y los reescala considerando su dependencia, el reescalamiento de Mahalanobis[11]. Con el resultado de este algoritmo se vuelven a sumar los valores reescalados y se consigue el nuevo indicador de priorización de ofertas.

Una vez implementados estos cambios en el motor, se realiza una nueva prueba para medir el impacto de estos cambios. Nuevamente se considera un grupo piloto y un grupo control, esta vez el tamaño del universo es mayor, considerando 3012 personas en el grupo control y 3015 en el grupo piloto. Las características entre los grupos son similares y nuevamente se aplica la prueba sobre una campaña neutra.

Los nuevos resultados denotaron el impactos de los cambios antes descritos. El primero y más notable es el aumento en el ratio de conversión, pasando de un aumento de 5.8 % conseguido por la primera prueba a un 15.3 % obtenido ahora, completando un 24.1 % de conversión del canal de venta. El PPU no vio gran variación y solo aumento en 0.05 soles con respecto a la prueba pasada, cerrando en un aumento de 1.73 soles. Un punto negativo que se vió en la primera versión fue la disminución del indicador PUP en 0.42 unidades, esta vez el indicador sigue siendo negativo con respecto al grupo control, pero la disminución fue sólo de 0.35 unidades. El P$P cerró finalmente en un aumento de 1.73 soles. Desde la mirada comercial estos resultados son considerados exitosos, aunque se plantea el desafío de mejorar la caída en el PUP.

Por el lado de los tiempos de ejecución, la eliminación de restricciones y la agregación de nuevos cálculos, como el reescalamiento de Mahalanobis, generaron un aumento en los tiempos de cálculo. Sin embargo, las optimizaciones de código realizadas llevaron a completar la ejecución del ciclo

completo del motor en un tiempo cercano a las 3 horas.

Finalmente, por el lado de la empresa, la solución del MDO representó una disminucion en el tiempo de generación de las ofertas de una campaña, bajando de 17 días a sólo 10. Se espera en un futuro integrar la solución al ciclo completo de producción de ofertas, estimando la reducción del tiempo hasta un proceso que se complete en 4 días, esto basado en los diferentes procesos manuales que se requieren para extraer la información para alimentar al MDO, así como el proceso de recepción de los resultados generados.

5.3. Comparación de Algoritmos Recomendadores

A continuación se presenta un(Tabla 5.1) resumen que muestra algunos de los puntos comparables entre la versión existente antes del MDO, así como la primera versión del MDO y la con los ajustes implementados.

Medición	ARP	MDO primera versión	MDO segunda versión
Tiempo de Ejecución	4 dias entre generación y asignación	5 horas	3 horas
Cantidad de ofertas Generadas	500	150.000	150.000
Calidad de Personalización	Sólo en la asignación. Ofertas generadas genéricamente	Generación Personalizada. Asignación no precisa.	Generación Personalizada. Asignación Precisa.
Recursos	Equipo de 4 personas(generadores de las ofertas) más servidor	un servidor	un servidor

Tabla 5.1: Tabla comparativa de el algoritmo preexistente, primera versión del MDO y la versión ajustada.

Capítulo 6

Conclusiones

El trabajo comprendido en esta tesis ha revelado la viabilidad de migrar algoritmos utilizados en el mundo del retail hacia el mundo de la venta directa, detectando similitudes y diferencias.

Entender el proceso de venta e interacción con las ofertas es la parte crucial de todo este desarrollo, el entender la relación de las consultoras con los clientes y el cómo ellas generan sus ganancias determina gran parte de la estructura del motor recomendador.

Paralelamente a esto, se presentaron formas de lidiar con la falta de información para modelar las diferentes etapas de la solución. Métodos de integrar restricciones propias de un negocio donde el cliene final no es directamente alcanzado por la empresa. Comprender la lógica comercial detrás de cada una de las reglas de negocio resulta fundamental para entender el impacto que puede tener en el resultado una implementación errónea ya sea de forma o en el lugar del motor donde se lleve a cabo.

Encontrar soluciones creativas pero efectivas para cubrir cada uno de los casos resulta fundamental. Encontrar oportunidades de utilizar mecanismos existentes en el mercado del retail y ajustarlos al mundo de la venta directa es la clave del éxito de esta tesis.

