



**Facultad de Ciencias Económicas y Empresariales**

Director: Antonio Tena Blázquez

**Aplicación de modelos de Business Analytics en  
la gestión del riesgo y rentabilidad del sector  
Buy Now Pay Later**

**Trabajo de Fin de Grado**

Autor: Gonzalo Cabañas Alonso

Madrid

2026

## Resumen

El presente Trabajo de Fin de Grado analiza la aplicación de modelos de *Business Analytics* en la gestión del riesgo y la rentabilidad dentro del sector *Buy Now, Pay Later* (BNPL), en un contexto caracterizado por el crecimiento acelerado de este modelo de financiación y el incremento simultáneo de las exigencias regulatorias derivadas de la nueva Directiva europea de crédito al consumo y la presión por parte del CFPB. El objetivo principal consiste en evaluar si es posible diseñar un motor analítico capaz de integrar evaluación de solvencia, estimación probabilística del riesgo y toma de decisiones operativas coherentes con criterios de sostenibilidad económica y cumplimiento del nuevo marco normativo.

Para ello, el trabajo desarrolla un modelo estructurado en cuatro capas: parte de una generación de una base de datos sintética representativa del comportamiento crediticio en entornos BNPL. A partir de estos datos se define un *ground truth* como referencia de riesgo subyacente. Después, se aplica un *affordability check* determinista alineado con el principio de préstamo responsable. Posteriormente, se estima la probabilidad de impago mediante modelos de *credit scoring*, comparando regresión logística y *gradient boosting*. Finalmente, se incorpora un motor de decisión capaz de traducir las estimaciones probabilísticas en límites operativos de exposición crediticia ajustados al riesgo individual de cada operación.

## Palabras clave

Buy Now, Pay Later (BNPL), *Business Analytics*, riesgo crediticio, *credit scoring*, *gradient boosting*, regresión logística, *affordability check*, probabilidad de impago, pérdida esperada (*Expected Loss*), regulación financiera, CCD II, solvencia *ex ante*, modelos predictivos, base de datos sintética, toma de decisiones crediticias.

## Abstract

This Final Degree Project analyzes the application of Business Analytics models to risk management and profitability in the Buy Now, Pay Later (BNPL) sector, within a context characterized by the rapid growth of this financing model and the simultaneous increase in regulatory requirements arising from the new European Consumer Credit Directive and the growing supervisory pressure from the CFPB. The main objective is to assess whether it is possible to design an analytical decision engine capable of integrating creditworthiness assessment, probabilistic risk estimation, and operational decision-making consistent with criteria of economic sustainability and compliance with the new regulatory framework.

To this end, the study develops a model structured in four layers. It begins with the generation of a synthetic dataset representative of credit behavior in BNPL environments. Based on these data, a ground truth is defined as a reference for underlying credit risk. Next, a deterministic affordability check aligned with the principle of responsible lending is applied. Subsequently, the probability of default is estimated through credit scoring models, comparing logistic regression and gradient boosting techniques. Finally, a decision engine is incorporated to translate probabilistic estimates into operational credit exposure limits adjusted to the individual risk level of each transaction.

## Keywords

Buy Now, Pay Later (BNPL), Business Analytics, credit risk, credit scoring, gradient boosting, logistic regression, affordability check, probability of default, Expected Loss, financial regulation, CCD II, ex ante creditworthiness assessment, predictive models, synthetic dataset, credit decision-making.

# Índice

Resumen.....	2
Palabras clave.....	2
Abstract.....	3
Keywords .....	3
1. Introducción .....	6
1.1 Contexto y crecimiento del modelo Buy Now Pay Later.....	6
1.2 Problema de investigación .....	10
1.3 Objetivos del trabajo .....	11
2. Marco conceptual.....	13
3. Modelo operativo .....	15
3.1 Generación de la base de datos sintética .....	15
3.2 Nivel 0 – <i>Ground Truth</i> .....	16
3.3 Nivel 1 – Capa determinística .....	17
3.4 Nivel 2 – Capa probabilística .....	19
3.5 Nivel 3 – Motor de decisión.....	23
4. Resultados y conclusiones .....	28
4.1 Conclusiones de los métodos de <i>Business Analytics</i> .....	29
4.2 Conclusiones del modelo operativo .....	30
5. Bibliografía .....	33
6. Anexos.....	35
7. Declaración de uso de Herramientas de Inteligencia Artificial Generativa.....	45

## Índice de ilustraciones

Ilustración 1. Porcentaje de usuarios de BNPL que han usado los siguientes proveedores .....	7
Ilustración 2. Ticket promedio de los siguientes proveedores .....	7
Ilustración 3. Tipología de clientes del servicio.....	8
Ilustración 4. Número de usuarios por generación .....	8
Ilustración 5. Comparación métodos Business Analytics .....	21
Ilustración 6. Resultados umbral.....	23
Ilustración 7. Tabla de conclusiones y resultados del trabajo .....	28

# 1. Introducción

## 1.1 Contexto y crecimiento del modelo Buy Now Pay Later

El Buy Now Pay Later (BNPL) se ha consolidado como un esquema de financiación de corto plazo integrado en el proceso de pago, principalmente en comercio electrónico, que permite fraccionar una compra en cuotas mediante un tercero. Habitualmente sin intereses explícitos, aunque sí se suelen aplicar en transacciones con planes largos o tickets altos, donde varían de un 10% a 30% anual (Deloitte, 2025).

Internamente el modelo funciona de la siguiente manera: los comercios incorporan el servicio de BNPL en la caja online como opción de pago. Cuando el cliente elige la opción, el importe total de la compra se divide en cuotas (entre 2 y 6 pagos) sin intereses. La aprobación de esta financiación es prácticamente instantánea ya que no sigue los procesos de crédito tradicionales. La empresa de BNPL asume el riesgo de crédito del consumidor y paga al comercio inmediatamente, reteniendo una comisión. De esta forma, la tienda paga una comisión mayor que las tarjetas de crédito tradicionales, a cambio de mayor conversión y tique promedio. Si el cliente llegara a incurrir en impago, se aplica una *late fee* (fija, acumulativa o con tope máximo).

Este modelo se aprovecha de distintas exenciones para poder quedarse fuera de la normativa de crédito tradicional. La primera sería la exención por crédito sin interés, al lograr un 0% APR <sup>1</sup>explícito no cuenta formalmente como crédito oneroso, a pesar de que materialmente sí genera deuda. La segunda sería por su corta duración, menos de 90 días, y bajo importe, que históricamente se excluían para reducir la carga regulatoria. Aquí el riesgo no surge de una operación si no de la acumulación. La tercera exención es por pago diferido, al existir una diferencia legal entre el pago aplazado y una concesión de crédito, las empresas BNPL se podían acoger bajo la primera definición, considerándose un método de pago y no un prestamista. Adicionalmente, al ser un incumplimiento contractual, las *late fees* no cuentan como intereses.

Este concepto tal y como lo conocemos hoy en día surge a mediados de 2010 cuando algunas empresas tecnológicas especializadas en pagos empiezan a ofrecer financiamiento a corto plazo sin pasar por canales bancarios tradicionales. Comienza en EEUU, aunque a finales de la década se empieza a expandir por Europa, Australia y partes de Asia. Su rápida y amplia

---

<sup>1</sup> APR – *Annual Percentage Rate* (Tasa Porcentual Anual)

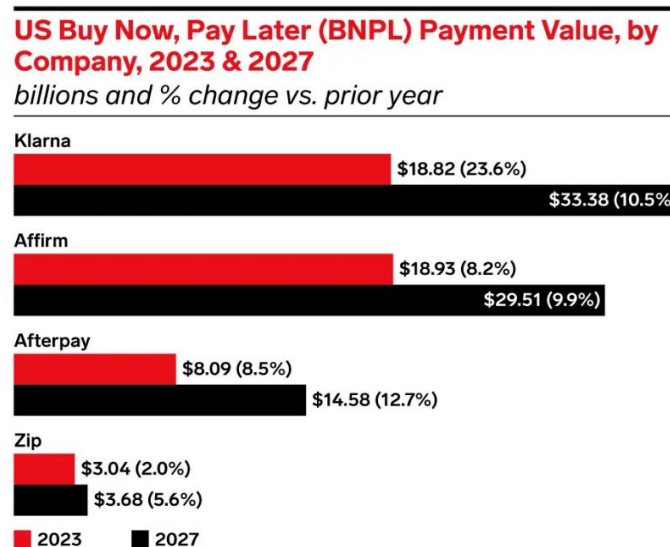
expansión se explica por las fricciones reducidas en *checkout*, la digitalización de pagos, la adopción masiva de canales móviles y en especial, la preferencia del consumidor por alternativas al crédito tradicional en importes medios y bajos. En términos de estructura competitiva, el mercado estadounidense ocupa una posición central en el desarrollo y consolidación del modelo BNPL, al concentrar algunos de los principales operadores y marcar tendencias que posteriormente se trasladan a otras regiones.

*Ilustración 1. Porcentaje de usuarios de BNPL que han usado los siguientes proveedores*



Fuente: Statista. (2023). *Chart: Buy Now, Pay(Pal) Later.*

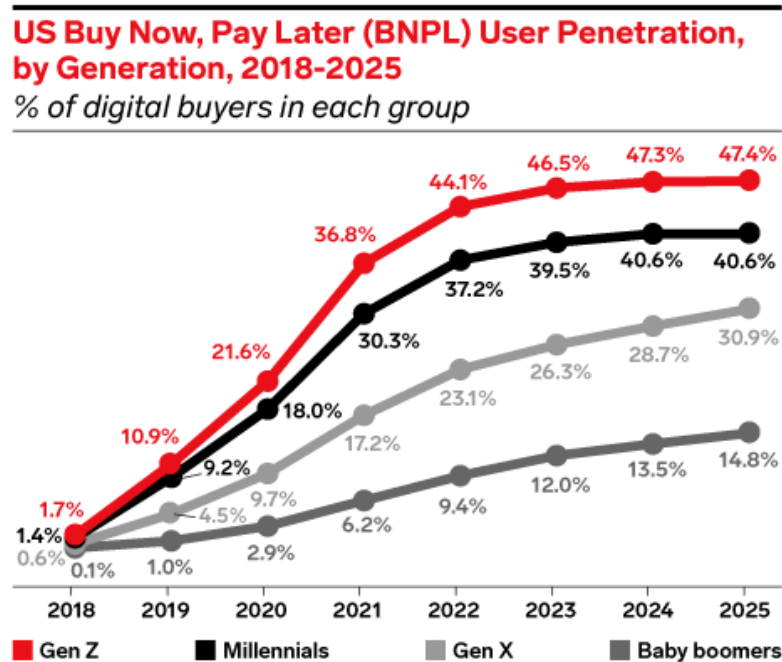
*Ilustración 2. Ticket promedio de los siguientes proveedores*



Fuente: Insider Intelligence (now EMARKETER). (2024). *Guide to Buy Now, Pay Later: Industry trends, regulation, and market data*

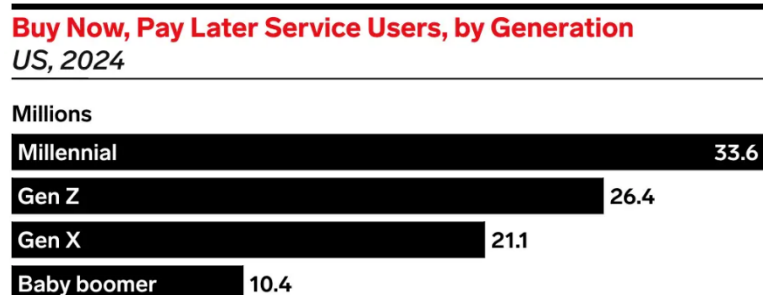
Destacan Affirm y Afterpay, con una fuerte presencia tanto en Estados Unidos como en Australia, mientras que Klarna lidera el mercado europeo. A escala global, PayPal mantiene una posición dominante gracias a su amplia base de usuarios y comercios, actuando además como vector de estandarización del BNPL en el ecosistema de pagos digitales).

