**MODELO ADMISIÓN BANK QUICK WIN**

Área Modelos y Metodologías

SQUAD BPExitos

**INTERBANK**

**Control de Versiones**

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Fecha** | **Realizado por** | **Actualización realizada** |
| 1 | 24/10/2023 | Ismael Centeno | --- |
| 2 | 02/11/2023 | Ismael Centeno | Levantamiento de observaciones |
| 3 | 14/11/2023 | Ismael Centeno | Se corrigió el nombre del campo para los filtros correspondientes. |
| 4 | 17/11/2024 | Ismael Centeno | Levantamiento de las observaciones de auditoria de variables y documento de creación de variables. |
| 5 | 27/11/2023 | Ismael Centeno | Levantamiento de observaciones indeterminados,correcion de filtros |
| 6 | 12/12/2023 | Ismael Centeno | Se ha agregado en el anexo 5 el descriptivos de las variables sin imputar como también el univariado de los flag |
| 7 | 07/02/2024 | Ismael Centeno | Se ha agregado la tabla de eficiencia, rango de hiperparámetros Anexo 7, pie de página definición de cliente y se cambió el nombre del modelo lo cual se renombrado en el comité de modelos, anexo 8 |

Contenido

[1. INTRODUCCIÓN 4](#_Toc117267280)

[2. POBLACIÓN OBJETIVO 4](#_Toc117267281)

[3. RECOPILACIÓN Y GENERACIÓN DE INFORMACIÓN 6](#_Toc117267282)

[3.1 Fuentes de información requeridas 6](#_Toc117267283)

[3.2 Softwares utilizados 7](#_Toc117267284)

[4. DESCRIPCION DE PROCESOS 7](#_Toc117267285)

[4.1 OBTENCION DE LA MUESTRA 7](#_Toc117267286)

[4.2 FITLTROS APLICADOS SOBRE LA BASES DE DATOS INICIALES 8](#_Toc117267287)

[4.3 PERIODOS DE OBSERVACIÓN 9](#_Toc117267288)

[4.2 PERIODO DE COMPORTAMIENTO 10](#_Toc117267289)

[5. MARCA DE DEFAULT 11](#_Toc117267290)

[5.1 Análisis descriptivo 12](#_Toc117267291)

[5.2 Análisis temporal a nivel variable 12](#_Toc117267292)

[5.3 Creación de Variables 13](#_Toc117267293)

[6. ETAPA DE MODELAMIENTO 13](#_Toc117267294)

[6.1 Metodología de Modelamiento 13](#_Toc117267295)

[Gradient Boosting 13](#_Toc117267296)

[Light Gradient Boosting (Ligthbm) **¡Error! Marcador no definido.**](#_Toc117267297)

[6.2 Partición de la muestra 15](#_Toc117267298)

[6.3 Modelo Hiperpámetros del Modelo Light Gradient Boosting **¡Error! Marcador no definido.**](#_Toc117267299)

[6.4 Selección de variables 19](#_Toc117267300)

[6.5 Análisis Bivariante/ Dependencia parcial con valores shap 19](#_Toc117267301)

[6.6 Resultados de Perfomance del modelo 20](#_Toc117267302)

[6.7 Tabla de Eficiencias **¡Error! Marcador no definido.**](#_Toc117069651)

[6.8 Sofware utilizado: 24](#_Toc117267303)

[7. ANEXO 24](#_Toc117267304)

[8. RESPONSABILIDAD DE ÁREA Y ÓRGANOS INTERVENIENTES 29](#_Toc117267305)

## INTRODUCCIÓN

La banca de Pequeña Empresa está definida con aquellas empresas, cuyo endeudamiento total en SF (sin incluir créditos hipotecarios para vivienda) es superior a S/. 20 mil pero no mayor a S/. 300 mil en los últimos seis meses

El modelo que actualmente evalúa a estos clientes el Buro BPE no está ordenando a nivel de tasa de malo y score por lo cual esta herramienta será complementaria al modelo buro, es decir trabajaran de forma conjunta con esta herramienta, lo que se busca es encontrar segmentos diferenciados a nivel mora.

Para la generación de los segmentos se hizo mediante arboles de decisión lo cual se encontraron 4 segmentos.

El siguiente documento contiene la descripción del modelo Clúster BPE. El cual corresponde a un modelo matemático que clasifica a un cliente según su perfil de riesgo en alguna de las siguientes categorías: bueno o malo.

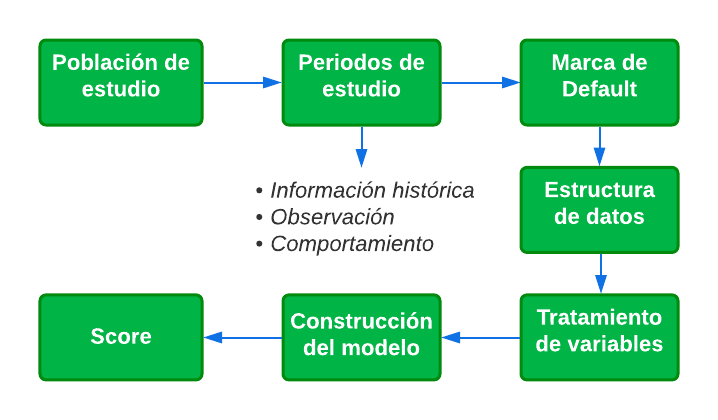
En la segunda etapa del proceso, se llevó a cabo una segmentación basada en las probabilidades proporcionadas por un modelo binario. El objetivo de esta segmentación era diferenciar los segmentos a nivel de morosidad. Para lograrlo, se utilizó un modelo de árbol. Este enfoque permitió clasificar de manera efectiva a los clientes en diferentes segmentos según su probabilidad de incumplimiento financiero.

