UNIVERSIDAD ADOLFO IBÁÑEZ

FACULTAD DE INGENIERÍA Y CIENCIAS

PROYECTO DE PASANTÍA

COSTOS POR COMPORTAMIENTO DE COBRO CRÍTICOS

POR

CATALINA DEL PILAR CUEVAS PINTO

PROF.: NICOLÁS PÉREZ BRIONES

SEPTIEMBRE 2023

Resumen ejecutivo

En el presente informe se detalla el desarrollo del proyecto Costos por Comportamientos de Cobro Críticos que surge en el contexto de pago de pensiones y beneficios sociales que realiza el Instituto de Previsión Social, IPS. El mecanismo de cobro de tarifas bancarias por parte de la Caja de Compensación y Asignación Familiar Los Héroes (CCAF) en contraste con el comportamiento de cobro de la población beneficiaria, da origen a un dolor que desencadena en una pérdida económica, afectando a los recursos de la Institución. Dicha situación tiene origen en una falta de análisis de información sobre el cobro de beneficios y el desajuste con los datos de Emisión, demostrando la necesidad de profundizar en la problemática para sustentar la toma de decisiones. Para ello, este proyecto busca reducir la brecha desajuste y está orientado a estimar criterios basados en evidencia proveniente de análisis robustos, cualidad de los modelos de Machine Learning. Se presentan las propuestas de solución junto a sus respectivas ponderaciones, las metodologías y medidas de desempeño empleadas para medir el éxito del proyecto. Junto a ello se muestra el diseño de la solución, los resultados de ejecución y se estiman los resultados considerando implicancias económicas y mitigación de riesgos para su implementación.

Abstract

This report details the development of the Costs for Critical Collection Behaviors project that arises in the context of payment of pensions and social benefits carried out by the Social Security Institute, IPS. The mechanism for collecting bank fees by the Los Héroes Family Compensation and Allowance Fund (CCAF), in contrast to the collection behavior of the beneficiary population, gives rise to pain that triggers economic loss, affecting resources. of the institution. This situation has its origin in a lack of analysis of information on the collection of benefits and the mismatch with the Issuance data, demonstrating the need to delve deeper into the problem to support decision-making. To this end, this project seeks to reduce the mismatch gap and is aimed at estimating criteria based on evidence from robust analyses, a quality of Machine Learning models. The solution proposals are presented along with their respective weightings, the methodologies and performance measures used to measure the success of the project. Along with this, the design of the solution, the execution results are shown, and the results are estimated considering economic implications and risk mitigation for its implementation.

Conceptos clave

Beneficio: Aporte económico que entrega el Estado a la ciudadanía y que está normado por Ley. Puede ser un beneficio en régimen o regímenes especiales.

Beneficios en régimen: Aquellos beneficios fijos que, al momento de acceder, se tiene derecho de recibir periódicamente.

Redes: Clasificación de los distintos servicios de atención y pago de beneficios que responde a los contratos establecidos con los prestadores.

Forma de pago: Modalidad asociada al servicio elegido por el usuario para materializar la entrega de su beneficio, alineada a las Redes existentes.

Emisión: Información que se entrega a las entidades pagadoras por período que incluye todos los datos relevantes para efectuar los pagos.

Rendición: Información que entregan las entidades pagadoras a la Institución luego de pagar los documentos de un período.

Servicio de Pago Rural: Corresponde a un servicio de pago entregado por CCAF, que consiste en una serie de rutas móviles hacia zonas específicas para acercar el pago de beneficios a la población.

Sucursal de Cabecera: Corresponde a la sucursal desde la que parte una ruta móvil. Acá se rinden los documentos pagados y se dejan disponibles los pagos no efectuados en ruta.

Cobro en ruta: Comportamiento de cobro en que la persona beneficiaria se dirige a cobrar en el punto de pago rural.

Cobro fuera de ruta: Comportamiento de cobro en que la persona beneficiaria con Forma de Pago Rural se dirige a cobrar en sucursal.

Introducción

El Instituto de Previsión Social (IPS) es un organismo público del Estado de Chile que administra los aportes económicos del sistema de pensiones solidarias y regímenes especiales, gestionando la entrega de estos beneficios a la población beneficiaria. Parte de su cadena de valor se concreta mediante el servicio de pago masivo concedido a entidades externas por medio de contrataciones periódicas. A día de hoy, la Institución realiza más de cinco millones de pagos mensuales, entre los cuales más de tres millones corresponden a pagos de beneficios en régimen.

La población que recibe estos beneficios tiene una Forma de Pago asignada (Anexo 1), ya sea de modalidad electrónica o presencial, y cada contratación presenta una comisión respectiva. El servicio de pago presencial está contratado por Caja Los Héroes (CCAF) y BancoEstado, sin embargo, solo el primer prestador presenta más de un tipo de servicio presencial, tarifas diferenciadas, y un mecanismo de cobro específico por Forma de Pago. Esta situación provoca que:

- Exista interacción entre las distintas Formas de Pago y lugares de cobro.
- Se cobre más en ciertas ocasiones perjudiciales para el IPS.
- Se requiera actualizar la información para el pago de beneficios.

En estos casos, se dan las siguientes situaciones:

Caso Crítico	**Emisión (Datos para el Pago)**			**Rendición (Datos de Cobro)**	
N°	Servicio	Red de Emisión	Forma de Pago	Servicio	Red de Cobro
1	Pago Rural	Red 3	FP2	Pago Sucursal	Red 1
2	Pago Rural	Red 3	FP2	Pago Sucursal	Red 4
3	Pago Sucursal	Red 4	FP7	Pago Sucursal	Red 1

Tabla 1: Casos Críticos.

La dimensión e impacto de estos Casos Críticos en un período es el siguiente:

Caso Crítico	**Ocurrencias (Q)**	**Costos**
1	9.124	(8.801-2.916) · Q = $53.694.740
2	862	(8.801-8.498) · Q = $261.186
3	4.430	(8.498-2.916) · Q = $24.728.260

Tabla 2: Casos críticos durante junio 2023.

Esto se produce debido a que las Formas de Pago del Servicio Presencial de CCAF tienen asociadas distintas tarifas, de modo que, al momento en que los beneficiarios de Pago Rural se cobran en sucursal o los de Pago en Sucursal de la Red 4 se cobran en una sucursal de la Red 1, se produce un desbalance entre las tarifas cobradas y las que "deberían" cobrarse. Ahora bien, al estar las tarifas establecidas por contrato, entendemos que los Casos Críticos en realidad consisten en un desajuste entre las Redes de Emisión (o FP) con la Rendición (Cobro). Dicha brecha da origen a este proyecto para comprender los casos a cabalidad y tomar acciones que permitan mejorar la gestión pública.

Esta problemática tiene aristas como la falta de personal dedicado a este análisis, ausencia de sistemas de medición y estudio del Comportamiento de Cobro de manera regular, como también de un procedimiento establecido que relacione las variables implicadas. Por otro lado, en cuanto a la oportunidad, el IPS cuenta con información interna y externa sobre la población beneficiaria a la que se entrega beneficios. Esta es un activo subutilizado y habilita la posibilidad de ahondar en tendencias de cobro, evolución y proyecciones estadísticas.

Objetivos SMART

El objetivo general del proyecto es reducir el desajuste en las Redes de Emisión por casos críticos en un 70%, antes de enero de 2024. Este objetivo busca promover la excelencia en la gestión en cuanto a pago de beneficios, en conformidad a la misión y objetivos estratégicos de la Institución. El abordaje a esta problemática significa, además, un gasto eficiente de los recursos públicos en la medida en que la solución sea capaz de proponer reajustes de las redes para reducir los costos.
Para alcanzar el objetivo general se establecieron los siguientes objetivos específicos:

- Objetivo específico 1: Identificar tendencias en los distintos comportamientos de cobro.
- Objetivo Específico 2: Identificar y definir variables relevantes dentro de los casos.
- Objetivo Específico 3: Establecer criterios de ajuste para los distintos casos.

