



Universidad Internacional de La Rioja
Escuela Superior de Ingeniería y
Tecnología

Máster Universitario en Inteligencia artificial

Integración de aprendizaje automático en software ERP

Trabajo fin de estudio presentado por:	Santiago Noé Bonay Leites
Tipo de trabajo:	Desarrollo software Comparativa de soluciones
Director/a:	Carlos Rubert Escuder
Fecha:	16/05/2024

Resumen

Los software de planificación de recursos empresariales (ERP) se han convertido en herramientas cruciales para la eficiencia y optimización de procesos en el entorno empresarial. Representan una solución integral que unifica y automatiza múltiples procesos de negocio en una única plataforma, englobando funciones críticas como finanzas, gestión de relaciones con clientes (CRM), abastecimiento, gestión de inventario, recursos humanos, producción y logística, entre otros. Al centralizar la información en una base de datos única, los ERP consiguen facilitar la gestión de la información que registra la empresa, consiguiendo datos de alta calidad en tiempo real. Integrar en el ERP soluciones de aprendizaje automático (ML) puede mejorar la productividad y la toma de decisiones, aprovechando la calidad de los datos almacenados. Sin embargo, la complejidad de implementar, adaptar y desplegar modelos de ML sigue suponiendo un desafío considerable. Este trabajo explora el uso de tecnologías de aprendizaje automático automatizado (AutoML) para superar estas barreras, desarrollando un cuadro de mando integrado en el ERP Libra con el objetivo de facilitar la implementación de algoritmos de clasificación automáticos. El objetivo es proporcionar un sistema que democratice el uso de ML en ERPs, haciéndolo accesible a usuarios no especializados. Para analizar la viabilidad y éxito del cuadro de mando desarrollado se han definido dos casos de uso de clasificación. El primero de los casos de uso se trata de una tarea de clasificación supervisada orientada a predecir el riesgo de impuntualidad en el pago de un cliente en el momento de realizar una venta. El segundo caso corresponde a clasificación no supervisada, cuyo objetivo será agrupar los productos en inventario en base a patrones de demanda y características de almacenaje para optimizar la gestión del inventario.

Palabras clave: ERP, aprendizaje automático, clasificación supervisada, clasificación no supervisada, AutoML

Abstract

Enterprise Resource Planning (ERP) software has become a crucial tool for efficiency and process optimization in the business environment. They represent a comprehensive solution that unifies and automates multiple business processes on a single platform, encompassing critical functions such as finance, customer relationship management (CRM), procurement, inventory management, human resources, production, and logistics, among others. By centralizing information in a single database, ERPs facilitate the management of the information recorded by the company, achieving high-quality real-time data. Integrating machine learning (ML) solutions into ERP can improve productivity and decision-making by leveraging the quality of the stored data. However, the complexity of implementing, adapting, and deploying ML models continues to pose a significant challenge. This work explores the use of automated machine learning (AutoML) technologies to overcome these barriers, developing an integrated dashboard in the Libra ERP with the goal of facilitating the implementation of automatic classification algorithms. The aim is to provide a system that democratizes the use of ML in ERPs, making it accessible to non-specialized users. To analyze the viability and success of the developed dashboard, two classification use cases have been defined. The first use case involves a supervised classification task aimed at predicting the risk of payment delinquency by a client at the time of making a sale. The second case corresponds to unsupervised classification, whose goal will be to group inventory products based on demand patterns and storage characteristics to optimize inventory management.

Keywords: ERP, machine learning, supervised classification, unsupervised classification, AutoML

Índice de contenidos

1.	Introducción	1
1.1.	Motivación	1
1.2.	Planteamiento del trabajo	2
1.3.	Estructura del trabajo	3
2.	Contexto y estado del arte	4
2.1.	Evolución del software ERP.....	4
2.2.	Machine Learning en ERP.....	5
2.3.	Clasificación supervisada	8
2.4.	Clasificación no supervisada.	10
2.5.	AutoML	11
2.6.	Conclusiones	13
3.	Objetivos concretos y metodología de trabajo	14
3.1.	Objetivo general.....	14
3.2.	Objetivos específicos	14
3.2.1.	Selección de algoritmos de ML	14
3.2.2.	Investigación y selección de librerías AutoML	14
3.2.3.	Desarrollo de cuadro de mando AutoML integrado en el ERP LIBRA	14
3.2.4.	Evaluación del resultado	14
3.3.	Metodología del trabajo	15
4.	Desarrollo específico de la contribución	18
5.	Conclusiones y trabajo futuro	19
5.1.	Conclusiones	19
5.2.	Líneas de trabajo futuro.....	19
	Bibliografía	20

Anexo A.	Código fuente y datos analizados	22
----------	--	----

Índice de figuras

Figura 1: Historia de los ERP	5
Figura 2: Aplicaciones de ML por módulo del ERP.....	7
Figura 3: Flujo de trabajo AutoML	12
Figura 4: Comparativa de soluciones AutoML	13
Figura 5: Etapas metodología CRISP-DM	16
Figura 6: Flujo de trabajo AGILE.....	16

Índice de tablas

Tabla 1: Comparación de algoritmos de clasificación supervisada.....	9
--	---

1. Introducción

1.1. Motivación

En el entorno empresarial actual, los sistemas de planificación de recursos empresariales (ERP) juegan un papel crucial en la optimización y eficiencia de los procesos operativos (Wieder, 2006). A pesar de su amplia adopción, la integración de soluciones avanzadas de aprendizaje automático (ML) en estos sistemas sigue siendo un desafío significativo, debido a la complejidad técnica y la necesidad de adaptación a contextos empresariales concretos (Godbole, 2023). Este trabajo se centra en abordar el problema de la accesibilidad y la implementación de técnicas de aprendizaje automático embebidas en el ERP mediante técnicas de aprendizaje automático automatizado (AutoML).

