Limpieza, transformación e integración de datos para modelos predictivos.

Breve descripción:

Este componente fortalece las competencias técnicas para analizar, validar y preparar datos, facilitando su conversión en conocimiento útil. Prepara a los aprendices para procesar información en entornos digitales dinámicos, con enfoque en su aplicación en modelos de Inteligencia Artificial (IA).

Junio 2025

Tabla de contenido

Introducción

En un entorno donde la inteligencia artificial y el análisis de datos son protagonistas, la calidad y preparación de los datos adquiere una importancia equivalente a la de los algoritmos utilizados. Los datos bien gestionados permiten construir modelos capaces de predecir, clasificar o generar información relevante para múltiples sectores productivos y sociales.

Este componente formativo desarrolla la comprensión y aplicación de técnicas esenciales para la limpieza, transformación, integración y reducción de datos. Se promueve el desarrollo de habilidades para gestionar valores ausentes, normalizar variables, codificar categorías, consolidar información procedente de distintas fuentes y aplicar técnicas como el análisis de componentes principales (PCA) con el fin de optimizar conjuntos de datos.

A lo largo del proceso de formación, se fortalecen competencias orientadas a asegurar la calidad, consistencia y utilidad de los datos, preparándolos adecuadamente para su uso en modelos de ciencia de datos, análisis predictivo o inteligencia artificial. Además, se emplean herramientas tecnológicas y casos prácticos que permiten una apropiación significativa de los contenidos. Para profundizar en la importancia de estos temas, se recomienda acceder al siguiente video:

Video 1. Limpieza, transformación e integración de datos para modelos predictivos

Enlace de reproducción del video

Síntesis del video: Limpieza, transformación e integración de datos para modelos predictivos
En la era de la inteligencia artificial, los datos se han convertido en uno de los recursos más valiosos para las organizaciones. Sin embargo, no basta con tener grandes volúmenes de información: la calidad, consistencia, completitud y estructura de los datos son factores determinantes en el éxito de cualquier modelo predictivo o sistema inteligente. Este componente formativo se centra en las técnicas esenciales para la preparación de datos en proyectos de ciencia de datos, análisis predictivo e inteligencia artificial. Se inicia con las técnicas de limpieza de datos, abordando las

dimensiones de la calidad, los tipos comunes de errores y métodos como la imputación, eliminación y corrección de inconsistencias.

Posteriormente, se exploran las técnicas de transformación de datos, donde se destacan la codificación, normalización y detección de duplicados, así como la adecuación de los datos para su uso en modelos de inteligencia artificial.

El recorrido continúa con la integración y el almacenamiento de datos, analizando procesos como el ETL (extracción, transformación y carga), el diseño de pipelines y la estructuración de modelos de datos para algoritmos de aprendizaje automático.

Finalmente, se introduce la automatización de los procesos de preparación y modelado de datos, utilizando herramientas especializadas que permiten optimizar y escalar las soluciones de inteligencia artificial.

Este componente no solo proporciona conocimientos técnicos, sino que fortalece la capacidad de diseñar soluciones eficientes para transformar datos en decisiones estratégicas.

1. Técnicas de limpieza de datos

La limpieza de datos es un proceso fundamental que consiste en identificar, corregir o eliminar datos incompletos, inexactos o irrazonables, con el objetivo de mejorar la calidad del conjunto de datos (Chen, 2014). Este proceso es determinante, ya que la calidad de los datos incide directamente en la calidad de la información generada, lo cual repercute en la precisión y efectividad de la toma de decisiones.

En el contexto del Big Data, donde el volumen y la variedad de los datos representan grandes desafíos, resulta esencial contar con métodos eficientes de limpieza. Un conjunto de datos bien depurado permite reducir el margen de error en los análisis y facilita la construcción de modelos predictivos más precisos.

Las técnicas de limpieza y preparación de datos permiten transformar los datos en bruto en formatos adecuados para su análisis, modelado y descubrimiento de conocimiento (Pyle, 1999; Wang, 2017). Estas técnicas son clave para garantizar la calidad del proceso analítico, ya que ayudan a detectar y corregir errores estructurales, inconsistencias y redundancias.

Una técnica ampliamente utilizada es la limpieza de datos o Data Cleaning, que abarca desde la identificación de valores faltantes o erróneos hasta la corrección o eliminación de registros redundantes, especialmente al integrar datos provenientes de múltiples fuentes. La supresión de datos duplicados y la estandarización de formatos, como la unificación de mayúsculas y minúsculas, son pasos comunes que contribuyen a la integridad del conjunto de datos.

La correcta aplicación de estas técnicas no solo mejora la calidad de los datos, sino que también reduce sesgos potenciales, aumentando la precisión de los análisis posteriores y fortaleciendo la toma de decisiones basada en evidencia.

1.1. Definición y dimensiones de la calidad de los datos

La calidad de los datos se refiere al grado en que un conjunto de datos resulta adecuado para su propósito, especialmente en contextos analíticos. Según Mesa Guerrero y Caicedo Zambrano (2020), esta calidad es una característica multidimensional que determina su utilidad y valor. Implica evaluar aspectos como la adecuación al propósito, precisión, confiabilidad, validez y ausencia de errores significativos. Garantizar datos de calidad es esencial para obtener información valiosa y tomar decisiones bien fundamentadas.

Las principales dimensiones para evaluar la calidad de los datos incluyen:

Coherencia: verifica que no existan contradicciones en los datos, ya sea dentro del mismo conjunto o entre diferentes fuentes. Asegura que los datos mantengan relaciones lógicas entre sí.

- ✓ **Precisión**: se refiere a qué tan cercanos están los valores registrados a los valores reales. Evalúa si los datos representan correctamente la realidad que intentan describir.
- ✓ **Integridad**: indica si todos los datos requeridos están presentes. Evalúa la existencia de valores completos, sin omisiones ni registros faltantes.
- ✓ **Validez**: determina si los datos cumplen con los formatos, tipos y reglas de negocio esperadas. Por ejemplo, que una fecha tenga el formato correcto o que una edad no sea negativa.

- ✓ **Puntualidad**: mide si los datos están disponibles cuando se necesitan. Evalúa si se actualizan con la frecuencia requerida para seguir siendo útiles.
- ✓ **Unicidad**: asegura que cada entidad o registro aparezca una sola vez. Verifica la ausencia de duplicados en el conjunto de datos.

Para garantizar esta calidad, es fundamental aplicar procesos de limpieza y transformación conforme a metodologías y estándares reconocidos. Estas metodologías comprenden un conjunto de principios y prácticas que aseguran que los datos sean apropiados para su análisis o para el desarrollo de modelos de inteligencia artificial.

Una metodología esencial es el Data Assay, descrita por Pyle (1999), que se enfoca en organizar los datos en un formato adecuado para minería y en evaluar su calidad. Sus objetivos principales son:

- ✓ Evaluar la calidad y detectar áreas problemáticas en los conjuntos de datos de entrada, salida, prueba y verificación.
- ✓ Analizar la calidad de variables individuales en todo su rango de valores.
- ✓ Estimar la independencia entre variables mediante la entropía.

El Data Assay permite identificar si los datos cumplen su propósito, al tiempo que revela limitaciones y brechas en el conocimiento disponible.

