



Escuela de Posgrados

**GESTIÓN DE INVENTARIOS TECNOLÓGICOS PARA UNA ORGANIZACIÓN DEL  
MERCADO MEDIANTE MODELOS PREDICTIVOS**

Hernández Rojas Aleida Cecilia

Pinzón Madrid Mariana

Quintana Quintana Luisa  
Fernanda

Trabajo de Grado presentado como requisito para optar al título de:

Especialista en Big Data e Inteligencia de Negocios

Asesor:  
Ingrid Durley Torres Pardo

Universidad Católica Luis  
Amigó  
Facultad de Ingenierías y  
Arquitectura  
Especialización en Big Data e Inteligencia de  
Negocios Medellín, Colombia  
2025

# Dedicatoria

Lo dedico principalmente a mí misma, como testimonio de mis capacidades y resiliencia para afrontar este reto. El camino recorrido durante este año, aunque no fue fácil, superó todas las adversidades, es la prueba fehaciente de que no debemos imponernos límites. Que las enfermedades mentales no son un obstáculo insuperable para continuar y alcanzar nuestras metas. Este logro es un acto de auto-reconocimiento y fortaleza inquebrantable.

Extiendo mi inmensa gratitud a mis compañeras de trabajo de grado, Luisa Fernanda Quintana Quintana y Mariana Pinzón Madrid. Su apoyo incondicional y acompañamiento constante fueron fundamentales. Sin su presencia y colaboración, el trayecto habría sido, sin duda, mucho más arduo.

Aleida Cecilia Hernández Rojas,

A mi familia, por ser el pilar constante que apoyo en cada etapa de este camino, por su amor incondicional, su paciencia y su fe en cada uno de mis pasos.

Gracias por enseñarme que el esfuerzo tiene sentido cuando se comparte, el valor de cada experiencia y que los logros se celebran mejor cuando nacen del corazón. Esta meta también es suya.

Mariana Pinzón Madrid

A mi esposo, por su compañía incondicional, su paciencia infinita y su comprensión generosa al respetar mis tiempos académicos y acompañarme en cada paso de este proyecto. Su presencia ha sido un sostén silencioso pero firme, que me ha permitido avanzar con serenidad y confianza.

A mi madre, por su apoyo constante y sus palabras siempre oportunas, que han sido faro en los momentos de duda y aliento en los días difíciles. Su fe en mí ha sido motor y refugio, recordándome que rendirse nunca es opción cuando se camina con amor y propósito.

Luisa Fernanda Quintana Quintana

# Agradecimientos

Queremos expresar nuestro sincero agradecimiento a los docentes que, de manera indirecta, aportaron su conocimiento y experiencia al desarrollo de este trabajo. Sus enseñanzas, aunque no siempre visibles en el proceso inmediato, fueron fundamentales para construir los fundamentos que sostienen esta estrategia.

En especial, agradecemos a la docente Ingrid Durley Torres Pardo por su acompañamiento cercano, su disposición constante y su guía respetuosa, que nos permitió avanzar con claridad y confianza en cada etapa del proyecto.

También extendemos nuestro reconocimiento a nuestro amigo Juan Pablo Duque Varela, por brindarnos su tiempo y apoyo en momentos clave. Sus asesorías y explicaciones fueron esenciales para aclarar dudas, fortalecer nuestro aprendizaje y enriquecer el enfoque del trabajo.

Este proyecto es el resultado de múltiples voces, gestos y conocimientos que se entrelazan. A todos quienes hicieron parte, directa o indirectamente, les extendemos nuestra profunda gratitud.

# Resumen

En el contexto actual, las organizaciones enfrentan el reto de gestionar de manera efectiva sus activos tecnológicos, lo cual exige herramientas que permitan centralizar la información, realizar seguimiento continuo y tomar decisiones basadas en datos. Esta necesidad se hace especialmente notoria en los departamentos de Gestión TIC o áreas tecnológicas, en donde la ausencia de mecanismos integrados ha dificultado el control, la actualización y la planificación estratégica de los recursos tecnológicos.

Este trabajo de grado surge como respuesta a dicha problemática, proponiendo el desarrollo de un sistema basado en modelado de datos y análisis predictivo que facilite la clasificación, identificación y monitoreo de activos tecnológicos, en cualquier organización. La solución contempla un modelo de regresión logística que permita hacer una identificación oportuna y centralizada de los datos para validar de manera eficiente si los activos tecnológicos requieren o no mantenimiento, optimizando los tiempos para la toma de decisiones y mejorando la vida útil del activo para la prestación del servicio requerido. Con el fin de lograr desarrollar esta práctica se toma como referencia la base de datos CORPOBOYACA tomada del portal de Datos Abiertos de Colombia. Los 1.439 datos facilitaron el análisis de los requerimientos con la aplicación de métodos y técnicas de machine learning a procesos que demandan un tiempo significativo en actividades manuales y recurrentes en las entidades.

El sistema busca superar las limitaciones de los procesos manuales y descentralizados, ofreciendo una base de datos centralizada que logre garantizar en tiempo real la veracidad de la información, la trazabilidad de los activos y la mejora continua en la prestación de servicios administrativos de las empresas. Además, representa un aporte académico y práctico al integrar conocimientos de análisis de datos y administración de la tecnología, fortaleciendo la capacidad institucional y fomentando los cimientos para una gestión proactiva y sostenible que perdure en el tiempo mientras se adapta y crece con las diferentes actualizaciones y alternativas del

mercado.

Palabras clave: Activos tecnológicos, bases de datos, regresión logística, redes neuronales, técnicas de machine learning.

**Nota aclaratoria:** Los datos empleados en este trabajo fueron tomados de la base de datos pública de la Gobernación de Colombia, específicamente del conjunto "Activos de Información" de la disponible en el portal de datos abiertos del gobierno colombiano.

**Fuente:** Datos Abiertos de Colombia [datos.gov.co](https://datos.gov.co) – Corpoboyacá, conjunto "Activos de Información".

## Tabla de contenidos

1.	Introducción.....	1
2.	Planteamiento del Problema.....	3
3.	Justificación.....	5
4.	Marco de Referencias .....	8
4.1	Marco Teórico .....	8
4.2	Marco conceptual.....	12
4.3	Flujograma PRISMA 2020.....	20
4.4	Marco Normativo .....	21
4.4.3	Integración del marco normativo con la investigación.....	23
5.	Antecedentes.....	24
6.	Objetivos .....	28
6.1	Objetivo General .....	28
6.2	Objetivos Específicos .....	28
7.	Viabilidad.....	29
8.	Metodología.....	32
8.1	Etapa teórica .....	32
8.2	Etapa de implementación .....	34
9.	Resultados.....	37
10.	Conclusiones .....	79
11.	Recomendaciones .....	82
12.	Referencias.....	84

## Contenido de Tablas

<b>TABLA 1:</b> FLUJOGRAMA PRISMA 2020.....	20
<b>TABLA 2:</b> METODOLOGÍA CRISP-DM .....	34
<b>TABLA 3:</b> PREPROCESAMIENTO DE ESCALADO Y NORMALIZACIÓN DE VARIABLES .....	37
<b>TABLA 4:</b> REGRESIÓN LOGÍSTICA.....	53
<b>TABLA 5:</b> CÓDIGO CASO DE USO 01 .....	54
<b>TABLA 6:</b> CÓDIGO DE CASO DE USO 02 .....	60
<b>TABLA 7:</b> CÓDIGO DE CASO DE USO 03 .....	65
<b>TABLA 8:</b> TABLA DE RED NEURONAL .....	72

## Tabla de Ilustraciones

<b>ILUSTRACIÓN 1:</b> VARIABLES DE RESUMEN DE CLUSTER COMPLETA .....	49
<b>ILUSTRACIÓN 2:</b> VARIABLES DE RESUMEN DE CLUSTER CON IMÁGENES AMPLIADAS.....	49
<b>ILUSTRACIÓN 3:</b> MÉTODO DEL CODO .....	50
<b>ILUSTRACIÓN 4:</b> MÉTODO DE SILUETA .....	51
<b>ILUSTRACIÓN 5:</b> GRAFICAR CLUSTERS.....	52
<b>ILUSTRACIÓN 6:</b> VARIABLES Y COEFICIENTES DE LA REGRESIÓN LOGÍSTICA.....	53
<b>ILUSTRACIÓN 7:</b> SENSIBILIDAD- REPORTE DE CLASIFICACIÓN CU01 .....	59
<b>ILUSTRACIÓN 8:</b> CONFIDENCIALIDAD- REPORTE DE CLASIFICACIÓN CU02.....	64
<b>ILUSTRACIÓN 9:</b> PRIORIZACIÓN DE REEMPLAZO EN TANGIBLES- REPORTE DE CLASIFICACIÓN CU03 .....	71
<b>ILUSTRACIÓN 10:</b> MATRIZ DE CONFUSIÓN CU03 .....	71
<b>ILUSTRACIÓN 11:</b> MODELO “SEQUENTIAL” .....	75
<b>ILUSTRACIÓN 12:</b> PRECISIÓN Y ÁREA DE ROC DE LA RED NEURONAL .....	76
<b>ILUSTRACIÓN 13:</b> CURVA DE PÉRDIDA - RED NEURONAL .....	77

# 1. Introducción

En un mundo donde cada vez la tecnología está más orientada a la eficacia de las actividades cotidianas y la optimización de los tiempos y recursos, la gestión de inventarios tecnológicos se ha convertido en uno de los grandes desafíos claves para las organizaciones del mercado, un ejemplo de ello es la Corporación Autónoma Regional de Boyacá - Corpoboyacá que comprometida con la excelencia en su infraestructura tecnológica enfrenta la necesidad de optimizar, validar y controlar la planificación de sus activos tecnológicos. Este reto no sólo radica en garantizar la disponibilidad de recursos para la organización, sino también en alertar el excedente de existencias, reducir costos asociados y prever incumplimientos normativos legales.

De acuerdo con la revisión de literatura, instituciones como la (Universidad de Antioquia et al., 2025) y otras organizaciones del mercado han buscado instrumentos y estrategias que les permita implementar sistemas para garantizar el equipamiento de los activos tecnológicos, su disponibilidad y reducción de costos innecesarios en compras y reparaciones.

Entre las herramientas más destacadas, los modelos predictivos han surgido como una solución innovadora respaldada en los avances de análisis de datos y la ayuda de inteligencia artificial que en la actualidad se incorpora en la cotidianidad de las labores profesionales. IBM (2025), describe cómo la inteligencia artificial predictiva utiliza análisis estadísticos y el machine-learning para lograr hacer una identificación de los eventos futuros, estas técnicas permiten analizar patrones históricos y proyectar escenarios anticipados con precisión, para la toma de decisiones estratégicas en la adquisición, mantenimiento y renovación de equipos de hardware y software.

Además, la literatura enfatiza que la optimización de inventarios tecnológicos no sólo está categorizada en los beneficios económicos, sino también en un mejor aprovechamiento de los recursos tecnológicos para el cumplimiento de los objetivos organizacionales o académicos.

Investigaciones como InsightSoftware (*Los 5 mejores modelos y algoritmos de análisis predictivo*, 2023) destacan como los algoritmos y modelos de análisis predictivo, pueden ser aplicados para evaluar datos pasados y predecir tendencias futuras en la gestión de activos.

En este contexto, el uso de modelos predictivos basados en herramientas de análisis de datos y algoritmos avanzados emerge como una solución innovadora para la problemática actual por la que transitan las organizaciones del mercado en relación con los inventarios tecnológicos (hardware y software). Estos modelos no sólo permiten anticipar las necesidades futuras de inventarios, sino también optimizar procesos de adquisición, mantenimiento y renovación de equipos. Es así como este trabajo busca desarrollar una propuesta que integre estas tecnologías con estrategias de gestión, con el objetivo de proporcionar una solución adaptable a la necesidad organizacional.

Finalmente, además de proponer una técnica para la optimización en la gestión de inventarios tecnológicos mediante modelos predictivos, se hace un rastreo bibliográfico para lograr identificar como otras instituciones de educación superior u organizaciones han creado y diseñado las validaciones, el control y custodia de los activos de hardware y software, logrando aprovechar los datos ya existentes para construir un modelo de gestión de inventarios de activos tecnológicos que no sólo sea utilizado como instrumento de consulta, sino que también sirva para la toma de decisiones estratégicas.

## 2. Planteamiento del Problema

La gestión de los activos tecnológicos representa grandes desafíos para las organizaciones en general, a pesar de la constante evolución de las tecnologías y la creciente complejidad de los sistemas, los procesos actuales siguen siendo manuales y descentralizados por las áreas tecnológicas, la cual genera diferentes problemáticas como errores en la veracidad de los datos, falta de visibilidad en la supervisión de los activos y la ausencia de mecanismos de predicción que contribuyan a la toma de decisiones y a respuestas oportunas e inmediatas a procesos de auditoría, particularmente a los departamentos de gestión TIC de las organizaciones del mercado.

La Corporación Autónoma Regional de Boyacá - Corpoboyacá enfrenta dificultades relacionadas con la administración de sus dispositivos tecnológicos, recursos humanos, la seguridad de la información y la necesidad de actualizar constantemente herramientas físicas e informáticas para sus colaboradores y la prestación de servicios a usuarios y clientes externos. La carencia de un sistema centralizado impide optimizar procesos administrativos, realizar un rastreo del estado de los dispositivos y planificar adecuadamente el ciclo de vida para los mismos. Esto repercute directamente en la calidad del servicio prestado al público administrativo y ciudadano, así como en la capacidad operativa de la organización.

Se ha demostrado que la implementación de modelos predictivos en la gestión de inventarios tecnológicos puede optimizar recursos, mejorar la toma de decisiones basada en datos y aumentar la precisión en las predicciones o actualizaciones que se requieran. Tecnologías como la inteligencia artificial y el aprendizaje automático, han transformado la manera en que se gestionan estos activos en organizaciones alrededor del mundo. Sin embargo, Corpoboyacá aún no ha integrado estas soluciones, lo que podría poner en riesgo su competitividad y sostenibilidad a largo plazo, por el incumplimiento normativo y regulatorio de las

organizaciones de control.

Además, una gestión ineficiente genera problemas de depreciación, desabastecimiento y pérdida de información valiosa de los mismos recursos. Estas carencias afectan negativamente la experiencia de los usuarios y el desarrollo organizacional, aumentando los riesgos operativos y limitando la posibilidad de innovación tecnológica. La falta de un enfoque estratégico para la gestión de activos tecnológicos no sólo afecta el plan de inversión, sino también la continuidad de la operación y la prestación del servicio.

Por lo tanto, resulta relevante desarrollar e implementar un modelo de gestión de inventarios tecnológicos basado en modelos predictivos, que permita centralizar la administración de los activos, prever futuras necesidades y reducir los riesgos asociados a la falta de control de estos elementos distribuidos en los diferentes centros de operación y centralizarse en la sede principal ubicada en municipio de Tunja en el Departamento de Boyacá. Esta solución contribuiría significativamente al fortalecimiento organizacional, ofreciendo beneficios tangibles y automáticos, lo que se convierte posiblemente en una mejora continua en la calidad de la prestación de sus servicios para la sociedad y la naturaleza.

### 3. Justificación

La gestión de activos es un aspecto esencial para los departamentos de tecnología de las diferentes organizaciones del mercado, esto debido a que se debe tener una herramienta que permita hacer análisis y predicciones para facilitar la clasificación e identificación adecuada de estos datos para garantizar la calidad de los servicios prestados.

Para realizar este estudio, basado en la necesidad de abordar los retos y dificultades vinculadas a la gestión tecnológica, se plantean diferentes factores a tener en cuenta para el análisis y desarrollo de una respuesta adecuada acorde a las problemáticas que se encuentren en el progreso de la investigación.

La falta de mecanismos centralizados ha generado una imperiosa necesidad de implementar herramientas predictivas y de control para monitoreo de los activos de software y equipos de hardware, de esta manera evitar problemas que afectan negativamente la prestación del servicio al público administrativo y ciudadano.

En la actualidad los departamentos de tecnología enfrentan desafíos como lo son la gestión de un gran número de dispositivos, la complejidad de las redes y sistemas de comunicación, el desarrollo acelerado de las tecnologías y la protección de la información. Sin embargo, las unidades aún utilizan procesos manuales y descentralizados para el almacenamiento de sus activos disponibles, lo que puede llevar a errores, inconsistencias en la veracidad de la data y falta de visibilidad en la supervisión de los mismos, por tener conocimientos aislados y no contar con una red interna que permita compartir y actualizar en tiempo real la nueva información de los recursos.

Todas estas problemáticas se pueden optimizar brindando un análisis de datos que permita que los procesos administrativos de TIC puedan realizar rastreos profundos y una visualización del estado de los dispositivos y equipos, además de la asignación de activos a usuarios, la baja de aparatos tecnológicos, la actualización de las herramientas físicas e

informáticas, y un control en la veracidad de la data. Para evitar retrocesos por medio de una base centralizada.

Como apoyo al personal de los departamentos de tecnología, las herramientas predictivas deben contemplar la integración de modelos de aprendizaje automático - Machine Learning. Estos modelos, entrenados mediante algoritmos como la regresión logística o las redes neuronales, ofrecen una solución más dinámica y eficiente. Su implementación permite la detección temprana de fallos, lo que facilita la optimización del mantenimiento y la prevención de deterioros o daños graves antes de que ocurran.

Con las redes neuronales se pueden crear patrones, detectar datos similares o realizar una serie de acciones que permitan realizar clasificaciones predictivas o exploratorias. Por su parte, la regresión logística aporta desde la explicación del comportamiento de una variable definida en función de una serie de factores o elementos que actúan como variables independientes y que están relacionadas con la variable principal.

En resumen, la implementación de una herramienta de modelo predictivo en la que se pueda llevar un control de activos tecnológicos puede proporcionar información valiosa y beneficios para administrar y anticipar escenarios futuros mediante el análisis de datos históricos y la centralización de las actividades para su rastreo, la actualización de información en la red interna y la disponibilidad del estado de cada equipo en la organización.

Por lo anterior, la creación de este sistema representa un aporte académico y práctico significativo, el cual aliviana la necesidad de actividades del día a día del público administrativo y ciudadano, ya que combina conocimientos de análisis de datos, modelado predictivo y gestión tecnológica en una solución tangible que contribuirá al desarrollo organizacional.

Con este enfoque, el proyecto no sólo atenderá las necesidades actuales del departamento, sino que también sentará las bases para una gestión proactiva y estratégica, que le aporta al bienestar institucional al verse beneficiado todos quienes lo conforman, al tiempo que

se garantizará la continuidad operativa y el fortalecimiento de la capacidad institucional en los años venideros.

## **4. Marco de Referencias**

### **4.1 Marco Teórico**

#### **4.1.1 Gestión tecnológica**

La gestión tecnológica constituye uno de los pilares estratégicos más relevantes dentro de las organizaciones modernas, en especial ante los cambios acelerados del entorno competitivo y la necesidad de adaptarse a los avances tecnológicos. Este proceso comprende la creación, desarrollo, transferencia, adaptación y aprovechamiento de tecnologías que permiten fortalecer la competitividad empresarial (Amador & Márquez, 2008). Según Dankbaar, citado por Escorsa y Valls, la gestión tecnológica involucra las actividades de identificación, adquisición, investigación, desarrollo y explotación de tecnologías para la producción de bienes y servicios (como se cita en Amador & Márquez, 2008).

En este contexto, los activos tecnológicos se consolidan como recursos estratégicos que sustentan la operación y la innovación organizacional. El rastreo de activos tecnológicos, entendido como el proceso de identificación, monitoreo y control de los bienes tecnológicos, resulta esencial para garantizar su disponibilidad y óptimo desempeño. De acuerdo con Gartner (2021), un sistema de rastreo adecuado permite anticipar fallas, planificar mantenimientos preventivos y minimizar los tiempos de inactividad, contribuyendo a mejorar la eficiencia operativa y la continuidad del servicio para las diferentes organizaciones independiente a su actividad económica.

Hamilton y Pezo definen la gestión tecnológica como “un sistema de conocimientos que tiene por función el desarrollo, la integración y el uso eficaz de los recursos tecnológicos para crear valor y satisfacer competitivamente las necesidades de los clientes” (como se cita en

Amador & Márquez, 2008, p. 50). En concordancia, Hidalgo, León y Pavón (2002) sostienen que la gestión tecnológica posibilita el uso eficiente tanto de la tecnología interna como de la adquirida externamente, al incorporar en los procesos de innovación de productos y servicios.

#### **4.1.2. Modelos de gestión tecnológica**

A lo largo del tiempo se han formulado diferentes modelos teóricos que orientan la administración de la tecnología dentro de las organizaciones. Sumanth, citado por Gaynor (1999), propone un modelo de carácter sistémico compuesto por cinco fases: percepción, adquisición, adaptación, avance y abandono. Este enfoque permite identificar tecnologías emergentes, incorporarlas, ajustarlas a las necesidades organizacionales y reemplazarlas cuando se vuelven obsoletas (Amador & Márquez, 2008).

Por su parte, Hidalgo, León y Pavón (2002) desarrollan un modelo basado en procesos específicos de gestión tecnológica que abarcan la identificación, evaluación, selección, adquisición, asimilación y utilización de tecnologías, los cuales deben ajustarse a las características de cada sector empresarial. Este enfoque resulta especialmente pertinente para la necesidad planteada, dado que permite estructurar de manera ordenada cómo una organización o institución puede reconocer sus requerimientos tecnológicos, valorar su pertinencia, incorporar soluciones acordes con su entorno y asegurar su adecuada integración en los procesos internos. En este sentido, el modelo orienta la toma de decisiones estratégicas y facilita la alineación entre las capacidades tecnológicas y los objetivos institucionales, así se garantiza que la adopción tecnológica sea efectiva, pertinente y sostenible dentro del contexto organizacional.