Se pretende aprovechar para ello el diseño de los software ERP, que se basan en la utilización de bases de datos relacionales que implementan el modelo ACID (*Atomicity, Consistency, Isolation, Durability*). Esto garantiza en gran medida una alta calidad de los datos registrados en el ERP, evitando duplicidad de datos y reduciendo la cantidad de datos erróneos o mal estructurados (Frank, 2008), lo que proporciona una base sólida para la explotación de los datos utilizando técnicas de aprendizaje automático

Mientras que los ERPs son capaces de gestionar y procesar grandes volúmenes de datos, muchas empresas no logran explotar completamente estas capacidades para mejorar la toma de decisiones basadas en datos y la automatización de tareas. Investigaciones previas han demostrado que la incorporación de ML puede transformar radicalmente la utilidad de los sistemas ERP, proporcionando un mayor conocimiento sobre los datos disponibles (Jawad, 2024). Sin embargo, existe una barrera tanto técnica como de coste para implementar soluciones de este tipo en pequeñas y medianas empresas (PyMEs), pues la puesta en marcha de soluciones de aprendizaje automático requieren una adaptación a las necesidades específicas de cada empresa, lo que supone un gasto en personal especializado o bien en servicios adicionales de consultoría externa (Juli, 2024). Para mitigar este hecho se pretende investigar la viabilidad de dotar al ERP de una serie de algoritmos estándar que puedan ser utilizados por la mayor parte de los clientes del ERP, aprovechando que todos comparten en gran medida la misma estructura de base de datos subyacente al ERP (Wieder, 2006).

En concreto, este trabajo busca ampliar la funcionalidad actual del ERP LIBRA¹, desarrollado por la empresa EDISA, que considera primordial mantener sus productos actualizados con las tendencias recientes en el ámbito del desarrollo de software, donde la inteligencia artificial destaca en los últimos años. EDISA pretende mantener LIBRA como un producto competitivo frente a ERPs desarrollados por empresas de un tamaño muy superior, como Oracle, Microsoft, SAP, Sage o Infor, que ya implementan o están desarrollando soluciones con técnicas de inteligencia artificial² (Josyula & Godbole, 2024). En este sentido el dotar a LIBRA de un cuadro de mando de aprendizaje automático puede suponer una ventaja competitiva frente a otros ERP y una base sobre la que incorporar nuevos algoritmos en el futuro.

1.2. Planteamiento del trabajo

El presente trabajo busca desarrollar un cuadro de mando de aprendizaje automático dentro del ERP LIBRA. En dicho cuadro de mando se permitirá a los usuarios avanzados del ERP, seleccionar un origen de datos previamente definido con las herramientas ya existentes en el ERP (*dataset*), una tarea a realizar sobre dichos datos (clasificación supervisada o no supervisada), y ajustar parámetros adicionales en función de la tarea seleccionada.

En el caso de seleccionar una tarea de clasificación supervisada, se deberá indicar cuál es la variable objetivo. Para las tareas de clasificación no supervisada se podrá indicar en cuántos grupos se quiere segmentar el *dataset*. Este enfoque busca una interfaz de usuario sencilla, donde cualquier tipo de usuario experto en el ERP, sea capaz de generar un modelo de ML en pocos pasos.

El proceso de AutoML recibirá el *dataset*, el tipo de tarea a realizar y los parámetros y en base a ello llevará a cabo las operaciones de pre-procesamiento de datos, selección de modelo o modelos ML a utilizar, *tuning* de hiperparámetros de cada modelo y evaluación de los resultados con el conjunto de datos de prueba. Finalmente el modelo con mejor resultado se guardará y podrá ser utilizado para la clasificación de nuevos registros dentro del ERP.

A modo de prueba y validación se llevará a cabo un caso de uso de cada tipo de clasificación, valorando la facilidad de uso, la calidad de los resultados, tiempos de desarrollo y

¹ <https://www.edisa.com/>

² <https://medium.com/@top10erp/7-top-erp-systems-with-artificial-intelligence-features-7dcaeb4ac756>

entrenamiento del modelo, y los requisitos hardware necesarios para llevar a cabo todo el proceso.

1.3. Estructura del trabajo

Este trabajo se estructura en cinco secciones principales que describen el desarrollo y evaluación de la integración de soluciones AutoML en el ERP LIBRA. La sección 2 “Contexto y estado del arte” proporciona una revisión de los usos actuales de algoritmos de aprendizaje automático en varios sistemas ERP del mercado, así como un análisis del estado del arte de técnicas de clasificación de machine learning y soluciones de AutoML. Este análisis se centra específicamente en identificar y describir los problemas de clasificación, que constituyen el foco inicial de este estudio.

En la sección 3 “Objetivos concretos y metodología de trabajo” se establecerán los objetivos concretos del estudio y el alcance del mismo, así como la metodología de trabajo adoptada para llevarlo a cabo. Dichos objetivos se van a desarrollar y describir en detalle en la sección 4 “Desarrollo específico de la contribución”.

Por último, en el apartado 5 “Conclusiones y trabajo futuro”, se analizan los resultados obtenidos en los escenarios de pruebas y se detallan las conclusiones del estudio y posibles ampliaciones futuras.