Otra metodología clave es el proceso de limpieza de datos o Data Cleaning, que se orienta a la corrección de errores y resolución de inconsistencias. Este proceso suele integrarse dentro del flujo ETL (extracción, transformación y carga), y se realiza comúnmente en un área de staging.

Las técnicas de limpieza pueden incluir:

- ✓ Aplicación de patrones y normas para detectar y corregir errores mediante transformaciones automáticas.
- ✓ Transformaciones en columnas y filas mediante operadores especializados.
- ✓ Reingeniería de datos a través de herramientas que identifican frecuencias y patrones comunes.
- ✓ Detección y consolidación de registros duplicados con técnicas de coincidencia estadística.

En resumen, la implementación de metodologías como el Data Assay y el Data Cleaning, junto con la estandarización y validación de datos, conforma un enfoque integral para garantizar su calidad. Este enfoque es indispensable para obtener resultados confiables tanto en el análisis estadístico como en aplicaciones basadas en inteligencia artificial.

1.2. Tipos comunes de errores

Durante la recopilación, integración o almacenamiento de datos, pueden presentarse diversos errores que afectan la calidad y utilidad de la información. Estos errores, si no se detectan y corrigen a tiempo, pueden generar interpretaciones erróneas y decisiones inadecuadas. Entre los más comunes se encuentran:

- ✓ **Valores nulos o ausentes**: se presentan cuando falta información en una o más variables de ciertos registros. Por ejemplo, una base de datos de clientes podría tener campos vacíos en la dirección de correo electrónico o en la fecha de nacimiento, lo cual limita su utilidad para análisis o segmentación.

- ✓ **Duplicados**: son registros repetidos que aparecen más de una vez en el conjunto de datos. Esto puede ocurrir, por ejemplo, cuando un mismo pedido se registra dos veces, inflando el total de ventas e introduciendo errores en los indicadores comerciales.
- ✓ **Ruido**: hace referencia a datos irrelevantes, inconsistentes o mal formateados. Un ejemplo común es cuando se ingresan nombres de clientes con diferentes convenciones como “Ana María”, “A. María” o “Ana M.”, lo que complica la unificación de registros y la calidad del análisis.
- ✓ **Outliers (valores atípicos)**: son observaciones que se desvían notablemente del resto de los datos. Por ejemplo, si en un conjunto de datos sobre salarios se encuentra un valor que triplica al siguiente más alto, podría tratarse de un error de digitación o de un caso excepcional que requiere atención especial.

Identificar y corregir estos errores es fundamental para asegurar la confiabilidad de los datos y evitar interpretaciones erróneas en el análisis posterior.

1.3. Técnicas de limpieza mediante imputación, eliminación y corrección

La limpieza o depuración de datos consiste en detectar, corregir o eliminar información incompleta, incorrecta o incoherente para mejorar su calidad. Este proceso es esencial en la integración de fuentes diversas y forma parte del ciclo ETL (extracción, transformación y carga) en entornos de almacenamiento de datos.

Antes de aplicar cualquier técnica, es necesario analizar los datos para detectar errores e inconsistencias. Este análisis puede realizarse de forma manual, mediante inspección de muestras, o con herramientas que generen metadatos e informes de calidad.

Rahm (2000) propone una clasificación de los problemas de calidad según su origen (una sola fuente o múltiples fuentes) y su nivel (esquema o instancia), lo cual facilita su detección y tratamiento. A continuación, se describe esta tipología:

Tabla 1. Problemas comunes de calidad de datos según su origen y nivel

Origen	Característica	Nivel de esquema	Nivel de instancia
Problemas de fuente única (Single-Source Problems)	Ocurren cuando los datos provienen de una sola fuente y presentan deficiencias internas.	Estos problemas están vinculados al diseño del modelo de datos o a la falta de restricciones: **Unicidad**: existencia de registros duplicados que deberían ser únicos, como dos clientes con el mismo número de identificación. **Integridad referencial**: claves foráneas sin correspondencia, lo que rompe las	Relacionados con los valores específicos almacenados, frecuentemente causados por errores de entrada o procesamiento: Errores ortográficos: por ejemplo, escribir “Colmbia” en lugar de “Colombia”. **Registros duplicados**: redundancia de información que

Origen	Característica	Nivel de esquema	Nivel de instancia
		relaciones entre tablas. **Otras restricciones**: definiciones inadecuadas de dominios, reglas de validación ausentes, entre otros.	puede afectar el análisis. **Otros**: valores fuera de rango, campos vacíos, formatos incorrectos.
Problemas de múltiples fuentes (Multi-Source Problems)	Se presentan al integrar datos de distintas fuentes, lo que introduce retos adicionales.	Los modelos de datos pueden diferir en estructura y denominación: Conflictos de nombres: como “CustomerID” frente a “ClientCode” para referirse al mismo concepto. Conflictos estructurales una misma entidad	Aparecen en los valores concretos provenientes de diversas fuentes: **Agregación inconsistente**: por ejemplo, una fuente muestra totales mensuales y la otra diarios. **Desincronización temporal**: los datos provienen de

Origen	Característica	Nivel de esquema	Nivel de instancia
		representada como tabla en una fuente y como objeto en otra. Otras diferencias: uso de distintos tipos de datos, estructuras jerárquicas frente a relacionales, etc.	momentos distintos, generando incoherencias. Otros: datos faltantes en alguna fuente, formatos incompatibles, distintas unidades de medida.

Una vez identificados los errores, es posible aplicar las siguientes técnicas:

- ✓ **Imputación de valores faltantes**: consiste en reemplazar los datos ausentes con estimaciones como el promedio, la mediana, la moda o valores predichos mediante algoritmos como KNN (K-Nearest Neighbors) o regresión.
- ✓ **Eliminación de registros o variables**: se utiliza cuando el número de errores es elevado o los datos no aportan valor analítico.
- ✓ **Corrección de inconsistencias**: incluye la estandarización de formatos (por ejemplo, unificar “Colombia” y “colombia”), la armonización de unidades de medida y la corrección ortográfica.

- ✓ **Tratamiento de valores atípicos (outliers)**: se puede realizar mediante técnicas estadísticas como el rango intercuartílico, el Z-score, o modelando estos valores de forma separada si son representativos

2. Técnicas de transformación de datos

La transformación de datos es una fase esencial dentro del proceso de preparación para el análisis, ya que convierte los datos en formatos adecuados para su posterior procesamiento o minería. Según Rahm y Hai Do (2000), esta fase incluye varias tareas como el análisis de datos, la definición del flujo de trabajo de transformación, la creación de reglas de mapeo, la verificación de los resultados transformados y, finalmente, el retroflujo de los datos limpios a los sistemas originales (citado por Almeida, 2013).

Entre las técnicas más comunes de transformación se encuentran:

a) **Codificación de variables categóricas**: en muchos casos, las herramientas analíticas requieren que todos los datos estén en formato numérico. Por esta razón, es necesario transformar los valores alfabéticos (por ejemplo, categorías como “bajo”, “medio”, “alto”) en representaciones numéricas. Esta técnica puede implementarse utilizando conocimientos del dominio o mediante el análisis de las frecuencias observadas en tablas de distribución conjunta (Pyle, 1999).

b) **Normalización**: esta técnica busca ajustar la escala de los valores de las variables numéricas para que puedan ser comparadas de forma justa y eficiente dentro de los algoritmos. Existen dos tipos principales:

- ✓ Normalización del rango: establece un rango fijo (como 0 a 1) para los valores, lo que facilita el análisis cuando las variables tienen diferentes escalas o unidades.
- ✓ Normalización de la distribución: su objetivo es que los datos de una variable se ajusten a una distribución específica (por ejemplo, normal o gaussiana), lo que resulta útil para algoritmos sensibles a la forma de la distribución.