Hemos constatado que el uso de tecnologías de machine learning logra mejorar tanto la precisión de las ofertas como la calidad de estas, apuntando a la personalización es posible mejorar los indicadores comerciales. La reducción de carga de trabajo al generar mediante algoritmos combinaciones más acertivas y en mayor cantidad permiten liberar tiempo para nuevas oportunidades.

Algoritmos estadísticos demuestran ser un gran complemento a soluciones ya probadas en el mundo del retail. La personalización es el camino para lograr aumentar la conversión de ofertas, mientras que aprovechar las ventajas que ofrece el mundo web permite utilizar herramientas como mapas de calor para conseguir informacón e identificar oportunidades de mejora que no son vistas desde el propio diseño del motor recomendador.

Finalmente se logró el objetivo principal de ésta memoria, el cual era construir una herramienta que permita generar promociones personalizadas a sus consultoras. El objetivo específico de ajustar algoritmos aplicados en el mundo del retail al mundo de la venta directa también fue logrado, así como la implementacion de las reglas de negocio impuestas y se generó un proceso integrado capaz

de funcionar solo con paráametros iniciales. Por el lado comercial, se lograron supurar casi todos los indicadores propuestos.

6.1. Trabajo Futuro

Quedan propuestos varios cambios que podrían impactar de manera positiva los resultados del MDO.

Desde el punto de desarrollo, utilizar un servidor de mayor capacidad permitiría probar algoritmos de machine learning que han demostrado ser más potentes como RandomForest o XGBoost. Así mismo, el obtener las características de los productos podría llevar a determinar probabilidades y cálculo de unidades más precisos, evitando la utilización de algoritmos para generar una distribución a menor nivel.

Por el tipo de cálculos realizados, resulta interesante explorar soluciones de programación paralela, donde se pudiesen ejecutar módulos por separado según perfil o algún otro elemento diferenciador.

Para aumentar el PUP se propone la generación de bundle mezclados con volumen, es decir, 2 unidades de un producto trigger recomiendan 1 unidad de otro producto, por ejemplo. El aumento del ratio de conversión por otro lado, va a tener directa relación con el proceso de digitalización que está sufriendo la empresa. El MDO contempla un primer paso de ayuda a entender las capacidades que ofrece el mundo del machine learning. Dentro de eso, se propone la generación de las otras ofertas mostradas en el sitio web a partir de sub módulos del MDO.

Tener un modelo de Pricing podría ser la mejor solución para el tema del descuento personalizado. Estimar un valor óptimo es más preciso que un descuento relativo. Por otro lado, como módulo adicional al MDO, podría ser de gran ayuda para la empresa contar con un modelo de estimación de unidades a vender, ya que es necesario contar con un stock de productos antes de lanzar las ofertas de una campaña y hoy este cálculo no refleja la nueva realidad que el MDO plantea.

Bibliografía

[1] "Fitting generalized linear models," https://stat.ethz.ch/R-manual/R-devel/library/stats/html/glm.html, [Online; accedida el 23/12/2018].

[2] "Package 'aer'," https://cran.r-project.org/web/packages/AER/AER.pdf, 2018, [Online; accedida el 23/12/2018].

[3] "Package 'arules'," https://cran.r-project.org/web/packages/arules/arules.pdf, 2018, [Online; accedida el 23/12/2018].

[4] "Package 'arules'," https://cran.r-project.org/web/packages/arules/arules.pdf, 2018, [Online; accedida el 23/12/2018].

[5] "Package 'rmysql'," https://cran.r-project.org/web/packages/RMySQL/RMySQL.pdf, 2018, [Online; accedida el 23/12/2018].

[6] "Package 'rpart'," https://cran.r-project.org/web/packages/rpart/rpart.pdf, 2018, [Online; accedida el 23/12/2018].

[7] V. D. J. Casters M., Bouman R., *Pentaho Kettle solutions*, 2010.

[8] M. A. Hartigan, J. A.; Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the Royal Statistical Society,Series C. 28*, vol. 28, no. 1, pp. 100–108, 1979.

[9] D. J., *Big Data, Data Mining, and Machine Learning*, 2014.

[10] M. T. M Khajvand, "Estimating customer future value of different customer segments based on adapted rfm model in retail banking context," *Procedia Computer Science*, vol. 3, p. 1327–1332, 2011.