2. Contexto y estado del arte

2.1. Evolución del software ERP

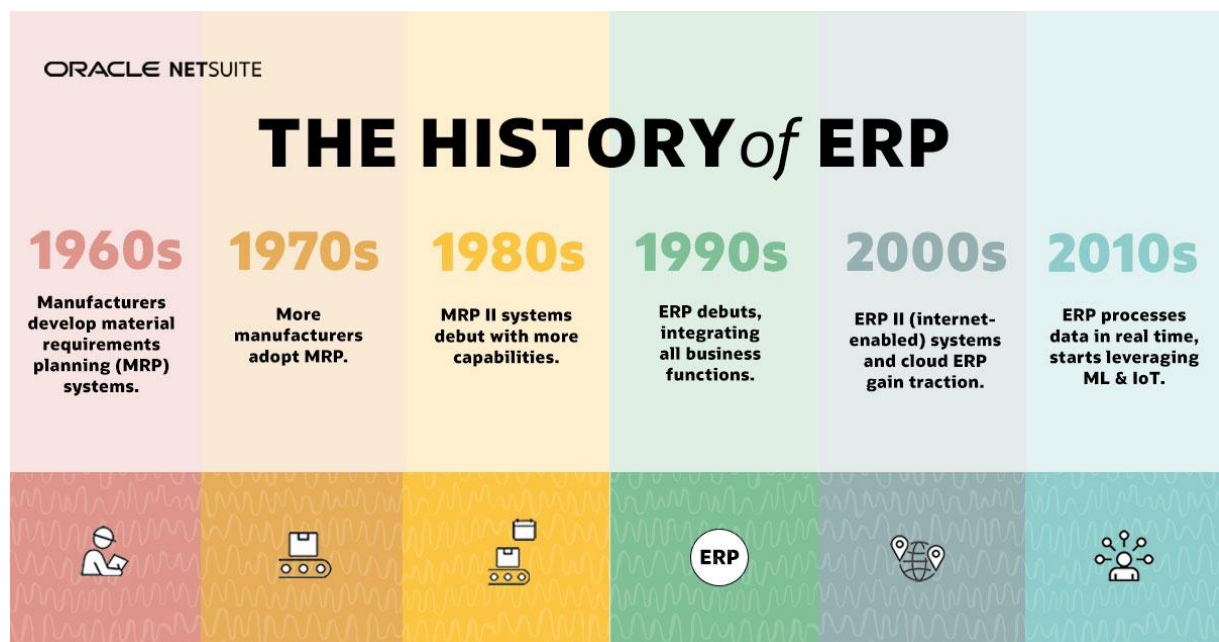
El origen de los ERP procede de los softwares de planificación de materiales de producción (MRP) surgidos en la década de 1960 para tratar la problemática existente en las fábricas respecto a la optimización de su cadena de suministro de materias primas y cálculo de necesidades (Katu, 2020). Como evolución de estos sistemas surgen los software denominados *Manufacturing Resource Planning* (MRP II) para gestionar la capacidad productiva disponible ajustándola a la planificación de recursos sugerida por el MRP. Estos softwares tuvieron un considerable éxito en empresas productivas de diversas industrias, lo que llevó a la paulatina aparición de diferentes módulos que añadían funcionalidades adicionales a los MRP. En la década de 1990 termina por imponerse la denominación ERP para los sistemas integrados de gestión empresarial, que añadían a la tradicional funcionalidad de la producción, almacenes e ingeniería otras áreas de la empresa como la contabilidad, finanzas, ventas o recursos humanos. A la par que los sistemas ERP surgen una serie de sistemas especializados en diversas áreas de negocio, como pueden ser los CRM (*Customer Relationship Management*) para la gestión de la relación con los clientes, SCM (*Supply Chain Management*) para gestionar la cadena de suministro, BPM (*Business Process Management*) para controlar los flujos de negocio o HCM (*Human Capital Management*) para la gestión de recursos humanos. La tendencia durante las dos últimas décadas ha sido la de integrar todas estas características dentro de los ERP, convirtiéndose en soluciones de software multidimensionales, constituidas por múltiples módulos o funcionalidades, cada una de las cuales atiende una función empresarial diferente (Jawad, 2024).

Los ERP han pasado por tanto a convertirse en una plataforma que unifica gran parte de los procesos empresariales, siendo el sistema que genera y almacena los datos de negocio de forma centralizada e interrelacionada. El siguiente paso en la evolución de los ERP fue expandir su conectividad con otros sistemas, para aprovechar el crecimiento de internet, comercio electrónico e intercambio de información con clientes, proveedores, bancos o entidades gubernamentales. Desde la década de 2010 en adelante, la tendencia ha sido la de convertirse en soluciones software basadas en entorno Cloud y ofrecer accesibilidad desde todo tipo de dispositivos móviles (Ularu, Puican, Suciu, Vulpe, & Todoran, 2013), lo que ha

permitido tanto una reducción de costes de infraestructura como la posibilidad de interactuar con el ERP desde móviles o tabletas, lo cual mejora la interacción en tiempo real con el sistema, permitiendo registrar y consultar los datos de la empresa en tiempo real desde cualquier localización.

En la actualidad la incorporación de ML en los sistemas ERP surge del crecimiento del entorno empresarial, caracterizado por el *big data*, el internet de las cosas (IoT), y la necesidad de tomar decisiones informadas en base a los datos (Wahab & Nor, 2023).

Figura 1: Historia de los ERP



Fuente: www.netsuite.com

2.2. Machine Learning en ERP

El aprendizaje automático es un subconjunto de la inteligencia artificial que permite a los sistemas informáticos aprender sobre un conjunto de datos aportado, sin necesidad de programación explícita. Las técnicas de ML son capaces de procesar grandes volúmenes de datos de forma precisa y rápida, pudiendo conectarse con las fuentes de datos que proporcionan los ERP. También sacan provecho de los avances en potencia de computación, sistemas distribuidos y computación en la nube que permiten construir sistemas complejos y de alta capacidad computacional si así lo requiere el volumen de datos o caso de uso (Jawad, 2024).

Los algoritmos de ML se han convertido en un paradigma que puede transformar cómo se perciben los sistemas ERP, convirtiéndolos en herramientas inteligentes, que apoyen la toma de decisiones en base al conocimiento extraído de los datos y permitan la automatización de procesos o tareas repetitivos, mejorando la productividad de la empresa (M3Systems, 2023).

Las principales empresas desarrolladoras de ERP a nivel mundial como Oracle³, SAP⁴ y Microsoft⁵, están integrando de forma progresiva técnicas de ML e inteligencia artificial en sus sistemas para mejorar diversas funcionalidades. Las principales áreas de aplicación para estas tecnologías son el análisis predictivo, análisis de datos, apoyo en la toma de decisiones y automatización de procesos internos del ERP, siendo aplicables en los principales módulos de los ERP (Godbole, 2023).

En la gestión de almacén e inventario son especialmente relevantes los modelos predictivos de demanda, que pueden adaptarse para tener en cuenta múltiples variables como las tendencias de consumo, precios de compra, o tiempos de reabastecimiento. Las técnicas de inteligencia artificial también pueden ayudar a optimizar el espacio del almacén y el envío de mercancía a los clientes (Godbole, 2023) (Qaffas, Ben HajKacem, Nasraoui, & Ben Ncir, 2023).

En cuanto al trato con los clientes de una empresa que utiliza un ERP, podemos encontrar aplicaciones para ML tanto en la gestión de las ventas, relación con los clientes o campañas de marketing. Los modelos de ML se pueden aprovechar tanto de información inherente al ERP como el historial de compras e interacciones de los clientes ya existentes lo que permite la categorización de los mismos, y también pueden obtener información del exterior, por ejemplo en redes sociales, para detectar posibles perfiles de clientes o tendencias de consumo a explotar (Kauppala, 2022).

En el módulo de recursos humanos, se han aplicado técnicas de clasificación automática de aplicaciones de candidatos a ofertas de trabajo, planificación de planes formativos dentro la empresa o ayuda en la redacción y publicación de ofertas laborales (Strang & Sun, 2022).