Es importante diferenciar estas normalizaciones del proceso de normalización en bases de datos, el cual se enfoca en la estructuración eficiente de las tablas y relaciones.

c) **Transformación de formatos y estructuras**: incluye tareas como convertir fechas en formatos estándar, fusionar columnas, dividir campos compuestos, convertir texto a minúsculas, eliminar espacios, entre otros.

d) **Derivación de variables**: consiste en crear nuevas variables a partir de las existentes, como calcular el índice de masa corporal a partir del peso y la estatura, o el rango etario a partir de la fecha de nacimiento.

Estas transformaciones no solo aumentan la eficiencia de los algoritmos, sino que también mejoran la calidad del análisis al facilitar comparaciones y descubrimientos de patrones ocultos.

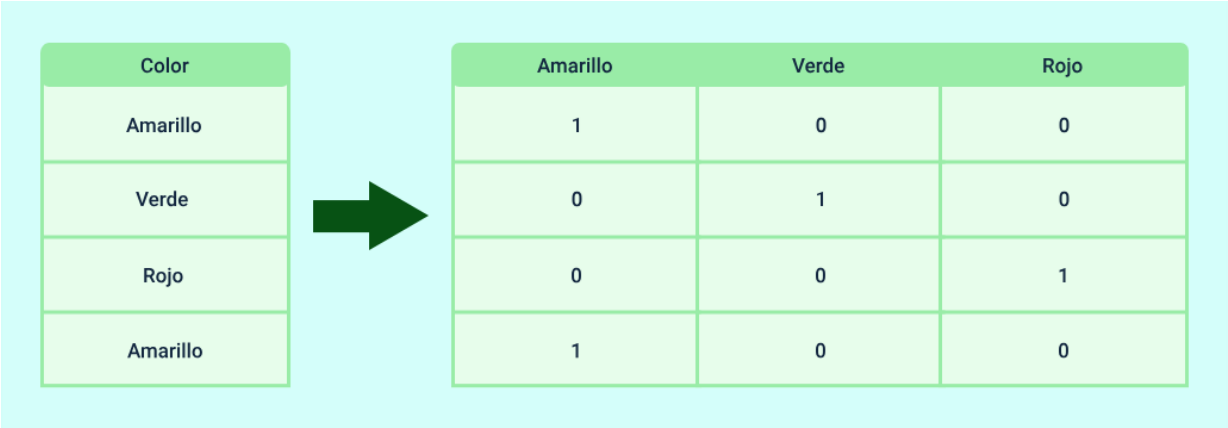
2.1. Codificación y normalización de datos

En el contexto de proyectos de aprendizaje automático, las tareas de codificación y normalización constituyen pasos fundamentales en la etapa de preparación y

transformación de los datos, ya que impactan directamente en el rendimiento y la eficacia de los algoritmos.

La codificación consiste en transformar datos no numéricos en representaciones numéricas comprensibles para los modelos. Dado que muchos algoritmos solo pueden operar con entradas numéricas, es indispensable convertir atributos categóricos (como etiquetas o colores) en valores numéricos. Una técnica ampliamente utilizada es el one-hot encoding, que genera una columna binaria por cada categoría posible, permitiendo representar la información sin introducir un orden artificial entre categorías. A modo de ejemplo, en la figura 1 se presenta cómo se transforma una variable categórica mediante la técnica de one-hot encoding.

Figura 1. Representación de una variable categórica aplicando one-hot encoding



Por su parte, la normalización busca ajustar la escala de las variables numéricas para evitar que aquellas con rangos más amplios dominen el comportamiento del modelo. Esta técnica es especialmente útil para algoritmos sensibles a la magnitud de los datos. Existen dos enfoques comunes: la normalización de rango, que transforma los valores para ubicarlos dentro de un intervalo específico (usualmente entre 0 y 1), y la

estandarización, que convierte los datos para que tengan una media de 0 y una desviación estándar de 1.

2.2. Gestión de inconsistencias y duplicados

La gestión de inconsistencias y duplicados es una etapa crítica en el proceso de transformación de datos para modelos de inteligencia artificial. Estos problemas, si no se abordan adecuadamente, pueden llevar a modelos de baja calidad y conclusiones erróneas, siguiendo el principio de “basura entra, basura sale” (Rahm, 2000).

Por tal razón, es importante identificar todo tipo de inconsistencias o duplicados. Las inconsistencias pueden manifestarse a nivel de esquema, como cuando el mismo nombre se asigna a diferentes objetos o se utilizan nombres distintos para un mismo objeto, o a nivel estructural, donde un objeto se representa de manera diferente en varias fuentes. También pueden presentarse a nivel de datos, donde los errores en el contenido real no son evidentes en el esquema, como errores tipográficos (Almeida, 2013).

A continuación, se presentan dos ejemplos de inconsistencias:

Ejemplo 1.

- ✓ Contexto: base de datos de ventas en una cadena de supermercados.
- ✓ Inconsistencia: “Coca-Cola 500ml”, “CocaCola 0.5L”, “Coke 500” y “Coca cola 500 ml”.
- ✓ El problema se presenta porque todos estos registros se refieren al mismo producto, pero están escritos de manera diferente, dificultando el análisis de ventas por producto.

Ejemplo 2.

- ✓ Contexto: base de datos de registro de clientes.
- ✓ Inconsistencia: “12/03/2024”, “2024-03-12”, “03-12-2024” y “12 Mar 2024”.
- ✓ El problema se presenta porque las fechas están registradas en distintos formatos (europeo, ISO, estadounidense y textual), lo que puede causar errores en la interpretación y procesamiento de datos cronológicos.

Las discrepancias también pueden originarse por razones arquitectónicas al combinar datos de distintos sistemas de bases de datos no compatibles o por problemas de sincronización entre flujos de datos.

Los duplicados, por su parte, se refieren a registros que representan la misma entidad del mundo real. Esto puede ocurrir por errores en la entrada de datos o durante la integración de múltiples fuentes. Una auditoría de datos reveló que casi el 90 % del volumen de datos era duplicado, aunque altamente inconsistente (Pyle, 1999).

2.3. Detección de inconsistencias y duplicados

Terminados los procesos de codificación y normalización de los datos, se debe verificar que todos tengan el formato correcto y sean del tipo adecuado para el entrenamiento del modelo. Para ello, se deben realizar diversas tareas:

- ✓ **Análisis de datos**: se requiere un análisis detallado para detectar errores e inconsistencias. Esto puede incluir inspección manual, uso de programas de análisis o generación de metadatos que revelen propiedades problemáticas.

- ✓ **Análisis de metadatos**: permite evaluar características auténticas de los datos y detectar valores atípicos o patrones inusuales comparando atributos entre diferentes tablas o conjuntos.
- ✓ **Minería de datos**: las técnicas de minería permiten identificar patrones y reglas. Las desviaciones de estas reglas podrían señalar errores potenciales, como en reglas de asociación ("total = cantidad * precio_unitario").
- ✓ **Algoritmos de emparejamiento**: se emplean para identificar registros similares que puedan ser duplicados, calculando puntuaciones de coincidencia automatizadas.

