CURSO SUPERIOR DE ADS

ETL, EDA e Machine Learning

Prof. Fernando Marlon Soares Figueiredo

Disciplina: Ciência de Dados e Bigdata

SECTRAS

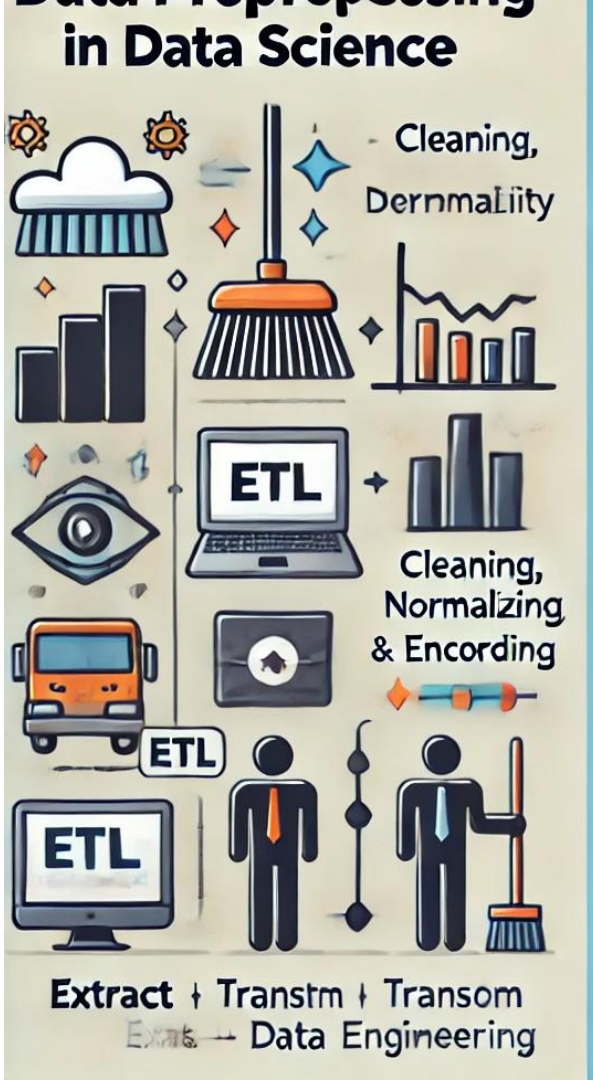
ADS

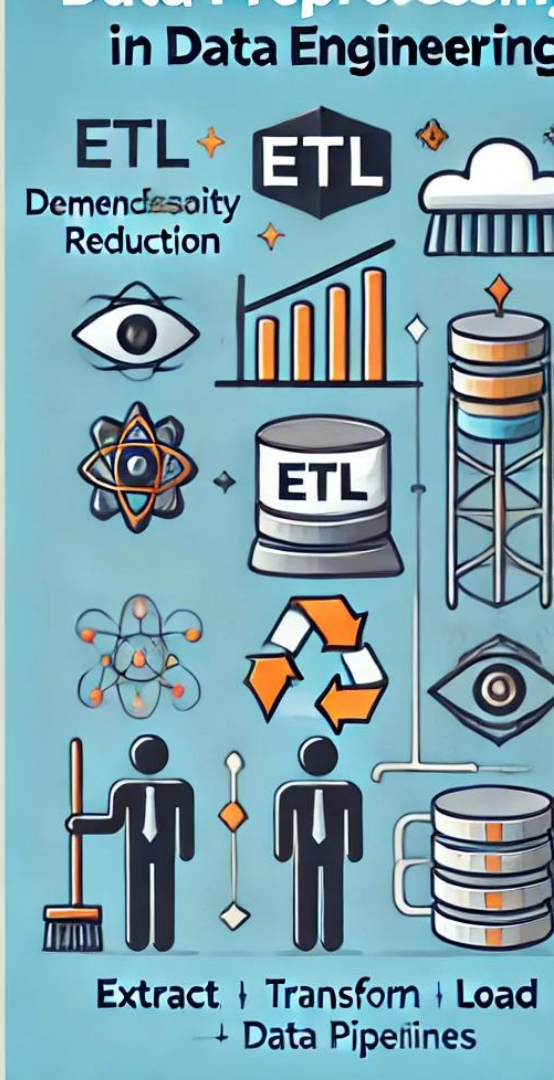

EXTRACT
TRANSFORM
LOAD
ETL
ETL

1.1. O que é ETL?

- **1.1. O que é ETL?**
- **Definição:** ETL significa "Extract, Transform, Load" e refere-se ao processo de extração de dados de várias fontes, transformação desses dados em um formato adequado para análise, e carregamento em um sistema de destino, como um Data Warehouse ou banco de dados.
- **Importância:** ETL é um processo essencial na preparação de dados, garantindo que os dados estejam limpos, organizados e prontos para serem analisados ou usados em modelos de aprendizado de máquina.

Qual a diferença entre ETL e Pré-processamento de Dados?





Contexto ETL:

- **Usado principalmente em Engenharia de Dados** para integrar e organizar dados de diferentes fontes e disponibilizá-los para análise.
- **Objetivo**: Mover dados entre sistemas de maneira estruturada e consistente para uso posterior por equipes de análise, ciência de dados ou por sistemas de BI (Business Intelligence).

Contexto Pré processamento de Dados

- **Usado principalmente em Ciência de Dados** e **Aprendizado de Máquina**, quando os dados estão sendo preparados para serem introduzidos em modelos preditivos.
- **Objetivo**: Garantir que os dados estejam limpos, consistentes e formatados de uma maneira que maximiza o desempenho de algoritmos de aprendizado de máquina.

Comparativo

Aspecto	ETL	Pré-processamento de Dados
Objetivo	Integrar e mover dados entre sistemas	Preparar dados para análise ou modelagem
Aplicação	Engenharia de Dados e Integração de Dados	Ciência de Dados e Aprendizado de Máquina
Fases	Extração, Transformação, Carregamento	Limpeza, Normalização, Codificação, etc.
Destino Final	Data Warehouses ou bancos de dados	Algoritmos de aprendizado de máquina
Transformação	Focado em integração e organização de dados	Focado em melhorar a qualidade e usabilidade dos dados