De manera complementaria, el modelo Temaguide (Cotec, 1998) propone un enfoque integral al estructurar la gestión tecnológica en cinco elementos: vigilar, focalizar, capacitarse, implantar y aprender, enfatizando el aprendizaje organizacional como componente central para

la mejora continua. Finalmente, Hidalgo (1999) plantea un modelo estructurado en funciones activas y de apoyo, orientadas a la evaluación del potencial tecnológico, el diseño de estrategias, el fortalecimiento del patrimonio tecnológico, la vigilancia del entorno y la protección de las innovaciones (Amador & Márquez, 2008).

Estos enfoques coinciden en que la gestión tecnológica debe ser un proceso dinámico, sistemático y adaptable a las necesidades de cada organización, integrando la innovación como motor de crecimiento y sostenibilidad.

#### **4.1.3. Herramientas predictivas aplicadas a la gestión de activos tecnológicos**

En la actualidad, las herramientas de analítica predictiva han transformado la manera en que las organizaciones administran sus activos tecnológicos. Estas combinan análisis de datos, aprendizaje automático (machine learning) y métodos estadísticos avanzados para anticipar eventos como fallas, obsolescencia o requerimientos de actualización (Chien & Chen, 2021). Mediante la integración de datos provenientes de sensores, plataformas de monitoreo remoto y sistemas empresariales, es posible obtener una visión completa del estado y desempeño de los activos para la toma de decisiones y la mejora continua.

De acuerdo con Aljumah et al. (2022), la incorporación de sistemas predictivos en la gestión de activos facilita la toma de decisiones basadas en datos, optimizando costos, reduciendo interrupciones operativas y elevando la calidad del servicio tanto para usuarios internos como externos. Estas herramientas fortalecen el control interno y contribuyen a una gestión proactiva del riesgo tecnológico (Institute of Internal Auditors [IIA], 2020).

La analítica predictiva se convierte también en un aliado de la mejora continua, entendida como un ciclo permanente de optimización de procesos, productos o servicios (Deming, 1986). Según Prajogo y Sohal (2001), las decisiones basadas en datos reales promueven una cultura

de innovación y excelencia, impulsando a las organizaciones hacia un desempeño más competitivo y sostenible.

En cuanto a las técnicas más empleadas, se destacan los modelos estadísticos tradicionales, los árboles de decisión, las máquinas de soporte vectorial (SVM) y, especialmente, las redes neuronales artificiales, por su capacidad de aprender patrones complejos en grandes volúmenes de datos. Por ejemplo, Kim et al. (2020) demostraron en un estudio sobre mantenimiento predictivo industrial que una red neuronal profunda alcanzó una precisión superior al 90 % en la detección de fallas. De forma similar, la empresa ABB (2019) aplicó redes neuronales para el monitoreo de activos eléctricos críticos, logrando mejoras significativas en la identificación temprana de anomalías.

Estas experiencias evidencian que el uso de modelos predictivos y redes neuronales fortalece la supervisión inteligente de los activos tecnológicos, permitiendo actuar de manera anticipada frente a posibles fallos y asegurando la continuidad del servicio en entornos empresariales altamente exigentes.

#### **4.1.4. Síntesis del marco teórico**

La gestión tecnológica, junto con la incorporación de herramientas predictivas, se consolida como un eje fundamental para la administración estratégica de los activos tecnológicos. Este enfoque no solo optimiza el uso de los recursos disponibles, sino que también impulsa la innovación, la toma de decisiones basada en evidencia y la sostenibilidad operativa. En consecuencia, las organizaciones que integran prácticas de gestión tecnológica y analítica predictiva logran una mayor eficiencia, competitividad y capacidad de adaptación frente a los retos de la transformación digital.

## 4.2 Marco conceptual

En el desarrollo del rastreo de las bases de datos se tuvieron en cuenta los siguientes conceptos básicos para la búsqueda de la información y el análisis de los datos requeridos.

- **Activos Tecnológicos:** Son recursos tangibles e intangibles, como hardware, software y sistemas digitales, que se utilizan para apoyar procesos organizacionales y educativos. Según García y López (2020), "los activos tecnológicos representan un componente esencial en la infraestructura de las organizaciones modernas, permitiendo la innovación y la competitividad en un entorno digital" (p. 45).

- **Hardware:** conjunto de componentes físicos y tangibles de un sistema informático, como circuitos, placas, memorias y dispositivos periféricos, que permiten el funcionamiento del equipo (Concepto.de 2025).

- **Software:** componente lógico de un sistema de cómputo, dividido en categorías como software de sistema, software de aplicación y software de programación, cada una con funciones específicas (Villalobos 2021).

- **Modelo Predictivo:** herramienta estadística utilizada para predecir el valor de una variable de interés mediante el análisis de datos pasados, siendo ampliamente aplicados en áreas como la detección de fraudes, la predicción de la demanda y el análisis de riesgos (Gamco 2025).

- **Algoritmo:** Este concepto hace referencia a un conjunto de pasos para realizar una tarea determinada. Las computadoras aplican los algoritmos para ejecutar procesos de manera más eficiente y ágil (GCF Global, s.f.).

- **Machine Learning (Aprendizaje automático:)** Es una de las ramas de la inteligencia artificial. Es la responsable de crear algoritmos capaces de aprender de los datos y los patrones encontrados, de tal manera que no tienen que ser programados en cada ejecución.

De esta manera, el algoritmo aprende cómo responder ante todos los posibles escenarios (Sandoval, 2018).

- **Big Data:** Se refiere a conjuntos de datos de un volumen, velocidad y variedad tan grandes que los métodos y herramientas tradicionales de procesamiento de datos no son suficientes para capturarlos, gestionarlos y analizarlos. (Gartner. s. f.).

- **Red Neuronal:** Es un modelo de Machine Learning inspirado en el funcionamiento del cerebro humano. Consiste en capas de "neuronas" interconectadas que procesan información para aprender patrones complejos y realizar tareas como clasificación o predicción. (Haykin, S. 1999).

- **Regresión Logística:** Un algoritmo de clasificación (no de regresión, a pesar del nombre) usado principalmente para predecir la probabilidad de un resultado binario (por ejemplo, sí/no, 0/1, spam/no spam) ajustando los datos a una función logística o sigmoide. (IBM. s. f.)

- **Precisión:** La proporción de predicciones positivas correctas (verdaderos positivos) respecto al total de predicciones positivas realizadas (verdaderos positivos + falsos positivos).

- **Recall (Sensibilidad o Cobertura):** La proporción de casos positivos reales que fueron identificados correctamente (verdaderos positivos) respecto al total de casos positivos reales (verdaderos positivos + falsos negativos).

- **Curvas ROC (Receiver Operating Characteristic):** Es una gráfica que ilustra el rendimiento de un clasificador binario. Muestra la tasa de verdaderos positivos (TPR) en función de la tasa de falsos positivos (FPR) para diferentes umbrales de decisión. Un área bajo la curva (AUC) mayor indica mejor rendimiento.

- **Matriz de Confusión:** Es una tabla que resume el rendimiento de un algoritmo de clasificación. Muestra cuántas predicciones fueron correctas e incorrectas, desglosando los resultados en Verdaderos Positivos, Verdaderos Negativos, Falsos Positivos y Falsos Negativos.

- **CRISP-DM (Cross Industry Standard Process for Data Mining):** Es una metodología estándar de la industria para guiar los proyectos de minería de datos o ciencia de datos, que consta de seis fases: Comprensión del Negocio, Comprensión de los Datos, Preparación de los Datos, Modelado, Evaluación y Despliegue.

- **Python:** Es un lenguaje de programación de alto nivel, interpretado y de propósito general, ampliamente utilizado en ciencia de datos y Machine Learning por su sintaxis clara y su gran ecosistema de librerías.

- **Google Colab (Colaboratory):** Es un servicio basado en la nube de Google que permite escribir y ejecutar código Python (y otros) a través de un Jupyter Notebook, sin necesidad de configuración.

- **Jupyter Notebook:** Es un entorno interactivo basado en web que permite crear y compartir documentos que contienen código ejecutable (celdas), visualizaciones y texto narrativo (Markdown), ideal para la experimentación y el análisis de datos.

- **Scikit-learn (sklearn):** Es la librería de Machine Learning más popular de Python. Proporciona una colección consistente y eficiente de herramientas para tareas como clasificación, regresión, clustering y reducción de dimensionalidad.

- **Pandas:** Es una librería de Python esencial para la manipulación y análisis de datos. Introduce estructuras de datos como los DataFrames (similares a tablas de bases de datos o hojas de cálculo), facilitando la limpieza y preparación de datos.

- **Numpy:** Es una librería fundamental de Python que permite trabajar de manera eficiente con arrays multidimensionales (matrices) y funciones matemáticas de alto nivel. Es la base para muchas otras librerías científicas.

- **Matplotlib:** Es una librería de trazado y visualización estática, animada e interactiva en Python. Es utilizada para crear una amplia variedad de gráficos, como histogramas, diagramas de dispersión y gráficos de líneas.

- **Tensorflow:** Es una biblioteca de software integral que facilita la creación y el entrenamiento de modelos de Machine Learning y Deep Learning, permitiendo su despliegue en diversas plataformas, desde servidores hasta dispositivos móviles y web.
- **Normalización:** Una técnica de preprocesamiento de datos que reescala las características (columnas) a un rango fijo, generalmente entre 0 y 1, útil cuando se usan algoritmos que asumen distribuciones acotadas.
- **Escalado (Estandarización):** Otra técnica de preprocesamiento que transforma los datos para que tengan una media de 0 y una desviación estándar de 1 (distribución normal estándar), útil cuando los algoritmos son sensibles a la magnitud de las características.
- **Cluster:** Es un grupo o conjunto de puntos de datos que son más similares entre sí (en términos de alguna medida de distancia) que con los puntos de otros grupos. Es el resultado de aplicar un algoritmo de clustering (agrupamiento).
- **Método del Codo:** Es un método heurístico utilizado para estimar el número óptimo de clusters (k). Consiste en graficar la varianza explicada (o el error cuadrático medio dentro del cluster) en función del número de clusters y buscar el punto donde la mejora disminuye drásticamente, formando una "rodilla" o "codo".
- **Método Silueta (Silhouette Score):** Es una métrica de evaluación para determinar qué tan bien se han separado los clusters. Mide la similitud de un objeto con su propio cluster (cohesión) en comparación con otros clusters (separación), con valores entre -1 (mal agrupado) y +1 (bien agrupado).
- **Pipeline (Tubería):** En Machine Learning, es una secuencia encadenada de transformaciones y estimadores (modelos), donde la salida de un paso se convierte en la entrada del siguiente. Simplifica y automatiza el flujo de trabajo de preprocesamiento y modelado, previniendo el "data leakage".

Para la búsqueda de documentos de referencia fueron utilizadas las siguientes palabras claves: Implementación, activos, Hardware, software, activos fijos, gestión activos fijos tecnológicos, inventario de activos tecnológicos, gestión de inventarios, big data, inteligencia empresarial, gestión de activos, gestión de la información, correlación, minería de datos, gestión de datos, integración de datos, análisis de los datos, detectabilidad de datos, algoritmos, clasificación, herramientas de análisis predictivo, redes neuronales, Machine Learning.

Dentro del ejercicio de investigación y rastreo bibliográfico se utilizaron cadenas o ecuaciones de la búsqueda para acortar la investigación de los artículos utilizados como referencia de la siguiente manera:

TITLE-ABS-KEY (assets) AND (fixed) AND PUBYEAR > 2014 AND PUBYEAR < 2025 AND PUBYEAR > 2018 AND PUBYEAR < 2025

TITLE-ABS-KEY (assets) AND (fixed) OR (hardware) AND (assets) OR (data") AND (analytics) AND PUBYEAR > 2014 AND PUBYEAR < 2025 AND PUBYEAR > 2018 AND PUBYEAR < 2025

TITLE-ABS-KEY (hardware) AND (data) AND (análisis) AND (Python) AND (sql) ( LIMIT-TO ( DOCTYPE, "ar") AND LIMIT-TO ( LANGUAGE , "English" ) ) AND PUBYEAR > 2014 AND PUBYEAR < 2025 AND PUBYEAR > 2018 AND PUBYEAR < 2025

TITLE-ABS-KEY (hardware) AND (acceleration) AND (data) AND (análisis) AND PUBYEAR > 2014 AND PUBYEAR < 2025 AND PUBYEAR > 2018 AND PUBYEAR < 2025

TITLE-ABS-KEY (Modelo) AND (Gestión) AND (Inventarios) AND PUBYEAR > 2014 AND PUBYEAR < 2025 AND PUBYEAR > 2018 AND PUBYEAR < 2025

TITLE-ABS-KEY (cooperative) AND (data) AND (big) AND (data) ( LIMIT-TO ( DOCTYPE , "ar") AND LIMIT-TO ( LANGUAGE , "English" ) ) AND PUBYEAR > 2014 AND PUBYEAR < 2025 AND PUBYEAR > 2018 AND PUBYEAR < 2025

TITLE-ABS-KEY (Valoración de activos tecnológicos) AND (metodologías de evaluación) AND PUBYEAR > 2014 AND PUBYEAR < 2025 AND PUBYEAR > 2018 AND PUBYEAR < 2025

TITLE-ABS-KEY (Activos intangibles) OR (capital intelectual) AND (herramientas tecnológicas) AND PUBYEAR > 2014 AND PUBYEAR < 2025 AND PUBYEAR > 2018 AND PUBYEAR < 2025

TITLE-ABS-KEY (Inversiones tecnológicas) AND (opciones reales) AND PUBYEAR > 2014 AND PUBYEAR < 2025 AND PUBYEAR > 2018 AND PUBYEAR < 2025

TITLE-ABS-KEY (Gestión de activos tecnológicos) AND (innovación empresarial) AND PUBYEAR > 2014 AND PUBYEAR < 2025 AND PUBYEAR > 2018 AND PUBYEAR < 2025

TITLE-ABS-KEY (Optimización) AND (activos) AND (tecnología) AND (aplicada) AND PUBYEAR > 2014 AND PUBYEAR < 2025 AND PUBYEAR > 2018 AND PUBYEAR < 2025

TITLE-ABS-KEY (fixed AND assets) AND (LIMIT-TO (SUBJAREA, "ENGI")) AND PUBYEAR > 2014 AND PUBYEAR < 2025 AND PUBYEAR > 2018 AND PUBYEAR < 2025

TITLE-ABS-KEY (fixed AND assets) AND (LIMIT-TO (SUBJAREA, "ENGI")) AND PUBYEAR > 2014 AND PUBYEAR < 2025 AND PUBYEAR > 2018 AND PUBYEAR < 2025

TITLE-ABS KEY (modelo AND gestión AND inventarios) AND PUBYEAR > 2014 AND PUBYEAR < 2025 AND PUBYEAR > 2018 AND PUBYEAR < 2025

TITLE-ABS-KEY (fixed AND assets) AND (LIMIT-TO (SUBJAREA, "ENGI")) AND (LIMIT-TO (LANGUAGE, "English") OR LIMIT-TO (LANGUAGE, "Spanish")) AND (LIMIT-TO (EXACTKEYWORD, "Fixed Assets") OR LIMIT-TO (EXACTKEYWORD, "Decision Making") OR LIMIT-TO (EXACTKEYWORD, "Asset Management") OR LIMIT-TO (EXACTKEYWORD, "Regression Analysis") OR LIMIT-TO (EXACTKEYWORD, "Risk Assessment") OR LIMIT-TO (EXACTKEYWORD, "Optimization") OR LIMIT-TO (EXACTKEYWORD, "Forecasting") OR LIMIT-TO (EXACTKEYWORD, "Cost Benefit Analysis") OR LIMIT-TO (EXACTKEYWORD, "Risk Management") OR LIMIT-TO (EXACTKEYWORD, "Strategic Planning") OR LIMIT-TO

(EXACTKEYWORD, "Data Envelopment Analysis") OR LIMIT-TO (EXACTKEYWORD, "Planning") OR LIMIT-TO (EXACTKEYWORD, "Risk Analysis") OR LIMIT-TO (EXACTKEYWORD, "Optimisations") OR LIMIT-TO (EXACTKEYWORD, "Information Technology") OR LIMIT-TO (EXACTKEYWORD, "Computer Simulation") OR LIMIT-TO (EXACTKEYWORD, "Infrastructure") OR LIMIT-TO (EXACTKEYWORD, "Fixed Assets Management") OR LIMIT-TO (EXACTKEYWORD, "Algorithms") OR LIMIT-TO (EXACTKEYWORD, "Big Data") OR LIMIT-TO (EXACTKEYWORD, "Database Systems")) AND (LIMIT-TO (DOCTYPE, "ar"))

TITLE-ABS-KEY (management AND technological AND inventories AND machine AND learning) AND (LIMIT-TO(SUBJAREA,"ENGI")) AND (LIMIT-TO (DOCTYPE,"ar")) AND (LIMIT-TO (LANGUAGE, "English"))

TITLE-ABS-KEY ( ANALISIS PREDICTIVO ) AND ( LIMIT-TO ( SUBJAREA , "ENGI" ) ) AND ( LIMIT-TO ( DOCTYPE , "AR" ) OR LIMIT-TO ( DOCTYPE , "CP" ) OR LIMIT-TO ( DOCTYPE , "RE" ) ) AND ( LIMIT-TO ( LANGUAGE , "SPANISH" ) ) AND ( LIMIT-TO ( EXACTKEYWORD , "BIG DATA" ) )

TITLE-ABS-KEY ( ANALISIS PREDICTIVO ) AND ( LIMIT-TO ( DOCTYPE , "AR" ) OR LIMIT-TO ( DOCTYPE , "CP" ) OR LIMIT-TO ( DOCTYPE , "RE" ) ) AND ( LIMIT-TO ( SUBJAREA , "ENGI" ) ) AND ( LIMIT-TO ( SUBJAREA , "ENGI" ) ) AND ( LIMIT-TO ( DOCTYPE , "AR" ) O LIMIT-TO ( DOCTYPE , "CP" ) O LIMIT-TO ( DOCTYPE , "RE" ) ) Y ( LIMIT-TO ( LANGUAGE , "SPANISH" ) )

Para la búsqueda de documentos de referencia fueron utilizadas las bases de datos científicas Science Direct, Scopus y Oxford academic. Así mismo, la base de datos académica google académico.

Con la finalidad de tener documentación actualizada la búsqueda desarrollada se generó

entre los rangos de años [2015] – [2025]

Lo que quiere decir vetustez inferior a 10 años, representado en un 80%, adicionando a esto una complementación de artículos que corresponden en un 20% a fechas un poco menos recientes, pero igual de válidas y sólidas en sus conceptos, trascendiendo el tiempo y siendo aplicables en la actualidad.

Y los criterios definidos para la inclusión de los artículos fueron:

Criterio 1. Documentos publicados que aborden la gestión de activos en universidades o instituciones de educación superior.

Criterio 2. Documentos que exploren la implementación de sistemas y herramientas de gestión de activos.

Criterio 3. Artículos que analicen la relación entre la gestión de activos y la toma de decisiones dentro de instituciones y organizaciones.

Criterio 4. Un estudio comparativo de inteligencia empresarial e inteligencia artificial con análisis de big data donde se desarrolla acciones de implementación en data de manera exitosa.

Al igual que los de exclusión los cuales fueron:

Criterio 1. Documentos que no exploren la relación entre gestión de activos y la toma de decisiones en instituciones y organizaciones.

Criterio 2. Documentos que están enfocados solo en los sectores de ciencia y salud.

Criterio 3. Documentos con textos incompletos, no verificados por bases de información académica o científica o no disponibles.

Criterio 4. Documentos que no aborden ciencia de datos, gestión de activos, big data o información relacionada a la investigación.

Criterio 5. No se menciona la implementación exitosa de herramientas y/o automatización en el proceso de Activos de inventarios.