Una vez detectadas las inconsistencias y duplicados, se deben aplicar técnicas como:

- ✓ **Limpieza de datos (data cleaning)**: consiste en corregir errores e inconsistencias antes de cargar los datos en un Data Warehouse. Esto incluye reestructurar esquemas, reformatear atributos, definir valores por defecto y unificar formatos de datos.
- ✓ **Eliminación de duplicados (duplicate elimination–merge-purge)**: consiste en identificar registros similares (matching), fusionarlos (merge) en un solo registro sin redundancias, y luego eliminar los registros redundantes (purge).
- ✓ **Integración de datos (data integration)**: al combinar fuentes diversas, se emplean procesos ETL (extracción, transformación y carga) o virtualización de datos, resolviendo inconsistencias y asegurando uniformidad entre esquemas y datos.

Finalmente, se destaca que la gestión de inconsistencias y duplicados requiere un balance entre la automatización y la intervención manual, dependiendo de la complejidad de los datos y de la criticidad del proyecto. Idealmente, los datos corregidos también deberían reemplazar los datos sucios en las fuentes originales para evitar trabajos repetidos en el futuro.

2.4. Transformación de datos para modelos de inteligencia artificial

La transformación de datos para modelos de IA es un proceso fundamental que consiste en convertir los datos brutos en un formato adecuado y efectivo para el entrenamiento y la aplicación de estos modelos (Pyle, 1999). Considerando que los modelos de inteligencia artificial, especialmente aquellos basados en el aprendizaje automático, son esencialmente matemáticos y trabajan con cifras, la conversión de datos a menudo requiere transformar información no numérica en formas numéricas (Pyle, 1999).

La transformación de datos para modelos de IA abarca varias técnicas y propósitos:

- ✓ **Preparación para el modelado**: el objetivo principal es transformar los datos brutos en una forma que el modelo pueda entender y utilizar para aprender patrones y realizar predicciones. Esto incluye la transformación de variables categóricas en numéricas (codificación), la normalización o escalado de variables numéricas para garantizar que tengan rangos y distribuciones comparables, así como la imputación o sustitución de valores faltantes.

- ✓ **Adaptación a los requisitos del modelo**: los distintos tipos de modelos de inteligencia artificial presentan requisitos específicos en cuanto al formato y la representación de los datos. Por ejemplo, las redes neuronales suelen requerir entradas estrictamente numéricas, mientras que ciertos algoritmos estadísticos pueden funcionar de manera óptima con datos que sigan una distribución particular. La transformación asegura que los datos cumplan con estos requisitos.
- ✓ **Mejora del rendimiento del modelo**: la transformación de datos puede mejorar la precisión, la eficiencia y la estabilidad del entrenamiento del modelo. Por ejemplo, la normalización ayuda a evitar que variables con rangos mayores dominen el proceso de aprendizaje (Wang, 2017). Técnicas como la reducción de dimensionalidad pueden simplificar los datos, disminuir el riesgo de sobreajuste y acelerar el entrenamiento.
- ✓ **Extracción de características relevantes**: la transformación puede incluir la creación de nuevas variables (ingeniería de características) a partir de las existentes para resaltar información más relevante para la tarea de predicción (Tsai, 2015). Normalmente, la extracción de características viene acompañada de un análisis de componentes principales (PCA), una forma de reducción de la dimensionalidad que simplifica conjuntos de datos con muchas variables. Su objetivo es convertir un espacio de características de alta dimensión en uno de menor dimensión, manteniendo la información esencial. Esto se refleja en una mayor eficiencia del modelo al minimizar el riesgo de sobreajuste. La efectividad de la reducción de dimensionalidad

puede compararse evaluando el rendimiento del modelo antes y después de aplicarla.

- ✓ **Manejo de datos de ejecución (live data)**: una vez entrenado el modelo, cualquier dato nuevo que se utilice para realizar predicciones debe ser transformado exactamente de la misma forma que los datos de entrenamiento y prueba. El PIE-I (Prepared Information Environment Input) es el encargado de esta transformación (Pyle, 1999).
- ✓ **Inversión de la transformación para la salida**: para que las predicciones del modelo, que a menudo están en un formato transformado (por ejemplo, valores numéricos escalados), sean comprensibles y útiles en el mundo real, es necesario revertir la transformación. Esta tarea la lleva a cabo el PIE-O (Prepared Information Environment Output) (Pyle, 1999).

En esencia, la transformación de datos para modelos de IA constituye un proceso fundamental que acondiciona los datos para el aprendizaje y la predicción, garantizando su compatibilidad con el modelo, optimizando su rendimiento y facilitando la interpretación de los resultados. Este proceso no es automático y, a menudo, requiere conocimiento del dominio y experimentación para determinar las transformaciones más efectivas.

3. Integración y almacenamiento de datos

La integración de datos consiste en combinar información proveniente de diversas fuentes heterogéneas en un conjunto de datos unificado y coherente. Este proceso busca ofrecer una visión completa y consistente de la información para su análisis y posterior explotación en modelos de inteligencia artificial o sistemas de apoyo a decisiones. Una de las tareas fundamentales en la integración es resolver las discrepancias en los nombres de atributos, estructuras y dimensiones, evitando duplicidades y conflictos que puedan comprometer la calidad del conjunto final (Wang, 2017).

Las herramientas ETL (Extract, Transform, Load) y ELT (Extract, Load, Transform) son estrategias ampliamente utilizadas en los procesos de integración. En el enfoque ETL, los datos son extraídos de sus fuentes, transformados para adecuarlos a los requisitos de destino “lo que puede incluir tareas de limpieza, normalización, enriquecimiento y catalogación” y, posteriormente, cargados en un repositorio, como un Data Warehouse o un Data Lake. En cambio, el enfoque ELT invierte parte del proceso: los datos son primero extraídos y cargados en el destino para luego ser transformados, aprovechando las capacidades de procesamiento de los sistemas modernos (Elgendy, 2014).

La integración de datos también puede incorporar técnicas como la reconciliación de entidades (entity resolution), el mapeo semántico y el uso de middlewares o plataformas de integración, como los Enterprise Service Bus (ESB) o los sistemas basados en API. Estos enfoques permiten gestionar no solo bases de datos estructuradas, sino también fuentes de datos semiestructuradas o no estructuradas.

El almacenamiento de los datos integrados depende de las necesidades específicas del proyecto y puede organizarse en estructuras tradicionales como bases de datos relacionales, en almacenes optimizados para el análisis (Data Warehouses) o en repositorios flexibles diseñados para manejar grandes volúmenes y variedad de datos (Data Lakes). La elección adecuada de la estrategia de almacenamiento resulta crucial para garantizar que los datos puedan ser accedidos, consultados y procesados de manera eficiente y segura.

En definitiva, la integración y almacenamiento de datos constituyen pasos estratégicos que aseguran la disponibilidad de información fiable, consistente y de calidad para su posterior análisis, modelado y toma de decisiones basada en datos.

