Procesamiento Avanzado de datos en Python

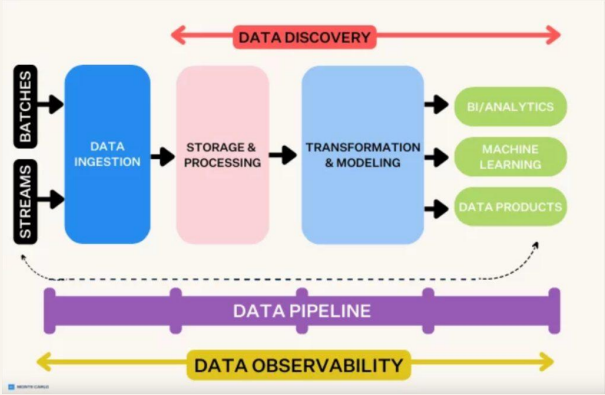
2025

RAFAEL MARTÍN R.

Procesamiento Avanzado de Datos en Python

De la Adquisición al Preprocesamiento

El procesamiento avanzado de datos en Python es fundamental para el análisis moderno. Esta sección abarca el pipeline completo de datos, desde la adquisición hasta el preprocesamiento, utilizando herramientas como pandas, numpy y scikit-learn. Aprenderemos técnicas efectivas para manejar datos estructurados y no estructurados, implementar limpieza automatizada y aplicar transformaciones avanzadas.

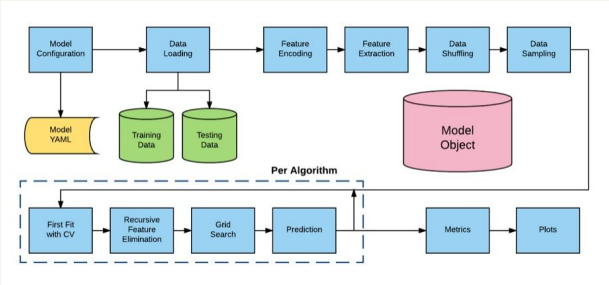


RAFAEL MARTÍN R.

Índice de Contenidos

Exploraremos seis módulos fundamentales del procesamiento de datos

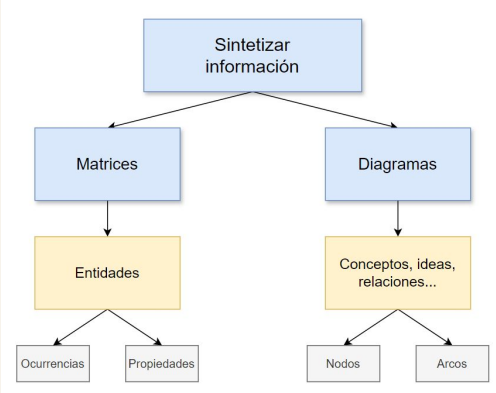
Introducción y Fundamentos



Construyendo bases sólidas para el procesamiento de datos

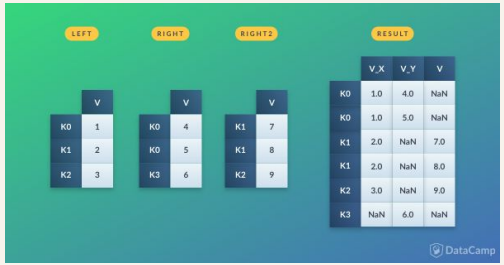
- Procesamiento de datos como pilar fundamental en análisis moderno
- Pipeline end-to-end: desde datos crudos hasta insights accionables
- Pandas: manipulación eficiente de datos estructurados
- NumPy: computación numérica y operaciones matriciales avanzadas
- Scikit-learn: herramienta esencial para preprocesamiento y modelado
- Jupyter Notebooks como entorno interactivo de desarrollo

Tipos de Datos: Datos Estructurados



¿Qué son los Datos Estructurados?