## POBLACIÓN OBJETIVO

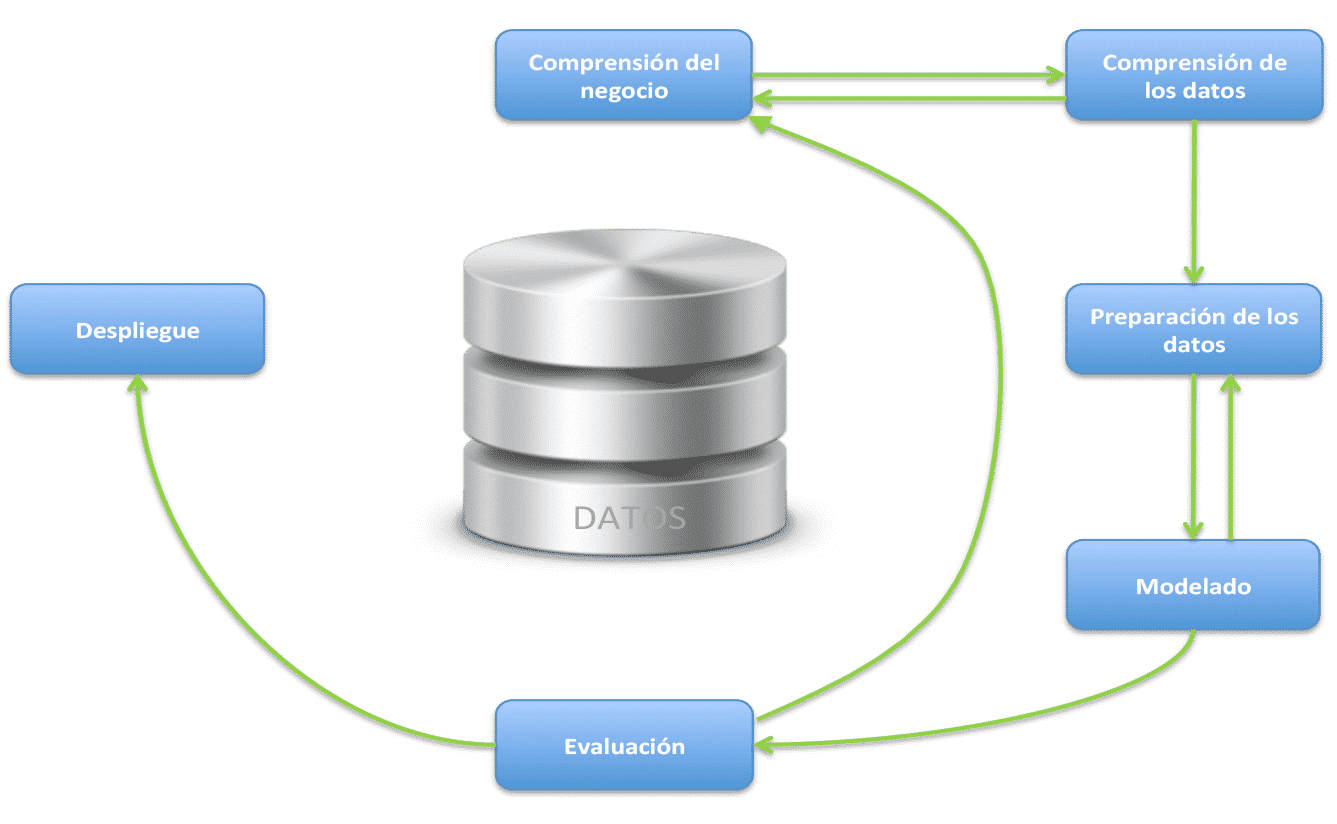
La población objetivo es altas del SF, aplicando la siguiente consideración:

Con el objetivo de evaluar el comportamiento de los clientes a futuro, se necesitará que los clientes en el periodo de observación no tengan un comportamiento negativo ni inconsistencias por lo que se optan por las siguientes exclusiones:

* Malos en el Origen, Indeterminados.
* Sin Ruc Activo, sin marca reactiva, los que no superaron las campañas, los que no tienen perfil interbank.
* Clientes reprogramados reactiva (*El programa Reactiva Perú, creado por el Estado mediante DL N°1455, está dirigido a las empresas afectadas por la emergencia sanitaria del COVID-19, y tiene como objetivo promover financiamiento de reposición de capital de trabajo de empresas que enfrentan pagos y obligaciones de corto plazo y así asegurar continuidad en la cadena de pagos*).



Metodología CRIPS-DM



## RECOPILACIÓN Y GENERACIÓN DE INFORMACIÓN

### Fuentes de información requeridas

La información para el desarrollo del modelo se obtiene de las siguientes fuentes mencionadas en el siguiente gráfico.

* RCC
* SUNAT
* SUNARP
* RENIEC
* PASIVOS
* ADEX
* TRX



* **Base RCC (Base Sistema Financiera):** Información de clientes en el sistema financiero.
* **Base SUNAT (Base Proveedor):** Base de contribuyentes para evaluar sus características. También contiene la información de representantes legales de los clientes, en su mayoría jurídicos.
* **Base SUNARP (Base Proveedor):** Base de personas que registran vehículos.
* **Base ADEX (Base Interna):** Base que registran información sobre transacciones como exportaciones e importaciones realizadas
* **Base TRX (Base Interna):** Base que registran información sobre transacciones en entidades como retiro y depósitos hechos.
* **Base PASIVOS (Base Interna):** Base que registran información sobre pasivos.
* **Base RENIEC (Base Interna):** Base que registran información sobre las personas naturales.

**Rutas de las fuentes usadas:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | |  |  |  |
|  | **ESQUEMA** | | **TABLA** | **ACTUALIZACIÓN** |  |
|  | **e\_perm\_aws** | | t\_fact\_report\_rcc\_rsk | MENSUAL |  |
|  | **e\_perm\_aws** | | t\_mst\_princ\_sunat\_rsk t\_mst\_sec\_sunat\_rsk | MENSUAL |  |
|  | **e\_perm\_aws** | | t\_agg\_mes\_vpc\_transacc\_sav\_imp | MENSUAL |  |
|  | **e\_perm\_aws** | | t\_mst\_sunarp\_rsk | MENSUAL |  |
|  | **e\_perm\_aws** | | t\_mst\_ctaibk\_rsk  t\_dim\_jerarquia\_productos  t\_dim\_tipo\_cambio\_rsk  t\_fem\_all\_instr\_rsk | MENSUAL |  |
|  | **e\_perm\_aws** | | mst\_vpc\_adex | MENSUAL |  |
|  | **e\_perm\_aws** | | t\_mst\_reniec\_rsk | MENSUAL |  |
|  | **e\_perm\_aws** | | tbl\_rcc\_agg01\_allsf\_mdl | MENSUAL |  |
|  |  |
|  |  | |  |  |  |

### Softwares utilizados

* Microsoft Excel, Word y Power Point usado para la construcción de la documentación
* AWS (Athena) para la construcción de la base de desarrollo y validación
* AWS (SageMaker) para la construcción del modelo de desarrollo
* Python, como lenguaje para la construcción del modelo donde se usaron paquetes como pandas, numpy, lightGBM.
* Pandas= '2.0.3'
* Numpy= '1.22.3'
* lightgbm= '3.3.5'
* Shap= '0.42.0'

## DESCRIPCION DE PROCESOS

### OBTENCION DE LA MUESTRA

Para los períodos de desarrollo, se tomaron en consideración tanto aquellos previos a la implementación del modelo Buro BPE como aquellos en los que se comenzó a implementar. Estos períodos abarcaron desde antes de que el nuevo modelo entrara en funcionamiento hasta el momento en que el nuevo modelo ya no estuvo generando resultados ordenados.