[11] P. Mahalanobis, "On the generalized distance in statistics," *Proc. Nat. Inst. Sci. India (Calcutta)*, vol. 2, p. 49–55, 1936.

Capítulo 7

Modelos Machine Learning

7.1. Perfilamiento

Para el perfilamiento se utilizó el algoritmo Kmeans de R. Para determinar el número óptimo de cluster se utiliza la técnica del codo.

```
wss <- (nrow(data_mct_norm)-1)*sum(apply(data_mct_norm,2,var))
for (i in 2:7)
    wss[i] <-kmeans(data_mct_norm,centers=i)$tot.withins
plot(1:7, wss, type="b",
        xlab="Number of Clusters",
        ylab="Within groups sum of squares")
```

El resultado de código anterior se muestra en al figura 7.1

Para la aplicación del algoritmo kmeans una vez elegido el número óptimo de cluster(k), se ejecuta el comando:

```
    kmeans_mct=kmeans(data_mct_norm,k)
```

7.2. Propensión

Para modelar se separa la base analítica a partir de la creación de una *llave*, la cual está compuesta por la concatenación de la MCT. Luego a cada uno de los dataset generados a partir de las llaves se les aplica el modelo ctree con la siguiente fórmula:

```
  tree<-ctree(var_target ~ recencia+des_comportamiento_rolling+
  factor(cod_region)+factor(flag_pedido_web)+ppu_venta_catalogo+
  pdp_venta_catalogo+frecuencia+ antiguedad+ mix_productos,
  data = train, control = ctree_control(maxdepth =0))
```

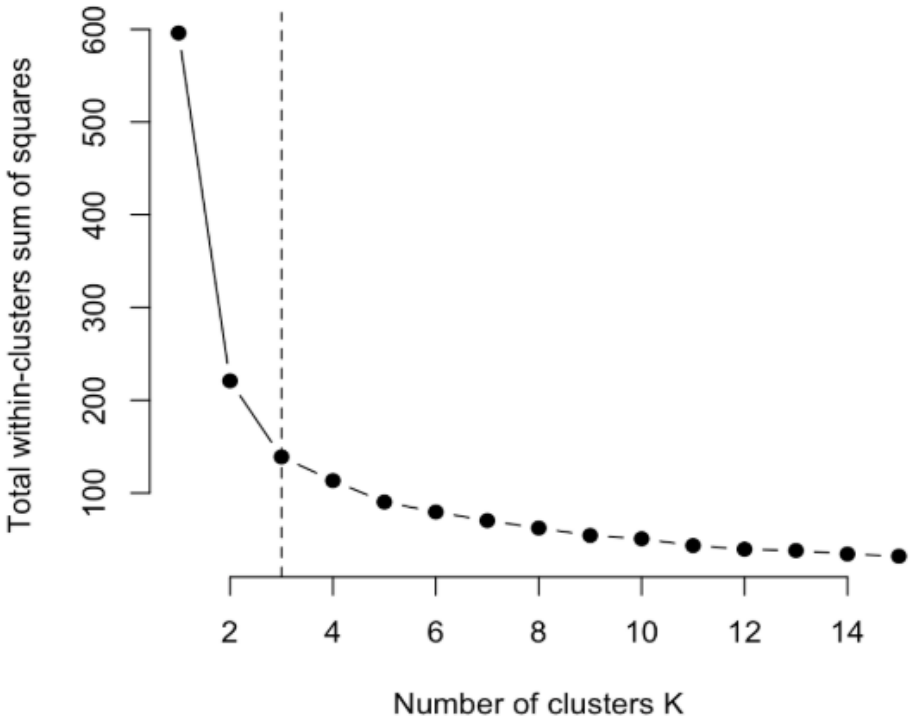

Figura 7.1: Regla de Codo

7.3. Volumen

Para modelar se consideran las siguientes variables por consultora:

- des_comportamiento_rolling: Etiqueta de segmentación utilizada por la empresa.
- unidades_1: Pedido en unidades de la última campaña en la MCT.
- unidades_2: Pedido en unidades de la penúltima campaña en la MCT.
- unidades_3: Pedido en unidades de tres campañas atrás en la MCT.
- unidades_otras_1: Pedido en unidades de la última campaña en las otras MCT.
- unidades_otras_2: Pedido en unidades de la penúltima campaña en las otras MCT.
- unidades_otras_3: Pedido en unidades de tres campañas atrás en las otras MCT.
- unidades_0: Cantidad de unidades que pedirá en la siguiente campaña.