### 4.3 Flujograma PRISMA 2020.

Para condensar lo anteriormente mencionado en el rastreo bibliográfico y las bases de datos utilizadas se ilustra a través del Flujograma Prisma que se relaciona a continuación:

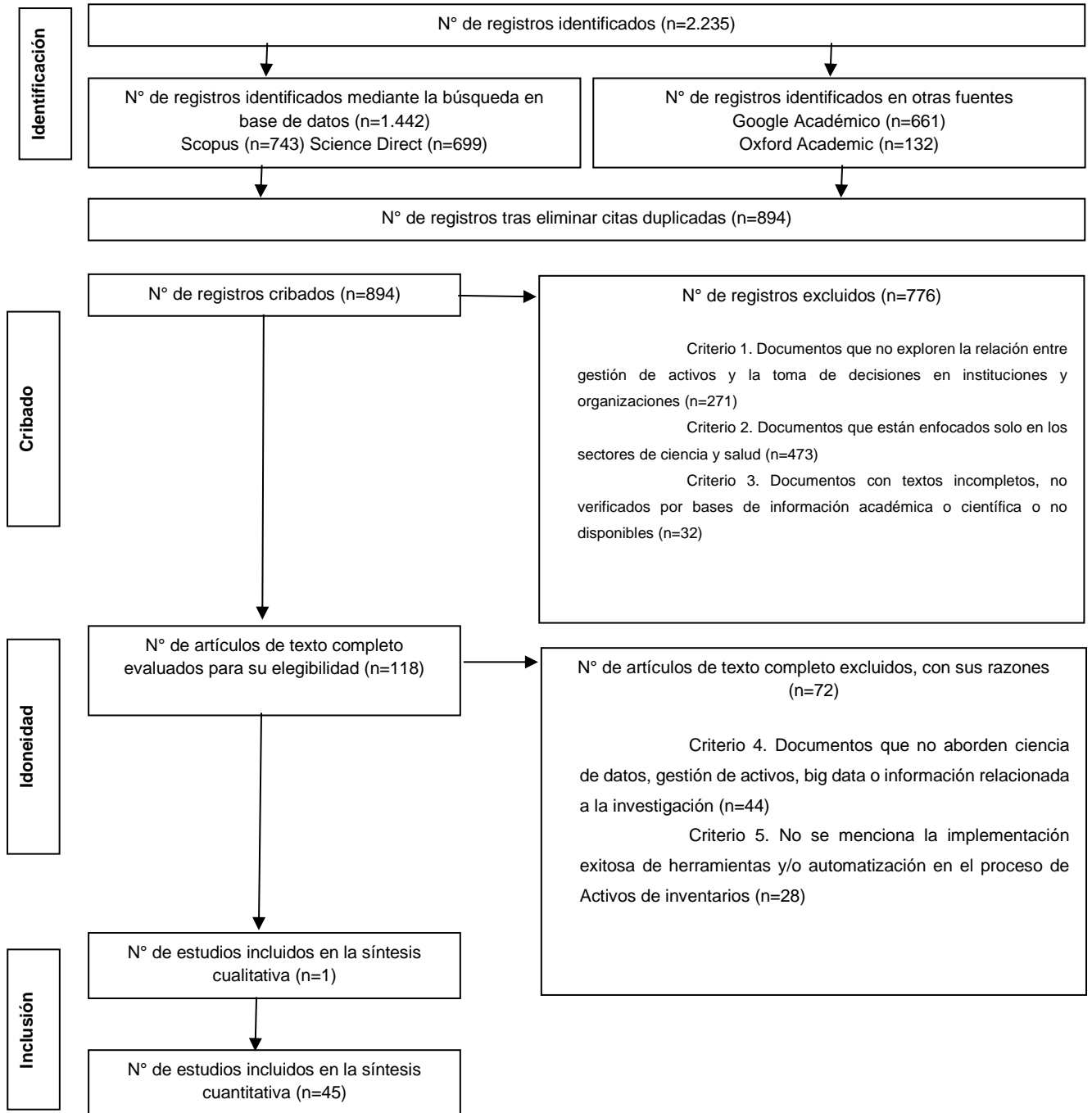


Tabla 1:Flujograma PRISMA 2020.

## **4.4 Marco Normativo**

La implementación de modelos predictivos en la gestión de activos tecnológicos dentro de las organizaciones empresariales debe sustentarse en un marco normativo que oriente las decisiones técnicas, garantice la seguridad de la información y promueva el uso ético de las herramientas de machine learning. Este marco establece las directrices que permiten operar bajo estándares internacionales y nacionales, asegurando que las soluciones tecnológicas desarrolladas sean confiables, responsables y sostenibles a largo plazo en cualquier organización.

### **4.4.1 Normas internacionales aplicables**

- **ISO 27001: Seguridad de la información**

La norma ISO/IEC 27001 establece los lineamientos para crear y mantener un Sistema de Gestión de Seguridad de la Información (SGSI). En las organizaciones, esta norma resulta esencial para proteger los datos relacionados con inventarios tecnológicos, registros de mantenimiento y sistemas predictivos. Su objetivo principal es garantizar la confidencialidad, integridad y disponibilidad de la información, elementos indispensables para la confiabilidad de los modelos de análisis. Aplicar ISO 27001 en una empresa, implica implementar controles de acceso, políticas de respaldo, auditorías internas y mecanismos de prevención de incidentes. Con esto se asegura que los datos utilizados por los algoritmos predictivos sean tratados bajo entornos seguros, evitando brechas o manipulaciones que puedan afectar su precisión o su confiabilidad en su resultado.

- **ISO 55001: Gestión de activos**

La norma ISO 55001 ofrece un marco integral para la gestión eficiente de los activos físicos, enfocándose en su valor estratégico dentro de la organización. En el contexto empresarial, su aplicación garantiza que los equipos tecnológicos sean administrados de manera

planificada, asegurando su operatividad, mantenimiento y renovación oportuna.

La adopción de esta norma permite vincular la gestión de activos con los objetivos corporativos, optimizando costos y mejorando la productividad. Además, cuando se integra con modelos predictivos, potencializan la capacidad de anticipar fallas, planificar mantenimientos y extender la vida útil del hardware, en coherencia con el principio de mejora continua establecido en la norma.

#### **4.4.2 Normatividad nacional vigente.**

- **Ley 1581 de 2012: Protección de datos personales**

La Ley 1581 de 2012 establece las disposiciones generales para la protección de datos personales en Colombia. En el entorno empresarial, su cumplimiento es fundamental para garantizar la confidencialidad de la información que alimenta los modelos predictivos y los sistemas de gestión de activos.

Toda empresa que procese datos ya sean de empleados, clientes o proveedores debe contar con la autorización correspondiente, aplicar medidas de seguridad y asegurar el uso legítimo de la información. En este sentido, los datos utilizados para entrenar modelos predictivos deben ser anonimizados o cifrados, evitando vulneraciones a la privacidad y fortaleciendo la confianza en las soluciones tecnológicas implementadas.

- **Decreto 1078 de 2025: Gestión de activos tecnológicos en entidades públicas y privadas**

El Decreto 1078 de 2025 compila las normas del sector de Tecnologías de la Información y las Comunicaciones (TIC), incorporando directrices sobre la gestión de activos tecnológicos en organizaciones públicas y privadas. Este decreto promueve la adopción de sistemas integrados de monitoreo, trazabilidad y control automatizado de los recursos tecnológicos.

En el ámbito empresarial, su aplicación favorece la modernización de los procesos internos y la eficiencia operativa. Los modelos predictivos responden a este marco legal al permitir decisiones basadas en evidencia, reducir costos y garantizar la transparencia en la administración de los activos tecnológicos.

#### **4.4.3 Integración del marco normativo con la investigación.**

La convergencia de normas internacionales, legislación nacional y principios éticos proporciona la base para una implementación responsable y sostenible de los modelos predictivos en las organizaciones empresariales. Este marco orienta la investigación hacia la construcción de soluciones que respeten la privacidad de los datos, fortalezcan la gestión de activos tecnológicos y promuevan la innovación bajo parámetros de seguridad y equidad.

Así, el cumplimiento de estas disposiciones garantiza que el modelo propuesto no sólo sea viable técnica y operativamente, sino que también actúe en coherencia con los valores corporativos y las exigencias regulatorias del mercado actual, además de la transparencia e implementación en bases de datos sin ninguna afectación en organizaciones de mercado.

## 5. Antecedentes

La gestión de inventarios ha sido un campo de creciente interés tanto para instituciones educativas como para organizaciones empresariales que buscan optimizar sus recursos y mejorar la eficiencia operativa. Gracias al desarrollo tecnológico y los avances en el aprendizaje automático (*machine learning*), han surgido múltiples modelos predictivos aplicados a la administración de inventarios tecnológicos, con el objetivo de optimizar procesos, fortalecer la toma de decisiones estratégicas, dar sentido y valor al tiempo de las actividades realizadas y, en muchos casos, racionalizar los presupuestos asignados. Estos modelos permiten a las organizaciones anticipar situaciones futuras de sus activos, detectar patrones en grandes volúmenes de datos y actuar preventivamente frente a posibles fallas o ineficiencias en los sistemas tecnológicos.

Diversas empresas del mercado global han adoptado modelos de *machine learning* como parte de su estrategia de gestión de activos. Un ejemplo destacado es el enfoque Edge ML, que permite ejecutar modelos de aprendizaje automático directamente en dispositivos de borde, reduciendo la latencia y mejorando la privacidad de los datos (Solución de inteligencia perimetral, 2023). En el ámbito corporativo, IBM desarrolló el modelo Predictive Analytics for Server Incident Reduction (PASIR), basado en algoritmos de clasificación y regresión para anticipar incidentes en servidores, logrando una disminución significativa de los tiempos de inactividad y los costos operativos (IBM, 2023).

De manera similar, Microsoft ha aplicado modelos de *machine learning* en la administración de sus centros de datos, con el fin de prever fallas en el hardware, optimizar el uso energético y extender el ciclo de vida de los equipos tecnológicos (Microsoft Sustainability Report, 2024)

En el contexto académico, Ramírez (2023) exploró modelos de *machine learning* aplicados a servicios en la nube, evaluando su efectividad mediante simulaciones estadísticas.

Por su parte, Hernández Rodríguez y Torres Aponte (2021) demostraron la utilidad de los modelos de aprendizaje supervisado y no supervisado en el pronóstico de demanda, destacando su papel en la reducción de costos operativos y la mejora de la toma de decisiones estratégicas. En el campo logístico, Jofra y Gómez (2018) señalaron la necesidad de especialización técnica en la adopción de herramientas predictivas, mientras que Rey y Valle (2024) resaltan cómo el uso de modelos de aprendizaje automático mejora la planificación de rutas, la trazabilidad y la eficiencia en empresas exportadoras.

En relación con los marcos regulatorios, Hernández García (2019) subraya que la ISO 27002:2013 es esencial para garantizar la seguridad y el tratamiento ético de los datos utilizados en modelos de *machine learning* orientados a la gestión de activos tecnológicos. En Colombia, los tableros de control desarrollados en Medellín mediante Power BI (Jiménez Aristizábal & Arango Arango, 2023) muestran cómo la analítica de datos y el aprendizaje automático pueden aplicarse para monitorear en tiempo real el rendimiento y la trazabilidad del hardware y software en las empresas, fortaleciendo los mecanismos de control interno.

Desde una perspectiva metodológica, Chaves Jiménez (2021) expone la aplicación del marco ITIL V4 en la gestión de activos tecnológicos, destacando su alineación con los modelos predictivos basados en *machine learning* y su capacidad de integrarse con la estrategia de negocio. En la misma dirección, Wahedi et al. (2023) plantean que la gestión de inventarios debe concebirse como una competencia estratégica, donde la combinación de *machine learning* y analítica predictiva permite a las organizaciones generar pronósticos precisos y visualizar con mayor claridad la demanda y el comportamiento de sus recursos.

Casos reales como los de Siemens (2024) evidencian la aplicación de redes neuronales y algoritmos de regresión para anticipar mantenimientos y optimizar la producción en sus plantas industriales, reduciendo paradas no planificadas y mejorando la confiabilidad de los equipos.

Investigaciones de Paula Vidal et al. (2022) y Guzmán et al. (2019) reafirman que la

combinación entre tecnología y modelos predictivos basados en *machine learning* es clave para reducir el riesgo de desabastecimiento y mejorar la eficiencia organizacional. De igual forma, Soori et al. (2025) y Falkner et al. (2024) muestran cómo la integración del *big data* con técnicas de aprendizaje automático permite mejorar el rendimiento de los activos y la adaptabilidad de las organizaciones a entornos cambiantes. En el sector empresarial, Accenture (2023) documenta que la adopción de *machine learning* en la gestión de activos reduce los costos de mantenimiento en un 20 % y mejora la productividad hasta en un 30 %, demostrando su impacto directo en la rentabilidad corporativa.

Villegas-Ch et al. (2024) evidencian el impacto positivo de las redes neuronales convolucionales (CNN) en la clasificación y conteo de productos, aumentando la precisión y reduciendo los tiempos de inventario. Por su parte, Kovárník y Janová (2025) comprobaron la efectividad de modelos predictivos en la gestión de recursos forestales, lo que refuerza la aplicabilidad transversal del *machine learning* en diferentes sectores. A nivel empresarial, Ecopetrol (2024) ha integrado modelos de aprendizaje automático en la detección temprana de anomalías en sus sistemas eléctricos, mientras que Enel ha adoptado algoritmos predictivos para optimizar la distribución energética, reduciendo pérdidas y mejorando la estabilidad operativa.

En materia de sostenibilidad, Banikya et al. (2024) destacan que los modelos de *machine learning* son herramientas clave para gestionar activos de baja rotación, permitiendo generar estrategias de aprovechamiento y extender la vida útil de los recursos. En la misma línea, Ortiz Panqueva et al. (2023) identifican la planificación estratégica, la eficiencia operativa y la sostenibilidad como factores motivadores para la adopción de modelos predictivos en la gestión empresarial. Estas iniciativas contribuyen a la reducción del impacto ambiental y fortalecen las políticas de economía circular dentro de las organizaciones.

Casos locales, como el de la Universidad Católica Luis Amigó descrito por Lucio Pillasagua (2025), demuestran la necesidad de estructurar sistemas de control eficientes para

conocer la disponibilidad y estado de los activos tecnológicos. Aunque este ejemplo proviene del ámbito académico, su metodología puede extrapolarse al sector empresarial, donde la trazabilidad de los activos determina la rentabilidad y sostenibilidad. Torres Estrella y Mendoza Arechua (2019) sostienen que disponer de registros contables precisos es fundamental para una gestión eficiente, mientras que Flores Tapia y Flores Cevallos (2023) advierten que la ausencia de estandarización en inventarios puede comprometer la planificación estratégica y la capacidad competitiva.

Fatorachian et al. (2025) y Chopra et al. (2020) coinciden en que la falta de modelos predictivos y de gestión basados en datos genera desorganización, sobrecostos y pérdida de información valiosa. Asimismo, Lins Espíndola (2019) e Iluore (2020) refuerzan la necesidad de contar con sistemas de conocimiento sobre los recursos disponibles para una planificación más eficiente. Rolón Ramírez et al. (2024) y Regulki (2024) alertan sobre las consecuencias de no tener bases de datos centralizadas, lo que limita la capacidad analítica y reduce la toma de decisiones fundamentadas. De acuerdo con Koster (2021), estas carencias tecnológicas disminuyen la competitividad de las empresas frente a los retos de la transformación digital.

Finalmente, experiencias empresariales recientes como la implementación de RFID y análisis predictivo mediante *machine learning* en compañías como Amazon, DHL y Microsoft (Tech Monitor, 2024) demuestran que la integración de estas tecnologías en la gestión de inventarios incrementa la trazabilidad, optimiza los tiempos operativos y mejora la sostenibilidad de los procesos. Estos avances, junto con los desarrollos aplicados en el sector energético y manufacturero, evidencian que el *machine learning* no solo fortalece la eficiencia interna, sino que impulsa la competitividad organizacional en mercados cada vez más dinámicos.

## **6. Objetivos**

### **6.1 Objetivo General**

Desarrollar un modelo de gestión de inventarios tecnológicos para aplicar en organizaciones del mercado, mediante el uso de herramientas predictivas basadas en análisis de datos, con el fin de entregar la información requerida para optimizar los procesos de administración y la toma de decisiones estratégicas en el ciclo de vida de estos activos.

### **6.2 Objetivos Específicos**

1. Caracterizar las variables críticas que están involucradas en el proceso de la gestión actual de los activos tecnológicos de una organización del mercado, a fin de que resulten efectivas y ajusta a las necesidades empresariales o institucionales.

2. Implementar un modelo predictivo que integre las variables identificadas en el primer objetivo específico utilizando aprendizaje automático para el manejo de activos tecnológicos.

3. Evaluar las fortalezas y debilidades del modelo en función de las métricas referenciadas para la valoración de los modelos de aprendizaje y que son comúnmente aceptadas en estos procesos para definir la efectividad de la herramienta.

## 7. Viabilidad

La viabilidad del presente proyecto se sustenta en la posibilidad real de desarrollar e implementar un modelo predictivo basado en técnicas de machine learning, orientado a determinar si los activos tecnológicos de una organización requieren o no mantenimiento, o por el contrario inició su periodo de obsolescencia y se debe dar de baja. Este enfoque permite anticipar posibles fallas o deterioros, reduciendo los tiempos de indisponibilidad, optimizando los recursos y fortaleciendo la eficiencia operativa.

En términos técnicos, el proyecto se considera factible gracias a la disponibilidad de herramientas digitales de acceso libre y bajo costo que facilitan el procesamiento, análisis y modelamiento de los datos. Entre las principales se encuentran Python, Google Colab y Jupyter Notebook, pero para el desarrollo de esta investigación se diseñó en el servicio en la nube de Google Colab, estas plataformas son acompañadas de librerías especializadas como scikit-learn, pandas, numpy y matplotlib, ampliamente utilizadas en entornos académicos u organizacionales. La infraestructura necesaria incluye un equipo de cómputo con un procesador de al menos ocho núcleos, memoria RAM mínima de 16 GB y una conexión estable a internet de 100 Mbps, especificaciones comunes en las áreas de tecnología de cualquier organización del mercado.

El desarrollo del modelo implica la recolección y análisis de los datos históricos de los activos tecnológicos, lo cual permite identificar patrones de comportamiento relacionados con el desgaste, la frecuencia de fallas y el tipo de mantenimiento realizado. Esta información servirá como base para el entrenamiento del modelo predictivo, garantizando su precisión en la clasificación de los equipos según su estado y la probabilidad de requerir mantenimiento preventivo o correctivo.

La ejecución del proyecto también considera aspectos legales y éticos, especialmente en relación con el tratamiento de la información. Por esta razón, se adoptarán las directrices establecidas en la Ley 1581 de 2012 sobre protección de datos personales, el Decreto 1377 de

2013, y los estándares de la ISO 27001, que promueven la gestión segura de la información. De igual forma, se seguirá la ISO 55001, orientada a la administración responsable de los activos tangibles e intangibles, asegurando que la gestión tecnológica cumpla con criterios de eficiencia, trazabilidad y sostenibilidad.

En cuanto a los recursos humanos, la propuesta requiere la participación de un equipo interdisciplinario conformado por profesionales en áreas de tecnología, ingeniería de sistemas y análisis de datos, con conocimientos en estadística o programación. La colaboración con el área de soporte técnico y la gestión administrativa es fundamental para validar los resultados y garantizar que el modelo se integre adecuadamente con las plataformas de inventario y mantenimiento existentes en la organización.

Desde la perspectiva económica, el proyecto presenta alta viabilidad. El uso de herramientas de código abierto elimina los costos de licenciamiento, mientras que la automatización de los procesos de seguimiento y diagnóstico de los activos tecnológicos contribuye a una reducción significativa en los gastos operativos. A mediano plazo, se espera que el modelo permita disminuir los mantenimientos correctivos, mejorar la disponibilidad de los equipos y optimizar la inversión en infraestructura tecnológica en la organización que sea implementado.

En términos operativos, la implementación del modelo generará beneficios directos como la automatización del control de activos, la detección temprana de posibles fallas y la priorización de intervenciones técnicas según el nivel de criticidad del equipo. Además, al contar con una herramienta predictiva que analice información en tiempo real, se facilitará la toma de decisiones estratégicas y se fortalecerá la cultura organizacional basada en el uso de datos para la mejora continua.

Por último, la aplicabilidad del modelo trasciende el contexto inmediato del proyecto, ya que su diseño modular permite adaptarlo a otros entornos empresariales. De este modo, puede

convertirse en un instrumento útil para diferentes sectores productivos interesados en mejorar la gestión de sus recursos tecnológicos y operativos.

En síntesis, el proyecto presenta viabilidad técnica, económica, operativa y automatizada, al basarse en recursos accesibles, metodologías comprobadas y un enfoque innovador orientado a la optimización de activos. Su implementación no solo contribuye al fortalecimiento de los procesos tecnológicos dentro de la organización, sino que también promueve la sostenibilidad, la eficiencia y la toma de decisiones inteligentes apoyadas en el análisis predictivo.

# 8. Metodología

## 8.1 Etapa teórica

Este caso de estudio adopta un enfoque cuantitativo de tipo aplicado, orientado a resolver una problemática recurrente en la gestión organizacional, la priorización eficiente de activos tecnológicos e informativos en contextos de alta heterogeneidad y dispersión de señales, en este caso la entidad objeto de estudio presenta una cartera diversificada de recursos compuesta por documentos, sistemas y equipos físicos, cada uno con diferentes atributos de criticidad, confidencialidad, disponibilidad, riesgo y amortización en casos concretos.

Ante esta complejidad, se plantea como objetivo metodológico construir un modelo predictivo que permita clasificar de forma dicotómica cada activo en las categorías para facilitar la asignación estratégica de recursos y la introducción de controles específicos; el diseño de la metodología se fundamenta en técnicas de aprendizaje automático supervisado, específicamente regresión logística y redes neuronales básicas aplicadas sobre un conjunto de datos estructurados provenientes de la base pública de “activos de información” de Corpoboyacá.

La muestra utilizada consta de 1.439 registros los cuales fueron sometidos a un proceso riguroso de preparación para imputación de valores faltantes, escalado de variables y división estratégica en conjuntos de entrenamiento y prueba, adicionalmente se implementaron controles explícitos contra fugas de información para garantizar la validez del modelo y evitar sobreajuste.

El análisis se estructuró en tres unidades de caso (CU01, CU02 Y CU03), cada una con objetivos específicos. En CU01 se evalúa la capacidad del modelo para anticipar la etiqueta de sensibilidad sin recurrir a variables que la codifican directamente, se mantuvo una arquitectura con enfoque aplicado constante centrado en la evaluación de métricas F1 y el área bajo la curva Precision-Recall, dada la naturaleza desbalanceada del problema, los resultados mostraron una señal robusta y estable, con desempeño superior a la línea base determinada por la prevalencia

y una matriz de confusión que preserve los verdaderos positivos sin incrementar falsos positivos de forma significativa.