Estado del Arte

Para comenzar, cabe mencionar que actualmente la medida tomada por parte del negocio frente a esta problemática ha sido un ajuste en las Redes de Emisión de los documentos de pago con distintos criterios dependiendo del caso crítico, lo que refleja un cambio en la Forma de Pago de los documentos. Estas se basan en la pérdida económica que se genera y consideran un horizonte de tiempo relativo al servicio de Pago Presencial.

Considerando que para el Pago en Sucursal un beneficiario puede continuar cobrando en cualquier oficina, independiente de si fue modificada su Forma de Pago de FP7 a FP1 (Caso Crítico n°3), se estima que este ajuste es una decisión transparente con el usuario y es apropiado para amortiguar la pérdida económica. Ahora bien, para el Pago Rural se deben considerar distintas variantes.

En primer lugar, las condiciones de este último servicio son distintas al contexto de las sucursales. Modificar la Forma de Pago desde el servicio de Pago Rural a Sucursal tiene un riesgo mayor, puesto a que, si un beneficiario desea volver a cobrar en el punto rural luego de haber modificado su FP, ya no estará disponible su pago en este servicio, sino que lo estará en cualquier sucursal del Pagador. Es por esto que se debe tener certeza de los casos en que corresponda ajustar la Red de Emisión, considerando un horizonte de tiempo mayor, para lo que se hace relevante acceder al comportamiento de cobro histórico de estos beneficiarios.

Como parte del estudio del comportamiento, la literatura menciona que existen distintas capacidades analíticas que entregan información a partir del análisis de datos. Por un lado, la capacidad descriptiva "representa la base de cualquier tipo de análisis de datos y, por ende, ningún proceso ni herramienta puede considerarse como íntegra sin ella." (Maldonado & Vairetti, 2022), de modo que responde a la pregunta de "¿qué sucedió?". Es por ello que, independiente de ser un análisis más simple, es indispensable para luego realizar estudios más elaborados. Por otro lado, la capacidad de diagnóstico o diagnostic analytics ayuda a responder "por qué sucedió". Esta capacidad permite revisar los datos para encontrar tendencias y relaciones causales más sofisticadas, de la misma forma en que el análisis o capacidad predictiva, habilita crear modelos que, mediante la clasificación, permitan extrapolar el análisis previo hacia el futuro, de modo que, "asumiendo que el pasado es un buen predictor del futuro, podemos inferir comportamientos de clientes u otros." (Maldonado & Vairetti, 2022).

Tomando esta información, se considera que la capacidad descriptiva entregará información relevante respecto del comportamiento de cobro, como, por ejemplo, cuánto tiempo una persona cobra fuera de ruta, y, por otro lado, la capacidad de diagnóstico y predictiva permitirá tomar decisiones asertivas para establecer criterios de ajuste, una vez que se logre establecer relaciones entre el análisis de la información histórica y la posibilidad en que esto se replique en períodos sucesivos. En este contexto, se pueden sugerir distintos planes de acción dependiendo de los casos.

Algunos casos presentes en la literatura para los que se aplica Ciencia de Datos tienen que ver con predicción de fraudes, riesgos de fuga de clientes y deudores de crédito. Estos casos son similares a la problemática actual mientras que se tengan dos tipos de datos: (1) características de los individuos y (2) la probabilidad de que estos cometan fraudes, se fuguen o no paguen un crédito. De tal forma, las primeras variables se pueden utilizar para determinar las segundas. Un estudio de la detección de fraude menciona la existencia de distintas fuentes de datos, como, por ejemplo, características socioeconómicas (edad, sexo, estado civil, nivel de ingresos, etc.) y cómo estas "pueden relacionarse significativamente con comportamientos fraudulentos." (Baesens et al., 2015). Esto evidencia la posibilidad de que este tipo de características puedan modelar otro tipo de tendencias de comportamiento.

En cuanto al análisis predictivo, se dan ejemplos de diversos modelos como: Regresiones lineales, Árboles de decisión, Redes neuronales, Support Vector Machine, Series de tiempo, entre otras. También se dice que "los modelos de Machine Learning tienen un muy buen historial de uso como modelos predictivos", además de que "según los parámetros de entrada, se puede predecir el futuro de cualquier valor. Ahora con el avance en el campo del aprendizaje automático (...) hay una tendencia hoy en día del uso de modelos de aprendizaje profundo en predicción." (Kumar & Garg, 2018).

En el mismo contexto de Data Science y Machine Learning, se distinguen los modelos de Aprendizaje Supervisado y No Supervisado. Por un lado, los primeros "se basan en entrenar una muestra de datos" considerando una variable a predecir y los segundos en "la capacidad de obtener aprendizaje y organizar la información sin proporcionar una señal de error a evaluar" (Sathya & Abraham 2013). De tal forma, ambos tipos de modelos pueden entregar información valiosa. En este caso, para enfocar la solución puede ser más conveniente utilizar el Aprendizaje Supervisado ocupando como foco el tipo de comportamiento de la población objetivo.

Concluyendo, a partir de las diversas fuentes de datos a las que se tiene acceso se puede obtener información de características propias de los beneficiarios, como también de su comportamiento de cobro, como frecuencia, lugar de cobro, y si se puede categorizar dentro de algún Caso Crítico. Para ello se hace necesario considerar el comportamiento histórico dentro de un horizonte de tiempo que sea representativo. Es por esto que se pueden revisar las tendencias por período para determinar una muestra que no altere el modelo.

En cuanto a la necesidad de muestreo, en el contexto de detección de fraude se señala que "(...) las transacciones de hoy son más similares a las transacciones de mañana que a las transacciones de ayer. La elección de la ventana temporal óptima de la muestra implica un equilibrio entre una gran cantidad de datos (y, por tanto, un modelo analítico más sólido) y datos recientes (que pueden ser más representativos)." (Baesens et al., 2015).

Por ende, si consideramos que el negocio utiliza como referencia los últimos 6 meses de cobro en el contexto de Pago Rural, para tener un primer acercamiento del comportamiento debemos tener un mínimo de muestra de este tamaño. Además, como se señala en el estudio de fraudes, cabe considerar "un período comercial promedio" lo que puede asimilarse a un año de pago de beneficios, de manera en que se generen la menor cantidad de sesgos posibles.

Propuestas de solución

En base al análisis comentado en el Estado del Arte y ajustándolo al contexto del negocio, se establecen las siguientes propuestas de solución junto a sus respectivos criterios de selección:

Solución / Criterio	Precisión del tiempo de ajuste	Necesidad para amortiguar costos	Tiempo de ejecución	Ponderación
Modelo de Aprendizaje Supervisado Caso Crítico n°1 y n°2	6	6	6	18
Modelo de Aprendizaje Supervisado Caso Crítico n°3	6	4	6	16
Modelo de Aprendizaje No Supervisado	3	3	6	12

Tabla 3: Ponderaciones de las propuestas de solución en base a los criterios seleccionados.

La primera propuesta consiste en aplicar un modelo de Aprendizaje Supervisado que permita clasificar a los beneficiarios con Forma de Pago Rural (Caso Crítico n°1 y n°2), y establecer como Función Objetivo el cobro fuera de ruta en los últimos 6 meses en un 100% de las instancias.