Los modelos se generan con los siguientes códigos:

Árbol de regresión

```
library(rpart)
arbol <- rpart(unidades_0 ~ des_comportamiento_rolling+
unidades_1+ unidades_2+ unidades_3+ unidades_otras_1+
unidades_otras_2+ unidades_otras_3,
data=datos, method = "anova")

arb_pred <- predict(arbol,df)
```

Regressión Tobit

```
library(AER)
tobit <- tobit(unidades_0 ~ 0+des_comportamiento_rolling+
unidades_1+ unidades_2+ unidades_3+ unidades_otras_1+
unidades_otras_2+ unidades_otras_3, data=datos,maxiter=500)

mu <- predict(tobit,newdata=datos,type="response")
sigma <- tobit$scale
p0 <- pnorm(mu/sigma)
lambda <- function(x) dnorm(x)/pnorm(x)
ey0 <- mu + sigma * lambda(mu/sigma)
ey <- p0 * ey0
ey <- round(ey,0)
```

Capítulo 8

Procesos de Kettle

En este anexo se presenta la composición de los diferentes módulos(jobs) en la herramienta de Kettle.

La figura 8.1 representa el job principal, este contiene sub módulos que representan lasa diferentess etapas del MDO. El flujo de flechas verdes representa el orden de ejecución de las diferentes etapas. Dentro de las etapas tenemos los SQL *dproducto_id* y *Update MCT Relevantes*. El objetivo del primero es crear un identificador único a cada combinación de marca-categoria-tipo presentes en la dimensión producto. Esto se hace para optimizar los cruces que se realizan en el proceso, dado que los cruces por campos de texto toman mucho más tiempo que los cruces por campos numéricos. El segundo SQL está para uptaderar el ID de las MCT relevantes que hacen el 80 % de la venta del país.

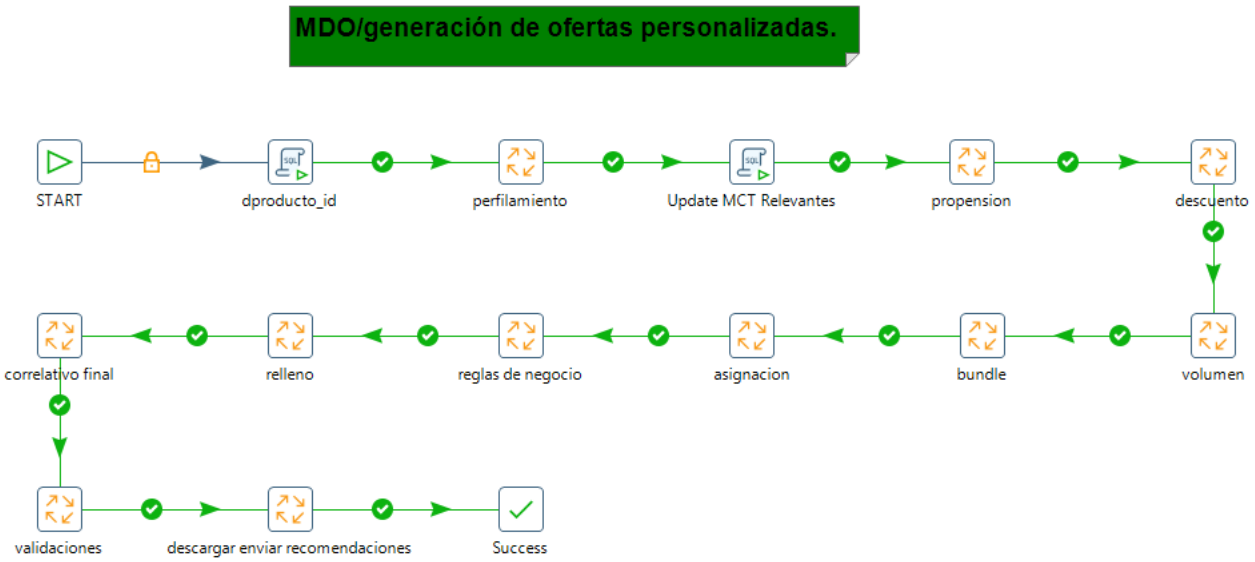

Figura 8.1: Job principal del MDO

Las figuras 8.7, 8.8 y 8.9 corresponden a la implementación del módulo de asignación. Por temas de entendimiento a la hora de explicar a la empresa se decidió separar en 3 jobs.