CU02 abordó la validación de una práctica operativa común, tratar activos con nivel de confidencialidad mayor o igual a tres con categoría diferenciada, se definió un objetivo binario con ese umbral y se excluyeron variables que replicarán la decisión, en este caso, el modelo mostró curvas ROC y Precision-Recall altas y estables, confirmando que una regla explícita o un clasificador ligero puede alcanzar el mismo desempeño, lo que refuerza la trazabilidad y comparabilidad entre unidades de análisis.

CU03 se centró en establecer un criterio de priorización que garantice un alto recall ( $\geq 0.80$ ), sin excluir casos relevantes, se ajustó el umbral operativo a 0.56 y se corroboró el comportamiento del modelo mediante coeficientes, curvas y matriz de confusión, las variables de criticidad, confidencialidad e índice de riesgo mostraron una contribución significativa a la señal positiva, mientras que otros atributos ajustaron el balance sin dominar el resultado.

Complementariamente, se exploró un modelo de red neuronal básica para evaluar la separabilidad del problema, las curvas de validación mostraron que la decisión es simple de predecir con las variables disponibles y la variable de confusión confirma la ausencia de errores. Esto sugiere que un clasificador sencillo o una regla operativa explícita puede alcanzar el mismo desempeño sin pérdida de calidad, no obstante, se conservó la red neuronal por motivos de comparabilidad en el esquema operativo, trazabilidad del proceso y utilidad futura en escenarios de mayor complejidad.

Para este caso, la metodología empleada permite transformar señales heterogéneas en decisiones operativas claras, mediante un flujo reproducible, auditable y centrado en métricas relevantes, este enfoque no solo responde a las necesidades del caso de estudio, sino que ofrece una solución eficiente, escalable y adaptable a otros contextos organizacionales que enfrentan retos similares en la gestión de activos tecnológicos.

## 8.2 Etapa de implementación

La metodología de esta investigación adopta un enfoque de métodos cuantitativos basado en el paradigma de CRISP-DM, éste adaptado a la gestión de activos tecnológicos, esta metodología consta de etapas que permiten elaborar los pasos para llegar a la resolución de la necesidad previamente planteada.

**Tabla 2:** Metodología CRISP-DM

<b>Objetivo específico</b>	<b>Actividad</b>	<b>Fase CRISP-DM</b>
<ul style="list-style-type: none"> <li>• Caracterizar las variables críticas que están involucradas en el proceso de la gestión actual de los activos tecnológicos de Corpoboyacá a fin de que resulten efectivas y determinar qué activos requieren atención inmediata o priorización.</li> </ul>	<ul style="list-style-type: none"> <li>• Identificar las necesidades específicas de Corpoboyacá, en relación con su inventario tecnológico que maneja el área de Gestión TIC, para evaluar la metodología que favorezca el ejercicio y que logre hacer una predicción de fallos, gestión de mantenimiento o ciclos de vida de los equipos o actualizaciones que los mismos requieran,</li> </ul>	<ul style="list-style-type: none"> <li>• CRISP-DM Fase 1 - Entendimiento del negocio</li> <li>• CRISP-DM Fase 2 - Entendimiento de los datos</li> </ul>

	<ul style="list-style-type: none"> <li>Recolectar los datos relevantes sobre el inventario e información de hardware y software, aplicando criterios de criticidad, riesgo y disponibilidad para etiquetar los activos como prioritarios o no a un análisis exploratorio preliminar donde se analicen posibles patrones, anomalías y tendencias en los datos que pueden influir en el modelo.</li> </ul>	
<ul style="list-style-type: none"> <li>Preparar los datos para entrenamiento de los modelos predictivos.</li> </ul>	<ul style="list-style-type: none"> <li>La etapa de la preparación de los datos, primero imputar valores faltantes, dividir el conjunto de datos y aplicar controles contra fuga de información, se nivelan para asegurar la calidad de los datos y poder pasar a estandarizar, escalar y codificar variables como confidencialidad, integridad y nivel de amortización.</li> </ul>	<ul style="list-style-type: none"> <li>CRISP-DM Fase 3 - Preparación de los datos</li> </ul>
<ul style="list-style-type: none"> <li>Implementar un modelo predictivo que integre las variables identificadas en el primer objetivo específico</li> </ul>	<ul style="list-style-type: none"> <li>Usando modelo de regresión logística, se ajustan hiper parámetros, se plantean CU01, CU02 Y CU03, aplicar validación cruzada y evaluar</li> </ul>	<ul style="list-style-type: none"> <li>CRISP-DM Fase 4 - Modelado</li> </ul>

<p>utilizando aprendizaje automático para el manejo de activos tecnológicos.</p>	<p>con métricas como F1 y área bajo curva PR, adicional, implementar red neuronal, revisar curvas ROC y PR, y determinar si aporta mejoras significativas.</p>	
<ul style="list-style-type: none"> <li>• Evaluar las fortalezas y debilidades del modelo en función de las métricas referenciadas para la valoración de los modelos de aprendizaje y que son comúnmente aceptadas en estos procesos para definir la efectividad de la herramienta.</li> </ul>	<ul style="list-style-type: none"> <li>• Acorde al análisis de las matrices de confusión, revisar falsos positivos y negativos, y ajustar el umbral operativo, se consolidan los pasos, las decisiones y métricas es un esquema reproducible para futuras adaptaciones.</li> </ul>	<ul style="list-style-type: none"> <li>• CRISP-DM Fase 5- Evaluación</li> </ul>

## 9. Resultados

En esta parte del trabajo de grado, se describe el desarrollo del análisis para alcanzar el objetivo general y los objetivos específicos establecidos previamente en el documento. Como punto inicial, se empieza con el entendimiento del negocio y las necesidades que tiene la organización y la comprensión de los datos para realizar un plan de desarrollo con las variables críticas, después de definir e identificar en la base de datos las columnas y registros, se procede a la etapa de preparación de datos con el preprocesamiento de la información suministrada con las características y variables para la construcción de los modelos.

**Objetivo General: Desarrollar un modelo de gestión de inventarios tecnológicos para aplicar en organizaciones del mercado, mediante el uso de herramientas predictivas basadas en análisis de datos.**

**Tabla 3:** Preprocesamiento de escalado y normalización de variables

Acciones	Código
Se generó la importación de las librerías usadas para el caso de estudio	<pre> from google.colab import files uploaded = files.upload() xlsx_path = next(iter(uploaded)) import pandas as pd, numpy as np, unicodedata, re, csv from rapidfuzz import fuzz                     </pre>
Define columnas en un diccionario con claves cortas	<pre> COLS= {     "id": "ID",     "fecha_registro": "FECHA DE REGISTRO",     "identificador": "IDENTIFICADOR",     "sede": "SEDE DEL ACTIVO",     "area": "AREA / DEPENDENCIA",     "proceso": "PROCESO",     "tipo_activo": "TIPO ACTIVO",     "nombre_activo": "NOMBRE DEL ACTIVO",                     </pre>

	<pre> "serie_doc": "SERIE DOCUMENTAL", "subserie_doc": "SUBSERIE DOCUMETNAL", "descripcion": "DESCRIPCION DEL ACTIVO", "ubicacion": "UBICACIÓN DEL ACTIVO", "resp_prop": "RESPONSABLE DEL ACTIVO (Propietario del activo)", "resp_cust": "RESPONSABLE DEL ACTIVO (Custodio del activo)", "idioma": "IDIOMA", "medio_cons": "MEDIO DE CONSERVACION", "formato": "FORMATO", "info_publicada": "INFORMACION PUBLICADA O DISPONIBLE", "enlace_datos_gov": "ENLACE A DATOS.GOV.CO", "clas_pd": "CLASIFICACION RESPECTO DE PROTECCION DE DATOS PERSONALES", "clas_conf": "CLASIFICACION RESPECTO A CONFIDENCIALIDAD", "clas_int": "CLASIFICACION RESPECTO A INTEGRIDAD", "clas_disp": "CLASIFICACION RESPECTO A DISPONIBILIDAD", "criticidad": "CRITICIDAD", } </pre>
<p>Inicia normalización de variables, se eliminan tildes y acentos dentro de los datos para depurar</p>	<pre> def strip_accents(s: str) -&gt; str:     if s is None or (isinstance(s, float) and pd.isna(s)):         return ""     s = str(s)     #quita tildes     s = "".join(c for c in unicodedata.normalize("NFD", s) if unicodedata.category(c) != "Mn")     return s </pre>
<p>Crea función de normalización, quita espacios entre registros para limpiar en otra capa</p>	<pre> def norm(s: str) -&gt; str:     s = "" if pd.isna(s) else str(s)     s = " ".join(s.split())     s = strip_accents(s).lower() </pre>

	<pre> return s def norm_drop_code(s: str) -&gt; str:     t = norm(s)     return re.sub(r"\s*([a-z0-9]+\s*\$)", "", t) </pre>
<p>Transforma los valores de texto de “alta, media, baja” en valores numéricos para ordinalidad</p>	<pre> def map_alta_media_baja(v):     t = norm_drop_code(v)     if t.startswith("alta"):         return 3     if t.startswith("media") or t.startswith("medio"):         return 2     if t.startswith("baja"):         return 1     n = pd.to_numeric(v, errors="coerce")     if pd.isna(n):         return int(np.clip(n, 1, 3))     return np.nan </pre>
<p>Lee el excel y clasifica los activos acorde a si son tangibles o intangibles, información, servicio y recursos humanos como categorías diferenciada</p>	<pre> df = pd.read_excel(xlsx_path) df["tipo_activo"] = df["tipo_activo"].astype(str).str.strip().str.upper() no_tangible = {"SERVICIOS", "INFORMACION", "RECURSO_HUMANO", "SOFTWARE"} df["TANGIBILIDAD"] = np.where(df["tipo_activo"].isin(no_tangible), "no tangible", "tangible") </pre>
<p>Transforma la columna de “fecha_registro” en formato datetime</p>	<pre> df[COLS["fecha_registro"]] = pd.to_datetime(     df[COLS["fecha_registro"]], errors="coerce", dayfirst=True ) </pre>
<p>Separa por sensibilidad de datos personales: normaliza los valores de la columna, define como “no sensibles” los casos “CONTIENE DATOS PÚBLICOS” y “NO CONTIENE DATOS PERSONALES”, y crea la columna PD_SENSIBILIDAD asignando “no sensible” a esos casos y “sensible” al</p>	<pre> pd_val = df[COLS["clas_pd"]].apply(norm) no_sens = {norm("CONTIENE DATOS PÚBLICOS"), norm("NO CONTIENE DATOS PERSONALES")} df["PD_SENSIBILIDAD"] = np.where(pd_val.isin(no_sens), "no sensible", "sensible") </pre>

resto.	
Normaliza la clasificación de confidencialidad (eliminando sufijos como "(A)/(B)/(M)", detecta el valor "no clasificada" y lo reemplaza por "Información publica sin datos sensibles" en la misma columna	<pre> conf_norm = df[COLS["clas_conf"]].apply(norm_drop_code) mask_nc = conf_norm.eq("no clasificada") df.loc[mask_nc, COLS["clas_conf"]] = "Información publica sin datos sensibles" </pre>
Asigna ordinalidad a la confidencialidad: normaliza el texto (quitando sufijos como "(A)/(B)/(M)") y devuelve 3 para "informacion publica clasificada", 1 para "informacion publica sin datos sensibles", 2 para "informacion publica", y NaN si no coincide con ninguno.	<pre> def map_conf_ord(v: str):     t = norm_drop_code(v)     if t.startswith("informacion publica clasificada"):         return 3     if t.startswith("informacion publica sin datos sensibles"):         return 1     if t.startswith("informacion publica"):         return 2     return np.nan </pre>
Reemplaza la columna de confidencialidad aplicando la ordinalidad definida, convierte INTEGRIDAD y DISPONIBILIDAD a valores numéricos con map_alta_media_baja (BAJA=1, MEDIA=2, ALTA=3), para finalizar, se transforma CRITICIDAD también para calcular	<pre> df[COLS["clas_conf"]] = df[COLS["clas_conf"]].apply(map_conf_ord) df[COLS["clas_int"]] = df[COLS["clas_int"]].apply(map_alta_media_baja) df[COLS["clas_disp"]] = df[COLS["clas_disp"]].apply(map_alta_media_baja) df[COLS["criticidad"]] = df[COLS["criticidad"]].apply(map_alta_media_baja) </pre>
Calcula el INDICE DE RIESGO sumando las ordinalidades de confidencialidad, integridad y disponibilidad (sustituyendo nulos por 0) y multiplica por criticidad; almacena el resultado en INDICE DE RIESGO	<pre> df["INDICE DE RIESGO"] = (     df[COLS["clas_conf"]].fillna(0)     + df[COLS["clas_int"]].fillna(0)     + df[COLS["clas_disp"]].fillna(0) ) * df[COLS["criticidad"]].fillna(0) </pre>
Mapeo de tipo de activo en clave corta	<pre> def map_primary_tipo(v: str) -&gt; str:     t = normalize_label(v)     if "COMPONENTES" in t and "RED" in t: return "RED" </pre>

	<pre> if t == "HARDWARE": return "HARDWARE" if t == "SOFTWARE": return "SOFTWARE" if t == "SERVICIOS": return "SERVICIOS" if t == "INFORMACION": return "INFORMACION" if t == "INSTALACIONES": return "INSTALACIONES" if t == "RECURSO_HUMANO": return "RECURSO_HUMANO" return t or "OTROS" </pre>
<p>Mapeo del nombre del activo acorde a la clase a la que pertenece</p>	<pre> def map_subtipo_from_nombre(v: str) -&gt; str   None:     n = normalize_label(v)     if "FIREWALL" in n or "CORTAFUEGOS" in n: return "FIREWALL"     if "ROUTER" in n: return "ROUTER"     if "SWITCH" in n:         if "CORE" in n: return "SWITCH_CORE"         if "BORDE" in n: return "SWITCH_BORDE"         if "SAN" in n: return "SWITCH_SAN"         return "SWITCH"     #HARDWARE / OTROS     if "COMPUTADOR PORTATIL" in n or "PORTATIL" in n: return "PORTATIL"     if re.search(r"\bCOMPUTADOR\b", n): return "CPU"     if "IMPRESORA" in n: return "IMPRESORA"     if "ESCANER" in n or "SCANNER" in n: return "ESCANER"     if "SERVIDOR" in n: return "SERVIDOR"     if "VIDEO BEAM" in n or "VIDEOBEAM" in n: return "VIDEO_BEAM"     if "WORKSTATION" in n: return "WORKSTATION"     if "ALMACENAMIENTO" in n and "MSA" in n: return "ALMACENAMIENTO_MSA"     #SOFTWARE     if "SISTEMA DE INFORMACION" in n: return "SISTEMA_INFO"     return None </pre>

<p>Analiza el subtipo a partir de la descripción cuando el nombre no basta. Inserta espacios de vigilancia (centinela) para coincidencias exactas y busca palabras clave con regex: portatil= PORTATIL, escaner= ESCANER, (impresora printer)=IMPRESORA, (all in one aio)=AIO, (monitor pantalla)=MONITOR, (xeon core i[3579] cpu torre desktop)=CPU, (switch routero access point ap firewall)=RED_MISC, en caso de no reconocer ningún patrón, esta devuelve None para dejar el subtipo sin definir.</p>	<pre>def map_subtipo_from_desc(t: str) -&gt; str   None:     t = " " + t + " "     if re.search(r"\bportatil\b", t): return "PORTATIL"     if re.search(r"\bescaner\b", t): return "ESCANER"     if re.search(r"\b(impresora printer)\b", t): return "IMPRESORA"     if re.search(r"\b(all\s*in\s*one aio)\b", t): return "AIO"     if re.search(r"\b(monitor pantalla)\b", t): return "MONITOR"     if re.search(r"\b(xeon core i[3579] cpu torre desktop)\b", t): return "CPU"     if re.search(r"\b(switch routero access\s*point ap firewall)\b", t): return "RED_MISC"     return None</pre>
<p>Ejecuta las funciones de normalización. Crea DESC_NORM normalizando la descripción. Mapea TIPO ACTIVO a clave corta en primary. Mapea el subtipo desde NOMBRE DEL ACTIVO en sub1. Si no lo obtiene con el nombre, consigue el subtipo desde la descripción normalizada en sub2</p>	<pre>df["DESC_NORM"] = df[COLS["descripcion"]].astype(str).apply(normalize_desc) primary = df[COLS["tipo_activo"]].apply(map_primary_tipo) sub1 = df[COLS["nombre_activo"]].apply(map_subtipo_from_nombre) sub2 = df["DESC_NORM"].apply(map_subtipo_from_desc)</pre>
<p>Construye la clave de bloqueo como TIPO SUBTIPO (usa ? si falta información). Fija UMBRAL=85 y, por cada BLOQUE_TIPO, forma clústeres con similitud token_set_ratio. Entonces compara la descripción del activo y si se asemeja al menos 85% a la descripción de otro activo, lo considera lo mismo, este asigna el primer ítem como representante y agrega los</p>	<pre>df["BLOQUE_TIPO"] = [ f"{p}{{{s1 or s2 or '?'}}}" for p, s1, s2 in zip(primary, sub1, sub2) ]  df["TIPO ACTIVO"] = df["TIPO ACTIVO"].str.strip().str.upper() df["BLOQUE_TIPO"] = df["BLOQUE_TIPO"].astype(str)  def forzar_tipo_activo_compuesto(bloque_compuesto: str, tipo_activo: str) -&gt; str:     if " " in bloque_compuesto:</pre>

demás si superan el umbral y si no, crea un nuevo clúster, también escribe CLUSTER\_ID y REPRESENTANTE\_CLUSTER con los resultados.

```
primary, sub = bloque_compuesto.split("|", 1)
else:
    primary, sub = bloque_compuesto, "?"
ta = (tipo_activo or "").strip().upper()
if ta in {"HARDWARE", "SOFTWARE"}:
    primary = ta
    return f"{primary}{{sub}}"
df["BLOQUE_TIPO"] = df.apply(
    lambda r:
forzar_tipo_activo_compuesto(r["BLOQUE_TIPO"], r["TIPO
ACTIVO"]), axis=1
)
df["BLOQUE_TIPO"] =
df["BLOQUE_TIPO"].str.strip().str.upper()
UMBRAL = 85
cluster_ids = np.full(len(df), -1, dtype=int)
rep_map = {}
next_cid = 0
for bloque, idxs in
df.groupby("BLOQUE_TIPO").groups.items():
    idxs = list(idxs)
    reps = []
    for i in idxs:
        s = df.at[i, "DESC_NORM"]
        if not reps:
            reps.append((next_cid, s))
            rep_map[next_cid] = s
            cluster_ids[i] = next_cid
            next_cid += 1
            continue

scores = [fuzz.token_set_ratio(s, r) for _, r in reps]
j = int(np.argmax(scores))
if scores[j] >= UMBRAL:
    cid = reps[j][0]
    cluster_ids[i] = cid
else:
```

	<pre> reps.append((next_cid, s)) rep_map[next_cid] = s cluster_ids[i] = next_cid next_cid += 1  df["CLUSTER_ID"] = cluster_ids df["REPRESENTANTE_CLUSTER"] = df["CLUSTER_ID"].map(rep_map) df["BLOQUE_TIPO"] = df["BLOQUE_TIPO"].astype(str).str.strip().str.upper() df["REPRESENTANTE_CLUSTER"] = df["REPRESENTANTE_CLUSTER"].astype(str).str.strip().str.u pper() </pre>
<p>Determina la vida útil por subtipo</p>	<pre> LIFE_YEARS = {     "PORTATIL": 4,     "CPU": 5,     "AIO": 4,     "MONITOR": 6,     "IMPRESORA": 5,     "ESCANER": 5,     "SERVIDOR": 5,     "WORKSTATION": 5,     "VIDEO_BEAM": 5,     "FIREWALL": 5,     "ROUTER": 5,     "SWITCH": 6,     "SWITCH_CORE": 7,     "SWITCH_BORDE": 6,     "SWITCH_SAN": 7,     "RED_MISC": 5,     "ALMACENAMIENTO_MSA": 6,     #"SISTEMA_INFO": 3, #A evaluar }  #uso del subtipo df["VIDA_UTIL"] = </pre>

	<code>df["_SUBTIPO"].map(LIFE_YEARS).astype("float")</code>
Calcula la tasa de depreciación restando costo - valor residual, divide por vida útil y obtiene la tasa de depreciación anual, se evitan las divisiones por cero con valid (solo filas con vida útil mayor a 0), a su vez, inicia en 0 y asigna el resultado, este redondea a múltiplos con STEP (10.000 COP) y guarda como un entero	<pre> valid = df["VIDA_UTIL"].fillna(0) &gt; 0 raw_dep = (df["COSTO DEL ACTIVO"] - df["VALOR RESIDUAL"]) / df["VIDA_UTIL"] df["TASA DE DEPRECIACION"] = 0 df.loc[valid, "TASA DE DEPRECIACION"] = (     np.round(raw_dep[valid] / STEP) * STEP ).astype(int) </pre>
Activa la sanitización de saltos, esto significa que si SANITIZAR_SALTOS es True, detecta las columnas de texto y reemplaza saltos por un espacio para que no haya saltos de línea y no se dañe el csv	<pre> SANITIZAR_SALTOS = True import csv  if SANITIZAR_SALTOS:     text_cols = df.select_dtypes(include="object").columns     df[text_cols] = df[text_cols].replace({r"[\r\n\t]+" : " "}, regex=True) </pre>
Instala librerías necesarias para análisis y visualización	<pre> !pip -q install scikit-learn matplotlib  from sklearn.preprocessing import StandardScaler from sklearn.cluster import KMeans from sklearn.metrics import silhouette_score import matplotlib.pyplot as plt </pre>
Se definen columnas numéricas para analizarlas	<pre> num_cols = [     COLS["clas_conf"],     COLS["clas_int"],     COLS["clas_disp"],     COLS["criticidad"],     "antiguedad",     "INDICE DE RIESGO",     "VIDA_UTIL",     "TASA DE DEPRECIACION", ] </pre>