La segunda propuesta consiste en aplicar el mismo tipo de modelo para los beneficiarios con Forma de Pago Presencial en Sucursal (Caso Crítico n°3), de tal forma en que se pueda predecir si un beneficiario con Red de Emisión 4 cobrará en una sucursal de la Red 1, sin embargo, la predicción tiene el riesgo de no ser totalmente precisa, debido a que desde el punto de vista de los beneficiarios todas las sucursales son similares, por ende la tendencia de cierto grupo de cambiarse a una sucursal de pago de otra Red tiene un mayor sesgo. Además, la decisión de ajuste de las Redes de Emisión que permita disminuir la pérdida económica de este caso no requiere de un estudio al nivel de un modelo de Aprendizaje Supervisado.

Finalmente, la tercera propuesta es aplicar Aprendizaje No Supervisado para agrupar la población con similares características. Esta solución permite interpretar los distintos segmentos para tomar decisiones, sin embargo, solo serviría para caracterizar este grupo más que para diferenciarlos a unos de otros.

Solución escogida

En base a la relevancia de utilizar modelos de Machine Learning, se ha tomado la decisión de determinar si existe una relación entre las características de los beneficiarios y su comportamiento de cobro dentro del Caso Crítico n°1 y n°2 que representan el servicio de Pago Rural. Esta decisión se tomó en conjunto de una parte interesada dentro del negocio utilizando una escala de 1 a 3.

Con esta propuesta se desea saber si es posible anticiparse al comportamiento de una clase de beneficiarios en base al estudio de datos históricos. Frente a esto se requiere contar con una Función

Objetivo que se pretende predecir, tomando en cuenta que esta es una variable dependiente que responde a la pregunta: "¿Cobró fuera de ruta en los últimos 6 meses?". Si la respuesta es positiva, la Función Objetivo toma el valor de 1, de lo contrario, toma el valor de 0. Esto permite clasificar grupos con diferente probabilidad de tomar un valor positivo en la variable a predecir.

Por otro lado, los riesgos y mitigaciones para esta solución son los siguientes:

#	Riesgo	Mitigaciones
1	Problemas en el código	Validación de variables riesgosas en el código
2	Muestra no representativa	Variación del muestreo de la Función Objetivo
3	Cobro en ruta en un período posterior	Mantener información actualizada respecto del comportamiento de cobro
4	No lograr contactarse con un beneficiario	Dejar en lista de espera y volver a contactar
5	Imprecisión del modelo	Utilizar datos de entrenamiento y testeo para validación; variar/crear nuevos atributos

Tabla 4: Riesgos de la solución escogida y mitigaciones.

Evaluación económica

Esta solución se desarrolla con Python e incluye información interna y externa. Para su ejecución e implementación se requiere del acceso a los datos correspondientes, a la herramienta Jupyter de Anaconda (o Google Collab en su reemplazo) y de personal capacitado. Los costos asociados a los primeros dos requerimientos son nulos o marginales, sin embargo, a nivel de personas es necesario contar con un profesional responsable de sostener esta implementación permanentemente. Se estima que el proyecto puede tomar un promedio de horas equivalente a una semana al mes para la actualización periódica si es que se desea aplicar modelamiento predictivo (solución completa), por ende, considerando un sueldo entre $1.600.000 a $2.000.000, la inversión varía entre $400.000 y $500.000 mensualmente. En cambio, si se desea mantener los resultados del modelo durante cierto período y únicamente actualizar la información de entrada, se requiere alrededor de 2-3 días al mes lo que es equivalente a $250.000 al mes.

Por otra parte, con el primer avance de la solución que modela el comportamiento de cobro, se tienen 720 beneficios que se cobraron en una Sucursal distinta a la Cabecera en los últimos 6 meses. Estos casos son potenciales para un ajuste en las Redes de Emisión y representan una posibilidad de ahorro de alrededor de $4.200.000 para el IPS. Por otro lado, existen 2.444 pagos que se cobraron en las Sucursales de Cabecera durante los últimos 12 meses, lo que implica un posible ahorro de $14.300.000 dependiendo de los criterios a escoger.

Metodologías

La primera parte consiste en la creación de nuevas variables y de una metodología definida que permita obtener el comportamiento de cobro de la población de estudio en el contexto de Pago Rural. Para ello, se trabaja con lenguaje de programación Python dada su versatilidad, variedad de librerías que habilitan desde análisis sencillos, como Análisis Exploratorio de Datos, hasta la ejecución de modelos de Machine Learning.

Para la obtención del comportamiento de cobro se toman en cuenta los beneficios en régimen, puesto que permiten obtener información histórica dada su continuidad, y se trabaja con los archivos de Emisión (datos para el pago) y Rendición (datos de cobro).

La implementación consistió en:

1. Recopilar Emisión y Rendición del período n, n+1 y n+2 (dado a que un beneficio se puede cobrar en el mes emitido y en los dos siguientes).
2. Filtrar por Forma de Pago Rural.
3. Utilizar la función pd.merge() de pandas de tipo 'left' cuya llave es el Rut, beneficio, número de inscripción y número de documento. Esto permite obtener los beneficios emitidos y rendidos por período.
4. Crear una función que establezca una revisión del tipo de movimiento (pagado, trasladado u otro), lugar de pago, lugar de cobro y tardanza en días hábiles en que se ingresó el pago.
5. Mediante el análisis de las variables mencionadas, se crea la variable COBRO cuyos valores son: {0: No Cobro; 1: Cobro en Ruta; 2: Cobro en Sucursal Cabecera; 3: Cobro en otra sucursal}.
6. Replicar lo anterior para la cantidad de períodos de interés (12 para este caso).
7. Ejecutar Matriz de Cobro Histórico y cálculo de la Función Objetivo.

La segunda parte consiste en el procesamiento de las variables independientes. Para su implementación se realizó lo siguiente:

1. Recolección de datos a partir de distintas fuentes (Emisión, Rendición, Registro Civil y Registro Social de Hogares).
2. Utilizar la función pd.merge() de pandas considerando como base los Rut presentes en la Matriz de Cobro Histórico, para luego unirlas por el Rut con los datos de las fuentes mencionadas.
3. Revisión de VP/I (Valores Perdidos e Inconsistencias) como parte del Preprocesamiento sugerido por la literatura.
    a. En base a esto, si los valores perdidos o faltantes están entre 2-5% se reemplaza por alguna MTC como la media, mediana o moda en caso de las variables categóricas. Si estos están entre 20-30% se sugiere realizar imputación múltiple (no hubo casos). Y si son mayores de 50% se elimina la variable.

    b. Las inconsistencias se trabajan con criterios Ad hoc, o bien, se consideran como un valor vacío.
4. Se evalúa la posibilidad de crear nuevas variables relevantes y se transforman las variables categóricas para acotar categorías (ej. Pasar de 300 comunas a 4 zonas).
5. Transformar variables categóricas a binarias (ej. SEXO = F:1; M:0).
6. Cálculo p-estadístico para selección de atributos relevantes.
7. Realizar normalización de variables numéricas dejando todas entre 0 y 1.
8. Ejecución de modelos de Aprendizaje Supervisado: Regresión Logística, Árboles de decisión, K-nearest neighbors, Random Forest, y Red Neuronal.
9. Medir: Accuracy y Recall (sensibilidad).

Al finalizar esta tarea se tiene una noción de los atributos más relevantes para el análisis, el modelo con mejor capacidad predictiva y, por ende, información relevante para la toma de decisiones en cuanto a criterios de ajuste.

Medidas de desempeño

KPI Objetivo Específico 1:

- Logrado/No Logrado (Cualitativa).

Creación de variables relevantes que modelen el comportamiento de cobro, identificar tendencias en las variables existentes y creadas, y utilizar Series de Tiempo, Análisis Univariado, Bivariado y Análisis Exploratorio de Datos.