Ilustración 3. Tipología de clientes del servicio



Fuente: Insider Intelligence (now EMARKETER). (2024). *Guide to Buy Now, Pay Later: Industry trends, regulation, and market data*

Ilustración 4. Número de usuarios por generación



Fuente: Insider Intelligence (now EMARKETER). (2024). *Guide to Buy Now, Pay Later: Industry trends, regulation, and market data*.

El sector presenta una concentración en segmentos de consumidores jóvenes, especialmente entre millennials y Generación Z (18 a 35 años), que representan más de 60 millones de clientes solo en Estados Unidos (Insider Intelligence, 2024). Este perfil de cliente se caracteriza por una elevada adopción de canales digitales y una menor preferencia por el crédito tradicional. Este perfil suele presentar historiales crediticios más limitados, ingresos medios inferiores y una mayor volatilidad en sus flujos de renta. Asimismo, diversos estudios señalan que los consumidores jóvenes tienden a utilizar el BNPL de forma recurrente y a mantener múltiples planes de pago activos de manera simultánea, incrementando el riesgo de acumulación de deuda y de estrés financiero.

Desde la perspectiva de método de pago, BNPL ha pasado de ser marginal a representar una cuota relevante en e-commerce global. Para 2021-2022 algunas estimaciones sitúan el mercado en 120.000 millones de USD, creciendo de forma exponencial desde valores mucho menores en 2015. Mientras que en 2024 informes sitúan el valor global en cifras cercanas a los 600.000 millones de USD. Con proyecciones de hasta 900.000 millones de USD en 2030 (Latinia, 2024). Este aumento se ha visto reforzado por la estandarización técnica (APIs<sup>2</sup>, embedded finance<sup>3</sup>) y por el interés comercial de los merchants (mayor conversión y ticket), a la vez que ha ampliado su alcance desde categorías discrecionales (moda, electrónica) hacia usos más cotidianos.

Esta expansión ha tenido como contrapartida una mayor atención de supervisores y reguladores por su naturaleza crediticia y por los riesgos asociados a sobreendeudamiento, opacidad del coste total, y acumulación de múltiples planes en distintos proveedores. En EE.UU. el CFPB (*Consumer Financial Protection Bureau*) ha intensificado la lectura de BNPL como producto con externalidades de riesgo comparables al crédito al consumo. En Europa, la expansión del BNPL se enmarca en un movimiento regulatorio explícito: la revisión del marco de crédito al consumo amplía el perímetro para cubrir formas de crédito gratuitas o de bajo importe que históricamente podían quedar fuera del alcance de la Directiva anterior (European Parliament & Council of the European Union, 2023). A escala internacional, organismos de

---

<sup>2</sup> APIs – Application Programming Interface. Protocolos que permiten que dos softwares se comuniquen.

<sup>3</sup> Embedded Finance – Integración directa de servicios financieros dentro de productos no financieros.

supervisión de conducta han señalado de forma consistente la necesidad de fortalecer evaluaciones de asequibilidad y prácticas de comercialización y cobro en productos de crédito digital de corto plazo, incluyendo BNPL (European Banking Authority, 2020). Es en esta intensificación de las regulaciones donde surge el principal problema del sector y el problema central de este trabajo.

## 1.2 Problema de investigación

La hipótesis de partida es que el BNPL está dejando de operar bajo excepciones históricas asociadas a productos sin interés o de corto plazo, y está siendo reencuadrado como crédito al consumo. Este cambio altera de forma estructural el equilibrio competitivo del sector: el crecimiento basado en aprobación ligera y fricción mínima entra en conflicto con exigencias regulatorias de préstamo responsable, evaluación de solvencia/asequibilidad, límites a penalizaciones y restricciones de comercialización.

La Unión Europea a través de la Directiva (UE) 2023/2225 (CCD II) actualiza el marco del crédito al consumo para abarcar nuevas formas de crédito digital y refuerza obligaciones de información, publicidad y evaluación de la capacidad del consumidor para atender sus obligaciones. De esta forma, la CCD II elimina la exclusión automática de crédito sin intereses (European Parliament & Council of the European Union, 2023).

Por otro lado, aunque el perímetro legal estadounidense difiere del europeo, el CFPB ha consolidado un enfoque de supervisión basado en análisis de mercado del BNPL (uso, late fees, desempeño, y riesgos de acumulación). Esto eleva el umbral de escrutinio sobre prácticas comerciales, transparencia del coste y gestión de cobros.

Esta reestructuración del sector tiene efectos reales en las operaciones de este tipo de modelos. Siendo los principales: la menor tasa de aprobación, reduciendo el volumen y crecimiento de clientes. Debido a la incorporación de *affordability checks* (evaluaciones de capacidad real de pago del consumidor) y la aplicación de estándares de préstamos, que inhabilitan la admisión de cliente con información incompleta o mayor riesgo. El aumento del coste operativo, ya que las nuevas obligaciones implican verificación de evidencia de decisión crediticia, trazabilidad del modelo, controles de gobernanza, además de, el rediseño de comunicación y marketing. La restricción de ingresos por penalizaciones y prácticas de cobro, impactando en el margen. Prohibiendo parcialmente las *late fees*, al ser penalizaciones acumulativas desproporcionadas que afectan a consumidores vulnerables y que incentivan

modelos laxos de *underwriting*. Ya que se había convertido en una fuente relevante de ingresos marginales.

Es en este nuevo contexto regulatorio donde surge la necesidad de las empresas BNPL de adoptar nuevos modelos de *business analytics* que optimicen el proceso de aprobación crediticia de acuerdo con las nuevas obligaciones (trazabilidad, *affordability checks*, gobernanza). Desde un punto de vista empresarial, la adaptación temprana a este marco no solo resulta necesaria para garantizar el cumplimiento regulatorio, sino que constituye una potencial ventaja competitiva frente a aquellos operadores que retrasen la transformación de sus modelos analíticos y operativos.

### 1.3 Objetivos del trabajo

La hipótesis de este trabajo es que una aplicación de un modelo de *Business Analytics* basado en una evaluación de la asequibilidad y un sistema de *credit scoring* explicable permite mantener una rentabilidad positiva en operaciones BNPL, incluso bajo un marco regulatorio que exige mayores requisitos de préstamo responsable. De esta forma, el objetivo principal de este trabajo es analizar si la aplicación de dicho modelo permite gestionar de forma eficiente el riesgo crediticio y mantener la rentabilidad del modelo BNPL en un contexto regulatorio más restrictivo.

A partir de una combinación de variables socioeconómicas y transaccionales, se busca construir un modelo capaz de predecir la probabilidad de incumplimiento de pago por parte de los clientes, permitiendo así mejorar los procesos de toma de decisiones en la concesión de microcréditos a corto plazo y siguiendo las nuevas obligaciones impuestas por los organismos regulatorios.

Como objetivos específicos se plantean: Analizar la evolución y particularidades del sector BNPL. Desarrollar e implementar un modelo de cuatro capas, compuesto por un módulo de generación de entorno sintético de riesgo, una capa determinista de evaluación de la asequibilidad, un modelo probabilístico de estimación de la probabilidad de impago mediante técnicas de *credit scoring* y un motor de decisión operativa que transforma la señal de riesgo en límites de exposición. De esta forma podemos, separar conceptualmente la capacidad de pago del consumidor, es decir, si puede asumir la operación, de su riesgo crediticio, es decir, la probabilidad de que incumpla sus obligaciones. Evaluar el rendimiento del modelo mediante métricas estándar de clasificación y calibración probabilística. Analizar las implicaciones

operativas de su implementación en términos de gestión conjunta del riesgo y la rentabilidad en un entorno regulatorio más restrictivo.

## 2. Marco conceptual

El presente trabajo se apoya en un conjunto de conceptos teóricos y operativos procedentes de la literatura sobre crédito al consumo, gestión del riesgo crediticio y *Business Analytics*. Este marco conceptual tiene como objetivo establecer las bases necesarias para comprender el problema de investigación, justificar el enfoque metodológico adoptado y delimitar los elementos clave que intervienen en la toma de decisiones en el sector.

El crédito al consumo se define como cualquier forma de financiación concedida a personas físicas para fines no profesionales, normalmente sin garantía real y con un horizonte temporal limitado. Según la Comisión Europea, este tipo de crédito incluye préstamos personales, créditos rotativos, pagos aplazados y otros instrumentos que permiten al consumidor diferir el pago de bienes o servicios (European Parliament & Council of the European Union, 2023). El rasgo distintivo del crédito al consumo no garantizado es que el prestamista asume el riesgo de impago basándose casi exclusivamente en la evaluación del prestatario, sin colateral que mitigue la pérdida en caso de incumplimiento. Actualmente el modelo BNPL, aunque históricamente presentado como un método de pago, es considerado por los cuerpos reguladores (Unión Europea, CFPB, Departamento del Tesoro de Australia), parte de esta categoría al permitir el acceso inmediato a un bien o servicio a cambio de pagos futuros, asumiendo el proveedor BNPL el riesgo de crédito del consumidor.