Además, se procuró incluir aquellos períodos que se vieron afectados por la coyuntura nacional. Esto permitió un análisis completo de los cambios en la morosidad y la influencia del modelo Buro BPE en diferentes contextos económicos y temporales Los períodos mostrados en los siguientes gráficos hacen referencia a la fecha de altas.

Para el modelamiento antes de calcular la variable objetivo (bueno, indeterminado, malo), se realizó una agrupación a nivel de codmes, tip\_doc, y key\_value y se calculó el máximo de días de atraso, para la exclusión de clientes duplicados. Ver Anexo 3

### FILTROS APLICADOS SOBRE LA BASES DE DATOS INICIALES

* *Indeterminados*

**Motivo**: Son clientes que no podemos estar seguros de que sean malos o buenos, debido a la alta tasa de recuperación de impago (de 8 a 30 días) que existe, se define como indeterminados los clientes que no alcanzan los 8 días, pero si los 30 días.

* *Días de atraso mayor a cero en periodo de observación (días\_atraso>0).*

**Motivo**: Son clientes que tienen días atraso al momento de la observación por lo que no hay nada que aprender de ellos.

* *No Presentan ruc activo*

**Motivo**: Son clientes que tienen ruc, pero no están activos, ya sea que dio de baja.

* *Reactiva*

**Motivo**: Se filtraron a todos clientes que tienen un perfil reactiva, debido a su periodo de gracia que presenta su producto.

* *No perfil ibk*

**Motivo**: indica que el cliente no ha superado todas las caídas de campaña (sin incluir las caídas por modelos Buró/Origen).

* *No Superaron las campañas*

**Motivo**: Se filtraron a todos clientes que no han tenido deuda negocio entre 15,000 y 1,700,000 soles en los últimos 12 meses.

* *Tenencia de garantía*

**Motivo**: identifica a los clientes que tienen garantías asociadas

* . *Sin Performance*

**Motivo**: identifica a los clientes que no tienen performance en la ventana de 6 meses.

|  |  |  |
| --- | --- | --- |
| **Exclusiones y filtros Target** | **Periodo** | |
| **202206-202302** | |
| **Total** | **Filtrados** |
| Altas SF | 98,691 |  |
| Perfil IBK | 69,763 | 28,928 |
| Caida Campaña | 30,612 | 39,151 |
| Tenencia Garantía | 30,087 | 525 |
| Sin ruc Activo | 29,085 | 1002 |
| reactiva | 26,604 | 2,481 |
| Malo en Origen | 26,527 | 77 |
| Sin performance | 26,451 | 76 |
| Clientes únicos | 26,134 | 317 |
| indeterminados | 24,349 | 1,785 |
| Base final | 24,349 | - |

En el anexo 3 se muestra el código de los filtros aplicados en código Python.

### PERIODOS DE OBSERVACIÓN

La muestra se desarrolló con una ventana de observación conformada por 9 cosechas tomando como periodos desde Junio22 a Febrero23.

A continuación una vista global:

|  |  |  |
| --- | --- | --- |
| **codmes** | **Total** | **TM(%)** |
| 202206 | 2714 | 5.0% |
| 202207 | 2693 | 4.9% |
| 202208 | 2720 | 6.8% |
| 202209 | 2704 | 5.6% |
| 202210 | 2797 | 5.8% |
| 202211 | 3138 | 7.1% |
| 202212 | 2451 | 6.8% |
| 202301 | 2493 | 8.7% |
| 202302 | 2639 | 8.7% |

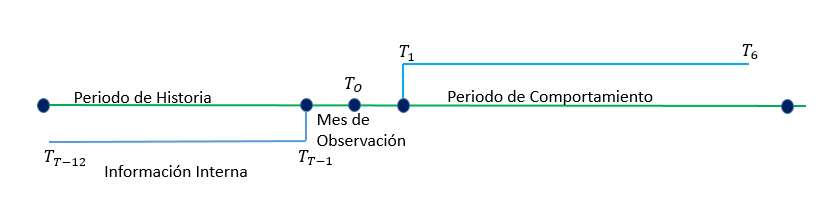
### PERIODO DE COMPORTAMIENTO

Los periodos de comportamiento hacen referencia a los meses en que se observa el desempeño del cliente, posteriores a un mes inicial de observación. Esto corresponde a los de 6 meses siguientes respecto a cada mes de observación, que comprenden los periodos de Junio22 a Febrero23.

Son en estos periodos en donde se evaluará el comportamiento de los clientes, determinándose una marca de comportamiento negativo, denominado Default, el cual será descrito posteriormente.

* 1. **PERIODO DE INFORMACIÓN HISTÓRICA**

Los periodos de información histórica para la construcción del modelo, hacen referencia a los periodos en donde se recogerán información de los clientes. Particularmente en este modelo, estos periodos comprenden desde un periodo precedente al mes de observación hasta 6 periodos como máximo.



## MARCA DE DEFAULT

La marca de incumplimiento, denominada Default, hace referencia al comportamiento negativo que un cliente presenta en su periodo de comportamiento en relación con todo el sistema financiero. Esta marca representa como malo, en un horizonte de 6 meses, a aquella persona (natural o jurídica) que presenta alguna de las siguientes condiciones

* Si su deuda con atraso superior a 30 días

El Default es denotado por el valor de 1 (clientes malos), y los clientes que no presentan comportamiento negativo son denotados con 0 (clientes buenos)

También se define la tasa de Default, la cual mide la proporción de clientes malos sobre el total de clientes

Para nuestro modelamiento se trabajó con la mora cliente que se calcula en todas las entidades que el cliente tiene algún producto lo cual por eso es la base (BASE\_INICIAL\_FILTRO.txt) se tendrá algunos duplicados en un mismo periodo pero diferenciados a nivel código institución, para obtener los clientes únicos se hizo una agrupación para cada cliente quedándose con el peor de los días de atraso lo cual se muestra en el anexo 3.