	<code>X = df_tec[num_cols].fillna(0)</code>
Implementamos un scaler para transformar las variables para que tengan media 0 y desviación estándar 1, esto para facilitar el análisis si hay valores demasiado diferentes (ej: antigüedad +100 días vs criticidad de 1 a 3)	<code>scaler = StandardScaler() X_scaled = scaler.fit_transform(X)</code>
Se lleva a cabo el uso del método del codo: Las inercias miden qué tan cerca están los puntos del centroide en cada cluster, menor inercia = más cerca del centroide	<code>inertias = [] K_range = range(1, 11) for k in K_range:     km = KMeans(n_clusters=k, random_state=42, n_init=10)     km.fit(X_scaled)     inertias.append(km.inertia_) plt.figure(figsize=(6,4)) plt.plot(K_range, inertias, "o-") plt.xlabel("Número de clusters (k)") plt.ylabel("Inercia (SSE)") plt.title("Método del codo") plt.show()</code>
Implementación de método de la silueta= entre más cercano sea a 1, los clústeres están mejor divididos unos de otros	<code>sil_scores = [] for k in range(2, 11):     km = KMeans(n_clusters=k, random_state=42, n_init=10)     labels = km.fit_predict(X_scaled)     sil = silhouette_score(X_scaled, labels)     sil_scores.append((k, sil)) plt.figure(figsize=(6,4)) plt.plot([k for k,_ in sil_scores], [s for _,s in sil_scores], "o-") plt.xlabel("Número de clusters (k)") plt.ylabel("Silhouette score") plt.title("Método de silueta") plt.show()</code>
Entrenamiento del modelo	<code>k_optimo = 3</code>

final con el K óptimo	<pre> km_final = KMeans(n_clusters=k_optimo, random_state=42, n_init=10) df_tec["CLUSTER_ANALISIS"] = km_final.fit_predict(X_scaled) </pre>
Se realiza el plano de PCA	<pre> pca = PCA(n_components=2, random_state=42) X_pca = pca.fit_transform(X_scaled) df_tec["PCA1"] = X_pca[:,0] df_tec["PCA2"] = X_pca[:,1] plt.figure(figsize=(8,6)) scatter = plt.scatter(df_tec["PCA1"], df_tec["PCA2"],c=df_tec["CLUSTER_ANALISIS"], cmap="tab10", alpha=0.7) plt.xlabel("PCA 1") plt.ylabel("PCA 2") plt.title(f"Visualización PCA de clusters (k={k_optimo})") plt.legend(*scatter.legend_elements(), title="Cluster") plt.show() </pre>
Graficar los Clusters	<pre> variables_resumen = [ COLS["clas_conf"], COLS["clas_int"], COLS["clas_disp"], COLS["criticidad"], "antiguedad", "INDICE DE RIESGO", "VIDA_UTIL", "TASA DE DEPRECIACION", "COSTO DEL ACTIVO", "VALOR RESIDUAL" ] resumen = df_tec.groupby("CLUSTER_ANALISIS")[variables_resumen].a gg( ["mean", "min", "max", "count"] ).round(2) pd.set_option("display.max_columns", None) </pre>

	display(resumen)
--	------------------

Consecuentemente en la fase de modelado se realiza la normalización, escalado y preparación requerida para implementar los modelos de regresión logística y redes neuronales para determinar el método más eficiente para la clasificación de los activos tecnológicos, finalmente para la fase de evaluación, en concordancia con los resultados de los casos de uso, matrices de confusión y ajustes de métricas en el paso previo, se determina que el modelo puede realizar la toma de decisiones estratégicas para tener un esquema aplicable a los requerimientos del dataset de Corpoboyacá.

Desde el preprocesamiento se establece una ruta clara y repetible que inicia con la depuración del inventario maestro y termina en la generación de insumos listos para modelos.

**Caracterizar las variables críticas que están involucradas en el proceso de la gestión actual de los activos tecnológicos de una organización del mercado, a fin de que resulten efectivas y ajustarse a las necesidades empresariales o institucionales.**

Primero se aplicó una normalización semántica del texto (limpieza de espacios, unificación de mayúsculas/minúsculas y eliminación de tildes) sobre campos claves de identificación del activo; con ello fue posible homogeneizar etiquetas heterogéneas y reducir ruido típico de digitación, se tipifican los activos en una clave compuesta de dos niveles de una cabecera primaria (hardware, software, red, información, servicios, etc) y un subtipo derivado del nombre técnico con la finalidad de construir el campo operativo de BLOQUE\_TIPO="primary/subtipo.

Esta taxonomía permite formar coherencia cuando el "tipo declarado" del activo estaba incompleto (la cabecera se impone cuando el registro lo exige) y para orientar el análisis hacia familias funcionales comparables. Con la estructura consolidada, se ejecutó una detección de duplicados y representaciones equivalentes basada en similitud de cadenas (tokenización y

comparación difusa con un umbral alto de aceptación), asignando un CLUSTER\_ID a cada grupo y un representante de cluster legible, el resultado fue un catalogo operativo con ejemplos por familias y conteos por grupo eficaz para una inspección manual, trazabilidad y soporte a precios o valuaciones.

### Ilustración 1: Variables de resumen de cluster completa

```

pd.set_option("display.max_columns", None) #muestra todas las columnas en la salida
display(resumen) #tabla resumen en notebook

```

	CLASIFICACION RESPECTO A CONFIDENCIALIDAD				CLASIFICACION RESPECTO A INTEGRIDAD				CLASIFICACION RESPECTO A DISPONIBILIDAD				CRITICIDAD				antigüedad				INDICE DE RIESGO				VIDA_UTIL				TASA DE DEPRECIACION				COSTO DEL ACTIVO				VALOR RESIDUAL			
	mean	min	max	count	mean	min	max	count	mean	min	max	count	mean	min	max	count	mean	min	max	count	mean	min	max	count	mean	min	max	count	mean	min	max	count	mean	min	max	count	mean	min	max	count
CLUSTER_ANALISIS																																								
0	1.87	1	2	177	1.00	1	1	177	1.00	1	1	177	1.00	1	1	177	401.8	105	442	177	3.87	3	4	177	5.00	5.0	5.0	177	99322.03	80000	800000	177	1117514.12	840000	8500000	177	620508.47	350000	5540000	177
1	3.00	3	3	770	1.00	1	1	770	1.00	1	1	770	2.00	2	2	770	442.0	442	442	770	10.00	10	10	770	4.91	4.0	7.0	770	168025.97	80000	3400000	770	2615025.97	1450000	50980000	770	1775285.71	940000	30580000	770
2	3.00	3	3	26	2.23	2	3	26	2.23	2	3	26	2.23	2	3	26	442.0	442	442	26	17.00	14	27	26	5.70	5.0	7.0	26	206153.85	0	610000	26	3320769.23	0	9930000	26	2156923.08	0	6450000	26

### Ilustración 2: Variables de resumen de cluster con imágenes ampliadas.

```

pd.set_option("display.max_columns", None) #muestra todas las columnas en la salida
display(resumen) #tabla resumen en notebook

```

	CLASIFICACION RESPECTO A CONFIDENCIALIDAD				CLASIFICACION RESPECTO A INTEGRIDAD				CLASIFICACION RESPECTO A DISPONIBILIDAD				CRITICIDAD				antigüedad				INDICE DE RIESGO							
	mean	min	max	count	mean	min	max	count	mean	min	max	count	mean	min	max	count	mean	min	max	count	mean	min	max	count	mean	min	max	count
CLUSTER_ANALISIS																												
0	1.87	1	2	177	1.00	1	1	177	1.00	1	1	177	1.00	1	1	177	401.8	105	442	177	3.87	3	4	177				
1	3.00	3	3	770	1.00	1	1	770	1.00	1	1	770	2.00	2	2	770	442.0	442	442	770	10.00	10	10	770				
2	3.00	3	3	26	2.23	2	3	26	2.23	2	3	26	2.23	2	3	26	442.0	442	442	26	17.00	14	27	26				

INDICE DE RIESGO				VIDA_UTIL				TASA DE DEPRECIACION				COSTO DEL ACTIVO				VALOR RESIDUAL			
mean	min	max	count	mean	min	max	count	mean	min	max	count	mean	min	max	count	mean	min	max	count
3.87	3	4	177	5.00	5.0	5.0	177	99322.03	80000	800000	177	1117514.12	840000	8520000	177	620508.47	350000	5540000	177
10.00	10	10	770	4.91	4.0	7.0	770	168025.97	80000	3400000	770	2615025.97	1450000	50980000	770	1775285.71	940000	30580000	770
17.00	14	27	26	5.70	5.0	7.0	26	206153.85	0	610000	26	3320769.23	0	9930000	26	2156923.08	0	6450000	26

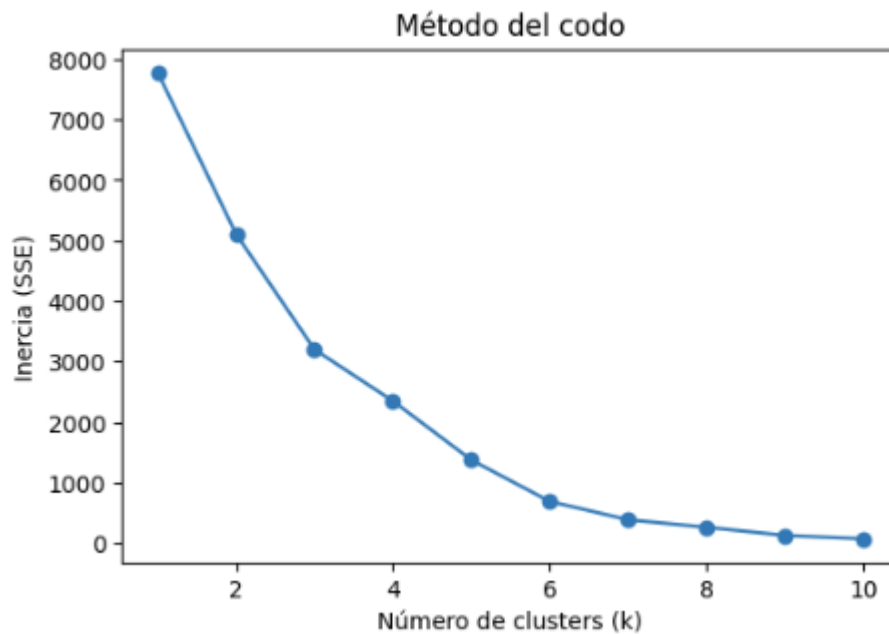
En resumen, el número óptimo de clústeres es 3, lo cual nos permite ver que lo que aparece es bastante intuitivo. (0) activos de baja C-I-D y criticidad=1, con costos contenidos, (1) confidencialidad fija en 3 y criticidad=2, sube el índice de riesgo y valores, (2) también aparece confidencialidad=3, pero con I/D más variables y criticidad y riesgo altos.

A partir de ello, se definieron tres preguntas que estructuran los casos de uso, CU01 comprueba si la sensibilidad puede anticiparse con variables operativas, sin fugas. CU02 formaliza la regla de confidencialidad 3, CU03 fija un criterio de priorización con recall controlado cuando criticidad+riesgo se disparan.

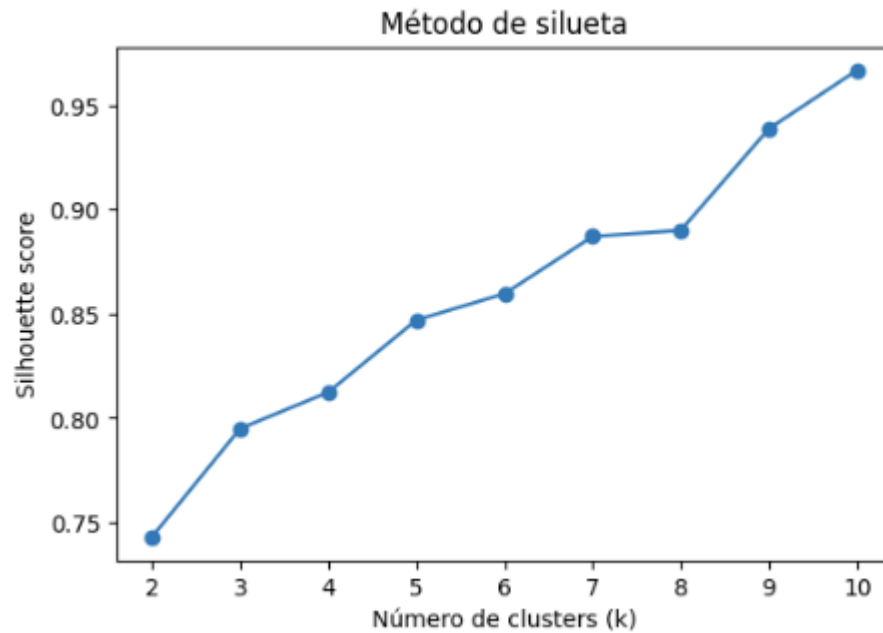
**Implementar un modelo predictivo que integre las variables identificadas en el primer objetivo específico utilizando aprendizaje automático para el manejo de activos tecnológicos.**

Para comprender la forma del inventario antes de formular hipótesis predictivas, se realizó una agrupación exploratoria sobre variables numéricas estandarizadas. Se compararon alternativas de k mediante el método del codo (inercia) y el método silueta (separación interna/externa), se estabilizó una partición concisa de tres segmentos que mostró patrones claros de un bloque C-I-D bajas y criticidad reducida, otro con confidencialidad alta y costos más elevados y un tercero intermedio con variación en integridad/disponibilidad y picos de criticidad y riesgo.

**Ilustración 3:** Método del codo



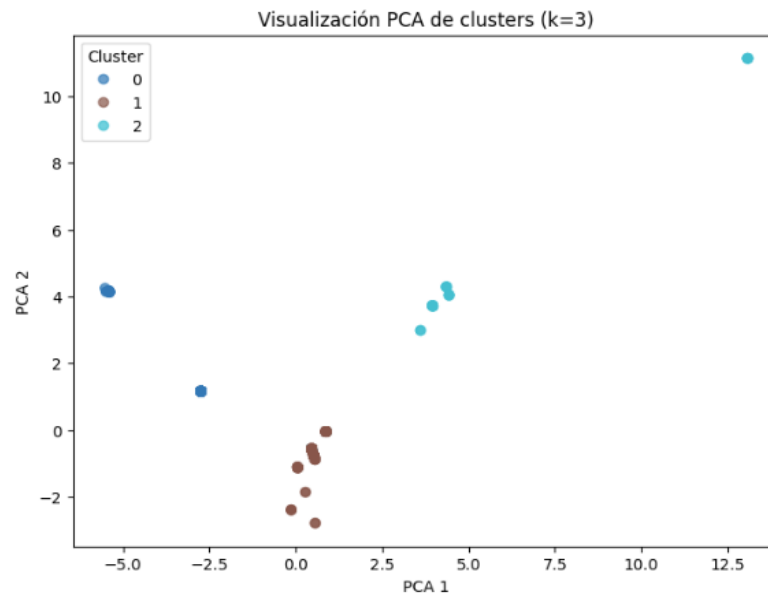
**Ilustración 4:** Método de silueta



Una proyección PCA (2D) corroboró la separabilidad visual de estos grupos, por tanto, a partir de esta necesidad de llevar estos patrones a decisiones operativas, se definieron tres casos de uso:

1. Anticipar sensibilidad de datos personales sin usar columnas que codifiquen la respuesta (CU01).
2. Confirmar y operativizar la regla de confidencialidad  $\geq 3$  (CU02).
3. Priorizar reemplazos en activos tangibles según la vida putin residual (CU03).

**Ilustración 5:** Graficar clusters



Sobre el análisis elaborado sobre la base de datos Corpoboyacá se obtuvieron tres resultados centrales, uno por cada caso de uso (CU01, CU02 Y CU03) y un resultado transversal con la red neuronal usada como contraste metodológico. El punto de partida fue un flujo único de preparación para garantizar comparabilidad y trazabilidad de limpieza y diagnóstico (NaN y varianza cero), exclusión explícita de variables con “fuga” de información (columnas que replican directa o indirectamente la etiqueta), particiones estratificadas en el entrenamiento, validación y prueba y un pipeline con imputación por mediana y estandarización ajustadas únicamente sobre el entrenamiento.

**Tabla 4:** Regresión Logística

Acciones	Código
Se comienza con la importación de librerías para Regresión Logística	<pre>from sklearn.linear_model import LogisticRegression from sklearn.model_selection import train_test_split from sklearn.metrics import classification_report, confusion_matrix, ConfusionMatrixDisplay</pre>
Variables predictoras de X y Y	<pre>X = df_tec[num_cols].fillna(0) y = (df_tec["PD_SENSIBILIDAD"] == "sensible").astype(int)</pre>
Aplicar el entrenamiento, train test y split test	<pre>X_train, X_test, y_train, y_test = train_test_split(     X_scaled, y, test_size=0.3, random_state=42, stratify=y )</pre>
Se aplica la regresión logística	<pre>log_reg = LogisticRegression(max_iter=1000, random_state=42, class_weight="balanced") log_reg.fit(X_train, y_train)</pre>
Tabla de coeficientes de la regresión logística, nombres de variables, pesos aprendidos ordenados de mayor a menor y gráfica	<pre>coef_df = pd.DataFrame({     "Variable": num_cols,     "Coeficiente": log_reg.coef_[0] }).sort_values(by="Coeficiente", ascending=False) display(coef_df)</pre>

**Ilustración 6:** variables y coeficientes de la regresión logística

	Variable	Coeficiente
3	CRITICIDAD	1.781699
0	CLASIFICACION RESPECTO A CONFIDENCIALIDAD	1.542500
5	INDICE DE RIESGO	1.146018
7	TASA DE DEPRECIACION	0.556808
4	antiguedad	0.018080
1	CLASIFICACION RESPECTO A INTEGRIDAD	-1.168744
2	CLASIFICACION RESPECTO A DISPONIBILIDAD	-1.168744
6	VIDA_UTIL	-3.810762

La evaluación se orientó al desbalance natural del problema: se priorizaron AUC-PR (AP), AUC-ROC, F1 y matrices de confusión, eligiendo umbrales por máximo F1 out-of-foldy y cuando se ameritaba, por función del coste con pisos de precision/recall, una primera exploración mostró tres perfiles operativos coherentes con la práctica (bajas C-I-D y una criticidad reducida; confidencialidad alta, mayor variabilidad en I/D y picos de criticidad y riesgo) y esa estructura guió el alcance de cad CU para anticipar sensibilidad, confirmar la regla de confidencialidad de  $\geq 3$  y priorizar reemplazos con cobertura suficiente.

CU01 (sensibilidad), en este caso el objetivo fue anticipar si un activo trata datos sensibles empleando sólo variables operativas, tras mapear la etiqueta a 0/1 y excluir columnas que la duplican como C-I-D, criticidad o índice de riesgo cuando corresponda, la regresión logística produjo un ordenamiento estable para las curvas precision-recall y ROC superaron holgadamente sus líneas base y el umbral fijado por máximo F1 en validación cruzada mantuvo el equilibrio entre capturar positivos y contener falsos. La interpretabilidad de los coeficientes permitió corroborar el sentido de las señales (variables con efecto positivo o negativo sobre la probabilidad), facilitando auditoría y traslado a operación, con esto, CU01 queda resuelto de forma práctica como un clasificador lineal, reproducible y trazable suficiente para detectar casos sensibles y recalibrable si cambian las distribuciones.