Uso de Ferramentas	Ferramentas de ETL como Talend, Apache Nifi	Ferramentas de ciência de dados como Pandas, Scikit-learn

- **1.2. Componentes do Processo ETL:**
- **Extract (Extração):** Coleta de dados brutos de uma ou várias fontes.
- **Transform (Transformação):** Processamento e modificação dos dados para atender aos requisitos de análise.
- **Load (Carregamento):** Inserção dos dados transformados em um sistema de armazenamento ou banco de dados para análise posterior.

2. Extract: Extração de Dados

- **2. Extract: Extração de Dados**
- **2.1. O que é Extração de Dados?**
- **Definição:** A etapa de extração envolve a coleta de dados de diversas fontes, que podem incluir bancos de dados, APIs, arquivos CSV, planilhas, e outros.
- **Desafios:** Dados podem estar em formatos variados e dispersos em diferentes fontes, exigindo ferramentas e técnicas adequadas para extrair os dados corretamente.

- **2.2. Técnicas de Extração**
- **Conexões com Bancos de Dados:** Utilização de SQL para extrair dados diretamente de um banco de dados.

```python
import pandas as pd
import sqlite3

# Conectar ao banco de dados SQLite
conn = sqlite3.connect('database.db')

# Executar uma consulta SQL e carregar os dados em um DataFrame
df = pd.read_sql_query("SELECT * FROM tabela_exemplo", conn)
print(df.head())
```

- **APIs:** Utilização de APIs para extrair dados em tempo real.

```
import requests

# Realizar uma requisição GET à API
response = requests.get('https://api.exemplo.com/dados')

# Converter a resposta JSON para um DataFrame Pandas
df = pd.DataFrame(response.json())
print(df.head())
```

- **Arquivos CSV/Excel:** Extração de dados a partir de arquivos locais.

```
df = pd.read_csv('dados.csv')
print(df.head())
```

3. Transform: Transformação de Dados

- **3.1. O que é Transformação de Dados?**
- **Definição:** Transformar dados envolve limpar, formatar e modificar dados extraídos para torná-los adequados para análise. Isso pode incluir a remoção de valores faltantes, normalização, agregação, e combinação de dados.