3.1. Proceso ETL

El proceso ETL (extracción, transformación y carga) constituye un componente esencial en el tratamiento de grandes volúmenes de datos y en la construcción de Data Warehouses. Según Almeida (2013), este proceso se compone de tres etapas principales que operan de forma secuencial:

a) **Extracción**: es el primer paso del proceso ETL y consiste en obtener datos relevantes de múltiples fuentes, que pueden incluir sistemas OLTP (Online Transaction Processing), hojas de cálculo, archivos de texto, bases de datos no estructuradas o contenido web. Para optimizar el tiempo de procesamiento, no siempre se extraen todos los datos, sino únicamente aquellos que han cambiado desde la última ejecución, principalmente registros nuevos o actualizados. La detección de cambios suele realizarse mediante la comparación entre dos instantáneas de los datos: una correspondiente a la última extracción y otra actual. Para este fin, las

herramientas ETL emplean conectividad directa con las fuentes mediante conectores, APIs (Application Programming Interfaces) o servicios de datos, asegurando un acceso eficiente y seguro.

b) **Transformación**: una vez extraídos los datos, estos se someten a una serie de rutinas de transformación que los adaptan, corrigen y estructuran para que sean compatibles con el esquema del Data Warehouse. Esta etapa contempla diversas operaciones, como reformatear atributos, recalcular valores, modificar estructuras clave, agregar elementos temporales, asignar valores por defecto, seleccionar información relevante y consolidar datos dispersos. Durante la transformación, también se aplican procesos de limpieza de datos para corregir errores, inconsistencias y valores atípicos. Esto puede implicar traducir esquemas, filtrar información no deseada, agregar datos resumidos o utilizar herramientas de limpieza especializadas. La transformación no solo prepara los datos para su almacenamiento, sino que también garantiza su calidad y uniformidad.

c) **Carga**: representa la etapa final del proceso ETL y tiene como objetivo incorporar los datos transformados en el repositorio de destino, generalmente un Data Warehouse. Uno de los principales retos durante esta fase es identificar correctamente los datos nuevos y los actualizados, para evitar duplicaciones o pérdidas de información. Los registros se clasifican entre filas nuevas, que deben insertarse, y filas existentes, que requieren actualización. Muchas herramientas ETL modernas facilitan esta tarea mediante predicados de lenguaje o reglas de integración específicas. Durante la carga, también se gestionan aspectos técnicos como los segmentos de rollback y los archivos de registro, para mantener la

integridad de los datos ante posibles fallos o interrupciones. Habitualmente, antes de su carga definitiva en el Data Warehouse, los datos pasan por un área de staging, donde se realizan validaciones finales y preparaciones específicas.

3.2. Preparación de datos para modelos de aprendizaje automático

El aprendizaje automático se define como un subcampo de la informática que se centra en el diseño y desarrollo de algoritmos capaces de aprender patrones a partir de datos, sin necesidad de ser programados de manera explícita (Maltby, 2011). Para que los modelos de aprendizaje automático sean efectivos, precisos y logren generalizar adecuadamente a datos no vistos, resulta esencial realizar una preparación cuidadosa de los datos.

El objetivo principal de la preparación de datos es facilitar el trabajo de los algoritmos de modelado, eliminando inconsistencias, reduciendo el ruido y destacando la información relevante. Este proceso implica la aplicación de diversas metodologías y técnicas, como Data Assay, PIE (Prepared Information Environment), PIE-O (Prepared Information Environment Output) y métodos de consistencia de datos, ya abordados en este programa de formación complementaria.

La preparación de datos es un proceso meticuloso que incluye comprender a fondo las características del conjunto de datos, evaluar su calidad, aplicar transformaciones específicas que se adecuen al algoritmo de modelado seleccionado y garantizar la coherencia en todos los subconjuntos de datos (entrenamiento, validación y prueba). El resultado esperado es obtener datos limpios, relevantes y en un formato óptimo para el desarrollo de modelos predictivos sólidos.

En términos generales, la preparación de datos para el aprendizaje automático sigue los siguientes pasos:

1. **Limpieza de datos**

Eliminación de duplicados, corrección de errores, tratamiento de valores atípicos y gestión de datos faltantes.

2. **Transformación de variables**

Normalización o estandarización de datos numéricos, codificación de variables categóricas (por ejemplo, mediante codificación one-hot o label encoding) y transformación de fechas o textos según sea necesario.

3. **Selección de características**

Identificación de las variables más relevantes para el modelo, reduciendo la dimensionalidad del conjunto de datos y eliminando variables redundantes o irrelevantes.

4. **Balanceo de clases**

En problemas de clasificación, aplicación de técnicas como sobremuestreo (oversampling) o submuestreo (undersampling) para equilibrar las clases y evitar sesgos en los modelos.

5. **División de datos**

Separación del conjunto de datos en conjuntos de entrenamiento, validación y prueba, con el fin de evaluar el rendimiento del modelo de manera objetiva.

6. **Aumento de datos (Data Augmentation)**

En algunos casos, especialmente en datos de imágenes o texto, generación de nuevas muestras a partir de variaciones de las existentes para mejorar la robustez del modelo.

La preparación de datos resulta particularmente crítica en aplicaciones de analítica predictiva, donde se utilizan modelos estadísticos y algoritmos de aprendizaje automático para detectar patrones en datos históricos y anticipar comportamientos y tendencias futuras (Zakir, 2015).

Una adecuada preparación de los datos no solo mejora el desempeño de los modelos, sino que también minimiza el riesgo de sobreajuste y maximiza la capacidad del modelo para ofrecer predicciones precisas en escenarios reales.

3.3. Diseño de modelos de datos para algoritmos de inteligencia artificial y aprendizaje automático

El diseño de modelos de datos para algoritmos de Inteligencia Artificial (IA) y aprendizaje automático (ML) consiste en construir estructuras organizadas de datos que permitan un entrenamiento efectivo de los algoritmos. Este diseño debe contemplar no solo la forma en que los datos son almacenados y procesados, sino también cómo se optimizan para extraer patrones significativos y realizar predicciones fiables (Sahoo, 2019).

El proceso de diseño de modelos de datos implica las siguientes etapas:

a) **Análisis del problema**: comprender el objetivo del proyecto, los tipos de predicciones requeridas y las características de los datos disponibles.

b) **Selección del modelo de IA**: elegir el algoritmo más adecuado (como regresión, clasificación, clustering, redes neuronales o árboles de decisión) de acuerdo con la naturaleza del problema y la estructura de los datos.

c) **Ingeniería de características**: crear o seleccionar variables relevantes que representen adecuadamente el problema para mejorar el desempeño del algoritmo.

d) **Entrenamiento del modelo**: ajustar los parámetros internos del modelo utilizando el conjunto de entrenamiento, buscando que el modelo aprenda patrones útiles.

e) **Evaluación del modelo**: medir el desempeño utilizando conjuntos de validación o prueba mediante métricas específicas como precisión, recall, F1-score o AUC.

f) **Ajuste de hiperparámetros (Hyperparameter Tuning)**: optimizar configuraciones como la tasa de aprendizaje, la profundidad de árboles o el número de capas en redes neuronales, con el fin de maximizar el rendimiento del modelo.

Posteriormente, es posible incorporar técnicas de automatización, como el AutoML, que permiten automatizar fases específicas del diseño, tales como la selección de modelos, el ajuste de hiperparámetros y la evaluación, acelerando el proceso de desarrollo de soluciones de IA.