KPI Objetivo Específico 2:

- Cálculo p-estadístico ($p < 0.05$).

Cálculo del valor de p-estadístico para determinar la relevancia de atributos que contribuyen al modelamiento del comportamiento de cobro.

Esto se realiza utilizando:

```
logit_model=srm.Logit(y_train_n, srm.add_constant(X_train_n))

result=logit_model.fit()

print(result.summary2())
```

Donde:

y_train_n, variable dependiente de entrenamiento.

X_train_n, variables independientes de entrenamiento.

KPI Objetivo Específico 3:

- Cálculo: *Accuracy y recall.*

Accuracy: Ac = model.score(X_test, y_test).

Donde:

X_test, variables independientes de testeo.

y_test, función objetivo de testeo.

Accuracy consiste en el total de valores que fueron predichos correctamente por el modelo. Otra manera de calcularlo es:

$$Accuracy = \frac{VerdaderosPositivos + VerdaderosNegativos}{VerdaderosPositivos + FalsosPositivos + VerdaderosNegativos + FalsosNegativos}$$

Recall: R= recall_score(y_test, Y_pred)

Recall, por otra parte, consiste en los valores positivos reales que fueron predichos por el modelo. También se calcula como:

$$Recall = \frac{VerdaderosPositivos}{VerdaderosPositivos + FalsosNegativos}$$

Donde:

y_test, función objetivo de testeo.

Y_pred, función objetivo de predicción.

Desarrollo de la solución

El primer producto de esta solución consiste en un código programado y adaptado a las restricciones del negocio. Este está almacenado en un cuadernillo de Jupyter considerando todas las variables anteriormente mencionadas junto a la respectiva descripción (Anexo 2). Además, se creó la variable DIAS_HABILES que consiste en la tardanza en días hábiles en que se ingresa el movimiento de un pago, dato relevante para luego construir el atributo COBRO, que representa el comportamiento de cobro. Por otra parte, se revisó la evolución del Cobro en Ruta y Fuera de Ruta utilizando Series Temporales (Anexo 3) para determinar si existe algún patrón relevante, y además se visualizaron ciertas

variables utilizando análisis univariado y bivariado, como, por ejemplo, la variable MONTO (Anexo 4), EDAD (Anexo 5) y REGION tanto para el Cobro en Ruta como Fuera de Ruta.

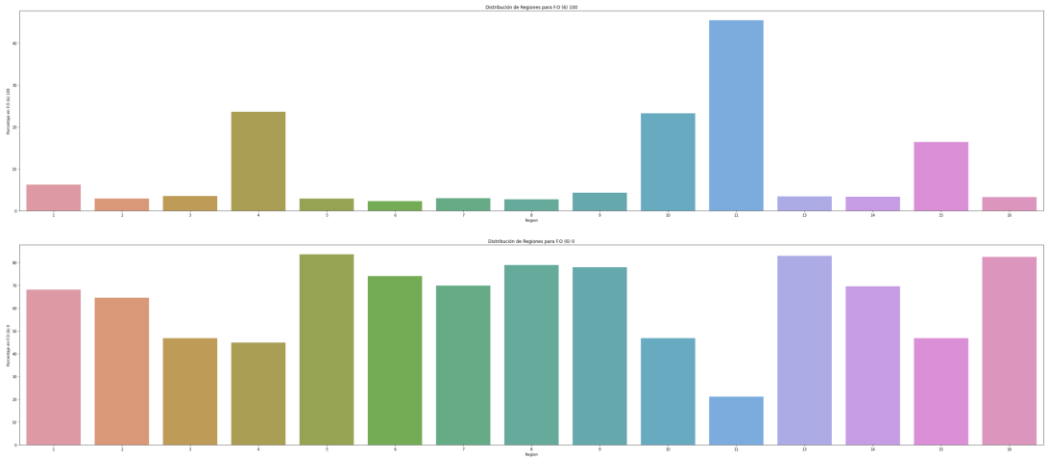


Imagen 1: Arriba se ve un análisis bivariado que relaciona las Regiones de Chile (1 a la 16) con la Función Objetivo de Cobro Fuera de Ruta en los últimos 6 meses. Abajo se ve el mismo análisis para el Cobro dentro de Ruta.

Se ve que la Región 10 y 11 correspondientes a la zona AUSTRAL tiene alta relación con la Función Objetivo, es por esto que se cambió el valor de zona NORTE, SUR, CENTRO y AUSTRAL por la variable AUSTRAL (1: Sí; 0: No), pues para el modelamiento se sugiere homogeneidad en los valores de los atributos y se determinó posteriormente que la variable AUSTRAL es un atributo relevante en el análisis.

Por otro lado, se transformaron las variables SEXO, BENEFICIO, PUNTAJE, ESTADO CIVIL y APODERADO a variables categóricas binarias, y se dejó como variables numéricas MONTO y EDAD.

La segunda parte de la solución correspondiente al preprocesamiento implica tanto la limpieza y transformación de atributos como la ejecución posterior de modelos de Aprendizaje Supervisado. Previo a ello se verificó que los valores de p-estadístico fuesen menores a 0.05 (Anexo 6), y luego se utilizaron los modelos de: Regresión logística, Árboles de decisión, K-nearest neighbors, Random Forest y Red Neuronal (Anexo 7). Cabe mencionar que, entre estos, los últimos tres requieren normalizar las variables numéricas. Finalmente, se calcula los valores de Accuracy y Recall para verificar la precisión del modelo predictivo.

Ahora bien, en cuanto a la implementación del proyecto, esta se divide en el diseño de la solución y ejecución del Comportamiento de Cobro y Modelos de Machine Learning, y en la entrega de esta solución a las partes interesadas del negocio. Para el traspaso de la solución se realizará una reunión inicial de presentación del proyecto, considerando objetivos, alcances y plazo de ejecución, hacia quienes serán los encargados de implementarla posteriormente. Se formalizarán los requerimientos (datos con sus respectivas fuentes, herramienta Jupyter, etc.), configuración (para asegurar su

replicabilidad en otros ambientes) y capacitaciones, con al menos una iteración de cada una, lo que se traduce en una prueba inicial, final y posibles pruebas intermedias de ser necesarias, para finalizar con la puesta en producción que da paso al cierre oficial del proyecto.

Resultados

Luego de haber desarrollado la solución se tienen las siguientes tendencias por comportamiento de cobro:

C. Cobro / Función Objetivo	Cobro Fuera de Ruta: Mixto (2,3)	Cobro Fuera de Ruta: Sucursal Cabecera (2)	Cobro Fuera de Ruta: Otra Sucursal (3)	Total
F.O. (6 meses)	216	3308	812	4336
F.O. (12 meses)	300	2871	597	3768
F.O. (18 meses)	460	2310	444	3214

Tabla 5: Clasificación del Comportamiento de Cobro.

Estos valores consideran todos los casos en que se cobraron los beneficios, pero sin utilizar el servicio designado en la Forma de Pago. Este dato debiese reflejar el método de pago preferido por los beneficiarios y requiere de actualización para disminuir los costos y reducir el desajuste estadístico.

Por otro lado, al haber medido la efectividad de los modelos, se tienen los siguientes valores de Accuracy y Recall:

Modelo	**Accuracy**	**Recall**
Regresión Logística	0.81	0.44
Árbol de decisión	0.83	0.42
K-nearest neighbors	0.92	0.03
Random Forest	0.80	0.45
Red Neuronal	0.82	0.43

Tabla 6: Medición de Accuracy y Recall para cada modelo.

Como se ve, el mayor valor de Accuracy lo tuvo el modelo K-nearest neighbors o K-vecinos más cercanos, con un valor de 0.92, lo que significa que el 92% del total de las predicciones realizadas por el modelo son correctas, sin embargo, el valor de Recall es significativamente bajo en comparación al resto de los modelos, lo que quiere decir que en realidad Knn no es el mejor predictor para esta problemática.