El riesgo de crédito se define como la probabilidad de que un prestatario no cumpla con sus obligaciones financieras según los términos acordados. En la literatura financiera, este riesgo se descompone habitualmente en tres componentes: probabilidad de impago (*Probability of Default*, PI), pérdida en caso de impago (*Loss Given Default*, LGD) y exposición en el momento del impago (*Exposure at Default*, EAD). En productos de crédito al consumo de corto plazo, como el BNPL, la atención se centra principalmente en la PI, dado que la LGD suele ser elevada (es un crédito no garantizado) y la EAD coincide generalmente con el importe financiado (Basel Committee on Banking Supervision, 2025). Por tanto, la capacidad del modelo para discriminar entre clientes con distinta probabilidad de impago resulta crítica para la sostenibilidad económica del producto.

El concepto de asequibilidad hace referencia a la capacidad del consumidor para asumir una nueva obligación crediticia sin comprometer su estabilidad financiera ni el cumplimiento de sus gastos esenciales. A diferencia de la mera estimación de probabilidad de impago, la asequibilidad incorpora una dimensión normativa y preventiva orientada a la protección del

consumidor. Los marcos regulatorios recientes en la Unión Europea, Reino Unido y Australia incorporan explícitamente el principio de préstamo responsable, que exige a los prestamistas evaluar si el crédito es adecuado para el consumidor, y no únicamente si es rentable para la entidad.

El *scoring* crediticio es el proceso mediante el cual se asigna una puntuación o probabilidad a un solicitante de crédito en función de sus características observables, con el objetivo de predecir su comportamiento futuro de pago.

Entre los modelos de *Business Analytics* analizados en este trabajo destacan la regresión logística como modelo explicable de estimación de la probabilidad de impago y los métodos de *gradient boosting* como alternativa no lineal orientada a mejorar la capacidad predictiva.

La regresión logística es un modelo estadístico de clasificación binaria que estima directamente la probabilidad de ocurrencia de un evento a partir de una combinación lineal de variables explicativas transformada mediante una función logística. Esta transformación permite acotar las predicciones entre 0 y 1, lo que facilita su interpretación.

El *gradient boosting* es un método de aprendizaje supervisado basado en la combinación secuencial de múltiples árboles de decisión poco profundos, construidos de forma iterativa para corregir los errores cometidos por los modelos anteriores. En cada iteración, el algoritmo ajusta un nuevo árbol sobre los residuos del modelo previo, mejorando progresivamente la capacidad predictiva del conjunto. Este procedimiento permite capturar relaciones no lineales e interacciones complejas entre variables que no pueden representarse mediante modelos lineales tradicionales.

### 3. Modelo operativo

El modelo ha sido construido siguiendo una arquitectura modular estructurada en cuatro niveles secuenciales. Esta organización permite separar conceptualmente la generación del entorno de riesgo, la evaluación de la capacidad de pago, la estimación probabilística del incumplimiento y la decisión operativa final. Además, previo a la construcción del modelo, existe la generación de la base de datos sintética, cuyas características se describen en un apartado independiente.

De forma conceptual, el Nivel 0 genera el entorno sintético de comportamiento de pago, el Nivel 1 evalúa la asequibilidad mediante reglas deterministas, el Nivel 2 estima la probabilidad de impago mediante modelos de *scoring* y el Nivel 3 transforma estas señales en decisiones de aceptación, limitación o rechazo junto con una estimación de pérdida esperada.

#### 3.1 Generación de la base de datos sintética

El primer paso para la construcción del modelo es encontrar una base de datos apropiada, con los datos necesarios. El primer problema del modelo surge aquí ya que encontrar datos con estas características es sumamente difícil debido a las limitaciones estructurales del BNPL. Este sector existe bajo un marco regulatorio muy estricto acerca de la confidencialidad y protección de datos ya que contienen información de carácter financiera y personal sensible, como pueden ser los ingresos, deuda o impagos. Por otro lado, la mayoría de la base de datos existentes pertenecen a entidades privadas que no permiten el acceso con fines académicos.

Por esta razón, he optado por el uso de una base de datos sintética. Diseñada para replicar distribuciones y relaciones económicas plausibles observadas en el riesgo crediticio, incorporando variables sociodemográficas, de endeudamiento y transaccionales relevantes, coherente con la literatura financiera. El uso de datos sintéticos permite un mayor control experimental sobre el proceso de generación del riesgo, facilita la reproducibilidad del estudio y resulta adecuado para el análisis comparativo de modelos de *Business Analytics* orientados a la gestión del riesgo y la rentabilidad, que constituye el objetivo principal de este trabajo.

Esta base de datos sintética ha sido generada a través de un código *Python*. A nivel conceptual las variables reflejan patrones observados en el mercado BNPL. Predominan perfiles jóvenes, Generación Z y *millennials*, con ingresos moderados, menor antigüedad laboral y mayor propensión al uso de pagos aplazados. Las variables de endeudamiento y capacidad de pago se construyen de forma coherente con los ingresos y los gastos fijos, evitando combinaciones irreales. La base de datos no se genera de forma completamente

aleatoria. Existen dependencias explícitas entre variables clave, por ejemplo: la edad condiciona el nivel de ingresos y la antigüedad laboral. Los ingresos influyen en los gastos fijos, el endeudamiento y el importe de las compras. El número de cuotas y el importe de cada cuota están directamente relacionados con el importe de la compra. El riesgo crediticio (impagos pasados, historial crediticio) se asocia a perfiles financieros más frágiles.

A nivel técnico el código utiliza las librerías, *NumPy* para la generación de valores aleatorios y el control de distribuciones estadísticas. *Pandas* para la construcción y gestión de la base de datos en formato tabular. Y *Datetime* para la simulación de fechas de compra realistas.

### 3.2 Nivel 0 – *Ground Truth*

El siguiente paso es la creación de un nivel 0 del modelo o *Ground Truth*, que representa una realidad simulada coherente, que actúa como referencia para el entrenamiento y validación de modelos posteriores. Es importante entender que este primer nivel no toma decisiones, sino que produce el *target (default)* del modelo. En este caso, el *default* es el incumplimiento de una obligación de pago según las condiciones pactadas, ya que es la variable clave que convierte el riesgo en algo medible. Esta variable se modela como binaria ya que esto permite entrenar modelos probabilísticos, calcular la PI y evaluar carteras y políticas. Un *default* de 1 significa que el cliente no completa el plan de pagos propuesto por su proveedor de BNPL o que la operación se considera fallida. Por simplicidad no tiene en cuenta retrasos leves ni pagos imparciales.

Esta primera capa descarga la base de datos del escritorio y construye dos ratios clave a partir de los datos. El primero es el *DSTI (debt service to income)*, este ratio mide presión financiera total que soporta el cliente. Es uno de los indicadores más utilizados en modelos de riesgo crediticio, ya que captura la sostenibilidad del endeudamiento. Se calcula sumando la deuda mensual existe y la cuota mensual y dividirlo por el ingreso mensual. El segundo es la cuota BNPL sobre el ingreso residual. Este ratio evalúa el peso de la cuota BNPL una vez descontados los gastos fijos, aproximando la liquidez real disponible. Cabe destacar que se introduce un pequeño término epsilon para evitar divisiones por cero, lo cual no altera la lógica económica, pero garantiza estabilidad numérica.

La siguiente parte del código se centra en asignar una penalización para los posibles valores de cada variable. Se tiene en cuenta: Estado de vivienda, un cliente con un estatus de

vivienda de alquiler se percibe como más peligroso que uno que tenga el estatus de vivienda en propiedad. La edad, existe un leve incremento en riesgo en los extremos, tanto los más jóvenes como los más mayores. Las categorías de producto, por ejemplo, los productos electrónicos o viajes se perciben como más arriesgados que productos farmacológicos o de alimentación. El historial crediticio, codificado con una función logarítmica de tal forma que haya rendimientos decrecientes, es decir, los primeros meses impactan más. Y los *defaults* pasados, que tienen un alto impacto. Estas penalizaciones no pretenden estimar efectos causales reales, sino introducir correlaciones plausibles observadas en el crédito al consumo, con el objetivo de generar un entorno sintético realista. Por otro lado, el código añade una pequeña cantidad de ruido para que los datos sintéticos sean lo más realistas posible.

Una vez ha asignado todos estos valores, se construye un *score* de riesgo agregado, combinando así en una sola magnitud numérica toda la información relevante de riesgo, creando así un índice latente de riesgo. El *score* se basa en una suma ponderada de los ratios financieros clave, ya que no todos tienen el mismo peso, la penalización estructural de cada variable y un intercepto, que actúa como base de riesgo del portfolio, fijado externamente para calibrar la tasa global de impago del entorno BNPL simulado.

A partir del *score*, se usa una función logística para que el valor esté entre 0 y 1, a este valor le denominamos PI (Probabilidad de impago). A su vez con un generador de números aleatorios, se asigna a cada cliente un número entre 0 y 1, al que denominamos  $u$ . Con estos dos datos se realiza una simulación del impago, comparando el número aleatorio  $u$  con la PI. Si  $u$  es menor que la PI, se asigna el impago 1 y viceversa. Este cálculo se realiza para convertir una probabilidad en un resultado binario. Cuando la simulación acaba se calcula la tasa de *default* sintética agregada y por segmentos, como mecanismo de validación del entorno generado. Esto sirve de validación del modelo, ya que nos permite asegurarnos que la frecuencia de defaults es coherente con un entorno BNPL realista. Finalmente se genera una nueva base de datos, añadiendo el *default* simulado.

### 3.3 Nivel 1 – Capa determinística

El Nivel 1 del modelo tiene como objetivo transformar la información bruta de la base de datos sintético en una representación estructurada del riesgo, previa a cualquier proceso de aprendizaje automático. A diferencia del Nivel 0, que define la realidad simulada mediante la generación del evento de default, este nivel actúa como una capa determinista de construcción

del riesgo. Donde se sintetiza la información económica y financiera de cada cliente en indicadores deterministas de capacidad de pago, previos a cualquier estimación probabilística del riesgo.

Esta capa parte de la base de datos generada. El primer paso es un simple chequeo de que las variables tengan los valores numéricos y que, si falta un dato no de error más adelante, adecuándolos a su procesamiento. El siguiente paso es clave para el modelo, crea un diccionario con los umbrales de decisión del proveedor BNPL. Su fácil acceso permite tanto cambiar los valores como facilitar la validación de las decisiones tomadas. Este bloque determinista traduce la estrategia de riesgo en reglas operativas. A partir de este diccionario hace *checks* deterministas a través de condiciones booleanas para comprobar que cada cliente cumple las condiciones. Es un motor de reglas previo a *scoring* que da trazabilidad a las decisiones de por qué se aprobó/denegó, clave para *governance*, y controla la exposición por *affordability*.