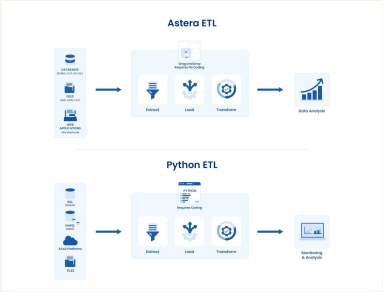
- Datos organizados en filas y columnas
- Formato predecible y consistente
- Fácilmente procesables por computadoras



Formatos y Manipulación

- CSV y Excel para almacenamiento simple
- SQL para bases de datos relacionales
- Pandas para análisis y transformación eficiente

Tipos de Datos: Datos No Estructurados



Definición y Tipos

- Datos sin formato o estructura predefinida
- Incluye texto, imágenes, audio y video
- Representa el 80% de datos empresariales

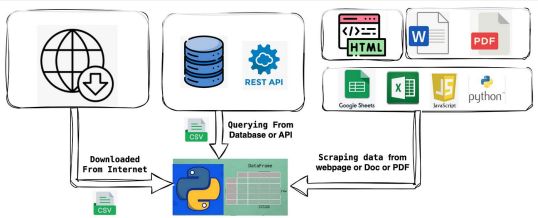
Formatos y Procesamiento

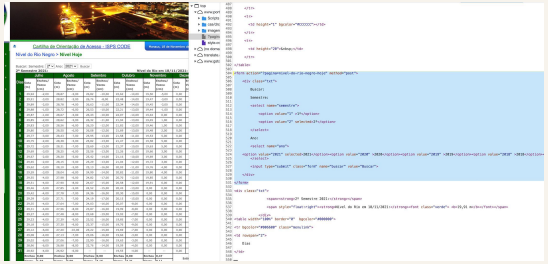
- JSON y XML para intercambio de datos
- NLP para procesamiento de texto natural
- Web scraping para extracción de datos

Herramientas Python

- BeautifulSoup para web scraping
- NLTK y spaCy para procesamiento de texto
- json y xmltodict para parseo

Adquisición de Datos





Fuentes y Métodos

Implementación Práctica

RAFAEL MARTÍN R.

Almacenamiento Eficiente

Formatos y Prácticas Recomendadas

- Uso de formatos comprimidos como Parquet o HDF5
- Particionamiento de datos por fecha o categoría
- Implementación de esquemas de indexación eficientes
- Optimización de tipos de datos y memoria

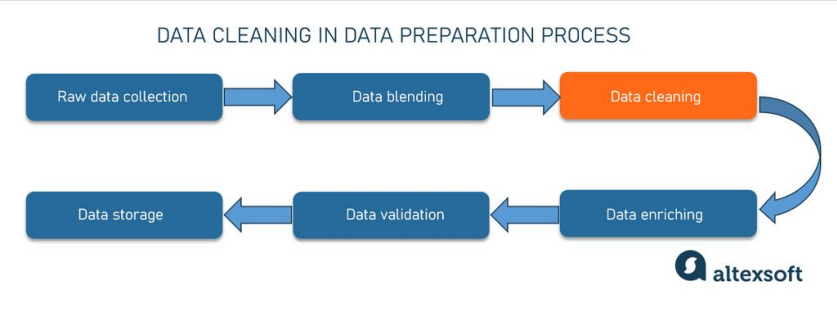
Prácticas a Evitar

- Almacenar datos sin comprimir en CSV
- Cargar datasets completos innecesariamente
- Ignorar la optimización de tipos de datos
- Descuidar la documentación del esquema

Limpieza de Datos: Fundamentos

- Identificar datos duplicados, inconsistentes y valores atípicos
- Estrategias efectivas para manejar valores faltantes (NaN)
- Técnicas estadísticas para detectar y tratar outliers
- Implementar validaciones de tipo, rango y formato
- Desarrollar flujos automatizados de limpieza con pandas
- Documentar y registrar todas las transformaciones realizadas

Limpieza de Datos: Técnicas Avanzadas




```python
if __name__ == '__main__':

    my_car = Car()
    print("I'm a car!")
    while True:
        action = input("What should I do? [A]ccelerate, [B]rake, "
                 "show [O]dometer, or show average [S]peed?").upper()
        if action not in "ABOS" or len(action) != 1:
            print("I don't know how to do that")
            continue
    while True:
        action = input("What should I do? [A]ccelerate, [B]rake, "
                 "show [O]dometer, or show average [S]peed?").upper()
        if action not in "ABOS" or len(action) != 1:
            print("I don't know how to do that")
            continue
```

Found duplicate code

Navigate to duplicate ⌥⇧⏎ More actions... ⌥⏎

Corrección y Estandarización

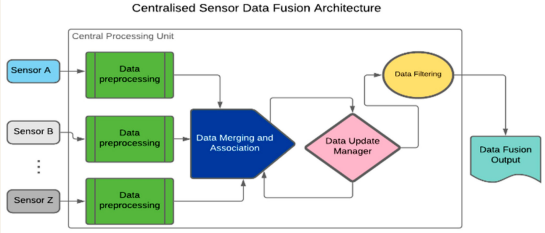
Implementa técnicas avanzadas para corregir inconsistencias y estandarizar formatos de datos utilizando pandas y expresiones regulares.