- **3.2. Técnicas Comuns de Transformação**
- **Limpeza de Dados:** Remoção de valores faltantes, correção de dados inconsistentes, e tratamento de outliers.

```
# Remover linhas com valores faltantes
df.dropna(inplace=True)
```

3.2. Técnicas Comuns de Transformação

- **Normalização e Padronização:** Escalonamento de dados para que estejam em um intervalo comum.

- **Agregação:** Resumo de dados, como calcular médias, somas, etc.

- **Combinação de Dados:** Mesclagem de várias fontes de dados em um único conjunto de dados.

Agregação: Resumo de dados, como calcular médias, somas, etc.

- Exemplo Prático:

```python
df_aggregated = df.groupby('categoria').mean()
print(df_aggregated)
```

Combinação de Dados: Mesclagem de várias fontes de dados em um único conjunto de dados.

- Exemplo Prático:

```python
df_combined = pd.merge(df1, df2, on='chave_comum')
print(df_combined.head())
```

Load: Carregamento de Dados

- **4.1. O que é Carregamento de Dados?**
- **Definição:** A etapa de carregamento envolve a inserção dos dados transformados em um sistema de destino, como um Data Warehouse, banco de dados relacional, ou uma estrutura de dados em memória para análise posterior.

- Carregamento em Bancos de Dados:
  - Exemplo Prático:

```python
# Inserir dados transformados em uma tabela de banco de dados
df.to_sql('tabela_transformada', conn, if_exists='replace', index=False)
```

- Carregamento em Arquivos CSV/Excel:
  - Exemplo Prático:

```python
df.to_csv('dados_transformados.csv', index=False)
```

- **Carregamento em Data Warehouses:** Uso de ferramentas como Apache Airflow, Talend, ou serviços de cloud como AWS Redshift para carregar grandes volumes de dados.

FERRAMENTAS DE ETL

Ferramenta	Licenciamento	Características Principais	Uso Comum
Apache NiFi	Open Source	Interface gráfica para arrastar e soltar, suporte a dados em fluxo	Processamento de dados em tempo real e integração de dados
Apache Airflow	Open Source	Gerenciamento de fluxo de trabalho, programação em Python	Agendamento e orquestração de pipelines ETL em larga escala
Informatica PowerCenter	Comercial	Ferramenta robusta e escalável, suporte a integração de dados empresariais	Integração de dados empresariais, migração de dados
SSIS (SQL Server Integration Services)	Comercial	Integrado ao SQL Server, ETL baseado em SQL, integração com ecossistema Microsoft	Integração e transformação de dados em ambientes SQL Server
AWS Glue	Comercial (Nuvem)	ETL totalmente gerenciado, integração nativa com outros serviços AWS	ETL para big data, integração com AWS Redshift e S3
Google Cloud Dataflow	Comercial (Nuvem)	Processamento de dados em fluxo e em lote, baseado no modelo Apache Beam	ETL na nuvem, processamento de dados em tempo real

Notas:

- **Open Source:** Apache NiFi e Apache Airflow são ferramentas flexíveis e amplamente usadas para fluxos de dados em tempo real e orquestração de pipelines.
- **Comercial:** Informatica PowerCenter e SSIS são escolhas populares para ambientes empresariais, oferecendo suporte robusto e integração com plataformas existentes.
- **Nuvem:** AWS Glue e Google Cloud Dataflow são ideais para quem trabalha em ambientes de nuvem, com forte integração com seus respectivos ecossistemas (AWS e Google Cloud).

Análise Exploratória de Dados (EDA)

Definição:

- A **Análise Exploratória de Dados (EDA)** é uma abordagem inicial para explorar e visualizar um conjunto de dados, antes de aplicar técnicas de modelagem preditiva ou algoritmos de aprendizado de máquina. O objetivo principal do EDA é **entender a estrutura** e as características dos dados, identificar padrões, e detectar possíveis anomalias ou outliers.
- **Principais Objetivos da EDA:**
- **Compreensão do conjunto de dados:** Entender quais variáveis estão presentes e seus tipos (categóricas, numéricas, temporais, etc.).