**Tabla 5:** Código Caso de Uso 01

Acciones	Código
Importar librerías	<pre>import numpy as np import pandas as pd import matplotlib.pyplot as plt from sklearn.model_selection import train_test_split, StratifiedKFold from sklearn.pipeline import Pipeline from sklearn.base import clone</pre>

	<pre> from sklearn.impute import SimpleImputer from sklearn.preprocessing import StandardScaler from sklearn.linear_model import LogisticRegression from sklearn.metrics import (     precision_recall_curve, classification_report,     confusion_matrix,     ConfusionMatrixDisplay, roc_auc_score,     average_precision_score ) </pre>
<p>Semilla para reproducibilidad y elige una columna del DF que coincida con alguno de los alias (case-insensitive), si no lo encuentra, retorna None</p>	<pre> SEED = 42 def pick_col(df, aliases):     cols_map = {c.casefold(): c for c in df.columns}     for a in aliases:         if a.casefold() in cols_map:             return cols_map[a.casefold()]     return None </pre>
<p>Busca la columna objetivo y normaliza el string del objetivo</p>	<pre> TARGET_ALIASES_CU1 = [     "SENSIBILIDAD",     "PD_SENSIBILIDAD",     "CLASIFICACION RESPECTO DE PROTECCION DE     DATOS PERSONALES", ] TARGET_COL_CU1 = pick_col(df_tec, TARGET_ALIASES_CU1) if TARGET_COL_CU1 is None:     print("CU01 no entrenable: no se encontró ninguna de las     columnas objetivo en el dataframe.")     print("Buscadas:", TARGET_ALIASES_CU1)     print("Disponibles:", list(df_tec.columns))     raise SystemExit raw_y = df_tec[TARGET_COL_CU1] if pd.api.types.is_numeric_dtype(raw_y):     y1 = pd.to_numeric(raw_y, errors="coerce").round().astype("Int64") else: </pre>

	<pre>sy = raw_y.astype(str).str.strip().str.upper()</pre>
<p>Mapa robusto para etiquetas</p>	<pre>mapping = {     "SENSIBLE": 1, "ALTA": 1, "ALTO": 1, "SI": 1, "SÍ": 1,     "TRUE": 1, "VERDADERO": 1, "1": 1, "MEDIA": 1,     "NO SENSIBLE": 0, "BAJA": 0, "BAJO": 0, "NO": 0,     "FALSE": 0, "FALSO": 0, "0": 0 } y1 = sy.map(mapping).astype("Int64")  if y1.isna().any():     print("CU01: se hallaron valores del target no mapeados a 0/1.")     print("Ejemplos de valores sin mapa:", pd.Series(raw_y[y1.isna()].unique()).head(10).to_list())     raise SystemExit y1 = y1.astype(int) vc = y1.value_counts() if set(vc.index) != {0, 1}:     print("CU01 no entrenable: el target no es binario 0/1. Distribución:", vc.to_dict())     raise SystemExit</pre>
<p>Definir semántica y realizar conteos por clase, se analizan columnas que pueden filtrar información del target (objetivo),</p>	<pre>print(f"Target CU01 = '{TARGET_COL_CU1}' → 1=sensible, 0=no sensible") print("Distribución de y1:", vc.to_dict()) print(f"Prevalencia clase positiva (sensible): {y1.mean():.3f}") leak_aliases = [     TARGET_COL_CU1,     "CLASIFICACION RESPECTO A CONFIDENCIALIDAD",     "CLASIFICACION RESPECTO A INTEGRIDAD",     "CLASIFICACION RESPECTO A DISPONIBILIDAD",     "CRITICIDAD",     "INDICE DE RIESGO",     "CLASIFICACION RESPECTO DE PROTECCION DE DATOS PERSONALES",     "PD_SENSIBILIDAD",</pre>

	]
<p>Verificación de columnas, corta ejecución por inconsistencia, advierte de columnas inútiles en el análisis, hace el split y aplica regresión logística</p>	<pre> drop_leaks = {pick_col(df_tec, [a]) for a in leak_aliases} drop_leaks.discard(None) feat_cols_cu1 = [c for c in num_cols if c not in drop_leaks] faltan = [c for c in feat_cols_cu1 if c not in df_tec.columns] if faltan:     print("Faltan columnas en df_tec para CU01:", faltan)     raise SystemExit X1_num = df_tec[feat_cols_cu1].apply(pd.to_numeric, errors="coerce").copy() n_nans = int(X1_num.isna().sum().sum()) if n_nans &gt; 0:     print(f" Se detectaron {n_nans} NaN en X1_num (se imputarán con mediana en el pipeline).") var0 = [c for c in X1_num.columns if X1_num[c].nunique(dropna=False) &lt;= 1] if var0:     print("Columnas con varianza cero:", var0) X1_train_raw, X1_test_raw, y1_train, y1_test = train_test_split(     X1_num, y1, test_size=0.30, random_state=SEED, stratify=y1 ) print(f"Split CU01 -&gt; train: {len(y1_train)}   test: {len(y1_test)} (estratificado)") base_lr = LogisticRegression(     max_iter=1000, random_state=SEED, class_weight="balanced", solver="liblinear" ) pipe_cu1 = Pipeline(steps=[     ("imputer", SimpleImputer(strategy="median")),     ("scaler", StandardScaler()),     ("clf", base_lr) ]) skf = StratifiedKFold(n_splits=5, shuffle=True, random_state=SEED) </pre>

	<pre> proba_oof = np.zeros(len(y1_train)) X1_train_idx = X1_train_raw.reset_index(drop=True) y1_train_idx = y1_train.reset_index(drop=True) for fold, (tr, va) in enumerate(skf.split(X1_train_idx, y1_train_idx), 1):     model = clone(pipe_cu1)     model.fit(X1_train_idx.iloc[tr], y1_train_idx.iloc[tr])     proba_oof[va] = model.predict_proba(X1_train_idx.iloc[va])[:, 1] prec_cv, rec_cv, th_cv = precision_recall_curve(y1_train_idx, proba_oof) f1_cv = 2 * (prec_cv * rec_cv) / (prec_cv + rec_cv + 1e-9) thr1 = 0.5 if len(th_cv) &gt; 0:     best_idx = int(np.argmax(f1_cv[1:]))     thr1 = float(th_cv[best_idx]) </pre>
Print de gráficas	<pre> print(f"\nUmbral OOF (max F1 en TRAIN-CV) para CU01: {thr1:.3f}") pipe_cu1.fit(X1_train_raw, y1_train) coef1 = pipe_cu1.named_steps["clf"].coef_[0] coef1_df = pd.DataFrame({"Variable": feat_cols_cu1, "Coeficiente": coef1}).sort_values("Coeficiente", ascending=False) print("\nTop coeficientes CU01 (mayor a menor):") print(coef1_df.to_string(index=False)) proba1 = pipe_cu1.predict_proba(X1_test_raw)[:, 1] y1_pred = (proba1 &gt;= thr1).astype(int) cm1 = confusion_matrix(y1_test, y1_pred, labels=[0, 1]) print("\nMatriz de confusión CU01 [filas=real 0/1, columnas=pred 0/1]:\n", cm1) print("\nReporte de clasificación CU01 (umbral OOF-F1):") print(classification_report(y1_test, y1_pred, digits=3)) roc_auc1 = roc_auc_score(y1_test, proba1) pr_auc1 = average_precision_score(y1_test, proba1) print(f"ROC-AUC CU01: {roc_auc1:.3f}   PR-AUC CU01: {pr_auc1:.3f}") </pre>

```

disp1 = ConfusionMatrixDisplay(confusion_matrix=cm1,
display_labels=["no sensible", "sensible"])
fig1, ax1 = plt.subplots(figsize=(5, 4))
disp1.plot(ax=ax1, colorbar=False)
ax1.set_title(f"CU01 — Sensibilidad (target:
{TARGET_COL_CU1})\nUmbral OOF-F1 = {thr1:.3f}")
ax1.set_xlabel("Predicción")
ax1.set_ylabel("Valor real")
plt.tight_layout()
plt.show()

```

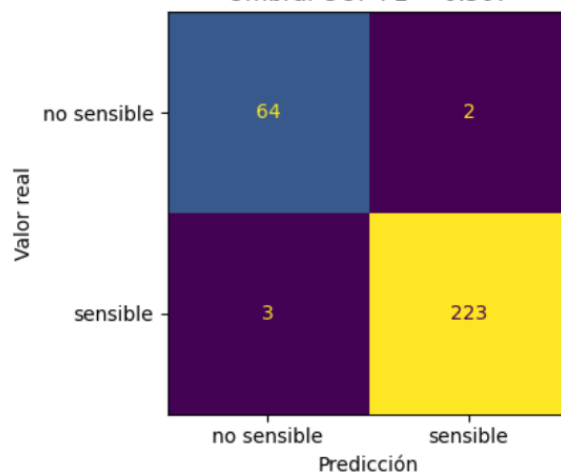
**Ilustración 7:** Sensibilidad- reporte de clasificación CU01

Reporte de clasificación CU01 (umbral OOF-F1):

	precision	recall	f1-score	support
0	0.955	0.970	0.962	66
1	0.991	0.987	0.989	226
accuracy			0.983	292
macro avg	0.973	0.978	0.976	292
weighted avg	0.983	0.983	0.983	292

ROC-AUC CU01: 0.979 | PR-AUC CU01: 0.992

**CU01 — Sensibilidad (target: PD\_SENSIBILIDAD)**  
Umbral OOF-F1 = 0.507



CU02 (confidencialidad), busca verificar con datos la práctica de tratar confidencialidad  $\geq 3$  como categoría diferenciada y operativiznvearla en un flujo reproducible, para ello se construyó el objetivo 0/1 con el umbral  $\geq 3$  ( usando  $\geq 2$  como reserva solo si no hubiera la

separación), se entrenó la logística excluyendo la propia C, y I-D y variables agregadas como criticidad de riesgo. Los resultados mostraron AUC-PR y AUC-ROC altas y estables, el umbral elegido por F1 OOF reprodujo la separación operativa ya intuida, en la práctica, una regla explícita alcanza el mismo desempeño pero para conservar el pipeline aporta compatibilidad entre CUs, trazabilidad y control de calidad (sin fugas, con particiones y métricas homogéneas).

**Tabla 6:** Código de Caso de Uso 02

<b>Acciones</b>	<b>Código</b>
Importar librerías	<pre>import numpy as np import pandas as pd import matplotlib.pyplot as plt from sklearn.model_selection import train_test_split, StratifiedKFold from sklearn.pipeline import Pipeline from sklearn.base import clone from sklearn.impute import SimpleImputer from sklearn.preprocessing import StandardScaler from sklearn.linear_model import LogisticRegression from sklearn.metrics import (     precision_recall_curve, classification_report,     confusion_matrix,     ConfusionMatrixDisplay, roc_auc_score,     average_precision_score )</pre>
Columna objetivo en crudo, se asigna la semilla para reproducibilidad y verificación de clases	<pre>TARGET_COL_CU2 = "CLASIFICACION RESPECTO A CONFIDENCIALIDAD" SEED = 42  umbral_candidatos = [3, 2] y2 = None umbral_usado = None for um in umbral_candidatos:</pre>

	<pre> y_tmp = (pd.to_numeric(df_tec[TARGET_COL_CU2], errors="coerce") &gt;= um).astype(int) vc_tmp = y_tmp.value_counts() if (0 in vc_tmp.index) and (1 in vc_tmp.index):     y2 = y_tmp     umbral_usado = um     break if y2 is None:     print("CU02 no entrenable: la columna de confidencialidad no produce ambas clases con umbral &gt;= 3 ni &gt;= 2.")     print("Distribución real de", TARGET_COL_CU2, ":\n", pd.to_numeric(df_tec[TARGET_COL_CU2], errors="coerce").value_counts(dropna=False))     raise SystemExit </pre>
<p>Definición del objetivo, definir columnas para evitar fugas, diagnósticos de NaN y varianza 0 y split estratificado</p>	<pre> print(f"Target CU02 = 1 si {TARGET_COL_CU2} &gt;= {umbral_usado}; en caso contrario 0.") print("Distribución de y2 (0=baja/media, 1=alta):", y2.value_counts().to_dict()) print(f"Prevalencia de clase positiva (alta): {y2.mean():.3f}") drop_leaks = {     "CLASIFICACION RESPECTO A CONFIDENCIALIDAD",     "CLASIFICACION RESPECTO A INTEGRIDAD",     "CLASIFICACION RESPECTO A DISPONIBILIDAD",     "CRITICIDAD",     "INDICE DE RIESGO" } feat_cols_cu2 = [c for c in num_cols if c not in drop_leaks] faltan = [c for c in feat_cols_cu2 if c not in df_tec.columns] if faltan:     print("Faltan columnas en df_tec para CU02:", faltan)     raise SystemExit X2_num = df_tec[feat_cols_cu2].apply(pd.to_numeric, errors="coerce").copy() n_nans = int(X2_num.isna().sum().sum()) if n_nans &gt; 0:     print(f" Se detectaron {n_nans} valores NaN en X2_num </pre>

	<pre>(se imputarán con mediana dentro del pipeline).") var0 = [col for col in X2_num.columns if X2_num[col].nunique(dropna=False) &lt;= 1] if var0:     print("Columnas con varianza cero (podrían no aportar):", var0) X2_train_raw, X2_test_raw, y2_train, y2_test = train_test_split(     X2_num, y2, test_size=0.30, random_state=SEED, stratify=y2 ) print(f"Split CU02 -&gt; train: {len(y2_train)}   test: {len(y2_test)} (estratificado)")</pre>
<p>Regresión logística robusta en desbalance, pipeline estándar, escalado, clasificador, muestra los coeficientes del modelo, predice el valor del fold, tabla de pesos (coeficientes) y matriz de confusión</p>	<pre>base_lr = LogisticRegression(     max_iter=1000, random_state=SEED, class_weight="balanced", solver="liblinear" ) pipe_cu2 = Pipeline(steps=[     ("imputer", SimpleImputer(strategy="median")),     ("scaler", StandardScaler()),     ("clf", base_lr) ]) skf = StratifiedKFold(n_splits=5, shuffle=True, random_state=SEED) proba_oof = np.zeros(len(y2_train)) X2_train_raw_idx = X2_train_raw.reset_index(drop=True) y2_train_idx = y2_train.reset_index(drop=True) for fold, (tr_idx, va_idx) in enumerate(skf.split(X2_train_raw_idx, y2_train_idx), 1):     model = clone(pipe_cu2) model.fit(X2_train_raw_idx.iloc[tr_idx], y2_train_idx.iloc[tr_idx]) proba_oof[va_idx] = model.predict_proba(X2_train_raw_idx.iloc[va_idx])[:, 1] prec2_cv, rec2_cv, th2_cv =</pre>

	<pre>precision_recall_curve(y2_train_idx, proba_oof) f1_2_cv = 2 * (prec2_cv * rec2_cv) / (prec2_cv + rec2_cv + 1e-9) if len(th2_cv) &gt; 0:     best_idx2 = int(np.argmax(f1_2_cv[1:]))     thr2 = float(th2_cv[best_idx2]) else:     thr2 = 0.5</pre>
<p>Gráficas de visualización con print</p>	<pre>print(f"\nUmbral OOF (max F1 en TRAIN-CV) para CU02: {thr2:.3f}") pipe_cu2.fit(X2_train_raw, y2_train) coef2 = pipe_cu2.named_steps["clf"].coef_[0] coef2_df = pd.DataFrame({     "Variable": feat_cols_cu2,     "Coeficiente": coef2 }).sort_values(by="Coeficiente", ascending=False) print("\nTop coeficientes CU02 (mayor a menor):") print(coef2_df.to_string(index=False)) proba2 = pipe_cu2.predict_proba(X2_test_raw)[:, 1] y2_pred = (proba2 &gt;= thr2).astype(int) cm2 = confusion_matrix(y2_test, y2_pred, labels=[0, 1]) print("\nMatriz de confusión CU02 [filas=real 0/1, columnas=pred 0/1]:\n", cm2) print("\nReporte de clasificación CU02 (umbral OOF-F1):") print(classification_report(y2_test, y2_pred, digits=3)) roc_auc2 = roc_auc_score(y2_test, proba2) pr_auc2 = average_precision_score(y2_test, proba2) print(f"ROC-AUC CU02: {roc_auc2:.3f}   PR-AUC CU02: {pr_auc2:.3f}") disp2 = ConfusionMatrixDisplay(confusion_matrix=cm2, display_labels=["no alta", "alta"]) fig2, ax2 = plt.subplots(figsize=(5, 4)) disp2.plot(ax=ax2, colorbar=False) ax2.set_title(f"CU02 — Confidencialidad ALTA (≥{umbral_usado})\nUmbral OOF-F1 = {thr2:.3f}")</pre>

```

ax2.set_xlabel("Predicción")
ax2.set_ylabel("Valor real") plt.tight_layout()
plt.show()

```

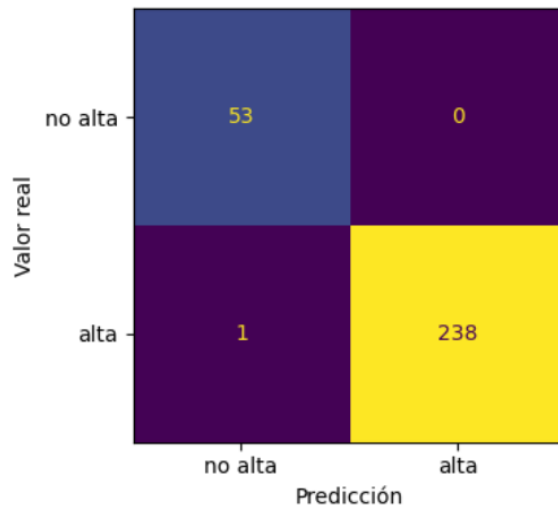
**Ilustración 8:** Confidencialidad- reporte de clasificación CU02

Reporte de clasificación CU02 (umbral OOF-F1):

	precision	recall	f1-score	support
0	0.981	1.000	0.991	53
1	1.000	0.996	0.998	239
accuracy			0.997	292
macro avg	0.991	0.998	0.994	292
weighted avg	0.997	0.997	0.997	292

ROC-AUC CU02: 0.997 | PR-AUC CU02: 0.999

**CU02 — Confidencialidad ALTA ( $\geq 3$ )**  
Umbral OOF-F1 = 0.488



CU03 (priorización de reemplazo en tangibles), se definió la variable residual en años a partir de vida útil y antigüedad y se construyó el objetivo como 1 si residual  $\leq u$  seleccionando el primero que generaba ambas clases, con la logística balanceada y el mismo pipeline se exploró una rejilla amplia de umbrales en validación y se consolidaron tres criterios

- (a). Máximo F1.
- (b). Precisión mínima con relajación si no había candidatos suficientes.

(c). Coste mínimo con piso recall para no dejar por fuera los casos relevantes.

Para este caso se adoptó el criterio (c) como un umbral oficial en ejercicios precios con recall  $\geq 80$ , el umbral operativo quedó alrededor de 0.56 el cual fue validado luego en prueba con matrices de confusión y F1 consistentes, de forma opcional se realizó el ensayo con una regla híbrida (predicción positiva  $\wedge$  criticidad  $\geq 3$ ) que reduce los falsos positivos cuando los recursos son escasos, asumiendo una leve caída de recall, el resultado es una pauta transparente y afinable por costes, adecuada para programar reemplazos con cobertura y control de falsos.

**Tabla 7:**Código de Caso de Uso 03

Acciones	Código
Importar librerías para aplicarlas al caso de uso	<pre> from sklearn.model_selection import train_test_split from sklearn.preprocessing import StandardScaler from sklearn.linear_model import LogisticRegression from sklearn.pipeline import make_pipeline from sklearn.metrics import (     precision_recall_curve, confusion_matrix,     ConfusionMatrixDisplay,     classification_report, roc_auc_score,     average_precision_score ) import numpy as np, pandas as pd, matplotlib.pyplot as plt </pre>
Semilla para reproducibilidad, división de test, porción de validación dentro del train y umbrales con regla híbrida opcional	<pre> SEED = 42 TEST_SIZE = 0.20 VAL_SIZE_IN_TRAIN = 0.20 P_FLOOR_INIT = 0.65 R_FLOOR_INIT = 0.60 STEP_RELAX = 0.05 P_FLOOR_MIN = 0.55 R_FLOOR_MIN = 0.50 COST_FP, COST_FN = 3.0, 1.0 </pre>

	<pre> MIN_PRED_POS    = 2 MIN_PRED_NEG    = 2 USE_HYBRID_RULE = True CRIT_MIN        = 3 </pre>
<p>Filtrar tangibles y prepara el objetivo y umbral definido</p>	<pre> if "TANGIBILIDAD" in df_tec.columns:     tang_mask = (df_tec["TANGIBILIDAD"] == "tangible") elif "df" in globals() and "TANGIBILIDAD" in df.columns and df_tec.index.equals(df.index):     tang_mask = (df["TANGIBILIDAD"] == "tangible") else:     raise RuntimeError("TANGIBILIDAD no se encuentra en df_tec ni alineada en df") df_tan = df_tec.loc[tang_mask].copy() print(f"[CU03] Activos TANGIBLES: {len(df_tan)}") vida_util_num = pd.to_numeric(df_tan["VIDA_UTIL"], errors="coerce") antig_num     = pd.to_numeric(df_tan["antiguedad"], errors="coerce") / 365.0 residual      = vida_util_num - antig_num y3, umbral_residual_anios = None, None for um in [1, 2, 3]:     y_tmp = (residual &lt;= um).astype(int)     if y_tmp.nunique() == 2:         y3 = y_tmp; umbral_residual_anios = um; break if y3 is None:     print(f"[CU03] No entrenable: residual no separa 0/1 ni con 1-3 años.")     raise SystemExit print(f"[CU03] y prevalence: {y3.value_counts().to_dict()}   umbral_residual_anios={umbral_residual_anios}") excluir = {"VIDA_UTIL", "antiguedad"} feat_cols = [c for c in num_cols if c not in excluir] faltan = [c for c in feat_cols if c not in df_tan.columns] if faltan:     print("Faltan columnas en df_tan:", faltan); raise SystemExit </pre>

<p>Se realiza el entrenamiento, escalado, coeficientes informativos</p>	<pre>X_num = df_tan[feat_cols].apply(pd.to_numeric, errors="coerce").fillna(0) X_train_full, X_test, y_train_full, y_test = train_test_split(     X_num, y3, test_size=TEST_SIZE, random_state=SEED, stratify=y3 ) print(f"[Split 80/20] Train_full n={len(y_train_full)}   Test n={len(y_test)}") X_tr, X_val, y_tr, y_val = train_test_split(     X_train_full, y_train_full,     test_size=VAL_SIZE_IN_TRAIN, random_state=SEED, stratify=y_train_full ) print(f"[Split interno] Train n={len(y_tr)}   Val n={len(y_val)} (Test n={len(y_test)}") pipe = make_pipeline(     StandardScaler(),     LogisticRegression(max_iter=1000, random_state=SEED, class_weight="balanced", solver="liblinear") ) pipe.fit(X_tr, y_tr) lr = pipe.named_steps["logisticregression"] coef_df = (pd.DataFrame({"Variable": feat_cols, "Coefficiente": lr.coef_[0]}) .sort_values("Coefficiente", ascending=False)) print("\nTop coeficientes CU03 (train):") print(coef_df.to_string(index=False))</pre>
<p>Se eligen umbrales en validación, además, se aplica una rejilla robusta de umbrales en val para filtrar umbrales válidos y auditables que brinden transparencia en el análisis</p>	<pre>proba_val = pipe.predict_proba(X_val)[:, 1] prec_v, rec_v, th_v = precision_recall_curve(y_val, proba_val) uniq = np.unique(proba_val) mid = (uniq[1:] + uniq[:-1]) / 2.0 thr_grid = np.unique(np.r_[0.0, uniq, mid, 1.0]) def metrics_at_threshold(t, y_true, p):</pre>