Si se realiza una ponderación, el resto de los modelos (excluyendo K-nearest neighbors) predicen de manera muy similar puesto a que cuentan con resultados cercanos entre sí. Sin embargo, el valor de Recall es muy bajo, por ende, estos últimos modelos tampoco son buenos predictores.

Visto lo anterior, si bien las variables con las que se cuenta actualmente no determinan la Función Objetivo de manera consistente, se recomienda considerar parte del análisis como un apoyo en la toma de decisiones.

Los criterios de ajuste sugeridos son de 6 meses, dado que la diferencia entre los 6 y 12 meses de Cobro Fuera de Ruta no es muy significativa, sin embargo, para llegar a este criterio se recomienda partir en el siguiente orden de prioridad:

#	Criterio de ajuste
1	Cobro Fuera de Ruta Otra Sucursal (18 y 12 meses)
2	Cobro Fuera de Ruta Mixto (18 y 12 meses)
3	Cobro Fuera de Ruta Sucursal Cabecera (18 y 12 meses)
4	Cobro Fuera de Ruta Otra Sucursal (6 meses) + Zona Austral
5	Cobro Fuera de Ruta Mixto (6 meses) + Zona Austral
6	Cobro Fuera de Ruta Sucursal Cabecera (6) + meses Zona Austral

Tabla 7: Criterios de ajuste potencial.

Estos criterios están definidos en base a variable como la distancia inferida del COBRO de valor '3'. Además, considerar las Rutas en la zona Austral como una clase de factor de riesgo permite poner atención a la gestión de los pagos en dicha zona, como también al comportamiento de los residentes de tal lugar, dada la relación notoria la Función Objetivo y dicha zona, y en segundo lugar la Región de Coquimbo.

Para finalizar, cabe mencionar el reajuste de las Formas de Pago debe pasar por el aviso y consentimiento de la población involucrada, por ende, el entregable de este proyecto corresponde a una lista de beneficiarios cuya FP es potencialmente ajustable.

Conclusión

Gran parte de este análisis se basó en el estudio del Libro *Analytics y Big Data* y sus laboratorios de Data Science aplicada a los negocios. Tanto las Series Temporales, como los Análisis Univariados y Bivariados son resultado del estudio de la aplicabilidad de la Ciencia de Datos, por lo que, independiente del resultado del modelamiento, se extrajeron conclusiones relevantes a partir del resultado preliminar.

En base a la primera parte de la solución se tiene una noción mucho más clara del Comportamiento de Cobro. Cabe destacar que todos los datos que entrega la variable COBRO son relevantes y pueden ser utilizados para otro tipo de estudios, o bien, complementados con otro tipo de variables. Por otra parte, el cumplimiento del objetivo general queda sujeto a gestiones internas en temas de contactabilidad, no obstante, es de utilidad aplicar estudios con lógica similar a otras Formas de Pago cuyos criterios de ajuste son menos restrictivos, de modo en que se pueda estabilizar el nivel de consistencia de la información en la Emisión con el Cobro de beneficios.

Referencias

Baesens, B., Van Vlasselaer, V., & Verbeke, W. (2015). Fraud analytics using descriptive, predictive, and social network techniques: A Guide to Data Science for Fraud Detection. John Wiley & Sons.

Kumar, V., & Garg, M. L. (2018). Predictive Analytics: A review of trends and techniques. International journal of computer applications, 182(1), 31-37. https://doi.org/10.5120/ijca2018917434.

Maldonado, S., Vairetti, C. (2022). Analytics y Big Data: Ciencia de los Datos aplicada al mundo de los negocios. RIL Editores.

Sathya, R., & Abraham, A. (2013). Comparison of supervised and unsupervised learning algorithms for pattern classification. International Journal of Advanced Research in Artificial Intelligence, 2(2). https://doi.org/10.14569/ijarai.2013.020206.

Anexos

Anexo 1: Tabla de Formas de Pago.

Forma de Pago	Servicio	Pagador
1	Pago Presencial (Red 1)	Caja Los Héroes
2	Pago Rural (Red 3)	Caja Los Héroes
3	Orden de Pago en CAPRI	BancoEstado
4	Orden de Pago a domicilio	BancoEstado
5	Pago Electrónico	BancoEstado
6	Pago en el Extranjero	Banco Scotiabank
7	Pago Presencial (Red 4)	Caja Los Héroes
9	Pago Presencial (Red 2 y 5)	BancoEstado

Anexo 2: Código Comportamiento de Cobro en Python (Primer Producto).

```
import pandas as pd
import os
import numpy as np
from datetime import date
import time

# Registra el tiempo de inicio
start_time = time.time()

#Ruta carpetas
carpeta_emision = 'C:/Users/56942/OneDrive - Chileatiende/Escritorio/IPS/Pagos/OneDrive_2023-08-21/Proyecto Comparatiivo Emisión-
carpeta_rendicion = 'C:/Users/56942/OneDrive - Chileatiende/Escritorio/IPS/Pagos/OneDrive_2023-08-21/Proyecto Comparatiivo Emisió

#Cambiar nombres a las columnas de las rendiciones para cruzar con la emisión
nuevos_nombres = {'instit': 'INST_COD','rut': 'RUT_BENEF','num_doc': 'NUM_DOC','tipo_mov': 'TIPO_MOV',
                  'cod_plaza_pago': 'COD_PLAZA_PAGO','num_inscripcion': 'NIS','fecha_mov': 'FECHA_MOV',
                  'mon_doc': 'MON_DOC','tipo_pension': 'TIPO_PENSION','hora_mov': 'HORA_MOV','fecha_pago_emi':'FECHA_PAGO_EMI',
                  'codins': 'INST_COD','numins':'NIS','ngrupa':'COD_AGRUPACION','tippen':'TIPO_PENSION',
                  'orileg':'ORIGEN_LEGAL','cilp':'CILP','forpag':'FP','nomben':'NOM_BENEF','nomapo':'NOM_APOD',
                  'rutben':'RUT_BENEF','dvrben':'DV_BENEF','rutapo':'RUT_APOD','dvrapo':'DV_APOD',
                  'frahor':'FRANJA_HORARIA','monliq':'MONTO_LIQ','codban':'PAGADOR','codsuc':'PLAZA_PAGO',
                  'numdoc':'NUM_DOC','fecpag':'FEC_PAGO','ciudad':'CIUDAD','numben':'NRO_BENEF','codmun':'COMUNA',
                  'codage':'SUCURSAL_IPS','canhab':'CANT_HD','tothab':'TOTAL_HAB','totdes':'TOTAL_DESC'
                 }

#Lista de feriados
feriados = [date(2019, 1, 1), date(2019, 4, 19), date(2019, 4, 20), date(2019, 5, 1), date(2019, 5, 21),
            date(2019, 6, 29), date(2019, 7, 16), date(2019, 8, 15), date(2019, 9, 18), date(2019, 9, 19),
            date(2019, 9, 20), date(2019, 10, 12), date(2019, 10, 31), date(2019, 11, 1), date(2019, 12, 25),

            date(2020, 1, 1), date(2020, 4, 10), date(2020, 4, 11), date(2020, 5, 1), date(2020, 5, 21),
            date(2020, 6, 29), date(2020, 7, 16), date(2020, 8, 15), date(2020, 9, 18), date(2020, 9, 19),
            date(2020, 10, 12), date(2020, 10, 31), date(2020, 11, 1), date(2020, 12, 8), date(2020, 12, 25),

            date(2021, 1, 1), date(2021, 4, 2), date(2021, 4, 3), date(2021, 5, 1), date(2021, 5, 21),
            date(2021, 6, 28), date(2021, 7, 4), date(2021, 7, 16), date(2021, 8, 15), date(2021, 9, 17),
            date(2021, 9, 18), date(2021, 9, 19), date(2021, 10, 11), date(2021, 10, 31), date(2021, 11, 1),
            date(2021, 11, 21), date(2021, 12, 8), date(2021, 12, 19), date(2021, 12, 25),

            date(2022, 1, 1), date(2022, 4, 15), date(2022, 4, 16), date(2022, 5, 1), date(2022, 5, 21),
            date(2022, 6, 21), date(2022, 6, 27), date(2022, 7, 16), date(2022, 8, 15), date(2022, 9, 16),
            date(2022, 9, 18), date(2022, 9, 19), date(2022, 10, 10), date(2022, 10, 31), date(2022, 11, 1),
            date(2022, 12, 8), date(2022, 12, 25),

            date(2023, 1, 1), date(2023, 1, 2), date(2023, 4, 7), date(2023, 4, 8), date(2023, 5, 1),
            date(2023, 5, 7), date(2023, 5, 21), date(2023, 6, 21), date(2023, 6, 26), date(2023, 7, 16),
            date(2023, 8, 15), date(2023, 9, 18), date(2023, 9, 19), date(2023, 10, 9), date(2023, 10, 27),
            date(2023, 11, 1), date(2023, 12, 8), date(2023, 12, 17), date(2023, 12, 25)]
```