### Análisis descriptivo

Para tener un conocimiento a rasgos generales de la información que se dispone y, al mismo tiempo, detectar posibles errores y falta de información, se realiza un “análisis descriptivo”.

Se diferencian dos tipos de análisis en función de si la variable es de tipo numérica o categórica.

En el caso de variables categóricas se calcula el número de casos por cada categoría de la variable y el porcentaje que representa del total de registros. El objetivo es detectar elevadas concentraciones en determinadas categorías y, si existen casos no informados, qué porcentaje suponen estos respecto al total de registros de la base de datos.

Para las variables numéricas, se realiza un análisis descriptivo en el que se muestra un conjunto de estadísticas de resumen para evaluar las medidas de tendencia central. Las estadísticas son:

* Nmissing: Es el número de missings.
* Fill rate: Es el porcentaje de casos válidos.
* Mean: Es la media.
* Min: Es el mínimo valor de la variable.
* P25 (1st Qu.): Es el percentil 25.
* Median: Es la mediana o segundo cuartil.
* P75 (3erd Qu.): Es el percentil 75.



En el Anexo 5 se ha agregado los descriptivos de las variables sin imputar.

### Análisis temporal a nivel variable

Con este análisis se trata de detectar posibles cambios estructurales en la información. El análisis consiste en, para cada mes observado, analizar la distribución de la variable.

* Para las **variables numéricas** se calcula la media, mediana y los percentiles 25 y 75 los cuales se visualizan en un mismo gráfico para evaluar el comportamiento de cada estadística a través del tiempo (codmes).



### Creación de Variables

Partiendo de las variables originales, se procede a definir nuevas variables de forma que la información a partir de la cual se califiquen las operaciones sea lo más enriquecedora posible.

El detalle de las variables creadas que finalmente se incluyen en el modelo se muestra en el documento de implementación adjunto:



Para el proceso de selección de variables a nivel de missing solo se quedó con variables que tenían un porcentaje de missing por debajo de 90% .

## ETAPA DE MODELAMIENTO

### Metodología de Modelamiento

### Gradient Boosting

Gradient Boosting utiliza un algoritmo de partición descrito en "Greedy Function Approximation: A Gradient Boosting Machine" y "Stochastic Gradient Boost" de Jerome H. Friedman. Un algoritmo de partición busca una partición óptima de los datos definidos en términos de los valores de una sola variable. El criterio de optimalidad depende de cómo otra variable, el objetivo, se distribuye en los segmentos de partición. Cuando los valores de la variable objetivo son más similares dentro de los segmentos, el valor de la partición es mayor. La mayoría de los algoritmos de partición particionan más cada segmento en un proceso llamado partición recursiva. Las particiones se combinan para crear un modelo predictivo. El modelo se evalúa mediante estadísticas de bondad de ajuste que se definen en términos de la variable objetivo. Estas estadísticas son diferentes de la medida del valor de una partición individual. Un buen modelo puede resultar de muchas particiones mediocres.

El Gradient Boosting es un enfoque de impulso que vuelve a muestrear el conjunto de datos de análisis varias veces para generar resultados que forman un promedio ponderado del conjunto de datos remuestreados. Los Arboles Boosting crea una serie de árboles de decisión que juntos forman un solo modelo predictivo. Un árbol de la serie se ajusta al residual de la predicción de los árboles anteriores de la serie. El residual se define en términos de la derivada de una función de pérdida.

Los objetivos de intervalo definen el residuo utilizando la función de pérdida de error al cuadrado. Para calcular el residual de un objetivo de intervalo utilizando la pérdida de error al cuadrado, simplemente reste el valor predicho del valor de destino. Los objetivos binarios definen el residuo utilizando la función de pérdida de probabilidad de registro binomial negativa. La función de pérdida de probabilidad binomial negativa también se conoce como pérdida logística.

Cada vez que se usan los datos para hacer crecer un árbol y se calcula la precisión del árbol. Las muestras sucesivas se ajustan para adaptarse a errores previamente calculadas. Debido a que cada muestra sucesiva se ponderó de acuerdo con la precisión de clasificación de los modelos anteriores, este enfoque a veces se denomina Gradient Boosting estocástico. El Boosting se define para objetivos binarios, nominales e intervalos.

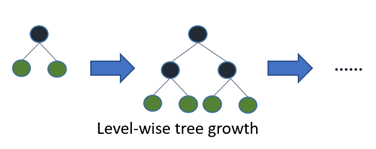
Al igual que los árboles de decisión, El Boosting no hace suposiciones sobre la distribución de los datos. Para una entrada de intervalo, el modelo solo depende de los rangos de los valores. Para un objetivo de intervalo, la influencia de una teoría del valor extremo depende de la función de pérdida. El nodo Gradient Boosting ofrece una pérdida de la estimación de Huber M que reduce la influencia de los valores objetivo extremo. El Boosting es menos propenso a adaptar los datos que un solo árbol de decisiones, y si un árbol de decisiones se ajusta bastante bien a los datos, el aumento a menudo mejora el ajuste.

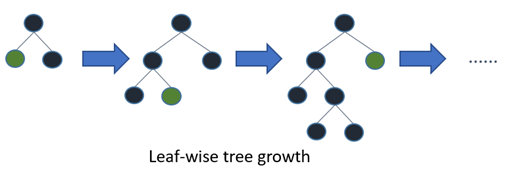
### Light Grandient Boosting Machines (LIGTHBM)

Como la variable dependiente es binaria, se procederá a realizar una estimación utilizando un modelo de machine learning de clasificación llamado LightGBM.

A continuación, se presenta la especificación del modelo definido:

LGBM es un modelo de boosting basado en árboles de decisión. El boosting es un tipo de método de ensamble que se basa en un proceso iterativo en el que en cada iteración del modelo se le asigna un mayor peso a las observaciones clasificadas erróneamente, para que se les pueda ir clasificando mejor. Esto implica que en cada iteración el algoritmo genera un árbol nuevo que depende de los pesos generados por el árbol previo. La diferencia del LGBM en comparación de otro algoritmo popular de Boosting como lo es el eXtreme Gradiente Boosting Machine (XGBoost) es que el segundo crea árboles que crecen por nivel, mientras que el primero crea árboles que crecen por hoja, esto se representa a continuación:





Esto hace que el modelo sea más complejo pero también más certero, lo que lo podría hacer propenso al sobre ajuste, pero este puede evitarse tomando los hiperparámetros adecuados que controlan el crecimiento del árbol para que este no crezca de forma desmedida.