	<pre> yhat = (p &gt;= t).astype(int) cm = confusion_matrix(y_true, yhat, labels=[0,1]) TN, FP, FN, TP = cm.ravel() prec = TP/(TP+FP) if (TP+FP) else 0.0 rec = TP/(TP+FN) if (TP+FN) else 0.0 f1 = (2*prec*rec)/(prec+rec+1e-9) cost = COST_FP*FP + COST_FN*FN return prec, rec, f1, cost, cm, yhat thr_valid = [] for t in thr_grid:     yhat = (proba_val &gt;= t).astype(int)     pp = int((yhat == 1).sum()); pn = int((yhat == 0).sum())     if (pp &gt;= MIN_PRED_POS) and (pn &gt;= MIN_PRED_NEG):         thr_valid.append(t) thr_valid = np.array(thr_valid) if len(thr_valid) else thr_grid rows = [] for t in thr_valid:     rows.append((t,) + metrics_at_threshold(t, y_val.values, proba_val)) df_th_val = pd.DataFrame(rows, columns=["thr", "prec", "rec", "f1", "cost", "cm", "yhat"]) thr_f1_val = float(df_th_val.loc[df_th_val["f1"].idxmax(), "thr"]) </pre>
<p>Se realiza la Precision en validación, coste mínimo con piso de recall en val, reentrenamiento en TRAIN_FULL y evaluación en TEST con estos umbrales determinados con anterioridad</p>	<pre> p_floor_eff = P_FLOOR_INIT; thr_prec_val = None while p_floor_eff &gt;= P_FLOOR_MIN and thr_prec_val is None:     cand = df_th_val.query("prec &gt;= @p_floor_eff")     if len(cand):         thr_prec_val = float(cand.loc[cand["rec"].idxmax(), "thr"]); break     p_floor_eff -= STEP_RELAX if thr_prec_val is None:     beta=0.5     fb = (1+beta**2)*(df_th_val["prec"]*df_th_val["rec"])/(beta**2*df_t </pre>

```

h_val["prec"]+df_th_val["rec"]+1e-9)
    thr_prec_val = float(df_th_val.loc[fb.idxmax(),"thr"])
r_floor_eff = R_FLOOR_INIT; thr_cost_val = None
while r_floor_eff >= R_FLOOR_MIN and thr_cost_val is
None:
    cand = df_th_val.query("rec >= @r_floor_eff")
    if len(cand):
        thr_cost_val =
float(cand.loc[cand["cost"].idxmin(),"thr"]); break
    r_floor_eff -= STEP_RELAX
if thr_cost_val is None:
    thr_cost_val = thr_f1_val
print(f"\n[VAL] Umbrales: F1={thr_f1_val:.3f} |
Prec≥{p_floor_eff:.2f}={thr_prec_val:.3f} | Coste
(R≥{r_floor_eff:.2f})={thr_cost_val:.3f}")
pipe.fit(X_train_full, y_train_full)
proba_test = pipe.predict_proba(X_test)[:, 1]
roc_auc = roc_auc_score(y_test, proba_test)
pr_auc = average_precision_score(y_test, proba_test)
print(f"\n[TEST] ROC-AUC={roc_auc:.3f} | PR-
AUC={pr_auc:.3f}")

def eval_on_test(tag, thr):
    yhat = (proba_test >= thr).astype(int)
    cm = confusion_matrix(y_test, yhat, labels=[0,1])
    pp = int((yhat==1).sum())
    print(f"\n[TEST-{tag}] thr={thr:.3f} |
pred_pos={pp}/{len(y_test)} ({100*pp/len(y_test):.2f}%)")
    print(cm) print(classification_report(y_test, yhat, digits=3,
zero_division=0))
    return cm, yhat
cm_f1, yhat_f1 = eval_on_test("F1(val)", thr_f1_val)
cm_prec, yhat_prec =
eval_on_test(f"Prec≥{p_floor_eff:.2f}(val)", thr_prec_val)
cm_cost, yhat_cost = eval_on_test(f"Coste
R≥{r_floor_eff:.2f}(val)", thr_cost_val)

```

<p>Regla híbrida en TEST, (predicción positiva <math>\wedge</math> criticidad <math>\geq 3</math>) esta regla permite reducir los falsos positivos cuando se evidencias menos recursos y se muestran las gráficas de visualización con print</p>	<pre> if USE_HYBRID_RULE:     crit_vals_test = pd.to_numeric(df_tan.loc[y_test.index, "CRITICIDAD"], errors="coerce").fillna(0)     crit_mask_test = (crit_vals_test &gt;= CRIT_MIN).astype(int).values     base_pred = yhat_cost    yhat_h = (base_pred &amp; crit_mask_test).astype(int)    cm_h = confusion_matrix(y_test, yhat_h, labels=[0,1])     print(f"\n[TEST-Híbrido] (Coste <math>\wedge</math> CRIT<math>\geq</math>{CRIT_MIN})")     print(cm_h) print(classification_report(y_test, yhat_h, digits=3, zero_division=0)) else:     cm_h = None fig, axes = plt.subplots(1, 2, figsize=(9,4)) for cm, title, ax in [     (cm_f1, f"F1(val) thr={thr_f1_val:.3f}", axes[0]),     (cm_cost, f"Coste(val) thr={thr_cost_val:.3f}", axes[1]), ]:     disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=["no reempl. =&lt; umbral ", " reempl. =&lt; umbral"])     disp.plot(ax=ax, colorbar=False)     ax.set_xlabel("Predicción"); ax.set_ylabel("Valor real") plt.suptitle(f"CU03 — OOF ROC-AUC={roc_auc:.3f}   PR- AUC={pr_auc:.3f}") #título global con AUCs plt.tight_layout(); plt.show() thr_used = thr_cost_val y_pred = (proba_test &gt;= thr_used).astype(int) cm_used = confusion_matrix(y_test, y_pred, labels=[0,1]) print(f"\n[OFICIAL] Umbral CU03 (desde VALIDACIÓN) = {thr_used:.3f}") print(cm_used) print(classification_report(y_test, y_pred, digits=3, zero_division=0)) </pre>
--	--

**Ilustración 9:** Priorización de reemplazo en tangibles- reporte de clasificación CU03

```

[CU03] Activos TANGIBLES: 965
[CU03] y prevalence: {0: 865, 1: 100} | umbral_residual_anios=3
[Split 80/20] Train_full n=772 | Test n=193
[Split interno] Train n=617 | Val n=155 (Test n=193)

Top coeficientes CU03 (train):
Variable Coeficiente
CRITICIDAD 0.764817
CLASIFICACION RESPECTO A CONFIDENCIALIDAD 0.668388
INDICE DE RIESGO 0.609341
TASA DE DEPRECIACION 0.097305
CLASIFICACION RESPECTO A INTEGRIDAD -0.433486
CLASIFICACION RESPECTO A DISPONIBILIDAD -0.433486

[VAL] Umbrales: F1=0.560 | Precz0.65=0.560 | Coste (Rz0.60)=0.560

[TEST] ROC-AUC=0.960 | PR-AUC=0.656

[TEST-F1(val)] thr=0.560 | pred_pos=27/193 (13.99%)
[[166 7]
 [ 0 20]]
precision recall f1-score support
0 1.000 0.960 0.979 173
1 0.741 1.000 0.851 20

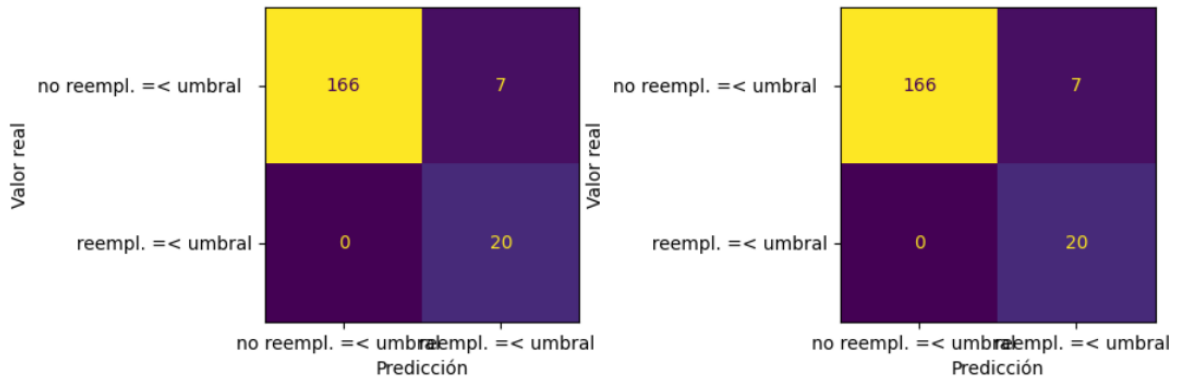
accuracy 0.964 193
macro avg 0.870 0.980 0.915 193
weighted avg 0.973 0.964 0.966 193

[TEST-Precz0.65(val)] thr=0.560 | pred_pos=27/193 (13.99%)
[[166 7]
 [ 0 20]]
precision recall f1-score support
0 1.000 0.960 0.979 173
1 0.741 1.000 0.851 20

accuracy 0.964 193
macro avg 0.870 0.980 0.915 193
weighted avg 0.973 0.964 0.966 193
    
```

**Ilustración 10:** Matriz de confusión CU03

CU03 — OOF ROC-AUC=0.960 | PR-AUC=0.656



```

[OFICIAL] Umbral CU03 (desde VALIDACIÓN) = 0.560
[[166 7]
 [ 0 20]]
precision recall f1-score support
0 1.000 0.960 0.979 173
1 0.741 1.000 0.851 20

accuracy 0.964 193
macro avg 0.870 0.980 0.915 193
weighted avg 0.973 0.964 0.966 193
    
```

## Red neuronal

Para el modelo de red neuronal (complementaria), se entrenó una red densa sencilla (32-16-1 con ReLu y sigmoid, early stopping), replicando splits y escalado, en al menos uno de los CUs las curvas de validación y prueba fueron perfectas (AP=1.000 y AUC-ROC=1.000, sin errores en la matriz), lo que implica que con las variables disponibles y sin fugas, el problema es separable y la decisión equivale a una regla clara (umbral o condiciones directas).

**Tabla 8:** Tabla de red neuronal

Acciones	Código
Instalación de tensorflow para proceder con el modelo de red neuronal	<pre>!pip -q install tensorflow scikit-learn matplotlib import tensorflow as tf from tensorflow import keras import matplotlib.pyplot as plt from tensorflow.keras import layers</pre>
Filtro para considerar solo activos tecnológicos	<pre>tecnologicos = {"HARDWARE", "SOFTWARE", "COMPONENTES_DE_RED"} df_tec = df[df[COLS["tipo_activo"]].isin(tecnologicos)].copy()</pre>
Toma columnas con valores numéricos	<pre>num_cols = [ COLS["clas_conf"], COLS["clas_int"], COLS["clas_disp"], COLS["criticidad"], "antiguedad", "INDICE DE RIESGO", "VIDA_UTIL", "TASA DE DEPRECIACION", ]</pre>
Objetivo de Sensibilidad, se escoge debido a que es una variable dicotómica, lo que ayuda a predecir y clasificar activos sensibles para decidir políticas de seguridad a futuro.	<pre>X = df_tec[num_cols].fillna(0).values y = (df_tec["PD_SENSIBILIDAD"] == "sensible").astype(int).values</pre>

<p>Escoger la semilla y se deciden las variables a entrenar</p>	<pre>SEED = 42 X_train, X_temp, y_train, y_temp = train_test_split(X, y, test_size=0.30, random_state=SEED, stratify=y) X_val, X_test, y_val, y_test = train_test_split(X_temp, y_temp, test_size=0.50, random_state=SEED, stratify=y_temp)</pre>
<p>Estandarización y scaler</p>	<pre>scaler = StandardScaler() X_train = scaler.fit_transform(X_train) X_val = scaler.transform(X_val) X_test = scaler.transform(X_test)</pre>
<p>Se escoge un modelo de perceptrón de varias capas</p>	<pre>tf.keras.utils.set_random_seed(SEED)</pre>
<p>Se implementan densidad, input y formas</p>	<pre>model = keras.Sequential([ layers.Input(shape=(X_train.shape[1],)), layers.Dense(32, activation="relu"), layers.Dropout(0.2), layers.Dense(16, activation="relu"), layers.Dense(1, activation="sigmoid") ])</pre>
<p>Se compila y resume el modelo aplicado</p>	<pre>model.compile(optimizer=keras.optimizers.Adam(learning_rate =1e-3), loss="binary_crossentropy", metrics=["accuracy"]) model.summary()</pre>
<p>Se entrena con early stopping</p>	<pre>cb = keras.callbacks.EarlyStopping(monitor="val_loss", patience=10, restore_best_weights=True) hist = model.fit( X_train, y_train, validation_data=(X_val, y_val), epochs=100, batch_size=32, callbacks=[cb], verbose=0 )</pre>

<p>Gráficas y FROM_LOGITS que indica si el modelo devuelve logits (true) o probabilidades (false)</p>	<pre>import numpy as np import matplotlib.pyplot as plt from sklearn.metrics import (     precision_recall_curve, roc_curve,     average_precision_score, roc_auc_score,     f1_score, confusion_matrix ) FROM_LOGITS = False</pre>
<p>Definición del sigmoide</p>	<pre>def _sigmoid(x):     return 1/(1+np.exp(-x))</pre>
<p>Predicciones</p>	<pre>def eval_pr_roc_keras(model, X, y, title_prefix="Validación"):     y_true = np.asarray(y).ravel().astype(int)     probs = model.predict(X, verbose=0).ravel()     if FROM_LOGITS:         probs = _sigmoid(probs)</pre>
<p>Curvas, AUCs, F1 y matriz de confusión</p>	<pre>p, r, thr_pr = precision_recall_curve(y_true, probs) fpr, tpr, _ = roc_curve(y_true, probs) auc_pr = average_precision_score(y_true, probs) auc_roc = roc_auc_score(y_true, probs) cand = np.unique(np.r_[thr_pr, 0.5]) cand = cand[(cand&gt;1e-6) &amp; (cand&lt;1-1e-6)] f1s = [f1_score(y_true, (probs&gt;=th).astype(int), zero_division=0) for th in cand] best_idx = int(np.argmax(f1s)) thr_f1 = float(cand[best_idx]) y_hat_best = (probs &gt;= thr_f1).astype(int) cm = confusion_matrix(y_true, y_hat_best)</pre>
<p>Gráficas de visualización con plt y print</p>	<pre>pos_rate = y_true.mean() plt.figure(figsize=(5,4)) plt.plot(r, p, label=f"PR (AP={auc_pr:.3f})") plt.hlines(pos_rate, 0, 1, linestyle="--", label=f"Base={pos_rate:.3f}") plt.xlabel("Recall"); plt.ylabel("Precision") plt.title(f"{title_prefix} · Precision-Recall"); plt.legend(); plt.tight_layout(); plt.show() plt.figure(figsize=(5,4))</pre>

	<pre>plt.plot(fpr, tpr, label=f"ROC (AUC={auc_roc:.3f})") plt.plot([0,1],[0,1], "--", label="Azar") plt.xlabel("FPR"); plt.ylabel("TPR (Recall)") plt.title(f"{title_prefix} · ROC"); plt.legend(); plt.tight_layout(); plt.show()  print(f"{title_prefix} -&gt; AP (AUC-PR): {auc_pr:.3f}   AUC- ROC: {auc_roc:.3f}") print(f"Umbral(F1): {thr_f1:.3f}   F1: {f1s[best_idx]:.3f}") print("Matriz de confusión (umbral F1):\n", cm) return {"probs": probs, "auc_pr": auc_pr, "auc_roc": auc_roc, "thr_f1": thr_f1, "f1": f1s[best_idx], "cm": cm}</pre>
Validación y Test	<pre>res_val = eval_pr_roc_keras(model, X_val, y_val, title_prefix="Validación") res_te = eval_pr_roc_keras(model, X_test, y_test, title_prefix="Test")</pre>

Modelo “sequential”

Se implementan densidad, input, formas

Ilustración 11: Modelo “sequential”

Model: "sequential"

Layer (type)	Output Shape	Param #
dense (Dense)	(None, 32)	288
dropout (Dropout)	(None, 32)	0
dense_1 (Dense)	(None, 16)	528
dense_2 (Dense)	(None, 1)	17

Total params: 833 (3.25 KB)

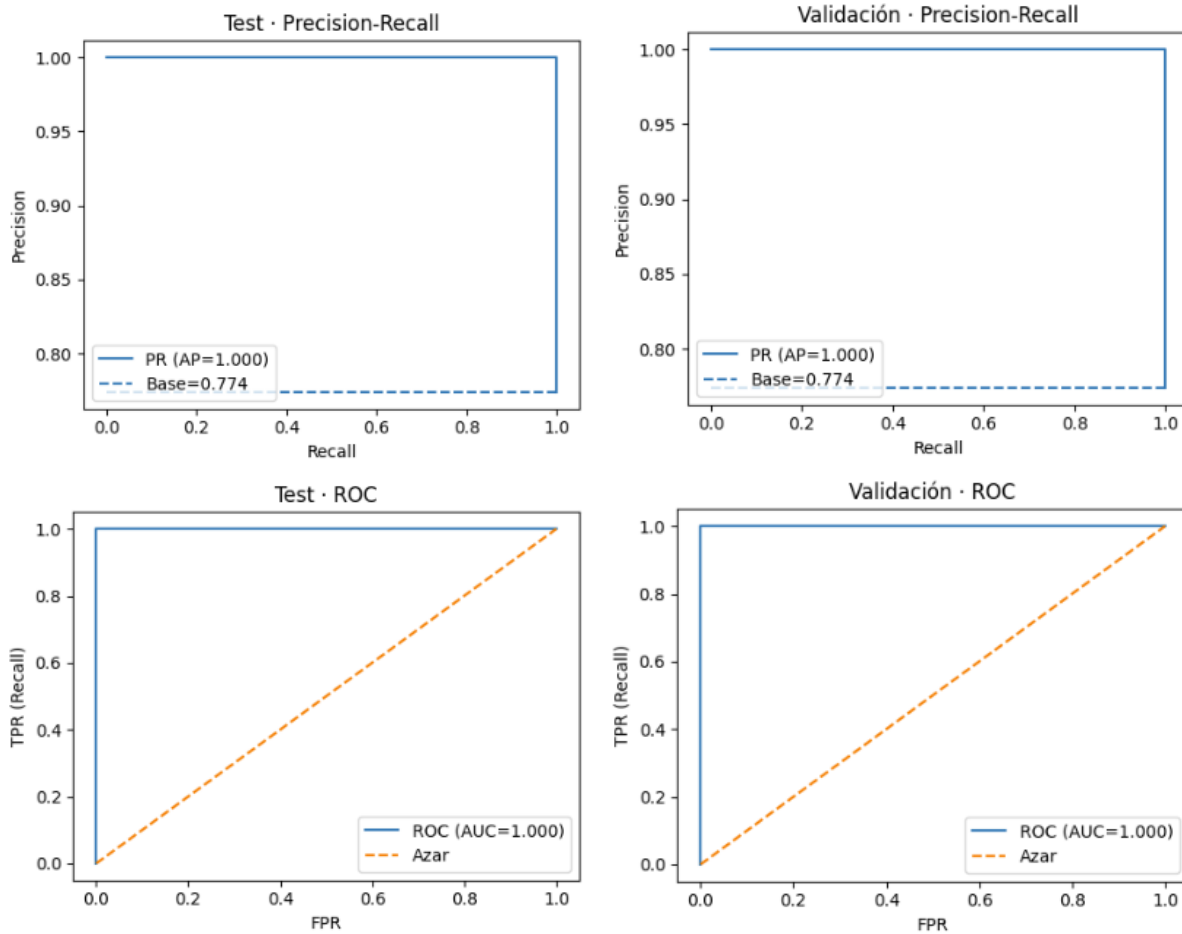
Trainable params: 833 (3.25 KB)

Non-trainable params: 0 (0.00 B)

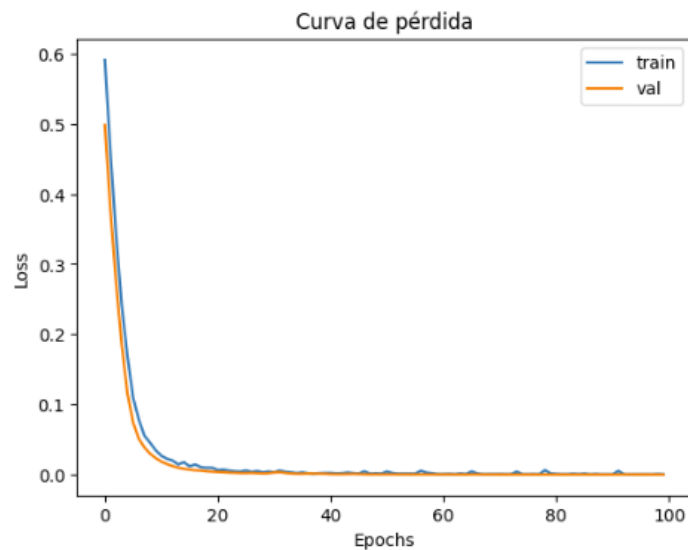
Sin embargo, esto no desplaza la regresión logística, simplemente confirma que no es

necesaria mayor complejidad para lograr el rendimiento observado y deja a la red neuronal como referencia metodológica y control independiente de que el flujo de datos está bien construido (preparación en entrenamiento, exclusiones correctas y umbrales coherentes con las curvas).