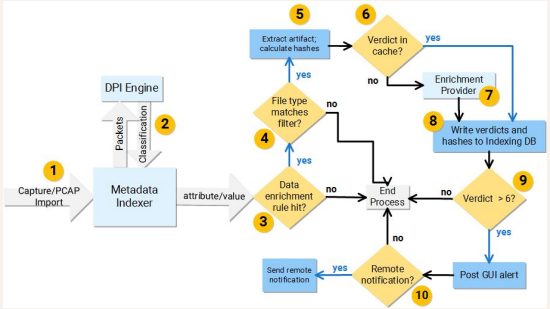
Eliminación de Duplicados

Aplica métodos eficientes para identificar y eliminar registros duplicados, considerando múltiples criterios de comparación.

RAFAEL MARTÍN R.

Enriquecimiento de Datos





Fusión y Feature Engineering

Agregación de Datos Externos

Variables Derivadas y Casos

RAFAEL MARTÍN R.

Preprocesamiento: Estandarización

Técnicas de Escalado

- StandardScaler: media 0, desviación estándar 1
- MinMaxScaler: escala datos entre 0 y 1
- RobustScaler: maneja outliers eficientemente

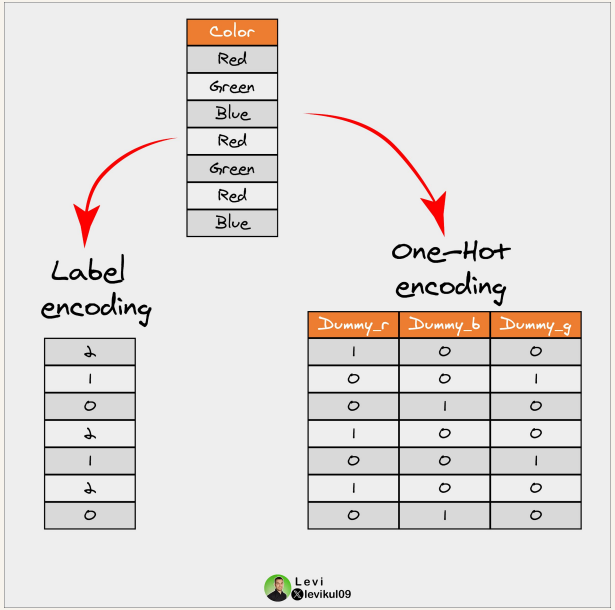
Cuándo Usar Cada Método

- Estandarización para distribuciones normales y ML
- Normalización para datos con rangos variables
- RobustScaler cuando hay outliers significativos

Preprocesamiento: Codificación

Transformando variables categóricas en formato numérico

- One-hot encoding: Crea columnas binarias para cada categoría
- Label encoding: Asigna valores numéricos a categorías ordinales
- Manejo de variables categóricas con alta cardinalidad
- Tratamiento de categorías desconocidas en datos nuevos
- Implementación eficiente usando pandas y scikit-learn
- Validación y verificación de la codificación correcta



Preprocesamiento: Discretización e Imputación

Métodos de Binning

- Binning uniforme por anchura o frecuencia
- Discretización basada en cuantiles
- Binning supervisado usando árboles de decisión

Estrategias de Imputación

- Imputación por media, mediana o moda
- KNN y métodos basados en similitud
- Imputación múltiple con modelos predictivos

Validación y Buenas Prácticas

- Evaluación de la calidad de discretización
- Métricas para validar imputaciones
- Documentación y reproducibilidad del proceso

Casos de Estudio y Mejores Prácticas

Caso de Estudio: E-commerce

Procesamiento de datos de ventas online, incluyendo limpieza de registros de transacciones y análisis de comportamiento del cliente.

Pipeline de Procesamiento

Implementación de un pipeline automatizado para datos financieros, desde la extracción hasta la validación con control de calidad.

Desafíos y Soluciones

Manejo de datos faltantes en registros médicos mediante técnicas avanzadas de imputación y validación cruzada.