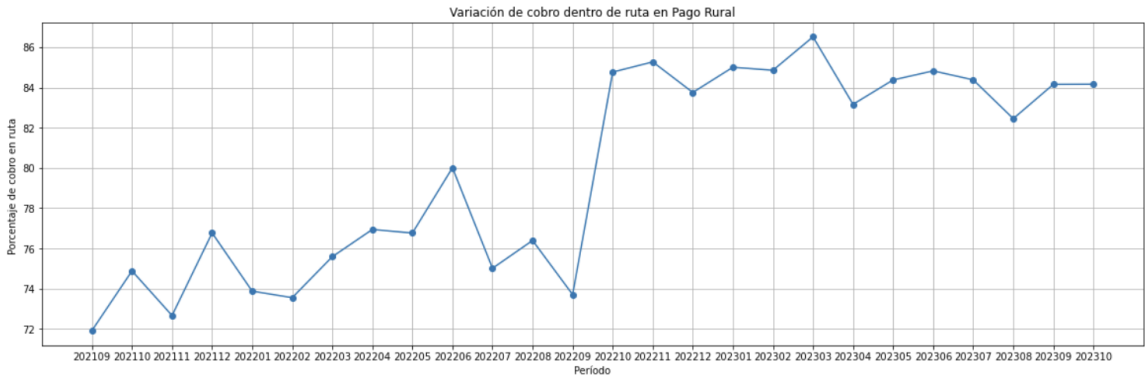
```python
#Función para convertir fechas a datetime64[D]
def convertir_fecha(fecha_num):
    if fecha_num == 's/r':
        return 's/r'
    fecha_str = str(fecha_num)
    dia = int(fecha_str[:-6])
    mes = int(fecha_str[-6:-4])
    anio = int(fecha_str[-4:])
    return pd.Timestamp(year=anio, month=mes, day=dia).date()

#Función para calcular días hábiles
def dias_habiles2(fecha_inicio, fecha_fin, feriados):
    if fecha_inicio == 's/r' or fecha_fin == 's/r':
        return 's/r'
    return np.busday_count(fecha_inicio, fecha_fin, holidays=feriados)

# Función para calcular días hábiles
def dias_habiles(fecha_inicio, fecha_fin, feriados):
    if fecha_inicio == 's/r' or fecha_fin == 's/r':
        return 's/r'

    # Convertir las fechas a datetime64[D]
    fecha_inicio = fecha_inicio.date()
    fecha_fin = fecha_fin

    return np.busday_count(fecha_inicio, fecha_fin, holidays=feriados)

#Definir una función para calcular el valor de COBRO basado en las condiciones dadas
def calcular_num(row):
    if row['TIPO_MOV'] == '30':
        if tolerancia >= int(row['DIAS_HABILES']): ### no puede ser -1
            return '1'
        elif tolerancia < int(row['DIAS_HABILES']):
            return '2'
    elif row['TIPO_MOV'] in ['10', '40']:
        if row['PLAZA_PAGO'] != row['COD_PLAZA_PAGO']:
            return '3'
    else:
        return '0'
```

```python
def procesar_COBRO(df, periodo):
    # Filtrar las filas con EMI_PERIODO igual a periodo y COBRO igual a 2
    filtro = (df['EMI_PERIODO'] == periodo) & (df['COBRO'] == '2')
    df_filtrado = df[filtro]
    # Recorremos las filas filtradas
    for indice, fila in df_filtrado.iterrows():
        # Obtener los valores de las columnas relevantes
        rut_benef = fila['RUT_BENEF']
        nis = fila['NIS']
        inst_cod = fila['INST_COD']
        tipo_mov = fila['TIPO_MOV']
        fecha_mov = fila['FECHA_MOV']
        # Filtrar las filas donde coinciden los valores de las columnas
        filtro_coincidentes = (df['RUT_BENEF'] == rut_benef) & (df['NIS'] == nis) & \
                              (df['INST_COD'] == inst_cod) & (df['TIPO_MOV'] == tipo_mov) & \
                              (df['FECHA_MOV'] == fecha_mov)
        # Excluir la fila actual del filtro
        filtro_coincidentes[indice] = False
        # Verificar si hay filas coincidentes con COBRO igual a 1
        if any(df[filtro_coincidentes]['COBRO'] == '1'):
            # Actualizar el valor de COBRO de la fila actual a 1
            df.at[indice, 'COBRO'] = '1'

    return df

dfs_rural = {}

# Definir la función para procesar un período
def procesar_periodo(periodo):
    global emi_periodos
    print(f"Procesando período {periodo}...")

    #Período emisión y rendiciones
    periodos = [elemento for elemento in os.listdir(carpeta_emision) if len(elemento) == 6]
    indice_periodo = periodos.index(periodo)
    periodos_emisiones = periodos[indice_periodo:indice_periodo+3]
    emi_periodos = list(periodos_emisiones)

    #Lectura emision
    dfs_emision = {}
    for periodo_emision in periodos_emisiones:
        nombre_archivo = f'Emi_{periodo_emision}_rural.csv'
        archivo_emision = os.path.join(carpeta_emision, periodo_emision, nombre_archivo)
        emision = pd.read_csv(archivo_emision, usecols=['rutben','codins','numdoc','codsuc','numins','fecpag','codmun','cilp','cc
        emision = emision.astype(str)
        print('Registros de emisiones cargadas del período',periodo_emision,':',len(emision))
        emision['EMI_PERIODO'] = periodo_emision
        dfs_emision[periodo_emision] = emision
```