En cuanto a los procesos de planificación de la producción, destaca un conjunto de nuevas tecnologías a aplicar en el ámbito industrial (Industria 4.0), que incluye el uso de sistemas ERP,

³ <https://www.oracle.com/applications/fusion-ai/>

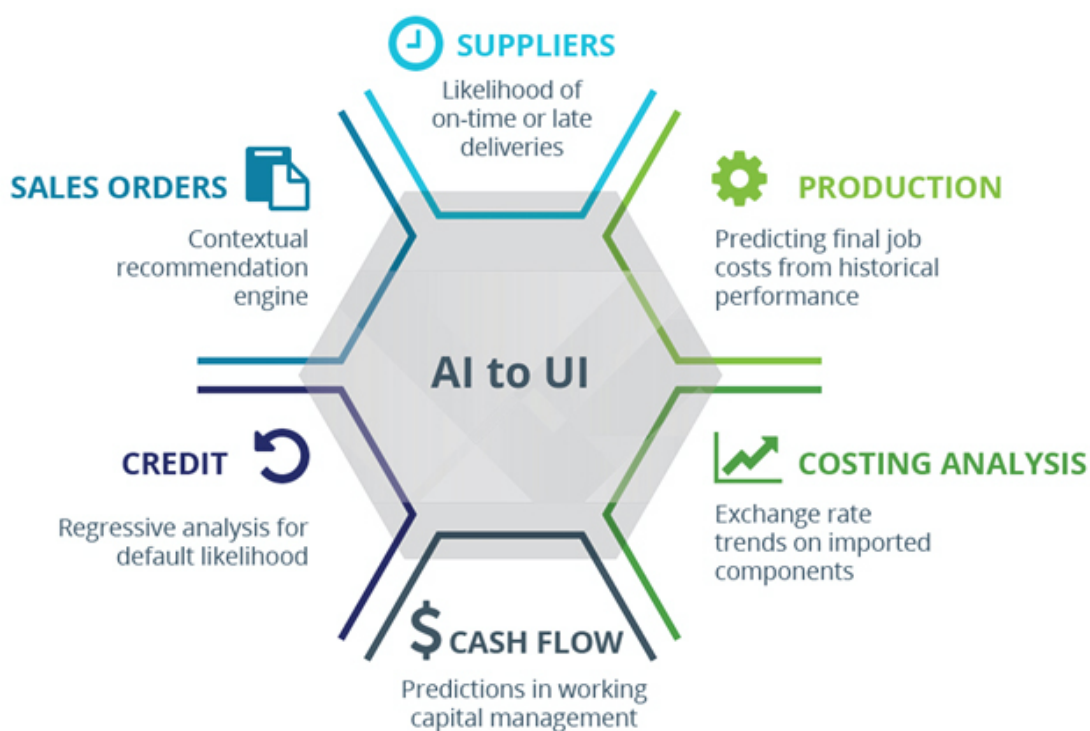
⁴ <https://www.sap.com/spain/products/artificial-intelligence.html>

⁵ <https://www.microsoft.com/es-es/dynamics-365/solutions/ai>

IoT, machine learning, automatización y robótica (Stadnicka, y otros, 2022). Con el uso de datos históricos de los procesos productivos de una empresa, se pueden crear modelos para monitorizar los costes de producción presentes, planificación de producción predictiva, detectar ineficiencias en la cadena de producción o prevenir problemas de abastecimiento de materias primas.

El módulo de los sistemas ERP que utilizan todas las empresas independientemente de su área de negocio es el de finanzas y contabilidad. En este ámbito podemos destacar la aplicación de modelos de clasificación de riesgo crediticio, detección de fraude en transacciones económicas o apuntes contables, análisis predictivo aplicado a balances contables o flujo de caja (Josyula & Godbole, 2024).

Figura 2: Aplicaciones de ML por módulo del ERP



Fuente: <https://www.syspro.com/product/business-digitalization/artificial-intelligence/>

Finalmente, dentro de las técnicas de inteligencia artificial utilizadas en los ERP, se deben destacar las soluciones de visión computacional basadas en inteligencia artificial, por ejemplo para captura y procesamiento de documentos (OCR), recuentos de inventario mediante imágenes o control de calidad mediante detección de anomalías. La tendencia actual pasa por implementar técnicas de inteligencia artificial generativa (GenAI) para aprovechar el auge de

los grandes modelos de lenguaje (LLM), aplicadas en *bots* conversacionales, generación de resúmenes de documentación o apoyo en la generación de textos.

En este estudio nos enfocaremos en algoritmos de machine learning tradicionales debido a su mayor interpretabilidad, eficiencia computacional y facilidad de implementación, características especialmente valiosas para democratizar el uso de ML en el entorno empresarial. Entre todas las técnicas de ML, nos centraremos en las tareas de clasificación que se pueden dividir en supervisada y no supervisada.

2.3. Clasificación supervisada

La clasificación supervisada es una técnica de aprendizaje automático que permite a los sistemas informáticos categorizar un conjunto de datos basándose en ejemplos previos que ya estaban correctamente etiquetados (Soofi & Awan, 2017). En un ERP pueden utilizarse para múltiples propósitos: segmentación de entidades: clientes (según un rating), proveedores (calidad de las compras, entregas y precios), artículos (índice de rotación de inventario) o candidatos a un puesto de trabajo (apto, no apto), clasificación de gastos e ingresos en las cuentas contables apropiadas o incluso para la detección de fraude en transacciones financieras.

Los algoritmos de clasificación supervisada clásica más importantes son:

- **Árboles de decisión (DT).** Estructuran las decisiones en formato de árbol, donde cada nodo del árbol se segmenta en base a una condición que busca obtener un subconjunto de datos homogéneo. Los nodos hoja de estos árboles representan las diferentes categorías en las que se segmentarán los datos. Son altamente interpretables, pues permiten conocer en base a qué criterio se ha determinado la categoría de cada registro. Permiten el manejo de variables o *features* tanto numéricas como categóricas (Soofi & Awan, 2017).
- **Random Forest (RF).** Técnica de *ensemble* que combina múltiples árboles de decisión, cada uno de los cuales ha tenido en cuenta diferentes *features*, para reducir el riesgo de sobreajuste (*overfitting*) y mejorar la precisión de la clasificación. Como desventaja principal frente a los árboles de decisión es que son menos interpretables, al utilizar múltiples árboles de decisión y ponderar los *outputs* de todos ellos (Ding, Huang, Wei, Tang, & Liu, 2021).