Este proceso termina con una decisión final de si el cliente se lo puede permitir (en cuyo caso devuelve 1) o no (devuelve 0 y la razón). Básicamente implementa un *hard cut*, si una condición crítica falla, se deniega. Aquí no solo obtenemos una variable 0 o 1 si no que también nos devuelve las razones de rechazo, pieza clave para valorar los ajustes de la política de decisión del proveedor BNPL.

El siguiente paso del modelo de *affordability* es calcular los límites de asequibilidad de cada cliente. Es decir, cuál es la cuota máxima que se puede permitir con sus finanzas y política de decisión actual. Se usan dos límites distintos, el primero por DSTI, es decir por su sobreendeudamiento, y el segundo por el residual mensual, es decir la liquidez neta del cliente. Elige el más restrictivo de los dos (el más bajo). Esto permite que los proveedores dimensionen las ofertas, ya que existen varias opciones de plazos. Esto abre vías alternativas de financiación y aumenta la base de clientes potenciales.

Finalmente muestra un resumen agregado del *pass rate* y de las principales razones de rechazo, mientras que a nivel individual devuelve el límite máximo de cuota y de importe financiable. Actúa como un *dashboard* para iterar políticas y documentar el impacto.

### 3.4 Nivel 2 – Capa probabilística

Este nivel implementa la capa probabilística: entrena modelos estadísticos de *Machine Learning* para estimar una PI a partir de variables ex-ante <sup>4</sup>. Lo primero que hace es verificar que existe la variable *default*, que viene de capas anteriores. Además, define *Y* como la variable objetivo binaria. Este paso es clave para asegurarse que el modelo sea entrenado con *target*.

A continuación, se construye la matriz de variables explicativas *X*, para esto define las columnas que no deben entrar en el modelo, por ejemplo, *default*, que es el *target*, *risk score*, o *affordability reasons*, que es texto. A partir de estas dos variables, se hace la división para entrenar el modelo, en este caso se entrena con los datos más antiguos, ya que están ordenados temporalmente, y se prueban con los más recientes. La división es 80% entrenamiento, 20% prueba, una convención empírica diseñada para maximizar el aprendizaje y preservar la capacidad de evaluación. En caso de que no hubiese suficiente información temporal, el código está diseñado para dividir los datos aleatoriamente.

El preprocesado continúa con una reclasificación de variables, las de texto se guardan como variables categóricas y el resto como numéricas. Además, cambia el formato de la fecha a *datetime* para que el modelo no se equivoque y lo interprete como un número “crudo”. Después, establece una serie de transformaciones secuenciales o *pipelines* para los dos tipos de variables. La primera de las numéricas es una imputación con mediana, es decir, sustituye los valores nulos por la mediana de la columna, evitando así que el modelo falle por valores nulos. La segunda es un escalado estándar de todos los valores numéricos, evitando que las variables con magnitudes grandes (ej. ingresos) dominen a otras pequeñas (ej. número de *defaults* pasados), este paso es clave ya que la regresión logística es sensible a escalas, aunque en modelos de *boosting* no es necesario.

La primera de las categóricas es una imputación con moda, mismo concepto que la imputación con mediana, pero orientada a textos, el que más se repita es el que reemplazara los huecos. La segunda es el One-Hot-Encoding, que convierte cada categoría en binaria, ya que los modelos supervisados no trabajan con texto. Este paso convierte información cualitativa en señal cuantitativa. Para hacer esto cogen una categoría, por ejemplo, *housing\_status* = {*rent*, *own*, *mortgage*} y la transforma en *housing\_status\_rent*, *housing\_status\_own*, *housing\_status\_mortgage*. De esta forma si un cliente es *rent* este sería {1, 0, 0}.

---

<sup>4</sup> Antes del impago

Una vez terminado el paso de preprocesado, se constituye el núcleo de la capa probabilística del modelo, ya que en ella se definen formalmente los algoritmos que estimarán la probabilidad de impago a partir de la información estructurada en las capas anteriores. Técnicamente, se construyen dos series de transformaciones secuenciales completas que integran el preprocesado definido previamente con dos clasificadores distintos: una regresión logística como modelo base y un *gradient boosting classifier* como alternativa no lineal de mayor capacidad predictiva.

La regresión logística modeliza directamente la probabilidad condicional de default mediante una función logística aplicada a una combinación lineal de las variables transformadas, lo que permite interpretar los coeficientes como impactos marginales sobre el *log-odds* del impago (en regresión logística se modeliza esta magnitud, y no directamente la probabilidad, porque permite expresar la relación con las variables explicativas de forma lineal y sin restricciones de rango) y mantiene coherencia con la tradición de la evaluación crediticia.

Por su parte, el modelo de *gradient boosting* construye de forma iterativa un conjunto de árboles de decisión poco profundos que corrigen progresivamente los errores residuales, capturando relaciones no lineales e interacciones complejas entre variables que la especificación lineal no puede representar.

Conceptualmente, esta parte del código materializa la transición desde una representación descriptiva del cliente hacia una función matemática que asigna a cada operación una probabilidad estimada de default, convirtiendo la información económica y comportamental en una medida cuantitativa de riesgo que posteriormente alimentará el motor de decisión y el análisis conjunto de riesgo y rentabilidad del modelo BNPL.

A continuación, el código ejecuta el entrenamiento efectivo de los modelos definidos previamente, ajustando tanto la regresión logística como el modelo de *gradient boosting* sobre el conjunto de entrenamiento mediante el método *fit*. Desde el punto de vista técnico, al estar encapsulados dentro de una serie de transformaciones secuenciales, el proceso no solo estima los parámetros del clasificador, sino que previamente ajusta el bloque de preprocesado, que incluye imputaciones, escalados y codificaciones categóricas, exclusivamente con los datos históricos de entrenamiento, generando una representación numérica coherente sobre la que se optimiza la función de pérdida correspondiente (máxima verosimilitud en el caso logístico y pérdida logarítmica iterativa en *boosting*).

En este momento, se calculan las probabilidades estimadas de impago sobre el conjunto de prueba utilizando los modelos previamente entrenados, mediante el método *predict\_proba*, que devuelve para cada observación la probabilidad asociada a cada clase del problema binario. Técnicamente, tras aplicar automáticamente el mismo preprocesado aprendido en entrenamiento, el modelo calcula para cada operación la probabilidad condicional de pertenecer a la clase positiva (default = 1), extrayendo la segunda columna del resultado para obtener directamente la PI estimada.

Conceptualmente, esta parte representa la aplicación real del Nivel 2 fuera del entorno de aprendizaje, ya que el modelo deja de ajustarse y pasa a comportarse como una función de *scoring* que asigna a nuevas transacciones una medida cuantitativa de riesgo, permitiendo evaluar su capacidad predictiva y sentando las bases para decisiones posteriores de aceptación, *pricing* o segmentación en el modelo BNPL.

El siguiente paso es calcular el desempeño predictivo de los modelos mediante métricas que analizan tanto su capacidad de discriminación como la calidad de sus probabilidades estimadas. Técnicamente, se calculan el AUC-ROC y el *Average Precision* (AUC-PR) para medir la capacidad del modelo de ordenar correctamente operaciones riesgosas frente a no riesgosas a lo largo de todos los posibles umbrales, lo que refleja su potencia de ranking en un contexto de clases desbalanceadas, y se calcula el *Brier Score* para cuantificar el error cuadrático medio entre las probabilidades predichas y el resultado real, evaluando así la calibración de las PI.

Conceptualmente, esta celda permite determinar si la capa probabilística no solo distingue adecuadamente entre clientes con distinto nivel de riesgo, sino si las probabilidades generadas son estadísticamente coherentes y utilizables para estimaciones económicas como pérdida esperada y análisis conjunto de riesgo y rentabilidad dentro del modelo BNPL.

#### *Ilustración 5. Comparación métodos Business Analytics*

```
=== Logistic Regression ===
```

```
AUC-ROC: 0.6840
```

```
AUC-PR : 0.2430
```

```
Brier  : 0.2104
```

```
=== HistGradientBoosting ===
```

```
AUC-ROC: 0.6742
```

```
AUC-PR : 0.3486
```

```
Brier  : 0.0992
```

---

Fuente: Elaboración propia

Los resultados muestran que ambos modelos tienen una capacidad de discriminación moderada, pero el *HistGradientBoosting* presenta una mejora en métricas predictivas relevantes para el uso operativo.

En discriminación global, la regresión logística obtiene un AUC-ROC de 0.6840 frente a 0.6742 del *boosting*, lo que indica que la capacidad de ranking general es muy similar e incluso ligeramente superior en el modelo lineal. Sin embargo, en contextos de desbalance como el riesgo crediticio, el AUC-PR es más informativo, y aquí el *boosting* mejora sustancialmente (0.3486 frente a 0.2430), lo que implica una mayor capacidad para identificar correctamente operaciones en default dentro del subconjunto de mayor riesgo.

La diferencia más relevante aparece en el *Brier Score*, donde el *boosting* obtiene 0.0992 frente a 0.2104 de la regresión logística. Dado que esta métrica mide el error cuadrático medio entre la probabilidad estimada y el resultado real, el *boosting* muestra una mejor calidad probabilística global, es decir, sus probabilidades están mucho más alineadas con la frecuencia real de impago. Dado su mejor desempeño en AUC-PR y *Brier Score* se selecciona *HistGradientBoosting* como modelo generador de la PI que alimenta el Nivel 3.

Después, se transforman las probabilidades estimadas por el modelo en una decisión binaria mediante la definición de un umbral basado en el percentil de la distribución de PI, concretamente seleccionando como operaciones de alto riesgo aquellas situadas por encima del percentil 90. Técnicamente, se calcula este umbral utilizando la función *quantile* sobre las probabilidades predichas y posteriormente se genera una variable clasificada que identifica como default predicho aquellos casos cuya PI supera dicho corte, a partir de esta clasificación se construyen la matriz de confusión y el informe de métricas (precision, recall y F1), lo que permite cuantificar errores tipo I y tipo II.

Conceptualmente, esta celda materializa el paso desde una estimación probabilística continua hacia una política operativa concreta de aceptación o rechazo, simulando una restricción de riesgo en la que solo se bloquea un determinado porcentaje de las operaciones más arriesgadas, y conectando así la capa probabilística con la lógica de decisión que impacta directamente en el equilibrio entre riesgo asumido y volumen de negocio.