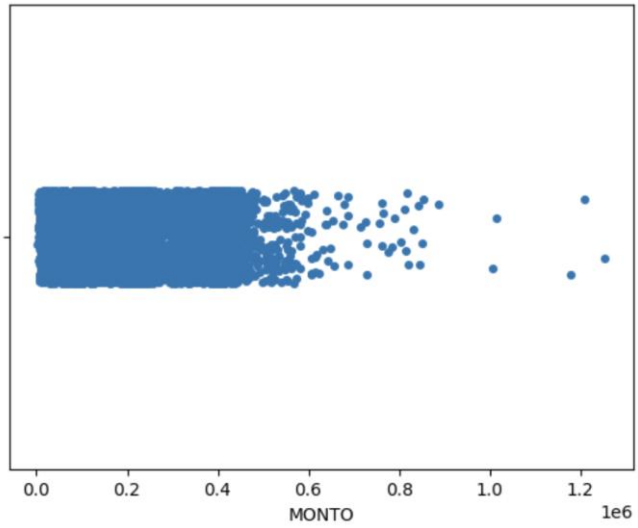
```
    #Lectura rendición
    dfs_rendicion = {}
    for periodo_rendicion in periodos_emisiones:
        nombre_archivo = f'rendiciones_{periodo_rendicion}_LLHH_303.csv'
        archivo_rendicion = os.path.join(carpeta_rendicion, periodo_rendicion, nombre_archivo)
        rendicion = pd.read_csv(archivo_rendicion, usecols=['rut','instit','num_doc','tipo_mov','cod_plaza_pago','num_inscripcion
        print('Registros de rendiciones cargadas del período', periodo_rendicion,':', len(rendicion))
        dfs_rendicion[periodo_rendicion] = rendicion

    #Arreglo emisión
    #Iterar a través del diccionario de DataFrames
    for df_name, df in dfs_emision.items():
        #Cambiar nombres
        df.rename(columns=nuevos_nombres, inplace=True)
        #Obs: Algunas columnas terminan en ".0"
        #Adecuar los datos al cruce eliminando los ".0" de las columnas
        df['NIS'] = df['NIS'].str.replace(r'\.0$', '',regex=True)
        # Convertir la columna FEC_PAGO al formato de fecha 'YYYY-MM-DD'
        df['FEC_PAGO'] = pd.to_datetime(df['FEC_PAGO'], format='%Y%m%d', errors='coerce')

    #Arreglo rendición
    #Iterar a través del diccionario de DataFrames
    for df_name, df in dfs_rendicion.items():
        #Cambiar nombres
        df.rename(columns=nuevos_nombres, inplace=True)
        #Obs: Algunas columnas terminan en ".0"
        #Adecuar los datos al cruce eliminando los ".0" de las columnas
        df['RUT_BENEF'] = df['RUT_BENEF'].str.replace(r'\.0$', '',regex=True)
        df['COD_PLAZA_PAGO'] = df['COD_PLAZA_PAGO'].str.replace(r'\.0$', '',regex=True)

    #Juntar los registros
    emisiones = pd.concat(dfs_emision.values(), ignore_index=True)
    rendiciones = pd.concat(dfs_rendicion.values(), ignore_index=True)

    ### CRUCE ###

    #Cruce de las rendiciones de los periodos indicados con las emisiones (con FP=2)
    cruce = pd.merge(emisiones, rendiciones, on=['RUT_BENEF','INST_COD','NUM_DOC','NIS'], how='left')
    #Registros de emisiones sin rendición
    sin_rendicion = cruce['TIPO_MOV'].isna().sum()
    print(f'Cantidad de emisiones sin coincidencia en el registro de rendiciones: {sin_rendicion}')

    #Cambiar los nombres de casillas vacías
    cruce['TIPO_MOV'].fillna('s/r',inplace=True)
    cruce['COD_PLAZA_PAGO'].fillna('s/r',inplace=True)
    cruce['NIS'].fillna('s/r',inplace=True)
    cruce['FECHA_MOV'].fillna('s/r',inplace=True)
    cruce['MON_DOC'].fillna('s/r',inplace=True)
```

```
    ### CRUCE RURAL ###
    cruce_rural = pd.DataFrame(cruce[cruce['FP']=='2'])
    # Aplicar la función para convertir fechas a las columnas
    #cruce_rural['FEC_PAGO'] = cruce_rural['FEC_PAGO'].apply(convertir_fecha)
    cruce_rural['FECHA_MOV'] = cruce_rural['FECHA_MOV'].apply(convertir_fecha)
    # Aplicar la función a las columnas y crear una nueva columna "tardanza_h"
    cruce_rural['DIAS_HABILES'] = cruce_rural.apply(
        lambda row: dias_habiles(row['FEC_PAGO'], row['FECHA_MOV'], feriados), axis=1)
    #Aplicar la función a lo largo de las filas del DataFrame y crear la columna "COBRO"
    cruce_rural['COBRO'] = cruce_rural.apply(lambda row: calcular_num(row), axis=1)
    cruce_rural = procesar_COBRO(cruce_rural, periodo)

    #Visualizar las frecuencias según tipo de movimiento
    print('Tipo de movimiento rural:')
    print(cruce_rural['TIPO_MOV'].value_counts())
    print('Tipo de cobro rural')
    print(cruce_rural['COBRO'].value_counts())
    dfs_rural[periodo] = cruce_rural

    df = pd.DataFrame(cruce_rural[cruce_rural['EMI_PERIODO']==periodo])
    print(f'Comportamiento de Cobro del periodo {periodo} :')
    print(df['COBRO'].value_counts())

    #Guardar un csv
    ruta = f'C:/Users/56942/OneDrive - Chileatiende/Escritorio/IPS/Pagos/OneDrive_2023-08-21/Proyecto Comparatiivo Emisión-Rendic
    print(f'Guardando archivo del período {periodo} ...')
    df.to_csv(ruta,sep=';',index=False)

#Establecer un nivel de tolerancia para la permitida en el cálculo
tolerancia = int(input('Ingrese tolerancia de tardanza en días hábiles: '))

#Definir una lista de periodos
periodos = ['202306']

#Iterar a través de la lista de periodos y ejecutar la función para cada uno
for periodo in periodos:
    procesar_periodo(periodo)

# Registra el tiempo de finalización
end_time = time.time()

# Calcula el tiempo total de ejecución
execution_time = end_time - start_time

print(f"Tiempo de ejecución: {execution_time} segundos")
```

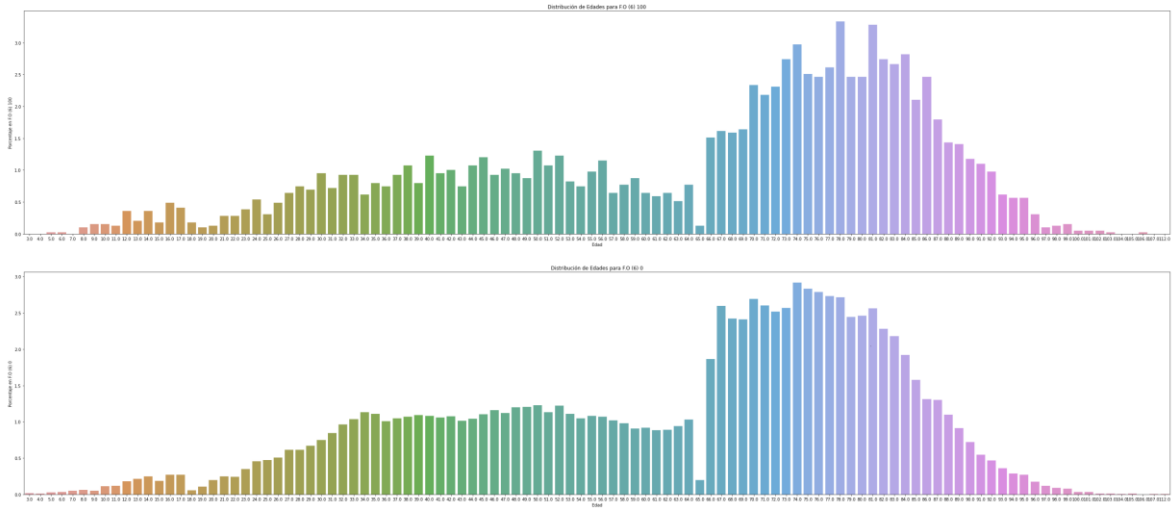
Anexo 3: Series de Tiempo de Cobro en Ruta histórico.