Otra diferencia del algoritmo LGBM es su velocidad, tiene un mayor grado de optimización porque utiliza un algoritmo basado en histogramas, que categoriza las variables para tener computaciones más rápidas, además presenta una técnica más eficiente de muestreo (Muestreo Unilateral Basado en Gradientes) y de reducción de dimensionalidad de tal forma que el algoritmo puede correr en una fracción del tiempo en comparación a otros algoritmos de boosting basados en árboles.

Se entrenó el Algoritmo con los siguientes parámetros en el lenguaje de programación Python 3.8

### Partición de la muestra

Esta etapa se trabajó con el 80% en train Y 20% en test. Por otro lado, se trabajó con una muestra out of time con el objetivo de desarrollar modelos que puedan aplicarse adecuadamente a nuevos casos, distintos de los utilizados para el desarrollo. En efecto, si toda la muestra de análisis se utilizará para ajustar los modelos, se podría llegar a modelos que separasen muy bien los clientes buenos de los malos, pero no se tendría la garantía de que los resultados fueran similares a la hora de aplicar dichos modelos a nuevos casos. A continuación presentamos los totales de las particiones:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  | |  |
|  |  | Train | Test | OOT |  |
|  | N | 17241 | 4311 | 2797 |  |
|  |  |  |  |  |  | |  |

Finalmente se utilizaron 9 cosechas para el desarrollo del modelo de los cuales, 8 cosechas fueron para el entrenamiento y 1 de validación fuera de tiempo (out of time).

En el proceso de train/test del modelo, se utilizan ocho cosechas de datos (de junio a septiembre de 2022, y noviembre de 2022 a febrero 2023), junto con un período de validación fuera de tiempo (OOT) correspondiente a octubre de 2022. La elección de octubre de 2022 como período OOT se basó en el hecho de que, en los meses de noviembre a febrero de 2023, la tasa de incumplimiento había mostrado un aumento significativo. Por lo tanto, se decidió dejar fuera de tiempo el mes de octubre para que el modelo pudiera aprender de estos perfiles y capturar mejor los patrones emergentes.

Los meses de enero y febrero de 2023 en el proceso de modelamiento. Estos dos últimos meses son especialmente relevantes debido a que experimentaron los picos más altos en la tasa de incumplimiento, influenciados por la coyuntura nacional.

Esta estrategia de entrenamiento se diseñó con el objetivo de considerar tanto los cambios históricos en la morosidad como los efectos de la coyuntura actual, permitiendo al modelo aprender de manera efectiva y adaptarse a las tendencias más recientes.

Por otro lado resulta relevante mencionar que se realizó una imputación para los valores missing (ver descripción de variables) a las variables numéricas para poder diferenciarlas y observar que el modelo los esté castigando.

**Muestra de train** (202206-202209, 202211, 202302):

|  |  |  |
| --- | --- | --- |
|  | **Total** | **%Total** |
| **Bueno** | 16106 | 93.4% |
| **Malo** | 1135 | 6.6% |
|  | 17241 | 100% |

**Muestra de test** (202206-202209, 202211, 202302):

|  |  |  |
| --- | --- | --- |
|  | **Total** | **%Total** |
| **Bueno** | 4005 | 92.9% |
| **Malo** | 306 | 7.1% |
|  | 4311 | 100% |

**Muestra de OOT** (202210):

|  |  |  |
| --- | --- | --- |
|  | **Total** | **%Total** |
| **Bueno** | 2634 | 94.1% |
| **Malo** | 163 | 5.9% |
|  | 2797 | 100% |

* 1. **Concepto de los Hiperparámetros**.
  + **objective:** Es el tipo de tarea a realizar, se especifica como binary si se trata de un problema de clasificación binaria, multiclass si la target más de dos clases y regression si la target es contínuo.
  + **boosting:** Es el tipo de algoritmo para la construcción de árboles, el más usado es gbdt que construye arboles secuenciales donde el aprendizaje se da en base a los errores de los aprendices anteriores existe otro algoritmo que es dart.
  + **metric:** Es métrica a monitorear durante el aprendizaje del modelo lo cual va depender de la tarea a realizar si es de clasificacion o regresion+
  + **max\_depth:** Limitar la profundidad máxima para el modelo de árbol. Esto se usa para tratar el ajuste excesivo.
  + **num\_leaves:**Es el número de hojas para el modelo de arbol .Esto se usa para tratar el ajuste excessivo.
  + **learning\_rate:** es la tasa de aprendizaje ,sirve para controlar el ajuste excessivo del modelo.
  + **min\_child\_samples:** mínimo de muestras de ejemplo que pueda existir en rama.
  + **min\_data\_in\_leaf:** un parámetro que se utiliza para controlar la cantidad mínima de datos (muestras) que se requieren en una hoja (leaf) del árbol
  + durante el proceso de entrenamiento del modelo de gradient boosting. Este parámetro es importante para la regularización y la prevención del sobreajuste (overfitting) en el modelo.
  + **feature\_fraction:**Fracción de variables a tomar en cuenta para la construccion del arbol.
  + **bagging\_fraction:**Fracción de muestras a tomar en cuenta para la construccion del arbol.
  + **early\_stopping\_rounds:**Especifica el número de veces hasta encontrar el mejor perfomance entre train-test.
  + **Random\_state:** Especificar la semilla para generar las submuestras aleatorias en cada árbol.
  + **num\_boost\_round:**Es un parámetro que representa el número de iteraciones o "boosting rounds" que el algoritmo ejecutará durante el entrenamiento del modelo. Cada iteración implica que se ajusta un nuevo conjunto de árboles débiles al modelo para mejorar su precisión.

|  |  |
| --- | --- |
| Hiperparametros | valores |
| objective | binary |
| boost | 'dart' |
| metric | ‘auc' |
| task | ‘train' |
| max\_depth | 6 |
| num\_leaves | 3 |
| learning\_rate | 0.047237010810954355 |
| min\_child\_samples | 120 |
| min\_data\_in\_leaf | 60 |
| feature\_fraction | 0.48685220113088357 |
| bagging\_freq | 5 |
| bagging\_fraction | 0.9010416150378611 |
| lambda\_l1 | 2 |
| lambda\_l2 | 5 |
| verbosity | -1 |
| random\_state | 42 |
| num\_boost\_round | 1000 |
| early\_stopping\_rounds | 10 |
| verbose\_eval | 5 |

En cuanto al parámetro de **valid\_sets** de utilizaron la data de train y test para el monitoreo del indicador de auc en cada proceso de iteración

1. **Hyperopt :**

Hyperopt se utiliza para encontrar la mejor combinación de hiperparámetros para un modelo de aprendizaje automático de manera automatizada. Consiste en un proceso de búsqueda y optimización de hiperparámetros mediante técnicas de búsqueda inteligente.