- **XGBoost (eXtreme Gradient Boosting)**. Se trata de una implementación optimizada de métodos de *boosting* de gradientes, diseñada para ser altamente eficiente, flexible y escalable. Puede trabajar con datos numéricos, categóricos y tratar datos faltantes. Incorpora técnicas de regularización para prevenir el overfitting. Al igual que Random Forest, sus resultados pueden ser complejos de interpretar al tratarse de un método de *ensemble* (Ren & Wang, 2023).
- **Máquinas de Soporte Vectorial (SVM)**. Busca el hiperplano óptimo para la separación de los datos en las diferentes clases posibles, empleando diferentes *kernels* para adaptarse a diferentes conjuntos de datos. Es especialmente útil para clasificación en espacios de alta dimensionalidad o datos no lineales (ElMadany, Alfonse, & Aref, 2021).
- **K Nearest Neighbours (K-NN)**. Se busca clasificar los diferentes registros en base a sus k vecinos más cercanos, siendo k un valor numérico a ajustar para mejorar la precisión del algoritmo (Soofi & Awan, 2017).
- **Naive Bayes**. Técnica estadística basada en el teorema de Bayes. Es efectivo cuando las diferentes variables del conjunto de datos son independientes. Utilizado ampliamente en clasificación de textos (Soofi & Awan, 2017).

Tabla 1: Comparación de algoritmos de clasificación supervisada.

Algoritmo	Interpretabilidad	Precisión	Escalabilidad	Uso Común
DT	Alta	Media	Media	Clasificación simple
RF	Baja	Alta	Alta	Clasificación compleja
XGBoost	Baja	Muy alta	Muy alta	Predicciones rápidas
SVM	Media	Alta	Media	Alta dimensionalidad
K-NN	Alta	Media	Baja	Clasificación simple
Naive Bayes	Alta	Media	Alta	Clasificación de texto

Fuente: Elaboración propia. Datos obtenidos de (Soofi & Awan, 2017), (ElMadany, Alfonse, & Aref, 2021), (Kauppala, 2022)

En este estudio se utilizará un conjunto de datos de ventas y cobros de los clientes de la empresa, para tratar de clasificar las nuevas ventas a los clientes según el riesgo de que el cliente no cumpla con el plazo de pago establecido. Evaluaremos diferentes algoritmos y cómo se integran en soluciones de AutoML, para valorar cuáles son más idóneos para la integración con el ERP LIBRA.

2.4. Clasificación no supervisada.

Los algoritmos de clasificación no supervisada se enfocan en identificar patrones o agrupaciones en un conjunto de datos que no está previamente etiquetado. Al contrario que en la clasificación supervisada, no conocemos a priori las categorías en las que se van a clasificar los registros que componen un determinado *dataset* (Iqbal, Elahi, & Shahzad, 2022). En el ámbito de un ERP pueden utilizarse para las mismas tareas de clasificación o detección de anomalías indicadas en el punto anterior, pero cuando no existe una variable objetivo en el conjunto de datos, es decir, se desconoce el patrón subyacente a los datos y el número de grupos a obtener.

Como algoritmos de clasificación no supervisada de amplio uso tenemos:

- **K-Means.** Tiene como objetivo dividir el dataset en k grupos o *clusters*, de manera que cada registro del conjunto de datos pertenezca al grupo con el centroide más cercano. Es importante definir qué valor de k es el idóneo, para lo cual existen varios métodos, siendo muy popular el método del codo (*Elbow*). Es efectivo con conjuntos de datos de tamaño moderado, aunque su rendimiento puede disminuir con datasets de alta dimensionalidad (Iqbal, Elahi, & Shahzad, 2022).
- **Clustering jerárquico.** Busca crear una jerarquía de clusters. Existen métodos que parten de los datos individuales y los van combinando creando clusters (aglomerativo o *bottom-up*) y los que parten de un único cluster que se va dividiendo en nuevos clusters más pequeños (divisivo o *top-down*). Este método no requiere establecer de antemano el número de clusters objetivo como K-Means (Sastry, Babu, & Prasada, 2013).
- **DBSCAN (Density-Based Spatial Clustering of Applications with Noise).** Permite identificar clusters que tienen formas arbitrarias y tiene un buen manejo de outliers. Se deben definir dos parámetros básicos, 'eps' que es la distancia máxima para

considerar que dos puntos son vecinos y 'minPts' que indica el mínimo número de puntos que debe tener un cluster. Es efectivo para datos con clusters con una densidad variable y es robusto frente a *outliers* (Iqbal, Elahi, & Shahzad, 2022).

En este estudio utilizaremos estas técnicas para la clasificación de artículos en base a su demanda, rotación y espacio de almacenamiento, para obtener una categorización de los mismos que sirva como base para la optimización del inventario del almacén.

2.5.AutoML

El ML automático o AutoML, se refiere al conjunto de técnicas y procesos que buscan automatizar los pasos que habitualmente siguen los humanos para construir un modelo de ML que se ajuste de forma óptima a un determinado conjunto de datos (Ferreira, Pilastri, Martins, Pires, & Cortez, 2021). En el área de machine learning se suelen seguir una serie de pasos secuenciales que tienen una gran parte de ensayo y error, siendo habitual el realizar varias iteraciones y ajustes en el proceso para lograr el mejor modelo posible para cada caso concreto.