Anexo 4: Análisis univariado de la variable MONTO para identificar valores atípicos.



Anexo 5: Análisis bivariado de la variable EDAD y Función Objetivo Cobro en Ruta (F.O (6) 0) y Fuera de Ruta (F.O (6) 100).

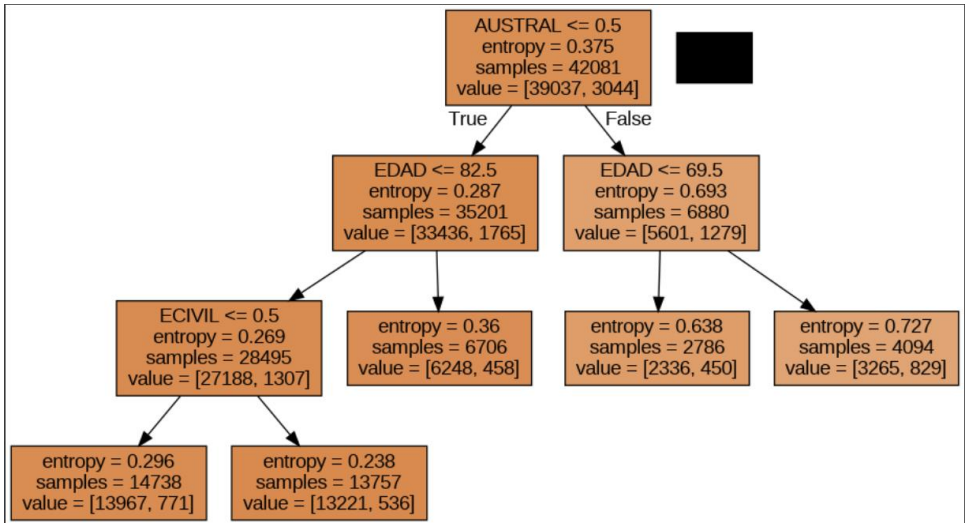


Anexo 6: Regresión logística con los valores de p-estadístico.

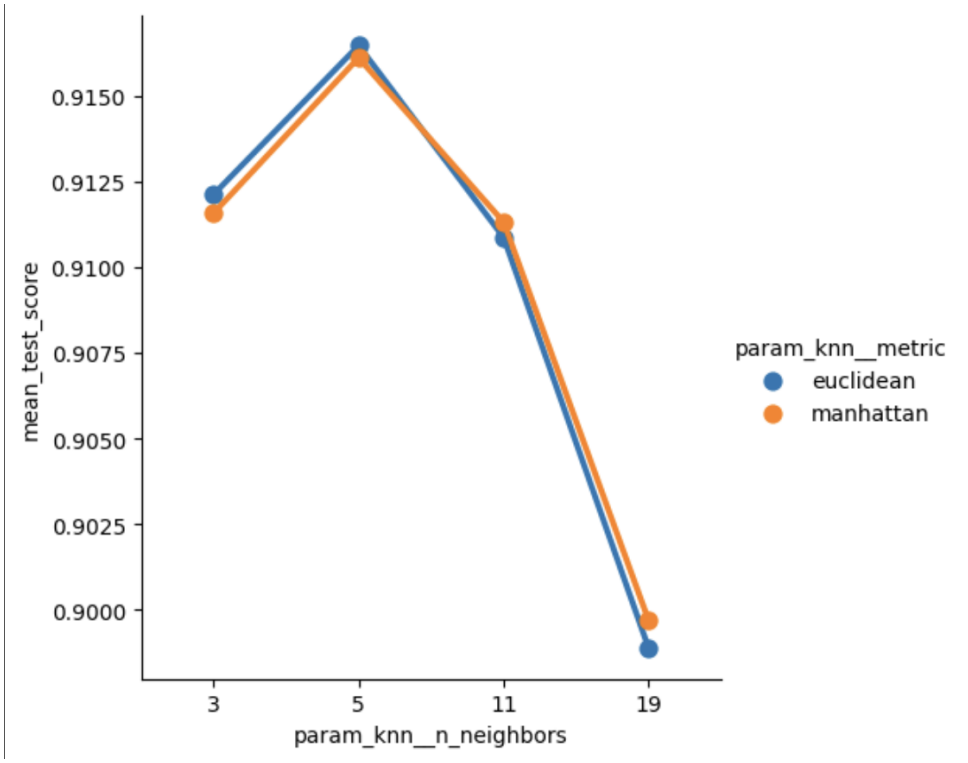
```
Optimization terminated successfully.
         Current function value: 0.646609
         Iterations 5
                         Results: Logit
================================================================
Model:              Logit            Method:           MLE
Dependent Variable: MIXTO FO 6       Pseudo R-squared: 0.067
Date:               2023-12-06 10:56 AIC:              100978.6624
No. Observations:   78074            BIC:              101034.2548
Df Model:           5                Log-Likelihood:   -50483.
Df Residuals:       78068            LL-Null:          -54117.
Converged:          1.0000           LLR p-value:      0.0000
No. Iterations:     5.0000           Scale:            1.0000
----------------------------------------------------------------
               Coef.   Std.Err.     z      P>|z|    [0.025   0.975]
----------------------------------------------------------------
const         -0.6463    0.0369  -17.5154  0.0000  -0.7186  -0.5740
AUSTRAL        1.3662    0.0182   75.2376  0.0000   1.3306   1.4018
ECIVIL        -0.3741    0.0162  -23.0722  0.0000  -0.4059  -0.3423
PUNTAJE       -0.2554    0.0176  -14.5315  0.0000  -0.2898  -0.2209
EDAD           0.0081    0.0004   18.0988  0.0000   0.0072   0.0089
BENEFICIO      0.1430    0.0212    6.7598  0.0000   0.1016   0.1845
================================================================
```

Anexo 7: Modelos de Aprendizaje Supervisado y sus mejores parámetros.

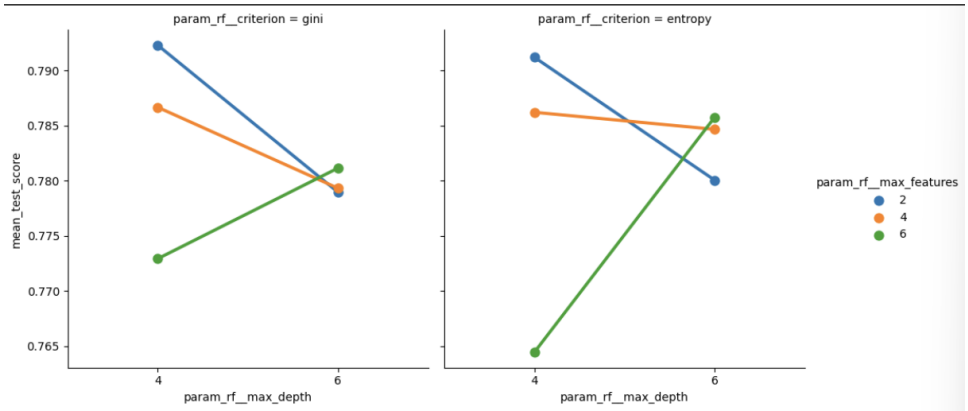
Árbol de decisión. Best params: {'tree__max_depth': 4, 'tree__max_leaf_nodes': 5}



K-nearest neighbors. Best params: {'knn__metric': 'euclidean', 'knn__n_neighbors': 5}



Random Forest. Best params: {'rf__criterion': 'gini', 'rf__max_depth': 4, 'rf__max_features': 2}



Red neuronal: Best params: {'red__alpha': 0.1, 'red__hidden_layer_sizes': 2, 'red__max_iter': 50}