**Las características clave de Hyperopt incluyen:**

**Algoritmos de Optimización:** Hyperopt ofrece varios algoritmos de optimización, como TPE (Tree-structured Parzen Estimator) y Random Search, que se utilizan para buscar el espacio de hiperparámetros en busca de la combinación óptima.

**Espacio de Búsqueda:** Permite definir el espacio de búsqueda de hiperparámetros, especificando los rangos o distribuciones de valores que pueden tomar los hiperparámetros.

**Función Objetivo:** Debes definir una función objetivo que evalúe el rendimiento del modelo para una combinación dada de hiperparámetros. Hyperopt buscará minimizar o maximizar esta función objetivo, según se especifique.

**Historial de Experimentos:** Lleva un registro de los resultados de las diferentes configuraciones de hiperparámetros probadas para que puedas analizar y seleccionar la mejor configuración.

**Paralelización:** Puede aprovechar la paralelización para evaluar múltiples configuraciones de hiperparámetros de manera eficiente, lo que es útil cuando tienes recursos computacionales limitados.

En el anexo 6 se tiene el rango de valores que se probaron en el proceso de encontrar los hiperparámetros “óptimos”.

### Selección de variables

En primer lugar, se retiraron todas las variables que tuviesen menos de 0.5% de importancia relativizada por gain ( gain (i)/ Sumatoria (gain (n))), siendo i la ganancia por variable y n total de variables)

Luego de las variables restantes se hizo una selección por correlación, donde si una variable estaba fuertemente correlacionada con otra (>80%) se retiraba aquella que tuviera el menor % de importancia relativizada por shap value mean, el proceso es el siguiente:

1. Se reordenan las variables en cuestión por importancia relativizada gain
2. Se calcula la matriz de correlación de Pearson.
3. Se retiran todas las variables asociadas a una columna de la matriz de correlación donde el valor absoluto de la correlación sea mayor a determinado umbral (80%). Nótese que, como la base fue previamente ordenada y la matriz es triangular superior, si una pareja de variables sobrepasa el umbral de correlación aquella de menor importancia gain será la que esté en la columna de la matriz, es decir que se excluyen variables con alta correlación y presenten menor importancia.

Esto nos deja con un número reducido de variables, de todas formas podemos volver a las variables retiradas en este paso si posteriormente eliminamos alguna variable de esta lista final por criterio de negocio o experto.

Para llegar a las variables finales se utilizó la importancia por ganancia de modelo de lightbm tomando en cuenta el principio de parsimonia para escoger un número óptimo de variables en el modelo final.

### Análisis Bivariante/ Dependencia parcial con valores shap

El análisis bivariante se realiza con 3 objetivos:

* Estudiar la distribución de cada una de las variables candidatas a puntuar en el modelo, observando la presencia de valores extraños, el porcentaje de valores ausentes, las concentraciones en determinados valores, etc. Este análisis permite por un lado estimar la calidad de la información y por otro identificar variables que, por distintas causas, no reúnen los requisitos necesarios para ser consideradas en el análisis posterior.
* Obtener una primera aproximación del poder predictivo de la variable respecto a la variable indicador de incumplimiento a nivel univariado.
* Para las variables numéricas -al igual que para las variables categóricas- se calcula el número de casos, número de buenos, número de malos, la tasa de malos y el porcentaje de casos para cada uno de los grupos definidos.

Además de analizar las posibles concentraciones y el porcentaje de valores ausentes, también se estudia si la tendencia de la tasa de malos se corresponde con la esperada.

Por otro lado, tenemos que los SHapley Additive exPlanations(SHAP) son una metodología innovadora creada para poder darle explicabilidad tanto local (utilizando LIME) como global, apoyándose de la teoría de juegos, a todo tipo de modelo, unificando varios métodos previos de tal forma que representa el único método posible de atribución de características aditivas consistente y localmente preciso. SHAP asigna a cada variable una medida de importancia en cada predicción que depende de cuánto impacta en la predicción esperada del modelo condicionado a dicha variable. Aquí se presentan los valores SHAP asignados a cada variable para cada predicción en la muestra de validación de manera gráfica, lo cual nos da una vista sobre cómo es que el modelo trabaja con cada variable(Eje X) realmente, valores SHAP positivos (eje Y) implican un incremento en log-odds respecto al valor esperado de la predicción, en este caso particular representan un incremento en la probabilidad de default (superar más de 90 días de atraso en 1 año), mientras que valores SHAP negativos representan un decremento en dicha probabilidad.

En el siguiente archivo Se detalla los gráficos de shaps (los cuales fueron elaborados con el 100% de la data de TRAIN)



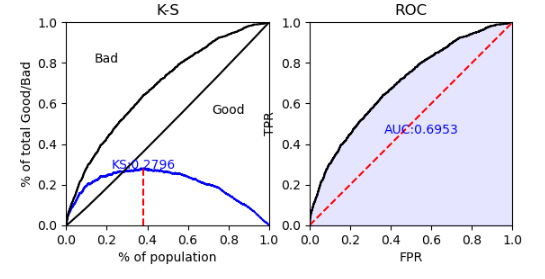
### Resultados de Perfomance del modelo

El gini del TRAIN 100% de la base, y para el OOT se usó el periodo mencionado anteriormente, cabe mencionar que se establecen como criterios de rendimiento indicadores como Gini, KS y AUC.