### Ilustración 6. Resultados umbral

```
=== Clasificación con umbral ===
Umbral PD : 0.2542
[[655  50]
 [ 65  30]]
      precision    recall  f1-score   support

     0       0.9097    0.9291    0.9193     705
     1       0.3750    0.3158    0.3429      95

 accuracy                   0.8562     800
 macro avg       0.6424    0.6224    0.6311     800
 weighted avg    0.8462    0.8562    0.8508     800
```

Fuente: Elaboración propia

La matriz de confusión muestra que, con un umbral de PI = 0.2542, el modelo clasifica correctamente 655 *no-defaults* y 30 *defaults*, mientras que comete 50 falsos positivos y 65 falsos negativos. Esto implica que, de los 95 defaults reales, solo detecta 30, lo que se refleja en un recall de 0.3158 para la clase 1. Técnicamente, el modelo mantiene una alta precisión en la clase 0 (0.9097) y una exactitud global del 85.62%, pero esta *accuracy* está fuertemente influida por el desbalance de clases, ya que la mayoría de las observaciones no son default. La precisión para la clase 1 es 0.3750, lo que indica que cuando el modelo bloquea una operación por alto riesgo, aproximadamente el 37.5% realmente termina en impago.

Conceptualmente, el umbral seleccionado está priorizando la contención de falsos positivos, es decir, evitar rechazar demasiadas operaciones sanas, a costa de dejar escapar una parte significativa del riesgo real. En un contexto BNPL, esto implica una política relativamente conservadora en términos de fricción comercial, pero todavía poco eficiente en captura de defaults, lo que sugiere que el punto de corte debería ajustarse en función del coste económico relativo entre pérdidas por impago y margen perdido por rechazos, en lugar de evaluarse únicamente con métricas de clasificación agregadas.

### 3.5 Nivel 3 – Motor de decisión

El Nivel 3 del modelo constituye la capa final de decisión operativa. Su función no es generar el default, como hacía el Nivel 0, ni construir reglas deterministas de *affordability*, como en el Nivel 1, ni estimar una probabilidad de impago, como en el Nivel 2. Su función es transformar

toda esa información previa en una resolución de negocio concreta para cada operación. En otras palabras, esta capa traduce el riesgo y la capacidad de pago en una decisión accionable para el proveedor BNPL: aprobar, aprobar con restricciones o denegar.

Conceptualmente, esta capa es la que convierte el sistema analítico en un motor de negocio. Hasta el Nivel 2 el modelo únicamente describe y puntúa el riesgo. En el Nivel 3, en cambio, se introduce una política explícita de aceptación, en la que el proveedor decide no solo qué operaciones está dispuesto a aceptar, sino también cuánto capital está dispuesto a exponer y bajo qué plazo. Por eso este nivel es clave, conecta la analítica con la rentabilidad y el control real sobre el cliente.

A continuación, el código define un diccionario de política llamado *POLICY*. Este bloque define las normas del motor de decisión. En él aparecen dos umbrales de PI: uno inferior, por debajo del cual la operación puede aprobarse sin restricciones, y otro superior, a partir del cual la operación se considera demasiado arriesgada y se rechaza. Entre ambos umbrales se sitúan las operaciones de riesgo intermedio, que pueden ser aprobadas, pero de forma limitada. Además, el diccionario fija una regla de reducción del límite aprobado en función de la PI, define el número de cuotas recomendado según el nivel de riesgo y establece un valor de *Loss Given Default* (Pérdida dado el incumplimiento) para calcular una pérdida esperada simple. La lógica económica de este diseño es que el proveedor no toma decisiones solo con una señal binaria de sí o no, sino que gradúa su exposición en función del riesgo estimado.

Después, el código realiza una validación mínima de columnas obligatorias y una conversión de tipos. Técnicamente, este bloque comprueba que todas las variables necesarias existen en la base de datos y transforma a formato numérico aquellas que pueden generar errores si llegan como texto o con valores vacíos. Los nulos se imputan con cero en variables clave como la PI, el límite de *affordability*, el importe solicitado y el número de cuotas, esto es una simplificación que busca aumentar la robustez operativa. Su función es asegurar que el motor pueda ejecutarse de forma estable y no falle por problemas de formato, algo esencial en cualquier sistema de decisión automatizada.

El siguiente elemento del código es una función auxiliar llamada *limit\_factor*. Esta función implementa una relación lineal decreciente entre la PI y el porcentaje del límite de *affordability* que realmente se aprueba. Cuanto mayor es la probabilidad de impago, menor es el factor aplicado al límite máximo asequible del cliente. Posteriormente, este factor se acota

dentro de un rango mínimo y máximo mediante *clip*, evitando reducciones extremas o valores incoherentes. Conceptualmente, esta función introduce una idea central del negocio crediticio, que no todos los clientes aceptados deben recibir la misma exposición. Dos clientes pueden ser formalmente aceptables, pero uno de ellos puede recibir un límite menor porque su riesgo esperado es superior.

El bloque principal del código es el motor de decisión que trabaja fila a fila. Para cada cliente u operación, el sistema extrae la señal de *affordability*, la PI estimada, el límite máximo asequible, el importe solicitado y el número de cuotas solicitadas. A partir de ahí, aplica una secuencia jerárquica de decisión.

La primera regla es un *hard cut* de *affordability*. Si el cliente no supera el filtro de capacidad de pago (*affordability\_pass* == 0), la operación se deniega automáticamente. No se evalúa la PI ni se intenta modular el importe. La lógica de esta prioridad tiene sentido desde el punto de vista económico y prudencial: antes de preguntarse si el cliente es riesgoso, el sistema comprueba si puede permitirse la operación. Si no puede pagarla, no tiene sentido seguir evaluando.

Si el cliente supera *affordability*, el motor pasa a la segunda capa, basada en la PI. Aquí el sistema segmenta las operaciones en tres grupos. Si la PI es inferior al umbral bajo, la decisión inicial es *APPROVE*. Si la PI cae entre el umbral bajo y el umbral intermedio, la decisión inicial es *APPROVE\_LIMITED*. Si la PI supera el umbral superior, la operación se deniega como *DECLINE*. Esta estructura refleja una política escalonada de riesgo. No toda operación con riesgo positivo se rechaza si no que se introducen zonas de aceptación total, aceptación condicionada y rechazo, lo cual es mucho más realista en un entorno BNPL que una política puramente binaria.

Una vez determinada la aceptabilidad básica de la operación, el motor calcula el límite aprobado. Para ello parte del límite máximo permitido por *affordability* y le aplica el factor de reducción dependiente de la PI. El resultado es un límite ajustado por riesgo (*limit\_by\_pd*). Luego compara ese valor con el importe solicitado y toma el mínimo entre ambos. Esto significa que el cliente nunca recibe más de lo que ha pedido ni más de lo que la política considera razonable exponer.

El código incorpora además una degradación de decisión. Si inicialmente una operación era *APPROVE*, pero el límite finalmente aprobado queda por debajo del importe solicitado, la decisión se transforma en *APPROVE\_LIMITED*. Esta parte busca evitar una incoherencia

operativa. No sería correcto etiquetar como aprobación plena una operación en la que en realidad se está recortando el monto solicitado. El cambio de etiqueta mejora la trazabilidad de la decisión y separa claramente una aprobación total de una aprobación condicionada por exposición.

En paralelo, el motor recomienda un número de cuotas en función del nivel de riesgo. Los clientes de menor riesgo pueden acceder a más cuotas, mientras que los perfiles más arriesgados reciben plazos más cortos. Después, el sistema toma el mínimo entre el plazo solicitado y el plazo recomendado por política. La lógica detrás de este paso es que un plazo más largo reduce la cuota mensual, pero también prolonga la exposición, aumenta la incertidumbre temporal y puede elevar el riesgo total de no pago. Por eso el modelo reduce el número de cuotas para perfiles más frágiles. Esta decisión conecta directamente riesgo y estructura del producto, no solo riesgo y aceptación.

A continuación, el código calcula una *Expected Loss* (EL) simplificada. Utiliza la fórmula estándar  $EL = PI \times LGD \times EAD^5$ , donde la exposición en caso de default se aproxima mediante el importe aprobado. Aunque aquí la *LGD* es fija y simplificada, el cálculo cumple una función conceptual importante ya que permite cuantificar la pérdida media esperada de cada decisión. Esto convierte el motor en algo más que un clasificador. Ya no solo decide si una operación entra o no, sino que genera una aproximación económica del coste esperado asociado a cada aprobación. Este apartado se incluye en este trabajo con el objetivo de aproximar el modelo, en la medida de lo posible, a la estructura que emplearía un proveedor real. De este modo, se busca analizar con mayor profundidad qué técnicas de *Business Analytics* resultan más eficaces en la gestión del riesgo y la rentabilidad.

Tras recorrer todas las operaciones, el código guarda los resultados en nuevas columnas de la base de datos. Esto hace que la salida del Nivel 3 no sea una simple etiqueta, sino un conjunto de variables operativas que permiten interpretar la decisión, cuantificarla y analizarla después. Desde el punto de vista de *governance*, esto es importante porque cada operación queda documentada con su motivo de aceptación o rechazo y con la magnitud exacta de la exposición autorizada.

El bloque final genera un resumen agregado por tipo de decisión. Agrupa las observaciones por *APPROVE*, *APPROVE\_LIMITED* y *DECLINE*, y calcula para cada grupo

---

<sup>5</sup> *Exposure at Default*: Importe de crédito que el prestamista tiene expuesto cuando el prestatario entra en default.

el número de casos, la PI media, el límite medio aprobado, la pérdida esperada media y la tasa de aprobación. Si además existe una variable real de default, el código compara la tasa de impago total, la de las operaciones aprobadas y la de las denegadas. Conceptualmente, esta parte actúa como un *backtesting* simple del motor de decisión. Permite comprobar si la política está filtrando correctamente el riesgo, es decir, si los aprobados terminan teniendo menor default que los rechazados y si la mezcla entre volumen aprobado y riesgo asumido resulta coherente.

## 4. Resultados y conclusiones

Como se ha mencionado previamente, el objetivo de este trabajo es la creación de un modelo de *Business Analytics* basado en una evaluación de asequibilidad y un sistema de *credit scoring*. Este modelo final es el producto de la unión de las cuatro capas explicadas en el apartado anterior.