A pesar de la existencia de herramientas que automatizan muchas de estas etapas, la intervención y supervisión de expertos sigue siendo indispensable para

interpretar adecuadamente los resultados y garantizar la validez y aplicabilidad del modelo en contextos reales.

3.4. Pipelines de procesamiento de datos

En el ámbito del procesamiento de datos, un pipeline es un flujo organizado que transporta los datos a través de varias etapas de procesamiento, transformando datos sin procesar en información valiosa o conocimientos útiles. Cada fase cumple una tarea específica, utilizando el resultado de la etapa anterior como entrada. Se podría afirmar que son una serie de procesos interconectados que transportan y transforman datos desde sus fuentes originales hasta un destino final, como un almacén de datos, un sistema de análisis o una aplicación de inteligencia artificial (Almeida, 2013).

La implementación de un pipeline generalmente involucra los siguientes pasos:

- ✓ Identificación de fuentes de datos.
- ✓ Selección de datos relevantes.
- ✓ Extracción de los datos.
- ✓ Preparación de los datos.
- ✓ Carga de los datos.
- ✓ Modelado y análisis de los datos.
- ✓ Evaluación e interpretación.
- ✓ Automatización y monitoreo.

A continuación, se presenta un caso de uso de un proyecto en el cual se pretende diseñar un modelo de aprendizaje automático para la predicción de abandono de aprendices en plataformas de educación virtual:

a) **Recolección de datos (Data Collection)**: obtener información sobre el comportamiento y perfil de los aprendices. A continuación, se presentan las fuentes típicas:
    - ✓ Logs de interacción de la plataforma (Moodle, Canvas, etc.).
    - ✓ Tiempo de conexión y actividad en la plataforma.
    - ✓ Participación en foros, chats, videollamadas.
    - ✓ Envío de evidencias y resultados de evaluaciones.
    - ✓ Datos demográficos (edad, género, país y nivel educativo).
    - ✓ Encuestas iniciales o de satisfacción.
    - ✓ Datos de navegación (rutas de clics, páginas visitadas, etc.).
b) **Ingesta y almacenamiento (Data Ingestion & Storage)**: consolidar y almacenar los datos para su posterior análisis. A continuación, se presentan las herramientas:
    - ✓ Extracción mediante API o exportación de logs (CSV y JSON).
    - ✓ Bases de datos SQL (PostgreSQL y MySQL) o NoSQL (MongoDB).
    - ✓ Sistemas de almacenamiento en la nube (Google BigQuery, AWS S3 y Azure).
c) **Limpieza de datos (Data Cleaning)**: eliminar inconsistencias y preparar datos confiables. A continuación, se presentan los procesos clave:
    - ✓ Imputación de valores nulos o ausentes.
    - ✓ Eliminación de registros duplicados.
    - ✓ Homogeneización de formatos de fecha, texto y numeración.
    - ✓ Corrección de etiquetas o errores ortográficos.
    - ✓ Identificación y manejo de outliers (por ejemplo, sesiones extremadamente largas).

d) **Integración de datos (Data Integration)**: unir datos desde diversas fuentes. A continuación, se presentan las acciones típicas:

- ✓ Unión por identificador único (ID de aprendiz o correo institucional).
- ✓ Alineación temporal de eventos de aprendizaje.
- ✓ Integración de datos históricos y contextuales (por ejemplo, historial académico).

e) **Transformación y enriquecimiento (Data Transformation & Enrichment)**: Crear variables significativas para el modelo. A continuación, se presentan las variables derivadas útiles:

- ✓ Número de días sin conexión.
- ✓ Porcentaje de actividades completadas.
- ✓ Participación en foros (% de mensajes enviados respecto al total).
- ✓ Tendencia de calificaciones (ascendente o descendente).
- ✓ Ratio de abandono previo.
- ✓ Tiempo promedio de sesión.

f) **Selección de características (Feature Selection)**: identificar las variables más predictivas del abandono. A continuación, se presentan los métodos aplicados:

- ✓ Análisis de correlación (Pearson o Spearman).
- ✓ Pruebas estadísticas (Chi-cuadrado para variables categóricas).
- ✓ Importancia de variables mediante modelos de árbol (Random Forest o XGBoost).

- ✓ Eliminación de variables redundantes o poco informativas.

g) **División del dataset (Train/Test Split)**: preparar los datos para el entrenamiento y validación del modelo. A continuación, se presentan las estrategias:

- ✓ División típica: 70 % entrenamiento / 30 % prueba.
- ✓ Validación cruzada (K-Fold) para mayor robustez.
- ✓ Estratificación para mantener la proporción de clases (abandono vs. no abandono).

h) **Entrenamiento del modelo**: aplicar algoritmos de clasificación supervisada. A continuación, se presentan los modelos recomendados:

- ✓ Regresión Logística (modelo base).
- ✓ Random Forest.
- ✓ Gradient Boosting (XGBoost y LightGBM).
- ✓ Redes neuronales simples (si existe suficiente volumen de datos).
- ✓ SVM o KNN como alternativas exploratorias.

i) **Evaluación del modelo (Model Evaluation)**: verificar el desempeño predictivo del modelo. A continuación, se presentan las métricas clave:

- ✓ Accuracy: porcentaje de predicciones correctas.
- ✓ Precision: proporción de predicciones positivas correctas.
- ✓ Recall: proporción de casos reales positivos correctamente detectados.
- ✓ F1-Score: balance entre precisión y recall.
- ✓ ROC-AUC: capacidad del modelo para diferenciar entre abandono y no abandono (especialmente relevante cuando las clases están desbalanceadas).

- ✓ Matriz de confusión para visualización de errores.

j) Despliegue y monitoreo: usar el modelo en producción para alertas tempranas de riesgo. A continuación, se presentan las opciones de despliegue:

- ✓ API REST con Flask o FastAPI para integración con el LMS.
- ✓ Dashboard de visualización (Power BI, Streamlit o Dash).
- ✓ Automatización del reentrenamiento periódico (por cohorte o semestre).
- ✓ Generación de alertas automáticas para instructores.
- ✓ Monitoreo de drift de datos: detección de cambios en los patrones de los datos para actualizar el modelo cuando sea necesario.

En resumen, un pipeline de datos es un camino establecido para el movimiento y transformación de datos, desde su origen hasta un punto de destino donde pueden ser utilizados para análisis, generación de informes o aplicaciones de IA. Su implementación incluye fases de extracción, preparación, carga y análisis, con un enfoque creciente en la automatización para mejorar la eficiencia y garantizar la adaptabilidad de los modelos a cambios futuros en los datos.

4. Automatización de modelos de inteligencia artificial

La automatización de modelos de inteligencia artificial busca reducir la intervención manual en las tareas de preparación, construcción, validación y despliegue de modelos predictivos. A través del uso de pipelines de datos y herramientas especializadas, se consigue acelerar el ciclo de vida de los proyectos de IA, mejorar la reproducibilidad de los resultados y facilitar la actualización continua de los modelos frente a nuevos datos. La automatización no solo optimiza los recursos humanos y computacionales, sino que también permite la detección temprana de desviaciones en el comportamiento del modelo, garantizando su vigencia en entornos dinámicos (Almeida, 2013).