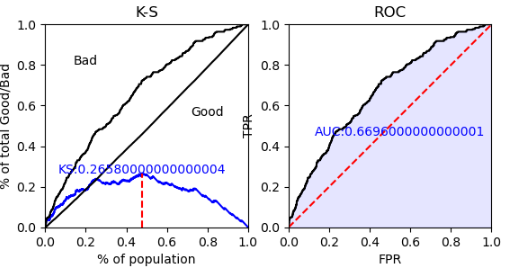
Por otro lado, cabe resalta que se considera un modelo estable si la diferencia entre los indicadores de la muestra de desarrollo y oot es menor a 5 puntos porcentuales[[1]](#footnote-1).En el anexo2 se pueden encontrar el grafico de gini a nivel periodo.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **KS** | **AUC** | **GINI** |
| **TRAIN** | 28% | 69.5% | 39% |
| **TEST** | 27% | 66.9% | 34% |
| **OOT** | 28% | 67.6% | 35% |

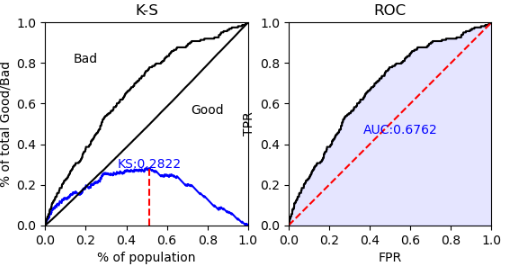
**Indicadores de train**



**Indicadores de test**

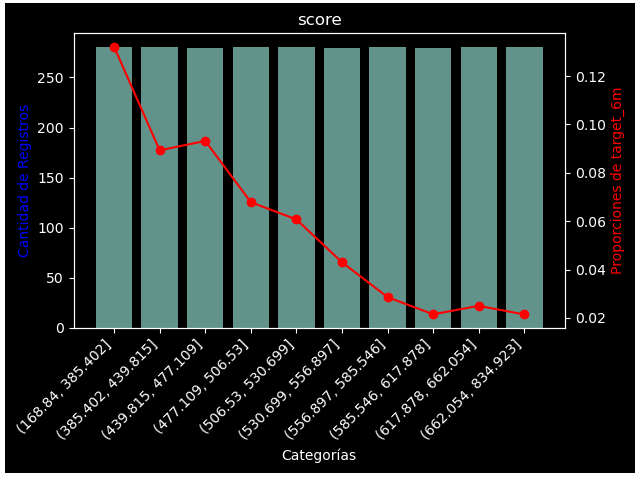


**Indicadores de oot**



**Tabla de eficiencia OOT.**

Para convertir las probabilidades al score se hizo de la siguiente manera



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Li** | **Ls** | **Buenos** | **Malos** | **Total** | **TM** | **%** | **TM ACUMULADO** |
| (168.84 | 385.402] | 243 | 37 | 280 | 13% | 10% | 5.83% |
| (385.402 | 439.815] | 255 | 25 | 280 | 9% | 10% | 5.01% |
| (439.815 | 477.109] | 253 | 26 | 279 | 9% | 10% | 4.49% |
| (477.109 | 506.53] | 261 | 19 | 280 | 7% | 10% | 3.84% |
| (506.53 | 530.699] | 263 | 17 | 280 | 6% | 10% | 3.34% |
| (530.699 | 556.897] | 267 | 12 | 279 | 4% | 10% | 2.81% |
| (556.897 | 585.546] | 272 | 8 | 280 | 3% | 10% | 2.43% |
| (585.546 | 617.878] | 273 | 6 | 279 | 2% | 10% | 2.17% |
| (617.878 | 662.054] | 273 | 7 | 280 | 3% | 10% | 2.36% |
| (662.054 | 834.923] | 274 | 6 | 280 | 2% | 10% | 2.17% |

**Importancia de variables por ganancia (gain)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Features** | **Descriptivo** | **Peso** | **Fuente** |
| max12um\_dm\_all | maximo dias de atraso total ult. 12 meses | 24.3 | RCC |
| ratio\_prm\_salrd\_1m\_3m | ratio del saldo deuda directa ult.mes sobre 3 ult.meses | 10.9 | RCC |
| min\_sldpdvtotal\_6um | mínimo pasivos en los 6um | 8.6 | PASIVOS |
| x6um\_saldo\_neg\_dct\_prin | Promedio de saldo directa en entidades principales ult. 6 meses | 7.9 | RCC |
| prm\_usotcrrstsf03m | promedio de uso de tc fuera de ibk | 6.6 | RCC |
| Edad | edad | 6.5 | RENIEC |
| prom\_dec\_12m\_total | Promedio de decremento saldo ult. 12 meses | 5.6 | RCC |
| ratio\_prm\_saldneg\_saldneg\_6m\_12m | ratio del promedio de saldo negocio ult.6 meses sobre saldo negocio .ult 12 meses | 5.6 | RCC |
| avg\_entidades\_dist\_rd\_u3m | Promedio de entidades distintas con saldo directo ult. 3 meses | 4.5 | RCC |
| x6\_numout\_trans | promedio del # de transacciones cash out en los 6um | 4.4 | TRX |
| ant\_negocio | antigüedad de negocio en meses | 4.2 | SUNAT |
| nro\_inc\_pv\_out12um | Numero de meses que incrementa el cash out en los ult. 12 meses | 3.1 | TRX |
| max\_d\_sldrctnegprin\_6um | máxima disminución de la deuda negocio en las principales entidades en los 6um | 1.9 | RCC |
| max\_d\_sldrctneg\_3um | máxima disminución de la deuda negocio en los 3um | 1.6 | RCC |
| flag\_veh | Indicador si presenta vehículo | 1.1 | SUNARP |
| flag\_x12\_valor\_impexp | tenencia de exportación e importación | 0.9 | ADEX |
| cuenta\_decre\_12m\_total | Numero de decremento saldo ult. 12 meses | 0.9 | RCC |
| x6\_avgtrpsv\_out | Promedio del promedio de la transacción del cash out | 0.6 | TRX |
| rat\_nroentidm01m24 | comparación numero entidades ult.mes sobre hace 24 meses | 0.5 | RCC |
| ct\_in\_6um | meses con cash in en los ult. 6 meses | 0.2 | TRX |
| num\_meses\_disldrctprin\_6m | # de meses que disminuye en las principales entidades en los 6um | 0.1 | RCC |

**Observación:** el proceso de segmentación se llevó acabo solo para poder entregar puntos de cortes referenciar al equipo de estrategias, ya que lo que se entrega como resultado final son las probabilidades del modelo mas no la segmentación para más detalle de los resultados se muestran en el Anexo 8

### Software utilizado:

A continuación se presenta el software utilizado para todos los procesos Mencionados:

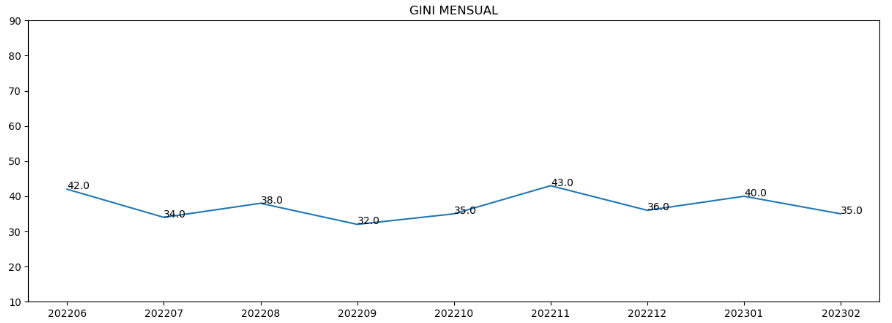
1. Bases en la nube: Archivo parquet S3 – AWS
2. Modelamiento: Jupyter Hub - Python 3

## ANEXO

* Anexo1

|  |  |
| --- | --- |
|  | Modelo |
| Target |  |
| Diccionario de variables |  |
| Análisis Bivariado |  |
| Importancia de variables |  |
| Objeto del Modelo LIGHTGM |  |

* Anexo 2



* Anexo 3



* Anexo 4
* Anexo 5



* Anexo 6

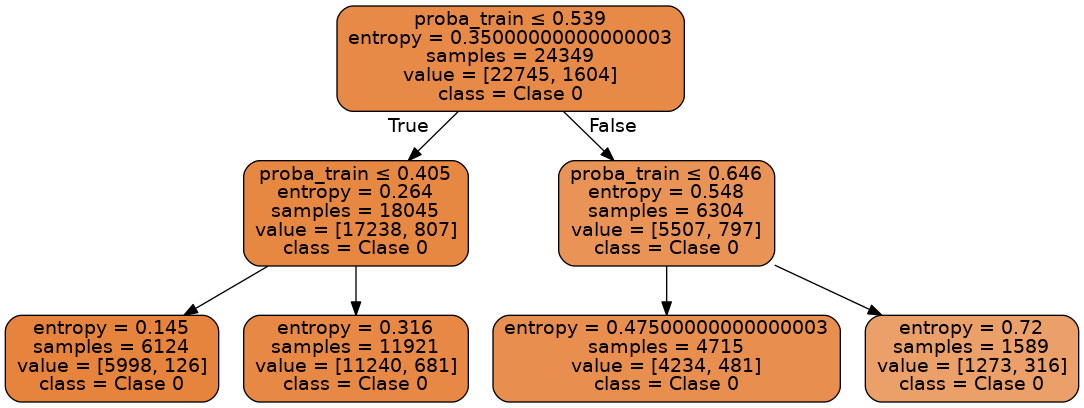
|  |  |
| --- | --- |
| **Hiperparámetros** | **Rango Valores** |
| max\_depth | 2,3,4,5 |
| num\_leaves | 3,4,5 |
| learning\_rate | Desde 0.01 hasta 0.1 |
| min\_child\_samples | Desde 100 hasta 400 |
| feature\_fraction | Desde 0.1 hasta 1 |
| bagging\_fraction | Desde 0.1 hasta 1 |
| lambda\_l1 | Desde 2 hasta 5 |
| lambda\_l2 | Desde 2 hasta 5 |
| bagging\_freq | 5 |

* Anexo 7

### Segmentación

Para la obtención de los grupos, se realizó usando arboles de decisión tomando como variable independiente la probabilidad del modelo binario y la target como dependiente, se entrenó el modelo con toda la información es decir con los 24,349 desde 202206 -202302

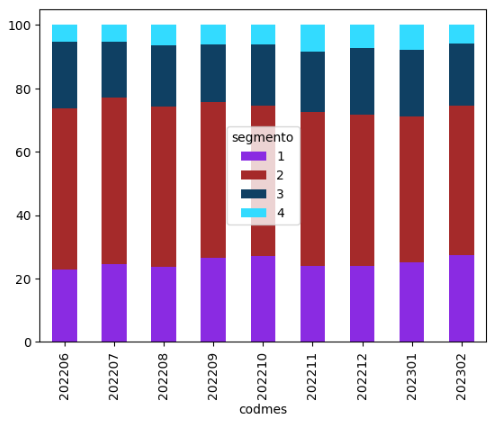
En el anexo 4 se muestra las librerías usadas para construir el árbol.



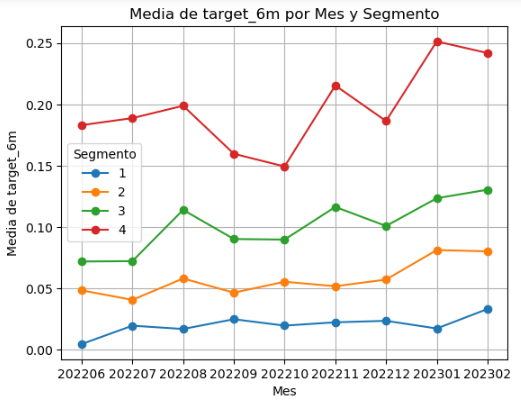
### Resultados

Se obtuvieron 4 segmentos en base a los resultados del árbol, para lo cual se presentan la materialidad a nivel segmentos y periodos, se incluyen los gráficos de la tasa de malos de dichos segmentos.

|  |  |  |  |
| --- | --- | --- | --- |
| **Segmento** | **Total** | **% Total** | **%Mora** |
| S1 | 6094 | 25% | 2% |
| S2 | 11917 | 49% | 6% |
| S3 | 4749 | 20% | 10% |
| S4 | 1589 | 7% | 20% |

**% de Materialidad**

% tasa de Malos



## RESPONSABILIDAD DE ÁREA Y ÓRGANOS INTERVENIENTES

En cada una de las etapas descritas, se han tenido reuniones periódicas presenciales entre las siguientes áreas:

* Áreas de Riesgos:
  1. Modelamiento de Riesgos.
  2. Departamento de Admisión de Riesgo Empresarial.

En dichas reuniones las decisiones de ejecución del desarrollo han sido acordadas por todas las partes.

1. Dentro de los márgenes que manejamos el banco aceptamos una caída máxima de 5 puntos. [↑](#footnote-ref-1)