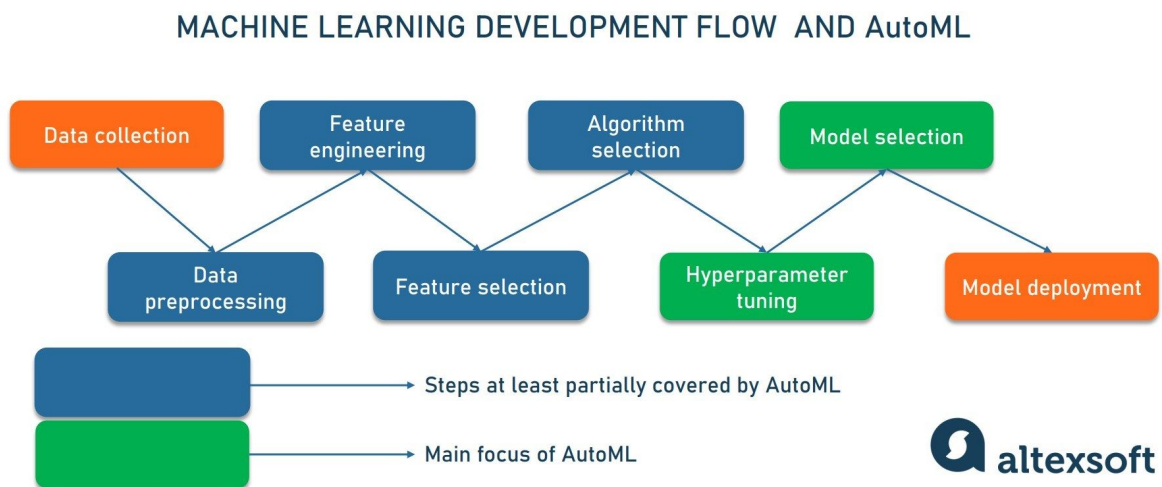
- **Análisis exploratorio de datos (EDA).** Consiste en hacer un estudio del dataset y aplicar ciertas técnicas, como tratar los valores faltantes, realizar la codificación de variables categóricas, normalizar las variables numéricas o seleccionar las características más relevantes para el entrenamiento del modelo. Existen librerías que se encargan de realizar un EDA automático sobre el dataset, como DataRobot o H2O.ai (LeDell & Poirier, 2020).
- **Selección de modelos.** Se pueden probar varios modelos diferentes para detectar cuál se ajusta mejor a un conjunto de datos determinado. La experiencia de un profesional del machine learning aporta conocimiento respecto a qué modelos pueden funcionar mejor, acotando el conjunto de modelos a validar. Con técnicas de AutoML, el propio proceso se encargará de probar múltiples modelos hasta encontrar el más adecuado (Waring, Lindvall, & Umeton, 2020).
- **Optimización de hiperparámetros.** Para cada uno de los modelos elegidos, es necesario hacer un ajuste de los hiperparámetros de los que hace uso cada modelo. Este es también un proceso iterativo de ajuste de dichos parámetros hasta encontrar la mejor combinación posible. AutoML ayuda con este proceso empleando técnicas de

búsqueda aleatoria (*random search*) o búsqueda en cuadrícula (*grid search*) y otras técnicas específicas de cada librería de AutoML (Real, Liang, So, & Le, 2020).

- **Evaluación de los modelos.** Para decidir qué modelos y qué ajustes son mejores debemos contar con una serie de métricas que nos permitan evaluar el desempeño de cada uno. Para esto se suele dividir el conjunto de datos entre un conjunto de datos de entrenamiento y otro para evaluar el modelo, de forma que podamos usar una parte de los datos originales para medir la precisión del modelo entrenado. Existen múltiples métricas a tener en cuenta según si la tarea a realizar es de regresión o clasificación (LeDell & Poirier, 2020).

Las soluciones de AutoML automatizan este proceso de selección, optimización y evaluación de modelos, realizando múltiples pruebas y seleccionando el modelo con un mejor comportamiento (ver Figura 3).

Figura 3: Flujo de trabajo AutoML



Fuente: <https://www.altexsoft.com/blog/automl>

Podemos diferenciar entre soluciones empresariales que ofrecen un entorno de desarrollo de machine learning, como Google Cloud AutoML, Oracle AutoML, Azure Machine Learning AutoML, IBM AutoML o AWS SageMaker Autopilot y librerías de código abierto que permiten desarrollar un entorno propio de machine learning automatizado como H2O, Auto-Sklearn, TPOT o PyCaret (Ferreira, Pilastrri, Martins, Pires, & Cortez, 2021). En este estudio nos

centraremos en soluciones open source para construir una solución de AutoML integrada en el ERP LIBRA.

Figura 4: Comparativa de soluciones AutoML

AutoML PRODUCTS COMPARED					
	Classification & regression	Time series forecasting	Neural networks and NAS	NLP	Computer vision
Tech giants					
Google Vertex AI	✓	✓	✓	✓	✓
Amazon SageMaker Autopilot	✓	✓	Limited to one option: MLP	✗	✗
Microsoft Azure AutoML	✓	✓	✓	✓	In preview
IBM Watson AutoAI	✓	✓	✗	✗	✗
End-to-end ML platforms					
H2O Driverless AI	✓	✓	NNs with default values	✓	✓
DataRobot	✓	✓	Automatically pre-tailored NNs	✓	✓
DataBricks	✓	✓	✗	✗	✗
Open source AutoML tools					
AutoGluon (Amazon)	✓	✗	✓	✓	✓
FLAML (Microsoft)	✓	✓	✗	✗	✗
H2O-3 AutoML	✓	✗	✗	✓	✗
Auto-Sklearn	✓	✗	✗	✗	✗
AutoKeras	✓	✓	✓	✓	✓
MLBox	✓	✗	✗	✗	✗



Fuente: <https://www.altexsoft.com/blog/automl>

2.6.Conclusiones

El futuro de los software ERP pasa por aplicar diferentes técnicas de inteligencia artificial, para ofrecer a las empresas usuarias de estos sistemas una herramienta que les permita ser más productivos y competitivos mediante la explotación de los datos internos nativos del ERP. La integración de técnicas de AutoML en un sistema ERP puede representar un avance sustancial en la democratización del uso de inteligencia artificial en entornos empresariales,

especialmente en empresas con recursos limitados, que no cuenten con personal especializado en el tratamiento y explotación de datos mediante machine learning.

3. Objetivos concretos y metodología de trabajo

3.1. Objetivo general

El objetivo principal de este trabajo es desarrollar un cuadro de mando de machine learning automatizado (AutoML) integrado en el ERP LIBRA, que permita la creación de algoritmos de clasificación de manera sencilla y efectiva, de forma que posteriormente se puedan utilizar estos modelos dentro del ERP.

3.2. Objetivos específicos

3.2.1. Selección de algoritmos de ML

Investigar y seleccionar los algoritmos que mejor se comportan con los *datasets* obtenidos de diversas bases de datos del ERP LIBRA en tareas de clasificación supervisada y no supervisada. Estos *datasets* tendrán la misma estructura de datos en todos los casos, pero variará el contenido de los mismos.