4.1. Automatización de procesos de preparación y modelado de datos

Los procesos de preparación y modelado de datos son tradicionalmente intensivos en tiempo y propensos a errores si se realizan manualmente. Para evitar estas limitaciones, se implementan flujos automatizados que permiten:

- ✓ La recolección periódica de nuevos datos desde diferentes fuentes (APIs, bases de datos y sistemas LMS).
- ✓ La ejecución sistemática de rutinas de limpieza, imputación de datos faltantes, detección de valores atípicos y normalización de variables.
- ✓ La generación automática de variables derivadas y enriquecimiento de los datasets.
- ✓ La actualización dinámica de particiones de entrenamiento y prueba conforme llegan nuevos datos.

- ✓ El reentrenamiento programado de modelos de machine learning, considerando validaciones cruzadas y evaluaciones de desempeño periódicas.

Esta automatización se logra mediante la construcción de pipelines que conectan las etapas de forma orquestada, utilizando herramientas de orquestación de flujos de trabajo como Apache Airflow, Kubeflow Pipelines o servicios integrados en plataformas en la nube.

4.2. Automatización mediante herramientas especializadas

La automatización de modelos de inteligencia artificial puede lograrse utilizando herramientas como Python, Scikit-learn, Jupyter Notebooks y Power BI, integradas en un flujo de trabajo secuencial. Cada herramienta cumple una función específica en las etapas de preparación de datos, modelado predictivo y visualización de resultados, permitiendo una orquestación eficiente y escalable mediante programación estructurada y procesos automatizados.

A continuación, se describe cómo estas herramientas contribuyen al proceso de automatización:

a) Preparación y análisis de datos con Python y Jupyter Notebooks

Python, junto con bibliotecas como Pandas, se utiliza para automatizar la extracción y carga de datos desde fuentes diversas, como bases de datos, archivos CSV o APIs. Jupyter Notebooks proporciona un entorno interactivo para el análisis exploratorio de datos (EDA), donde se pueden desarrollar scripts reutilizables para tareas de limpieza, imputación de valores faltantes, eliminación de duplicados, transformación y codificación de variables. Esta etapa inicial, aunque puede incluir

análisis manual, se automatiza mediante la generación de funciones que estandarizan el procesamiento de nuevos datos (Sahoo, 2019).

b) Modelado de inteligencia artificial con Python y Scikit-learn

Scikit-learn permite automatizar el entrenamiento de modelos mediante scripts que manejan la selección algorítmica, la división de datos en conjuntos de entrenamiento y prueba, y la capacitación del modelo. Herramientas como GridSearchCV o RandomizedSearchCV facilitan la búsqueda automatizada de los mejores hiperparámetros, optimizando así el rendimiento predictivo. Posteriormente, la evaluación del modelo se realiza de forma sistemática utilizando métricas predefinidas, y el modelo entrenado se guarda utilizando bibliotecas como Joblib o Pickle para su reutilización futura sin necesidad de reentrenamiento.

c) Visualización y presentación de resultados con Power BI

Power BI se integra como herramienta de visualización para automatizar la presentación de resultados. Puede conectarse a las fuentes donde se almacenan los datos procesados o las predicciones del modelo, permitiendo la creación de informes y dashboards que se actualizan automáticamente. Además, Power BI soporta la ejecución de scripts en Python, lo que amplía las capacidades de análisis y visualización avanzadas directamente dentro de los informes, asegurando la actualización continua de los resultados presentados.

d) Automatización completa del flujo de trabajo

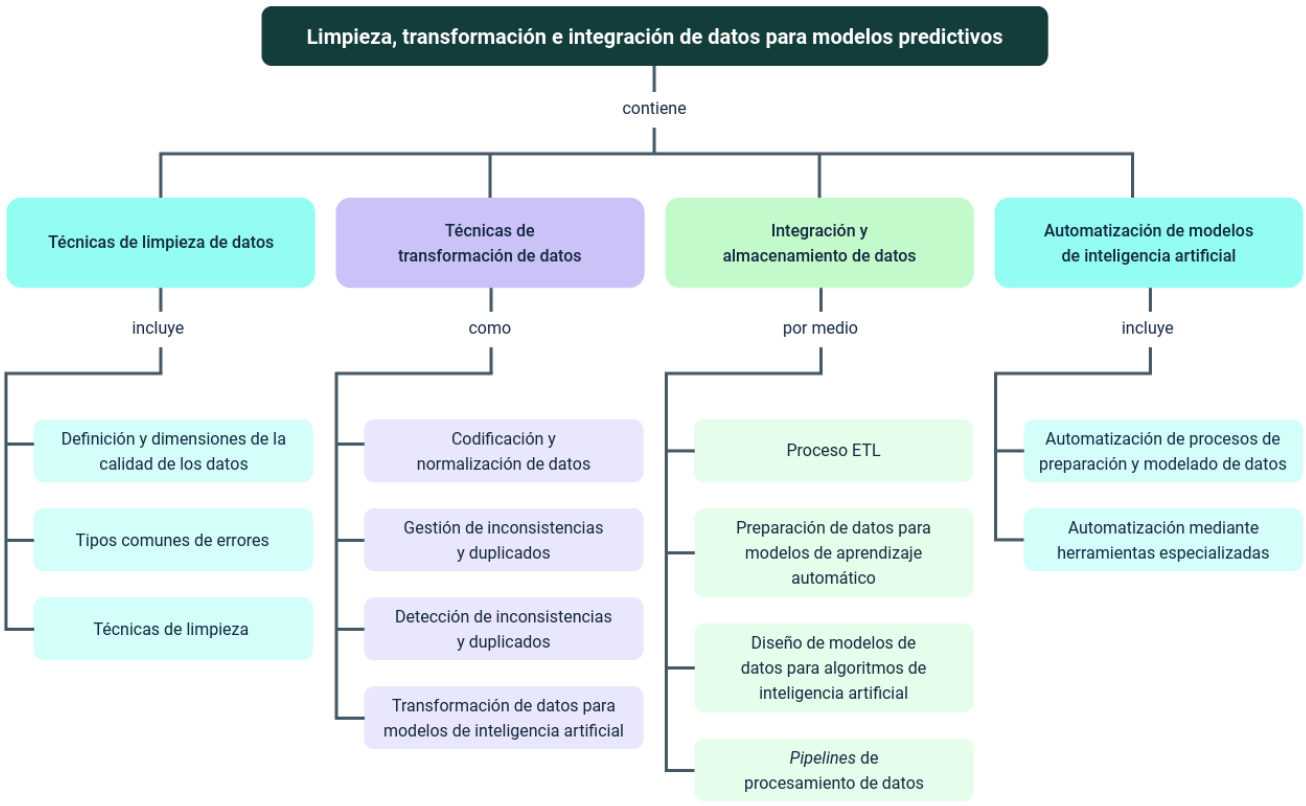
La automatización total se logra mediante la escritura de scripts de Python que orquestan todo el proceso, desde la recolección de datos hasta el despliegue de informes actualizados. Los Jupyter Notebooks actúan como entornos de desarrollo y

documentación, mientras que en producción se pueden emplear herramientas de gestión de flujos de trabajo, como Apache Airflow o schedulers de tareas, para ejecutar los procesos de forma programada, garantizando la actualización periódica de los modelos y reportes.