Esto se debe a que los distintos niveles construyen una arquitectura analítica completa que reproduce, de forma simplificada pero conceptualmente coherente, el funcionamiento de un sistema de gestión de riesgo adaptado a las nuevas regulaciones en un proveedor de BNPL. El nivel 0 es la generación de una realidad simulada de impago. El nivel 1 es la construcción determinista de la capacidad de pago del cliente. El nivel 2 la estimación probabilística del riesgo mediante técnicas de *Machine Learning*. Finalmente, el nivel 3 es la traducción de toda esta información en una política operativa de aceptación, dimensionamiento de la exposición y estimación de pérdida esperada.

Paralelamente, el modelo buscaba analizar si era más adecuado usar una regresión logística o el *Gradient Boosting*. Es en este apartado donde podemos llegar a la primera conclusión. Ocurre en la capa probabilística (nivel 2), donde se mide la eficacia de cada método a través de tres métricas clave: el AUC-ROC, el AUC-PR y el Brier.

Partiendo de estos dos objetivos y gracias a este trabajo se han llegado a dos conclusiones principales que sintetizan tanto la validez del enfoque metodológico propuesto como la utilidad práctica de los modelos de *Business Analytics* aplicados

*Ilustración 7. Tabla de conclusiones y resultados del trabajo*

Hipótesis	Evidencia	Resultado
Creación de un modelo de <i>Business Analytics</i> adaptado a las nuevas exigencias regulatorias del sector BNPL	Arquitectura con trazabilidad, solvencia ex-ante y separación operativa entre capacidad de pago y probabilidad de impago	Confirmado
Traducción resultados del modelo en decisiones operativas aplicables en entornos reales	Motor con límites dinámicos de exposición ajustados según la probabilidad estimada de impago y restricciones de <i>affordability</i>	Confirmado

Análisis y comparación de métodos de BA aplicados al objeto de trabajo	Mejor desempeño del <i>Gradient Boosting</i> en la medida del AUC-PR y el <i>Brier score</i> .	Confirmado
Evaluación de la capacidad predictiva del modelo	Capacidad discriminatoria moderada AUC-ROC de aprox 0.68, suficiente para segmentación operativa del riesgo, aunque con margen de mejora en detección de <i>defaults</i>	Parcialmente confirmado

Fuente: Elaboración propia

#### 4.1 Conclusiones de los métodos de *Business Analytics*

El AUC-ROC representa la probabilidad de que una operación en *default* tenga mayor probabilidad estimada que una operación elegida al azar. No evalúa si la probabilidad es correcta en valor absoluto, sino si el ranking de riesgo es bueno. Es decir, mide la capacidad del modelo para ordenar correctamente operaciones con y sin *default* para todos los posibles umbrales de decisión. Normalmente valores por debajo de 0.5 se descartan al considerarse que discriminan al azar y a partir de 0.6 se dice que empieza a discriminar correctamente. En este caso, ambos modelos presentan valores prácticamente equivalentes, lo que indica una capacidad discriminatoria similar entre ellos.

Esta métrica se usa ya que nuestro motor de decisión se basa en percentiles de la PI para segmentar operaciones. Por lo tanto, depende de un ranking relativo. Al usar el AUC-ROC, podemos comparar distintos modelos de *scoring*.

El AUC-PR evalúa el equilibrio entre *precision* (cuántos *defaults* predichos lo son realmente) y *recall* (cuántos *defaults* reales detecta el modelo). Es decir, mide la capacidad del modelo para identificar correctamente *defaults* cuando la clase positiva es rara. Es más informativa que la métrica anterior cuando hay desbalance de clases (el no *default* aparece muchas más veces que el *default*), como ocurre en riesgo crediticio BNPL. Al comparar ambos métodos aquí se empieza a notar la diferencia, ya que el *Gradient Boosting* es bastante superior.

Como en la mayoría de las operaciones de usuarios de BNPL no acaban en *default*, el *accuracy* (total de aciertos del modelo, sin distinguir el tipo) deja de ser informativo. Por eso, el AUC-PR sirve para ver si el modelo identifica correctamente operaciones realmente riesgosas dentro de la cola de la distribución.

El Brier Score mide el error cuadrático medio entre la probabilidad estimada y el resultado real. Por ejemplo, si predices que  $PI = 0.20$  y ocurre  $default = 1$  entonces el error será alto. Si predices  $PI = 0.80$  y ocurre  $default = 1$ , entonces el error será bajo. Indica si las probabilidades pueden usarse para *Expected Loss, pricing* o política de límites. Una vez más, el *Gradient Boosting* obtiene un mejor valor de esta métrica que la regresión lineal.

El modelo calcula:  $Expected Loss = PI \times LGD \times EAD$ . Esto exige probabilidades calibradas. Un modelo puede tener buen AUC-ROC pero malas probabilidades, en ese caso el motor económico funciona mal. El Brier Score sirve precisamente para saber si la PI es usable económicamente.

Por todos estos motivos, podemos llegar a dos conclusiones principales: la primera es que el modelo de *Gradient Boosting* es la alternativa más eficaz dentro del marco analítico planteado. Al ofrecer un mejor equilibrio entre capacidad predictiva, calibración de probabilidades y robustez frente al desbalanceo de clases, creando una base más sólida para la toma de decisiones orientadas a optimizar la rentabilidad en el nuevo contexto regulatorio del sector *Buy Now Pay Later*.

La segunda conclusión es que el modelo confirma que la utilización de técnicas no lineales permite capturar interacciones complejas entre variables socioeconómicas y transaccionales que no pueden representarse adecuadamente mediante especificaciones lineales tradicionales. Esta idea realza la necesidad del *scoring* en contextos de microcrédito digital.

## 4.2 Conclusiones del modelo operativo

El modelo desarrollado permite extraer las siguientes conclusiones:

Una de las más relevantes del modelo es la confirmación del papel estructural clave del *affordability check*, como filtro previo a cualquier estimación probabilística del riesgo. La incorporación de esta capa determinista refleja directamente el principio de préstamo responsable introducido en la Directiva (UE) 2023/2225 (CCD II), que exige evaluar la capacidad real del consumidor para asumir nuevas obligaciones financieras antes de conceder crédito.

Otro aspecto que podemos observar es el análisis del umbral de decisión en la capa probabilística. Este muestra, que existe un equilibrio entre reducir el riesgo asumido y mantener

el volumen de operaciones aprobadas. El percentil seleccionado reduce los falsos positivos, pero permite que parte del riesgo siga entrando en cartera. Esto indica que el punto de corte no debe elegirse solo con métricas estadísticas, sino a partir de un criterio económico que considere el coste del impago frente al margen perdido por rechazar operaciones.

Como se explica en apartados anteriores el motor de decisión permite pasar de una estimación estadística del riesgo a una decisión crediticia aplicable en la práctica. Su función no consiste solo en aceptar o rechazar operaciones, sino en ajustar el nivel de exposición en función de la probabilidad de impago y de los límites definidos por el análisis de asequibilidad. Haciendo que el importe aprobado y el número de cuotas se adaptan al perfil de riesgo del cliente. De este modo, el modelo demuestra que la combinación de reglas deterministas y *scoring* probabilístico permite construir políticas de aceptación más flexibles y coherentes con la nueva realidad BNPL.

Desde la perspectiva regulatoria, el modelo demuestra su capacidad para adaptarse conceptualmente a las exigencias introducidas por la CCD II, especialmente en lo relativo a la evaluación de solvencia, trazabilidad de decisiones y separación entre capacidad de pago y probabilidad de impago. La estructura en capas permite documentar de forma transparente cada etapa del proceso de concesión de crédito, facilitando la explicación del modelo y su alineación con los requisitos de gobernanza exigidos en el nuevo entorno regulatorio europeo.

La conclusión más relevante del trabajo es que el modelo cumple su objetivo principal, al demostrar que una arquitectura de *Business Analytics* basada en la combinación de *affordability checks*, *scoring* probabilístico calibrado y motor de decisión operativo permite gestionar de forma coherente el riesgo crediticio manteniendo la viabilidad económica del modelo BNPL en un contexto regulatorio más restrictivo. Este resultado se sustenta en tres elementos: la capacidad del modelo para separar conceptualmente capacidad de pago y probabilidad de impago, la mejora obtenida mediante técnicas de *Gradient Boosting* en la estimación probabilística del riesgo y la incorporación explícita de la pérdida esperada en la lógica de decisión.

Por otro lado, a diferencia de enfoques centrados exclusivamente en la clasificación de clientes según su probabilidad de impago, el sistema desarrollado incorpora explícitamente la estimación de pérdida esperada y el ajuste de los límites de exposición como parte de la decisión. Esta decisión permite aproximarse al funcionamiento real del modelo de un proveedor BNPL.

No obstante, el modelo presenta limitaciones estructurales que deben considerarse al interpretar sus resultados. La utilización de datos sintéticos, aunque metodológicamente justificada, impide contrastar directamente el comportamiento del sistema con carteras reales. Asimismo, la simplificación de la LGD como parámetro fijo reduce la precisión económica del cálculo de pérdida esperada, y la ausencia de un módulo de *pricing* dinámico limita la capacidad del modelo para optimizar simultáneamente riesgo y margen. Adicionalmente, el umbral de decisión no ha sido determinado mediante un proceso formal de optimización económica. Esto no permite la evaluación completa del equilibrio entre volumen aprobado y pérdidas esperadas.

En conjunto, el modelo desarrollado constituye una arquitectura analítica replicable que ilustra cómo los proveedores BNPL pueden adaptar sus modelo de crédito a entornos regulatorios más exigentes sin renunciar a la gestión activa de la rentabilidad. Integrando principios de préstamo responsable y políticas de decisión trazables dentro de un mismo sistema operativo.