3.2.2. Investigación y selección de librerías AutoML

Evaluar las diferentes opciones de librerías open source existentes de AutoML en combinación con los algoritmos seleccionados previamente y los *datasets* disponibles.

3.2.3. Desarrollo de cuadro de mando AutoML integrado en el ERP LIBRA

Después de seleccionar una de las librerías AutoML e investigar las opciones que ofrece, se desarrollará un cuadro de mando dentro del ERP LIBRA que permita la interacción sencilla con la librería.

El diseño de los *datasets* a utilizar se realizará con herramientas ya existentes dentro del ERP, que cuenta con diseñador de informes que permite la realización de consultas SQL complejas sobre la base de datos del ERP.

3.2.4. Evaluación del resultado

Se realizarán pruebas con dos casos de uso específicos:

- Clasificación de clientes en función del riesgo de impuntualidad en el pago, utilizando algoritmos de clasificación supervisada.
- Clasificación de productos según rotación y espacio que ocupan en el inventario, empleando algoritmos de clasificación no supervisada.

Evaluaremos la facilidad de uso, calidad de los resultados, tiempos de desarrollo y entrenamiento de los modelos y requisitos de hardware necesarios para el despliegue en el entorno del ERP.

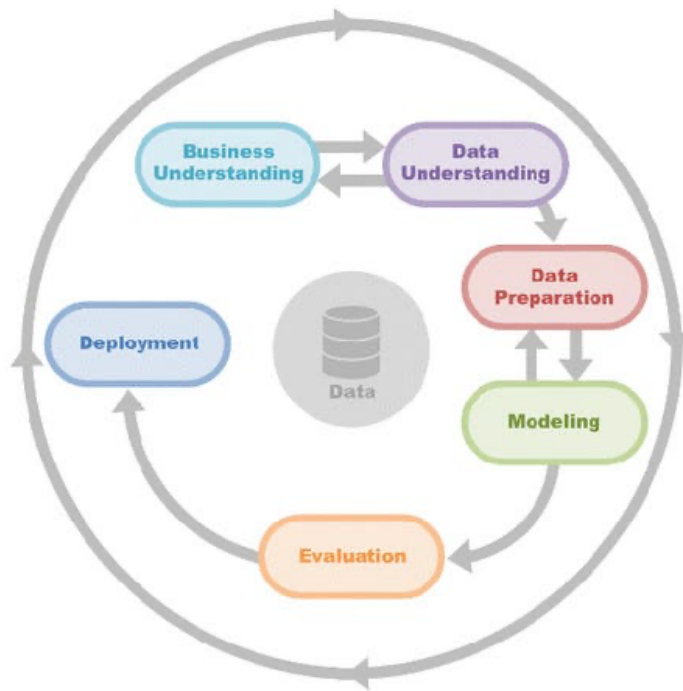
3.3. Metodología del trabajo

Para alcanzar los objetivos propuestos, se adoptará una combinación de la metodología CRISP-DM con principios de desarrollo AGILE para adaptarse a las necesidades específicas de un desarrollo de machine learning que también engloba un apartado de desarrollo de software tradicional (Gerhart, Torres, & Giddens, 2023). CRISP-DM se basa en cuatro pasos (ver Figura 5):

- Comprensión del negocio. Definir los objetivos junto a expertos en el dominio y *stakeholders*.
- Entendimiento de los datos. Obtener, explorar y catalogar los datos disponibles.
- Preparación de los datos. Limpieza y depuración de los datos disponibles para su preparación para su utilización posterior en el entrenamiento de modelos.
- Modelado de datos. Aplicación de uno o varios algoritmos sobre los datos ya preparados.

CRISP-DM se ajusta mejor que otras metodologías a los proyectos de machine learning dada su orientación a los datos.

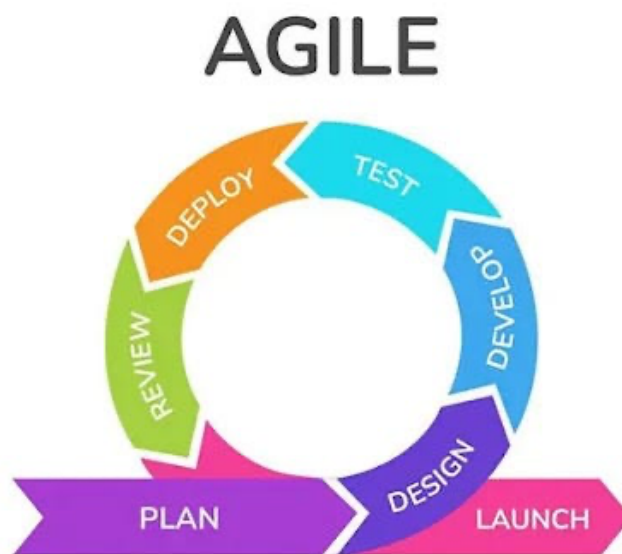
Figura 5: Etapas metodología CRISP-DM



Fuente: Medium.com

Como en este proyecto también se abordan otro tipo de tareas, emplearemos también parte de la metodología AGILE para definir *sprints* en los que abordar las tareas necesarias para el cuadro de mando y su integración en LIBRA de manera iterativa.

Figura 6: Flujo de trabajo AGILE



Fuente: Medium.com

Se estructurará el desarrollo en varias fases iterativas:

1. **Estudio y comprensión de las necesidades propias de un ERP.** Establecer junto a personal de la empresa desarrolladora del ERP LIBRA los objetivos, entregables y criterios de éxito del proyecto.
2. **Preparación y obtención de los *datasets*.** Creación de la estructura de los *datasets* para los dos casos de uso a estudiar. Una vez definidos, se obtendrán N *datasets* de diferentes bases de datos. Los datos deberán estar total anonimizados para descartar cualquier posibilidad de tratar datos reales de alguna entidad.
3. **Evaluación de diferentes modelos de ML.** En primer lugar se realizará un entrenamiento con los *datasets* obtenidos para seleccionar qué modelos se comportan mejor con las estructuras de datos obtenidas del ERP.
4. **Investigación y pruebas de librerías AutoML.** Investigación de las diferentes librerías open source disponibles y su adecuación a los *datasets* obtenidos y algoritmos seleccionados. Se realizarán pruebas con una selección de las librerías que se consideren más adecuadas, para evaluar su desempeño.
5. **Desarrollo del cuadro de mando integrado en el ERP LIBRA.** Se creará una interfaz desde la cual poder ejecutar el entrenamiento de un modelo de ML que haga uso de un *dataset* seleccionado por el usuario e invoque a la librería de AutoML. En este punto se determinará qué opciones de configuración se le ofrecerá a los usuarios, por ejemplo la posibilidad de establecer ciertos hiperparámetros o un tiempo máximo determinado para el cálculo del modelo. Esta sección es la más específica de desarrollo de software tradicional, se abordará mediante *sprints* de duración semanal.
6. **Despliegue y evaluación de resultados.** Se desplegará el cuadro de mando en un entorno controlado y se ejecutará con los *datasets* de dicha base de datos en concreto, para evaluar tanto la facilidad de uso, tiempos de entrenamiento y calidad de los resultados obtenidos.