Esta integración coordinada de herramientas facilita la construcción de pipelines robustos de inteligencia artificial, mejorando la eficiencia, reduciendo errores y permitiendo una respuesta ágil ante cambios en los datos o en las necesidades del negocio.

Síntesis

La preparación de datos para modelos de inteligencia artificial involucra diversas técnicas de limpieza, transformación e integración. La limpieza de datos abarca la identificación de dimensiones de calidad, la detección de errores comunes y la aplicación de métodos como la imputación, eliminación y corrección. La transformación de datos incluye procesos de codificación, normalización, gestión de inconsistencias y duplicados, así como su adecuación para algoritmos de inteligencia artificial. En cuanto a la integración y almacenamiento, se aborda el uso de procesos ETL, la preparación de datos para el aprendizaje automático, el diseño de modelos de datos y la construcción de pipelines de procesamiento. Finalmente, se contempla la automatización de todo el flujo mediante herramientas especializadas que permiten optimizar las etapas de preparación, modelado y actualización de los datos de manera eficiente.


Limpieza, transformación e integración de datos para modelos predictivos
contiene
Técnicas de limpieza de datos
Técnicas de transformación de datos
Integración y almacenamiento de datos
Automatización de modelos de inteligencia artificial
incluye
como
por medio
incluye
Definición y dimensiones de la calidad de los datos
Tipos comunes de errores
Técnicas de limpieza
Codificación y normalización de datos
Gestión de inconsistencias y duplicados
Detección de inconsistencias y duplicados
Transformación de datos para modelos de inteligencia artificial
Proceso ETL
Preparación de datos para modelos de aprendizaje automático
Diseño de modelos de datos para algoritmos de inteligencia artificial
Pipelines de procesamiento de datos
Automatización de procesos de preparación y modelado de datos
Automatización mediante herramientas especializadas

Material Complementario

Tema	Referencia	Tipo de material	Enlace del recurso
2.1 Codificación y normalización de datos	Ecosistema de Recursos Educativos SENA. (2022). Proceso de normalización de datos [Video]. YouTube.	Video	https://www.youtube.com/watch?v=hKwuc-JJisI
3. Integración y almacenamiento de datos	Ecosistema de Recursos Educativos SENA. (2024). Proceso de integración de datos y ETL [Video]. YouTube.	Video	https://www.youtube.com/watch?v=ilmPbQSBBoM
4.1 Automatización de procesos de preparación y modelado de datos	Ecosistema de Recursos Educativos SENA. (2023). Fundamentos de modelamiento de datos [Video]. YouTube.	Video	https://www.youtube.com/watch?v=tmYngyjHmbc

Glosario

Calidad de los datos: conjunto de atributos que determinan la utilidad de los datos, como precisión, completitud, consistencia y actualidad.

Codificación de variables categóricas: conversión de variables no numéricas en formatos numéricos mediante técnicas como one-hot encoding o label encoding.

Duplicados: registros repetidos dentro de un conjunto de datos que deben identificarse y eliminarse para evitar distorsiones en el análisis.

ETL (Extract, Transform, Load): proceso que consiste en extraer datos de diversas fuentes, transformarlos para análisis o modelado, y cargarlos en un sistema de almacenamiento.

Imputación: técnica utilizada para reemplazar valores faltantes en un conjunto de datos mediante estimaciones basadas en otros valores disponibles.

Normalización: proceso que ajusta la escala de los datos numéricos para que estén dentro de un mismo rango, facilitando el entrenamiento de modelos de aprendizaje automático.

Pipeline de procesamiento de datos: secuencia automatizada de pasos para preparar los datos, que puede incluir limpieza, transformación, modelado y evaluación.

Referencias bibliográficas

Almeida, F., & Calistru, C. (2013). The main challenges and issues of big data management. International Journal of Research Studies in Computing, 2(1), 11–20.

Chen, M., Mao, S., & Liu, Y. (2014). Big data: Una encuesta. Redes y Aplicaciones Móviles, 19(2), 171–209.

Elgendy, N., & Elragal, A. (2014). Big data analytics: A literature review paper. In Advances in Data Mining. Applications and Theoretical Aspects: 14th Industrial Conference, ICDM 2014, St. Petersburg, Russia, July 16–20, 2014. Proceedings 14 (pp. 214–227). Springer International Publishing.

FreeCodeCamp.org. (2024). Learn to code - for free. Build projects. Earn certifications (SQL, Tableau, Power BI, Python, Excel, Pandas, Projects, more). https://www.freecodecamp.org

Gehani, A., & Tariq, D. (2012). SPADE: Support for provenance auditing in distributed environments. In International Provenance and Annotation Workshop (pp. 101–120). Springer. https://doi.org/10.1007/978-3-642-35170-9_6

Maltby, D. (2011, October). Big data analytics. In 74th Annual Meeting of the Association for Information Science and Technology (ASIST) (pp. 1–6).

Mesa Guerrero, J. A., & Caicedo Zambrano, S. J. (2020). Introducción a la estadística descriptiva.

Pyle, D. (1999). Data preparation for data mining. Morgan Kaufmann.

Rahm, E., & Do, H. H. (2000). Data cleaning: Problems and current approaches. IEEE Data Engineering Bulletin, 23(4), 3–13.

Sahoo, K., Samal, A. K., Pramanik, J., & Pani, S. K. (2019). Exploratory data analysis using Python. International Journal of Innovative Technology and Exploring Engineering, 8(12), 4727–4735.

Tsai, C. W., Lai, C. F., Chao, H. C., & Vasilakos, A. V. (2015). Big data analytics: A survey. Journal of Big Data, 2, 1–32.

Wang, L. (2017). Heterogeneous data and big data analytics. Automatic Control and Information Sciences, 3(1), 8–15. https://doi.org/10.12691/acis-3-1-3

Zakir, J., Seymour, T., & Berg, K. (2015). Big data analytics. Issues in Information Systems, 16(2).

Créditos

Nombre	Cargo	Centro de Formación y Regional
Milady Tatiana Villamil Castellanos	Responsable Ecosistema de Recursos Educativos Digitales (RED)	Dirección General
Diana Rocío Possos Beltrán	Responsable de línea de producción	Centro de Comercio y Servicios - Regional Tolima
Deivis Eduard Ramírez Martínez	Experto temático	Centro de Comercio y Servicios - Regional Tolima
Viviana Esperanza Herrera Quiñonez	Evaluadora instruccional	Centro de Comercio y Servicios - Regional Tolima
Oscar Ivan Uribe Ortiz	Diseñador web	Centro de Comercio y Servicios - Regional Tolima
José Jaime Luis Tang Pinzón	Diseñador web	Centro de Comercio y Servicios - Regional Tolima
Veimar Celis Meléndez	Desarrollador fullstack	Centro de Comercio y Servicios - Regional Tolima
Gilberto Junior Rodríguez Rodríguez	Animador y productor audiovisual	Centro de Comercio y Servicios - Regional Tolima
Jorge Eduardo Rueda Peña	Evaluadora de contenidos inclusivos y accesibles	Centro de Comercio y Servicios - Regional Tolima

Nombre	Cargo	Centro de Formación y Regional
Jorge Bustos Gómez	Validador y vinculador de recursos educativos digitales	Centro de Comercio y Servicios - Regional Tolima