Figura 8.2: Job de Perfilamiento

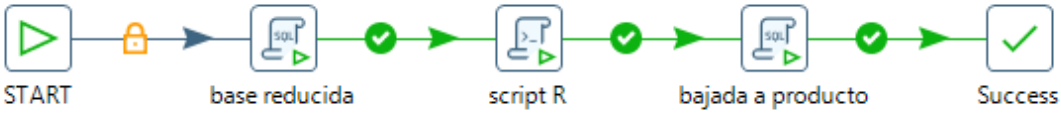

Figura 8.3: Job de Modelo de Propensión de Compra

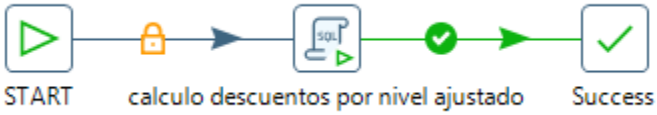

Figura 8.4: Job de Descuento

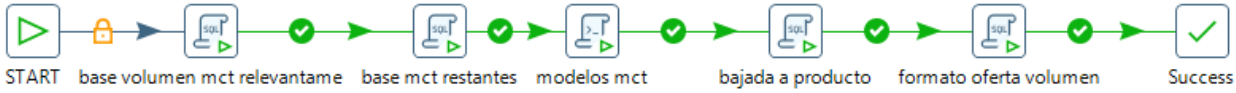

Figura 8.5: Job de Modelo de Volumen

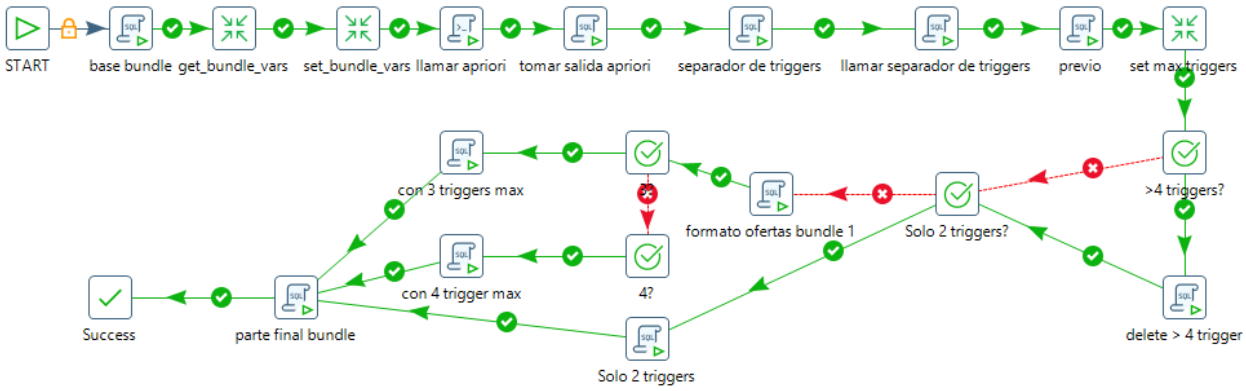

Figura 8.6: Job de Modelo de Bundle

Los job *correlativo final*, *validaciones* y *descargar enviar recomendaciones* corresponde a job operativos con el objetivo de hacer llegar las ofertas a la empresa en el formato acordado.

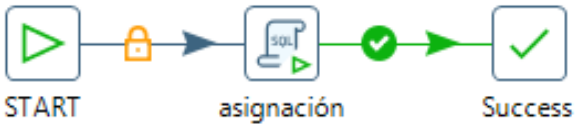

Figura 8.7: Job Primera Etapa de Asignación

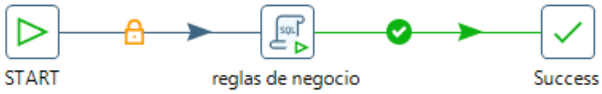

Figura 8.8: Job Segunda Etapa de Asignación

Figura 8.9: Job Tercera Etapa de Asignación