## 5. Bibliografia

- Australian Government Treasury. (2022, November 21). Regulating Buy Now, Pay Later in Australia. <https://treasury.gov.au/sites/default/files/2022-11/c2022-338372-op.pdf>
- Basel Committee on Banking Supervision. (2025, February 5). Principles for the management of credit risk. Bank for International Settlements. <https://www.bis.org/bcbs/publ/d595.pdf>
- Consumer Financial Protection Bureau. (2024, August 2). 12 CFR Part 1002 - Equal Credit Opportunity Act (Regulation B). <https://www.consumerfinance.gov/rules-policy/regulations/1002/>
- Deloitte. (2025). Reinventing consumer finance: Buy Now, Pay Later encourages banks to transform. Deloitte Luxembourg. <https://www.deloitte.com/lu/en/Industries/banking-capital-markets/perspectives/buy-now-pay-later-encourages-banks-transformation.html>
- European Banking Authority. (2020, May 29). Guidelines on loan origination and monitoring. <https://www.eba.europa.eu/activities/single-rulebook/regulatory-activities/credit-risk/guidelines-loan-origination-and-monitoring>
- European Parliament and Council of the European Union. (2023, October 18). Directive (EU) 2023/2225 on credit agreements for consumers and repealing Directive 2008/48/EC. EUR-Lex. <https://eur-lex.europa.eu/eli/dir/2023/2225/oj/eng>
- Financial Conduct Authority. (2026, February 11). PS26/1: Regulation of Deferred Payment Credit (unregulated Buy Now Pay Later). <https://www.fca.org.uk/publications/policy-statements/ps26-1-regulation-deferred-payment-credit>

- Insider Intelligence (now EMARKETER). (2024). Guide to Buy Now, Pay Later: Industry trends, regulation, and market data. <https://www.emarketer.com/learningcenter/guides/buy-now-pay-later-industry-challenges/>
- Latinia. (2024). BNPL: Tendencias y perspectivas para 2025. <https://latinia.com/recursos/bnpl-tendencias-y-perspectivas>
- Statista. (2023). Chart: Buy Now, Pay(Pal) Later.

## 6. Anexos

- Anexo 1 – Generador de base de datos sintética, condiciones para aumentar el realismo. Entre estas condiciones se encuentran: la edad de los clientes con una mayor probabilidad de perfiles jóvenes. Estatus de hogar, si alquilan, tienen hipoteca o no pagan nada. Ingresos, valores aleatorios entre una horquilla para cada categoría de edad. Duración del empleo en meses, distribuido de acuerdo con la edad. Gastos fijos mensuales, teniendo en cuenta el estatus de vivienda y la edad. Entre otras variables tanto socioeconómicas como transaccionales.

```
mix = weighted_choice(["young", "mid", "older", "senior"], [0.62, 0.28, 0.08, 0.02], size=N)
age = np.empty(N, dtype=int)
for i, g in enumerate(mix):
    if g == "young":
        age[i] = clip_int(rng.normal(26, 5), 18, 35)
    elif g == "mid":
        age[i] = clip_int(rng.normal(42, 7), 30, 60)
    elif g == "older":
        age[i] = clip_int(rng.normal(58, 4), 50, 70)
    else:
        age[i] = clip_int(rng.normal(69, 3), 65, 75)
```

```
housing = []
for a in age:
    if a <= 23:
        housing.append(weighted_choice(
            ["dependent", "rent", "mortgage", "own"],
            [0.65, 0.30, 0.03, 0.02]
        )[0])
    elif a <= 35:
        housing.append(weighted_choice(
            ["dependent", "rent", "mortgage", "own"],
            [0.18, 0.55, 0.20, 0.07]
        )[0])
    elif a <= 55:
        housing.append(weighted_choice(
            ["rent", "mortgage", "own"],
            [0.25, 0.50, 0.25]
        )[0])
    else:
        housing.append(weighted_choice(
            ["rent", "mortgage", "own"],
            [0.30, 0.25, 0.45]
        )[0])
housing = np.array(housing, dtype=object)
```

```

income = np.empty(N, dtype=int)
for i, a in enumerate(age):
    if a <= 22:
        mu, sd = 850, 220
    elif a <= 30:
        mu, sd = 1300, 320
    elif a <= 45:
        mu, sd = 2200, 550
    elif a <= 64:
        mu, sd = 2500, 600
    else: # 65+
        mu, sd = 1100, 350
    val = rng.normal(mu, sd)
    income[i] = clip_int(val, 400, 4000)

```

```

empl_len_months = np.empty(N, dtype=int)
for i, a in enumerate(age):
    if a <= 22:
        empl_len_months[i] = int(rng.integers(0, 7)) # 0-6 meses
    elif a <= 30:
        m = int(np.clip(rng.gamma(1.6, 10), 0, 60))
        empl_len_months[i] = max(m, 12) # mínimo 12 meses => 1 año mínimo
    elif a <= 45:
        empl_len_months[i] = int(np.clip(rng.gamma(2.2, 18), 0, 240))
    elif a <= 64:
        empl_len_months[i] = int(np.clip(rng.gamma(2.6, 22), 0, 360))
    else:
        empl_len_months[i] = int(np.clip(rng.gamma(1.8, 18), 0, 360))
employment_length = empl_len_months

```

```

fixed_exp = np.empty(N, dtype=int)
for i, a in enumerate(age):
    inc = income[i]
    hs = housing[i]
    if a <= 23:
        base_ratio = rng.uniform(0.35, 0.85)
    elif a <= 35:
        base_ratio = rng.uniform(0.45, 0.90)
    elif a <= 55:
        base_ratio = rng.uniform(0.50, 0.85)
    elif a <= 64:
        base_ratio = rng.uniform(0.50, 0.90)
    else:
        base_ratio = rng.uniform(0.55, 0.95)

    # ajustes vivienda
    if hs == "dependent":
        base_ratio -= rng.uniform(0.10, 0.25)
    elif hs == "mortgage":
        base_ratio += rng.uniform(0.05, 0.15)
    elif hs == "own":
        base_ratio += rng.uniform(0.00, 0.08)
    else: # rent
        base_ratio += rng.uniform(0.02, 0.10)

    val = inc * base_ratio + rng.normal(0, 80)
    fixed_exp[i] = clip_int(val, 100, 3000)

```

- Anexo 2 – Generador de base de datos sintética. Categorización de cada fila (1 cliente 1 transacción) en categorías de producto.

```
merchant_categories = ["electronics", "travel", "fashion", "home", "beauty", "groceries", "sports", "other"]
merchant_probs = [0.18, 0.08, 0.20, 0.18, 0.10, 0.10, 0.08, 0.08]
merchant = weighted_choice(merchant_categories, merchant_probs, size=N).astype(object)

installments = np.empty(N, dtype=int)
purchase_amount = np.empty(N, dtype=int)

inst_options = [4, 6, 8, 10, 12]
inst_probs = [0.55, 0.30, 0.08, 0.04, 0.03]

for i in range(N):
    cat = merchant[i]

    if cat == "electronics":
        val = rng.normal(650, 220)
    elif cat == "travel":
        val = rng.normal(800, 260)
    elif cat == "home":
        val = rng.normal(450, 180)
    elif cat == "fashion":
        val = rng.normal(180, 90)
    elif cat == "beauty":
        val = rng.normal(120, 60)
    elif cat == "groceries":
        val = rng.normal(90, 35)
    elif cat == "sports":
        val = rng.normal(220, 100)
    else:
        val = rng.normal(250, 140)

    val *= np.clip(income[i] / 2000, 0.7, 1.4)

    purchase_amount[i] = clip_int(val, 50, 1500)
    installments[i] = int(weighted_choice(inst_options, inst_probs)[0])

installment_amount = np.round(purchase_amount / installments, 2)
```

- Anexo 3 – Nivel 0. Asignación de penalizadores principales como el estatus de vivienda o la categoría de producto.

```

# Penalización por vivienda (proxy estabilidad)
house_penalty_map = {
    "own": -0.10,
    "mortgage": 0.05,
    "rent": 0.15
}
house_penalty = df["housing_status"].map(house_penalty_map).fillna(0.05)

# Penalización por categoría de merchant (muy suave)
mc = df["merchant_category"].astype(str).str.lower()

mc_penalty = np.select(
    [
        mc.str.contains("fashion|moda"),
        mc.str.contains("electronics|electronica|electrónica"),
        mc.str.contains("travel|viaje"),
        mc.str.contains("grocer|supermarket|aliment"),
        mc.str.contains("health|salud|pharma|farm"),
    ],
    [0.10, 0.05, 0.08, -0.05, -0.03],
    default=0.00
)

# Edad: Leve incremento en extremos jóvenes y muy mayores
age = df["age"].astype(float)
age_bump = np.where(age < 23, 0.10, 0.0) + np.where(age > 65, 0.05, 0.0)

# Historial: más historial reduce riesgo
# Log para rendimientos decrecientes -> Los primeros meses impactan más que los segundos
hist = df["credit_history_length"].astype(float).clip(lower=0)
hist_effect = -0.25 * np.log1p(hist)

# Defaults pasados: impacto fuerte
past_def = df["num_past_defaults"].astype(float).clip(lower=0)

```

- Anexo 4 – Nivel 0. Fórmula *risk score* e intercepto

```

# Intercepto
intercept = -3.3

risk_score = (
    intercept
    + 2.10 * DTI
    + 2.60 * DSTI
    + 2.80 * inst_inc
    + 1.50 * inst_res
    + 0.85 * past_def
    # + emp_penalty
    + house_penalty
    + mc_penalty
    + age_bump
    + hist_effect
    + noise
)

```

- Anexo 5 – Nivel 0. Calculo de la tasa de *default*

```

PD = 1 / (1 + np.exp(-risk_score))
df["PD_synthetic"] = PD

u = rng.random(len(df)) # Valor random entre 0 y 1
df["default"] = (u < df["PD_synthetic"]).astype(int) # Si u es menor que PD entonces el cliente impaga

# Control de tasa de default
default_rate = df["default"].mean()
print(f"Tasa de default sintética: {default_rate:.3%}")

Tasa de default sintética: 13.225%

```

- Anexo 6 – Nivel 1. Política del modelo para cumplir ls snuevas regulaciones

```

POLICY = {
    "min_income": 450.0,           # ingreso mínimo mensual
    "max_DTI": 0.65,              # límite DTI
    "max_DSTI": 0.40,            # límite DSTI (deuda + BNPL) / ingreso
    "min_residual_after_installment": 100, # residual_income - cuota
    "max_installment_to_residual": 0.50, # cuota / residual_income
    "max_purchase_to_income": 0.35, # purchase_amount / income
}

```

- Anexo 7 – Nivel 1. Razones de rechazo, tienen en cuenta: ingresos mensuales, cualquier incumplimiento de la política de rechazo y *affordability*.

```

def build_reasons(row):
    reasons = []
    if row.get("monthly_income", 0) < POLICY["min_income"]:
        reasons.append("Income Below Minimun Requirement")
    dti_val = row.get("DTI", np.nan)
    if pd.isna(dti_val):
        dti_val = row.get("existing_debt_monthly", 0) / (row.get("monthly_income", 0) + eps)
    if dti_val > POLICY["max_DTI"]:
        reasons.append("DTI Above Limit")
    if row.get("DSTI", 0) > POLICY["max_DSTI"]:
        reasons.append("DSTI Above Limit")
    if (row.get("residual_income", 0) - row.get("installment_amount", 0)) < POLICY["min_residual_after_installment"]:
        reasons.append("Residual too low")
    if row.get("installment_amount / residual_income", 999) > POLICY["max_installment_to_residual"]:
        reasons.append("Installment to residual Ratio above limit")
    pai_val = row.get("purchase_amount / income", np.nan)
    if pd.isna(pai_val):
        pai_val = row.get("purchase_amount", 0) / (row.get("monthly_income", 0) + eps)
    if pai_val > POLICY["max_purchase_to_income"]:
        reasons.append("Purchase to income ratio above limit")
    return "|".join(reasons) if reasons else "pass"

df["affordability_reasons"] = df.apply(build_reasons, axis=1)