**Ilustración 12:** Precisión y área de ROC de la red neuronal



**Ilustración 13:** Curva de pérdida - red neuronal



**Evaluar las fortalezas y debilidades del modelo en función de las métricas referenciadas para la valoración de los modelos de aprendizaje y que son comúnmente aceptadas en estos procesos para definir la efectividad de la herramienta.**

En conjunto, los resultados cumplen con los objetivos al entregar información accionable para optimizar la administración y apoyar decisiones estratégicas del ciclo de vida de los activos (clasificar sensibilidad, confirmar confidencialidad alta, programar reemplazos con cobertura y coste explícitos), a su vez, se da cumplimiento a los objetivos específicos:

1. Se caracterizaron variables críticas y se controló la calidad del dato (limpieza, varianza y fugas).
2. Se implementó un modelo predictivo reproducible como lo son la regresión logística y la red neuronal (NN) como contraste en el análisis para integrar las variables operativas en la base de datos aplicada.
3. La evaluación de fortalezas y debilidades con métricas aceptadas (AUC-PR, AUC-ROC,

F1, coste y matrices) priorizando aquellas robustas al desbalance y documentando el umbral operativo.

La conclusión técnica es clara, con el estado actual del dataset, la señal de negocio es fuerte, la regresión logística resuelve los tres CUs con suficiencia, interpretabilidad y trazabilidad y la red neuronal corrobora que no se requiere complejidad adicional para sostener el desempeño del caso de estudio, esto deja a la organización con una herramienta estable, auditable y ajustable gracias a que se le puede hacer una recalibración periódica del umbral y afinación de costes lista para integrarse a los procesos de gestión de inventario tecnológico.

## 10. Conclusiones

Los modelos predictivos basados en *machine learning* han transformado la manera en que las organizaciones gestionan sus activos tecnológicos y equipos de cómputo. Estas herramientas permiten anticipar fallas, planificar mantenimientos y optimizar el uso de recursos, al tiempo que facilitan la toma de decisiones sustentadas en datos reales. La implementación de algoritmos como la regresión logística y las redes neuronales artificiales ha demostrado ser especialmente útil para identificar patrones de comportamiento y evaluar la vida útil de los equipos, adaptándose a las necesidades específicas de cada entidad, sin importar su tamaño o actividad económica.

En la actualidad, las instituciones están orientando sus esfuerzos hacia la adopción de modelos de análisis predictivo que integren grandes volúmenes de datos provenientes de sus inventarios tecnológicos. Estos modelos no solo buscan automatizar procesos de mantenimiento, sino también mejorar la eficiencia de los recursos disponibles, reducir costos y aumentar la satisfacción de los usuarios internos. Al incorporar técnicas de *machine learning*, como la regresión logística o las redes neuronales, se fortalece la capacidad de detección temprana de fallas, generando una cultura organizacional basada en la prevención y la mejora continua de los procesos tecnológicos.

A partir de la revisión bibliográfica y del análisis de las necesidades identificadas en diferentes organizaciones, se evidenció la importancia de desarrollar una herramienta predictiva que apoye la gestión de activos de hardware y software. Este tipo de solución permite concentrar la información en una base de datos centralizada, favoreciendo la integración entre áreas y garantizando decisiones oportunas respecto al mantenimiento, reposición o actualización de los equipos. Con ello, se minimizan los tiempos de inactividad, se evitan gastos innecesarios y se optimiza la inversión en tecnología para la prestación del servicio.

El desarrollo del código en este trabajo de grado permitió comprobar la efectividad de la regresión logística como modelo principal para determinar el estado de los equipos y establecer si requieren mantenimiento o presentan señales de obsolescencia. La elección de este modelo respondió al tamaño y la estructura de la base de datos disponible, logrando un equilibrio entre precisión, interpretabilidad y eficiencia computacional. Los resultados obtenidos reflejaron que la regresión logística ofrece una clasificación confiable, fácilmente comprensible y replicable en diferentes entornos u organizaciones.

Durante la fase de experimentación se observó que variables como la antigüedad del equipo, el historial de mantenimiento y el nivel de rendimiento tienen un impacto directo en la predicción del estado del activo. El modelo permitió establecer una relación clara entre estos factores y la probabilidad de falla, proporcionando una herramienta práctica para la planificación de mantenimientos preventivos. Las métricas de evaluación, como la matriz de confusión y la precisión, evidenciaron un desempeño adecuado del modelo para apoyar la toma de decisiones tecnológicas.

Aunque se analizó también la posibilidad de utilizar redes neuronales como modelo alternativo, se determinó que la regresión logística resultaba más apropiada para el volumen de datos y los objetivos del proyecto. Sin embargo, el código fue diseñado de forma modular, lo que permite su futura adaptación a modelos más complejos, incluyendo redes neuronales, siempre que se cuente con un conjunto de datos más amplio. Esta flexibilidad convierte la propuesta en una herramienta versátil y escalable para distintos escenarios organizacionales.

Finalmente, la implementación de técnicas de *machine learning* dentro del marco del trabajo de grado demuestra el potencial de estas tecnologías para fortalecer la gestión predictiva en las instituciones. Al aprovechar modelos como la regresión logística y las redes neuronales, las organizaciones pueden anticipar necesidades, optimizar recursos y tomar decisiones

estratégicas fundamentadas en evidencia, consolidando así una gestión tecnológica más inteligente, sostenible y eficiente.

## 11. Recomendaciones

En atención a los resultados obtenidos y al análisis desarrollado durante la implementación del modelo de *machine learning*, se presentan las siguientes recomendaciones orientadas a fortalecer la gestión predictiva de los activos tecnológicos dentro de las organizaciones:

1. Se recomienda ampliar la base de datos empleada en el entrenamiento del modelo, incorporando información histórica y nuevos registros de los equipos. Esto permitirá mejorar la capacidad predictiva del algoritmo y abrir la posibilidad de implementar modelos más complejos, como las redes neuronales, cuando se disponga de un volumen de datos mayor.

2. Es aconsejable evaluar modelos complementarios de aprendizaje automático: tales como, máquinas de soporte vectorial o redes neuronales, con el fin de comparar su desempeño frente a la regresión logística y seleccionar el más adecuado según las características del conjunto de datos con los que se cuente para su implementación, además de la necesidad que requieran.

3. Se sugiere desarrollar un tablero de monitoreo o panel de control: que integre las predicciones del modelo presentado, permitiendo visualizar el estado de los equipos en tiempo real e integrando diferentes analíticas. Esto facilita la planeación de mantenimientos preventivos y la priorización de intervenciones según el nivel de riesgo detectado.

4. Fortalecer las competencias del personal técnico, mediante procesos de capacitación en análisis de datos y fundamentos de *machine learning*, de manera que los equipos encargados de la gestión tecnológica comprendan y apliquen adecuadamente los resultados generados por el modelo presentado y puedan hacer los cambios que requieren de manera satisfactoria, integrando rutinas que permitan la carga continua de datos y la generación periódica de nuevas

predicciones.

5. Finalmente, se sugiere promover en la organización una cultura de toma de decisiones basada en los datos, utilizando los resultados de este modelo como insumo estratégico para planificar presupuestos, definir políticas de mantenimiento preventivo y proyectar la renovación tecnológica de manera sustentable para la mejora continua.

## 12. Referencias

ABB. (2019). Predictive maintenance with machine learning: ABB's experience. ABB White Paper.

Accenture. (2023). *The predictive enterprise: Using machine learning for smarter asset management*. Accenture Insights

Ahmed, N., Saha, A. K., Al Noman, M. A., Jim, J. R., Mridha, M. F., & Kabir, M. M. (2024). Deep learning-based natural language processing in human-agent interaction: Applications, advancements and challenges. *Natural Language Processing Journal*, 9(100112), 100112.

Al-Amin, S. T., & Ordonez, C. (2021). Efficient machine learning on data science languages with parallel data summarization. *Data & Knowledge Engineering*, 136(101930), 101930.

Aljumah, A., Ayesh, A., 'Alshaikh, M., 'Alghamdi, A., 'Alrashidi, M., 'Alsulami, B., & Alotaibi, A. (2022). Predictive maintenance for asset management: A review. *International Journal of Prognostics and Health Management*, 13(1), 1-12.

Alcivar Castro, J. E., Viscarra Zambrano, D. R., & ESPOL, F. (2024). Sistema de gestión de inventario y seguimiento automatizado de elementos en bodega. ESPOL. FIEC, Guayaquil.

Amador, B., & Márquez, A. (2008). *Un modelo conceptual para gestionar la tecnología en la organización*. *Revista Espacios*, 29(1), 1–22. Recuperado de <http://www.revistaespacios.com/a09v30n01/09300121.html>

Banikya, A. K., Ranade, P., & Verma, I. K. (2024). Gestión de stock muerto en la industria de la moda y la confección: Una exploración de causas, soluciones e intervenciones tecnológicas. *Engineering Proceedings*, 66(9). <https://doi.org/10.3390/engproc2024066009>

Bellido Mantilla, R., Bsc, L. P., Bsc, V., & Msc, C. (s/f). Prospectiva y tendencias en tecnología y competencias para el desarrollo social sostenible. En Aprovechando las tecnologías emergentes para construir el futuro: Actas de la 19.ª Multiconferencia Internacional LACCEI de Ingeniería.

Belman-López, C. E., Jiménez-García, J. A., Vázquez-Lopez, J. A., & Camarillo-Gómez, K. A. (2023). Diseño de una arquitectura para sistemas y aplicaciones en Industria 4.0 basada en computación en la nube y análisis de datos. *Revista Iberoamericana de Automática e Informática Industrial*, 20(2), 137–149.

BM. (2023). El análisis predictivo de IBM reduce los tiempos de inactividad del servidor.

Chaves Jiménez, A. F. (2021). Análisis de la gestión y control de bienes tecnológicos en el área de activos TI, aplicando la metodología ITIL V4 ítem 5.2.6 “IT Asset Management” para la empresa Corbeta S.A. [Trabajo de grado, Corporación Universitaria Remington].

Chopra, S. (2019). *Supply chain management: Strategy, planning, and operation* (8th ed.). Pearson.

Chien, C. F., & Chen, L. F. (2021). Data-driven approach for predictive maintenance of complex systems. *Annual Reviews in Control*, 51, 242-256. <https://doi.org/10.1016/j.arcontrol.2020.12.002>.

Cotec. (1998). *Temaquide: Pautas metodológicas en gestión de la tecnología y de la innovación para las empresas*. Fundación Cotec para la Innovación Tecnológica.

Concha, M., & Selene, M. (2012). Área de tecnología: Orden y control de inventario de insumos tecnológicos. Recuperado de <http://repositorio.umayor.cl/xmlui/handle/sibum/3135>

De Paula Vidal, G. H., Caiado, R. G. G., Scavarda, L. F., Ivson, P., & Garza-Reyes, J. A. (2022). Decision support framework for inventory management combining fuzzy

multicriteria methods, genetic algorithm, and artificial neural network. *Computers & Industrial Engineering*, 174(108777), 108777. <https://doi.org/10.1016/j.cie.2022.108777>

Deming, W. E. (1986). *Out of the crisis*. MIT Press.

Decreto 1078 de 2025. Por el cual se compilan las normas del sector de Tecnologías de la Información y las Comunicaciones. Presidencia de la República de Colombia.

Ecopetrol. (2024). *Informe de sostenibilidad y transformación digital 2024*. Ecopetrol S.A.

Falkner, D., Bögl, M., Gattinger, A., Stainko, R., Zenisek, J., & Affenzeller, M. (2024). Integración del aprendizaje automático en la gestión de la cadena de suministro: Desafíos y oportunidades. *Procedia Computer Science*, 232, 1779–1788. <https://doi.org/10.1016/j.procs.2024.01.176>

Fatorachian, H., Kazemi, H., & Pawar, K. (Year). Title of the article in sentence case. *Sustainability*, 17(5), 1996.

Flores Tapia, C., & Flores Cevallos, K. (2023). Optimización de inventarios aplicando investigación de operaciones. *RECAI Revista de Estudios en Contaduría, Administración e Informática*.

Fragapane, G., de Koster, R., Sgarbossa, F., & Strandhagen, J. O. (2021). Planning and control of autonomous mobile robots for intralogistics: Literature review and research agenda. *European Journal of Operational Research*, 291(1), 1–20.

Gartner. (2021). *Market guide for IT asset management tools*. Gartner Research.

González, A. (2020). Un modelo de gestión de inventarios basado en estrategia competitiva. *Ingeniare. Revista Chilena de Ingeniería*, 28(1), 133–142.

Guzmán, P., Montalvo, F., Carvallo, E., & Raymundo, C. (2019). Implementación de un modelo de gestión de procesos y control de inventarios para incrementar el nivel de servicio en el área de postventa de equipos industriales. 17th LACCEI International Multi-

Conference for Engineering, Education, and Technology.  
<https://doi.org/10.18687/LACCEI2019.1.1.147>

Guzmán, P., Montalvo, F., Carvallo, E., & Raymundo, C. (2023). Escuela de Ingeniería Industrial, Universidad Peruana de Ciencias Aplicadas, Lima, Perú: Dirección de Investigaciones. Lima, Perú.

Hernández García, D. G. (2019). Diseño e implementación de un esquema de seguridad de nivel 0 a nivel 1 basado en las normas ISO 27002:2013. [Trabajo de maestría, ESPOL]. FIEC, Guayaquil.

Hernández Rodríguez, I. Y., & Torres Aponte, L. M. (2021). Modelo de pronóstico de demanda para productos del sector eléctrico. Tesis de Postgrado, Universidad de los Andes, Bogotá. Recuperado el 3 de febrero de 2024, de <https://repositorio.uniandes.edu.co/server/api/core/bitstreams/7c5bd9f6-6dc2-437f-90ea2d298a4628f4/content>

Hidalgo, A., León, S., & Pavón, J. (2002). *La gestión de la innovación y la tecnología en las organizaciones*. Ediciones Pirámide.

Iluore, O. E., Mamudu Onose, A., & Emetere, M. (2020). Development of asset management model using real-time equipment monitoring (RTEM): Case study of an industrial company. *Cogent Business & Management*, 7(1), 1763649. <https://www.tandfonline.com/doi/full/10.1080/23311975.2020.1763649#abstract>

Institute of Internal Auditors (IIA). (2020). International professional practices framework (IPPF). <https://na.theiia.org>

International Organization for Standardization (ISO). (2014). ISO 55000: Asset management — Overview, principles and terminology.

ISO/IEC 27001:2022. *Information Security Management Systems – Requirements*. International Organization for Standardization.

ISO 55001:2014. *Asset Management – Management Systems – Requirements*. International Organization for Standardization.

Jiménez Aristizábal, J., & Arango Arango, A. (2023). Aplicativo web responsive con tablero de control para la gestión de activos TI hardware y software: Caso de estudio Secretaría de Educación de Medellín. Tecnológico de Antioquia, Institución Universitaria.

Jofra, X. B., & Gómez, A. P. (2018). Innovaciones tecnológicas en la cadena de suministro aplicadas al eCommerce. *Oikonomics Revista de Economía Empresa y Sociedad*.

*Journal of the American Medical Informatics Association*, Volume 29, Issue 4, April 2022, Pages 631–642.

Kim, J., Park, S., & Lee, S. (2020). Deep learning-based predictive maintenance for industrial machinery: A case study. *Journal of Manufacturing Systems*, 56, 317-324. <https://doi.org/10.1016/j.jmsy.2020.06.011>

Kausik, A. K., Rashid, A. B., Baki, R. F., & Maktum, M. M. J. (2025). Machine learning algorithms for manufacturing quality assurance: A systematic review of performance metrics and applications. *Array*, 26, 100393. <https://doi.org/10.1016/j.array.2025.100393>

Kovárník, R., & Janová, J. (2025). Validation of sentinel 2 based machine learning models for Czech National Forest Inventory. *Ecological Informatics*, 87, 103133. <https://doi.org/10.1016/j.ecoinf.2025.103133>

Lins Espíndola, S. C. N. (2019). Standardization of administrative processes: A case study using continuous improvement tool. *Brazilian Journal of Operations & Production Management*.

Lucio Pillasagua, A. del J., Arteaga Toala, N. J., Bumbila Anzules, J. E., Marcillo Salazar, V. C., & Garcia Constante, V. V. (2023). Optimización de inventarios en empresas retail a través de análisis predictivo [Inventory optimization in retail companies through predictive analysis]. ResearchGate.

Ley 1581 de 2012. Por la cual se dictan disposiciones generales para la protección de datos personales. Diario Oficial de la República de Colombia.

Maria Selene Monsalve Concha. (s/f). Ingeniera en Administración con Mención en Finanzas y Control Presupuestario Facultad de Ciencias Económicas y Empresariales. Universidad Mayor.

Microsoft. (2024). *Sustainability and machine learning in datacenter management. Microsoft Sustainability Report.*

Montañez Murillo, O. A. (2025). Pronóstico del comportamiento de los activos administrados por los fondos de inversión colectiva de las sociedades fiduciarias en Colombia mediante modelos de inteligencia artificial y herramientas econométricas. Universidad Nacional Abierta y a Distancia UNAD.

Morales Castaño, O. A. (2018). Diseño de un esquema para la gestión y control de inventarios de activos fijos y acceso de colaboradores para la organización Carvajal S.A.

Moreno-Ponce, M. R. (2024). Valoración empresarial moderna: Un análisis del estado del arte y sus innovaciones. *MQRInvestigar*, 8(3), 3186–3205.

Ortiz Panqueva, J. K., Jaramillo Soto, V. M., & Serrano Rodríguez, X. P. (2023). Propuesta de implementación de un modelo predictivo para la gestión de inventarios de la compañía IMECOL S.A.S. Universidad EAN.

Prajogo, D. I., & Sohal, A. S. (2001). TQM and innovation: A literature review and research framework. *Technovation*, 21(9), 539-558. [https://doi.org/10.1016/S0166-4972\(00\)00044-0](https://doi.org/10.1016/S0166-4972(00)00044-0)

Pantoja España, L. D., & Galvis Mendedelso, Z. A. (2024). Mejora de la gestión en inventarios a través de modelos predictivos con machine learning. [Trabajo de grado, Corporación Universitaria Remington].

Praful Bharadiya, J. (2023). Un estudio comparativo de inteligencia empresarial e inteligencia artificial con análisis de big data. *Revista Americana de Inteligencia Artificial.*

Ramírez, F. (2023). Una Perspectiva De Las Pruebas De Software En La Nube. Academia. Edu.

Regulski, K., Rauch, L., Hajder, P., Bzowski, K., Opaliński, A., Pernach, M., Hallo, F., Piwowarczyk, M., & Kalinowski, S. (2024). Optimizing continuous casting through cyber-physical system. *Processes*, 12(8), 1761.

Rey, I., & Valle, J. E. (2024). Transformación digital en la logística internacional: Estrategias y desafíos de la inteligencia artificial para los inventarios y cadena de suministro en las empresas exportadoras colombianas.

Rolón Ramírez, D. A. (2023). Transformación tecnológica en el modelo de gestión de inventarios. [Tesis de especialización, Universidad Nacional de Pilar]. Repositorio Institucional.

Siemens. (2024). *Machine learning for predictive maintenance in industrial operations. Siemens Annual Report 2024.*

Solución de inteligencia artificial perimetral. (2023). Solución de inteligencia artificial perimetral que ahorra recursos para el mantenimiento preventivo.

(S/f-b). Iop.org. Recuperado el 26 de marzo de 2025, de <https://iopscience.iop.org/article/10.1088/1742-6596/1085/4/042035>

Soori, M., Karimi Ghaleh Jough, F., Dastres, R., & Arezoo, B. (2025). Additive manufacturing modification by artificial intelligence, machine learning, and deep learning: A review. *Additive Manufacturing Frontiers*. <https://doi.org/10.1016/j.amf.2025.200198>

Tech Monitor. (2024). *Machine learning and RFID integration in logistics management. Tech Monitor Journal*, 19(2), 55–68.

Torres Estrella, P. E., Mendoza Arechua, G. R., & Ramírez Yépez, P. (2019). Control de los inventarios y su incidencia en los estados financieros. *Observatorio de la Economía Latinoamericana*, (6). <https://dialnet.unirioja.es/servlet/articulo?codigo=9003963>

UNESCO. (2021). *Recomendación sobre la Ética de la Inteligencia Artificial*.

París: Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura

Vega, C., Kfoury, E. F., Gomez, J., Pezoa, J. E., Figueroa, M., & Crichigno, J. (2024).

Machine learning controller for data rate management in science DMZ networks. *Computer Networks*, 242(110237), 110237. <https://doi.org/10.1016/j.comnet.2024.110237>

Villegas-Ch, W., Navarro, A. M., & Sanchez-Viteri, S. (2024). Validation of sentinel 2 based machine learning models for Czech National Forest Inventory. *Sistemas Inteligentes con Aplicaciones*, 24, 200438.

Wahedi, H. J., Heltoft, M., Christophersen, G. J., Severinsen, T., Saha, S., & Nielsen, I. E. (2023). Forecasting and inventory planning: An empirical investigation of classical and machine learning approaches for Svanehøj's future software consolidation. *Applied Sciences*, 13(15), 8581. <https://doi.org/10.3390/app13158581>