4. Desarrollo específico de la contribución

Tipo 2. Desarrollo de software

Tipo 3. Comparativa de soluciones

5. Conclusiones y trabajo futuro

5.1. Conclusiones

5.2. Líneas de trabajo futuro

Bibliografía

- Wieder, B. B. (2006). The impact of ERP systems on firm and business process performance. *Journal of Enterprise Information Management*, 19(1), 13-29.
- Godbole, M. (2023). Revolutionizing Enterprise Resource Planning (ERP) Systems through Artificial Intelligence. *International Numeric Journal of Machine Learning and Robots*, 7(7), 1-15.
- Frank, L. (2008). Smooth and flexible ERP migration between both homogeneous and heterogeneous ERP systems/ERP modules. *Proceedings of the 2nd Workshop on 3rd Generation Enterprise Resource Planning Systems, Copenhagen business School*.
- Katuu, S. (2020). Enterprise Resource Planning: Past, Present, and Future. *New Review of Information Networking*, 25(1), 37-46.
- Jawad, Z. B. (2024). Machine learning-driven optimization of enterprise resource planning (ERP) systems: a comprehensive review. *Beni-Suef University Journal of Basic and Applied Sciences*, 13(1), 1.
- Juli, M. (2024). AI-Powered ERP: Revolutionizing Usability and Innovation in Enterprise Resource Planning. *EasyChair*, No.12750.
- Josyula, H., & Godbole, M. (2024). Navigating The Future: A Comprehensive Analysis of AI, ML, ERP, And Oracle Integration in Financial Digital Transformation Article. *International Journal of Computer Engineering & Technology*, 61-70.
- Wahab, N., & Nor, R. (2023). The Role of Enterprise Resource Planning (ERP) Systems in Facilitating Sustainable Business Practices. *AI, IoT and the Fourth Industrial Revolution Review*, 29-39.
- Ularu, E., Puican, F., Suci, G., Vulpe, A., & Todoran, G. (2013). Mobile Computing and Cloud maturity - Introducing Machine Learning for ERP Configuration Automation. *Informatica Economică*, 17, 40-52.
- M3Systems. (Mayo de 2023). *m3.systems*. Obtenido de <https://m3.systems/understanding-the-role-of-ai-and-machine-learning-in-erp-systems/>

- Yathiraju, N. (2022). Investigating the use of an Artificial Intelligence Model in an ERP Cloud-Based System. *International Journal of Electrical, Electronics and Computers*, 1-26.
- Kauppala, J. (2022). Classifying customer companies in an enterprise resource planning system using machine learning methods. *Lappeenranta-Lahti University of Technology LUT*.
- Qaffas, A. A., Ben HajKacem, M. -A., Nasraoui, O., & Ben Ncir, C. -E. (2023). Explainable Artificial Intelligence Approach for Multi-Criteria ABC Item Classification. *J. Theor. Appl. Electron. Commer.*, 848-866.
- Strang, K. D., & Sun, Z. (2022). ERP Staff versus AI recruitment with employment real-time big data. *Discov Artif Intell*.
- Stadnicka, D., Sęp, J., Amadio, R., Mazzei, D., Tyrovolas, M., Stylios, C., . . . Navarro, J. (2022). Industrial Needs in the Fields of Artificial Intelligence, Internet of Things and Edge Computing. *Sensors*, 22(12).
- Soofi, A. A., & Awan, A. (2017). Classification Techniques in Machine Learning: Applications and Issues. *Journal of Basic & Applied Sciences*, 459-465.
- ElMadany, H., Alfonse, M., & Aref, M. (2021). A Proposed Approach for Production in ERP Systems Using Support Vector Machine algorithm. *International Journal of Intelligent Computing and Information Services*, 49-58.
- Ding, B. Y., Huang, B., Wei, J., Tang, Y., & Liu, C. (16 de 11 de 2021). Enterprise Risk Assessment Based on Machine Learning. *Computational Intelligence and Neuroscience*.
- Ren, Q., & Wang, J. (2023). Research on Enterprise Digital-Level Classification Based on XGBoost Model. *Sustainability*, 15(2699).
- Iqbal, T., Elahi, A., & Shahzad, A. (2022). Exploring Unsupervised Machine Learning Classification Methods for Physiological Stress Detection. *Frontiers in Medical Technology*.
- Sastry, S. H., Babu, P., & Prasada, M. S. (2013). Analysys & Prediction of Sales Data in SAP-ERP System using Clustering Algorithms. *International Journal of Computational Science and Information Technology*, 1312.2678.

- Ferreira, L., Pilastri, A., Martins, C. M., Pires, P. M., & Cortez, P. (2021). A Comparison of AutoML Tools for Machine Learning, Deep Learning and XGBoost. *International Joint Conference on Neural Networks*, 1-8.
- LeDell, E., & Poirier, S. (2020). H2O AutoML: Scalable Automatic Machine Learning. *7th ICML Workshop on Automated Machine Learning*.
- Waring, J., Lindvall, C., & Umeton, R. (2020). Automated machine learning: Review of the state-of-the-art and opportunities for healthcare. *Artificial Intelligence in Medicine*.
- Real, E., Liang, C., So, D. R., & Le, Q. V. (2020). AutoML-Zero: Evolving Machine Learning Algorithms From Scratch. *37th International Conference on Machine Learning*.
- Gerhart, N., Torres, R., & Giddens, L. (2023). Towards and Agile CRISP-DM Process. *Association for Information Systems*.

Anexo A. Código fuente y datos analizados