```

- Anexo 8 – Nivel 2. *Train* y *test* del modelo con los datos sintéticos

```

use_time_split = True

if use_time_split and X["purchase_date"].notna().mean() > 0.8:
    # ordenar por fecha
    idx = X["purchase_date"].sort_values().index
    X = X.loc[idx].reset_index(drop=True)
    y = y.loc[idx].reset_index(drop=True)

    split = int(0.8 * len(X))
    X_train, X_test = X.iloc[:split].copy(), X.iloc[split:].copy()
    y_train, y_test = y.iloc[:split].copy(), y.iloc[split:].copy()
else:
    X_train, X_test, y_train, y_test = train_test_split(
        X, y, test_size=0.2, random_state=42, stratify=y
    )

```

- Anexo 9 - Nivel 2. Selección de columnas numéricas y categóricas

```

cat_cols = X_train.select_dtypes(include=["object"]).columns.tolist()
num_cols = [c for c in X_train.columns if c not in cat_cols]

# convertir a variables útiles y eliminar la fecha cruda
def add_date_parts(df_in: pd.DataFrame) -> pd.DataFrame:
    df_out = df_in.copy()
    if "purchase_date" in df_out.columns:
        dt = pd.to_datetime(df_out["purchase_date"], errors="coerce")
        df_out["purchase_dow"] = dt.dt.dayofweek
        df_out["purchase_month"] = dt.dt.month
        df_out = df_out.drop(columns=["purchase_date"])
    return df_out

X_train = add_date_parts(X_train)
X_test = add_date_parts(X_test)

# Recalcular columnas tras transformación
cat_cols = X_train.select_dtypes(include=["object"]).columns.tolist()
num_cols = [c for c in X_train.columns if c not in cat_cols]

```

- Anexo 10 – Nivel 2. Preprocesado de los datos

```
numeric_pipe = Pipeline(steps=[
    ("imputer", SimpleImputer(strategy="median")),
    ("scaler", StandardScaler())
])

categorical_pipe = Pipeline(steps=[
    ("imputer", SimpleImputer(strategy="most_frequent")),
    ("onehot", OneHotEncoder(handle_unknown="ignore"))
])

preprocess = ColumnTransformer(
    transformers=[
        ("num", numeric_pipe, num_cols),
        ("cat", categorical_pipe, cat_cols)
    ],
    remainder="drop"
)
```

- Anexo 11 – Nivel 2. Ejecución de los modelos PI

```
# Modelo 1: Logística
logit = Pipeline(steps=[
    ("prep", preprocess),
    ("clf", LogisticRegression(
        max_iter=2000,
        class_weight="balanced",
        solver="lbfgs"
    ))
])

# Modelo 2: Gradient Boosting
gb = Pipeline(steps=[
    ("prep", preprocess),
    ("clf", HistGradientBoostingClassifier(
        max_depth=6,
        learning_rate=0.06,
        max_iter=300,
        random_state=42
    ))
])
```

- Anexo 12 – Nivel 2. Cálculo de predicciones y evaluaciones

```
pd_logit = logit.predict_proba(X_test)[: , 1]
pd_gb = gb.predict_proba(X_test)[: , 1]
```

```
def evaluate(y_true, pd_hat, name):
    auc = roc_auc_score(y_true, pd_hat)
    ap = average_precision_score(y_true, pd_hat)
    brier = brier_score_loss(y_true, pd_hat)
    print(f"\n=== {name} ===")
    print(f"AUC-ROC: {auc:.4f}")
    print(f"AUC-PR : {ap:.4f}")
    print(f"Brier : {brier:.4f}")

evaluate(y_test, pd_logit, "Logistic Regression")
evaluate(y_test, pd_gb, "HistGradientBoosting")
```

```
=== Logistic Regression ===
AUC-ROC: 0.6840
AUC-PR : 0.2430
Brier : 0.2104
```

```
=== HistGradientBoosting ===
AUC-ROC: 0.6742
AUC-PR : 0.3486
Brier : 0.0992
```

- Anexo 13 – Nivel 3. Parámetros de política

```
POLICY = {
    # UmbralEs PD
    "pd_approve": 0.03,          # por debajo = APPROVE
    "pd_limited": 0.10,        # entre approve y limited = APPROVE_LIMITED
    # por encima = DECLINE

    "limit_factor_a": 1.00,
    "limit_factor_b": 2.00,
    "limit_factor_min": 0.30,
    "limit_factor_max": 1.00,

    "installments_low_risk": 6,
    "installments_med_risk": 4,
    "installments_high_risk": 3,

    # Parámetros Expected Loss
    "LGD": 0.75,               # pérdida dado default
}
```

- Anexo 14 – Nivel 3. Creación y ejecución del motor de decisión

```

decisions = []
reasons = []
approved_limits = []
approved_installments = []
expected_losses = []

for _, r in df.iterrows():
    aff = int(r["affordability_pass"])
    pd_val = float(r["PD_model"])
    aff_limit = float(r["max_affordable_purchase_amount"])
    req_amount = float(r["purchase_amount"])
    req_inst = int(r["installments"])

    # affordability
    if aff == 0:
        decisions.append("DECLINE")
        reasons.append(f"affordability_fail:{r.get('Affordability Reasons','')}")
        approved_limits.append(0.0)
        approved_installments.append(0)
        expected_losses.append(0.0)
        continue

    #PI-based decision
    if pd_val < POLICY["pd_approve"]:
        decision = "APPROVE"
        reason = "pd_low"
        rec_inst = POLICY["installments_low_risk"]
    elif pd_val < POLICY["pd_limited"]:
        decision = "APPROVE_LIMITED"
        reason = "pd_medium"
        rec_inst = POLICY["installments_med_risk"]
    else:
        decisions.append("DECLINE")
        reasons.append("pd_high")
        approved_limits.append(0.0)
        approved_installments.append(0)
        expected_losses.append(0.0)
        continue

    #Límite aprobado
    f = limit_factor(pd_val)
    limit_by_pd = aff_limit * f

    # El límite aprobado final no debe ser menor que mínimo operativo
    approved_limit = min(req_amount, limit_by_pd)

    if approved_limit + 1e-9 < req_amount:
        if decision == "APPROVE":

```

- Anexo 15 – Nivel 3. Evaluación del modelo

```
summary = df.groupby("decision").agg(
    n=("decision", "size"),
    avg_PD=("PD_model", "mean"),
    avg_limit=("approved_limit", "mean"),
    avg_EL=("expected_loss", "mean"),
    approval_rate=("approved_flag", "mean")
).sort_values("n", ascending=False)

print("\n=== Resumen decisiones ===")
print(summary)

print("\nTasa de aprobación total:", df["approved_flag"].mean())

# Si existe default real, evaluar tasa de default en aprobados
if "default" in df.columns:
    print("\nDefault rate (total):", df["default"].mean())
    print("Default rate (aprobados):", df.loc[df["approved_flag"] == 1, "default"].mean())
    print("Default rate (denegados):", df.loc[df["approved_flag"] == 0, "default"].mean())
```

```
=== Resumen decisiones ===
              n  avg_PD  avg_limit  avg_EL  approval_rate
decision
DECLINE      1774  0.256573    0.000000  0.000000           0.0
APPROVE      1190  0.012632  185.865546  1.964927           1.0
APPROVE_LIMITED 1036  0.057130  273.581610  12.078633           1.0

Tasa de aprobación total: 0.5565

Default rate (total): 0.13225
Default rate (aprobados): 0.0
Default rate (denegados): 0.29819616685456596
```

## 7. Declaración de uso de Herramientas de Inteligencia Artificial Generativa

**ADVERTENCIA:** Desde la Universidad consideramos que ChatGPT u otras herramientas similares son herramientas muy útiles en la vida académica, aunque su uso queda siempre bajo la responsabilidad del alumno, puesto que las respuestas que proporciona pueden no ser veraces. En este sentido, NO está permitido su uso en la elaboración del Trabajo fin de Grado para generar código porque estas herramientas no son fiables en esa tarea. Aunque el código funcione, no hay garantías de que metodológicamente sea correcto, y es altamente probable que no lo sea.

Por la presente, yo, Gonzalo Cabañas Alonso, estudiante de E2-Analytics de la Universidad Pontificia Comillas al presentar mi Trabajo Fin de Grado titulado "Aplicación de modelos de Business Analytics en la gestión del riesgo y rentabilidad del sector Buy Now Pay Later", declaro que he utilizado la herramienta de Inteligencia Artificial Generativa ChatGPT u otras similares de IAG de código sólo en el contexto de las actividades descritas a continuación:

1. **Brainstorming de ideas de investigación:** Utilizado para idear y esbozar posibles áreas de investigación.
2. **Referencias:** Usado conjuntamente con otras herramientas, como *Science*, para identificar referencias preliminares que luego he contrastado y validado.
3. **Corrector de estilo literario y de lenguaje:** Para mejorar la calidad lingüística y estilística del texto.
4. **Sintetizador y divulgador de libros complicados:** Para resumir y comprender literatura compleja.
5. **Revisor:** Para recibir sugerencias sobre cómo mejorar y perfeccionar el trabajo con diferentes niveles de exigencia.
6. **Traductor:** Para traducir textos de un lenguaje a otro.

Afirmo que toda la información y contenido presentados en este trabajo son producto de mi investigación y esfuerzo individual, excepto donde se ha indicado lo contrario y se han dado los créditos correspondientes (he incluido las referencias adecuadas en el TFG y he explicitado para que se ha usado ChatGPT u otras herramientas similares). Soy consciente de las implicaciones académicas y éticas de presentar un trabajo no original y acepto las consecuencias de cualquier violación a esta declaración.

Fecha: 21 de abril de 2026

Firma:

A handwritten signature in blue ink, consisting of several fluid, overlapping strokes that are difficult to decipher as specific text.