Principais Objetivos da EDA:

Compreensão do conjunto de dados: Entender quais variáveis estão presentes e seus tipos (categóricas, numéricas, temporais, etc.).

Identificação de padrões: Explorar relações entre variáveis e comportamentos gerais dos dados.

Detecção de outliers e valores faltantes: Identificar valores extremos ou dados ausentes que podem influenciar a análise futura.

Resumo estatístico: Gerar medidas de tendência central, dispersão e outras estatísticas descritivas.

2. Importância da EDA

- A EDA é uma etapa fundamental em qualquer projeto de ciência de dados porque:
- **Prevenção de Erros:** Identificar problemas nos dados, como valores incorretos ou faltantes, ajuda a evitar que esses erros afetem a modelagem.
- **Geração de Insights:** Descobrir padrões ocultos, correlações, e insights que podem não ser evidentes à primeira vista.
- **Melhorias na Modelagem:** A EDA ajuda a orientar a escolha de algoritmos e técnicas adequadas para os dados, além de melhorar a performance dos modelos.

3. Técnicas Comuns de EDA

- **3.1. Estatísticas Descritivas**
- As estatísticas descritivas oferecem um resumo rápido dos dados e incluem:
- **Média, Mediana e Moda:** Medidas de tendência central que descrevem o valor médio ou mais frequente de uma variável.
- **Desvio Padrão e Variância:** Medidas de dispersão que indicam como os dados estão distribuídos em relação à média.
- **Quartis e IQR (Intervalo Interquartil):** Usados para entender a distribuição dos dados e detectar outliers.

3.2. Visualização de Dados

- Visualizações são fundamentais na EDA para identificar padrões e relações entre variáveis.
- **Gráficos Comuns:**
- **Histogramas:** Exibem a distribuição de variáveis numéricas.
- **Gráficos de Barras:** Usados para variáveis categóricas.
- **Boxplots:** Muito úteis para detectar outliers e entender a dispersão dos dados.
- **Gráficos de Dispersão:** Revelam relações entre duas variáveis numéricas.

3.3. Matriz de Correlação

- A matriz de correlação mostra a relação entre variáveis numéricas. A correlação varia entre -1 (correlação negativa perfeita) e 1 (correlação positiva perfeita). Uma correlação próxima de 0 indica que as variáveis não estão relacionadas.

```
# Matriz de correlação
corr_matrix = df.corr()

# Mapa de calor da correlação
sns.heatmap(corr_matrix, annot=True)
plt.title('Matriz de Correlação')
plt.show()
```

3.4. Análise de Outliers

- Outliers podem distorcer análises e influenciar modelos de forma indesejada. Detectar e, se necessário, lidar com outliers é uma etapa crucial.
- **Exemplo:**
- **Boxplot:** Utilizado para identificar outliers, que aparecem como pontos fora das "caixas" do boxplot.
- **Z-Score:** Uma técnica comum para detectar outliers é o cálculo do Z-score, que mostra quantos desvios padrão um valor está da média.

Aprendizado de Máquina

Estou sempre disposto a aprender apesar de nem sempre gostar de ser ensinado.
—Winston Churchill

Muitas pessoas imaginam que data science é, em maior parte, aprendizado de máquina e que os cientistas de dados constroem, praticam e ajustam modelos de aprendizado de máquina o dia inteiro. E, novamente, muitas desas pessoas não sabem o que *é* aprendizado de máquina. Na verdade, data science é mais transformar problemas empresariais em problemas de dados e coletar, entender, limpar e formatar os dados, após o que aprendizado de máquina é praticamente uma consideração subsequente. Mesmo assim, é uma referência interessante e essencial que você deve saber a fim de praticar data science.

1. O que é Aprendizado de Máquina?

- Aprendizado de máquina (Machine Learning) é a técnica de usar algoritmos para analisar dados, aprender com esses dados e, em seguida, fazer previsões ou tomar decisões com base no aprendizado.
- A ideia central é construir modelos que possam generalizar a partir de dados observados para novas entradas.

2. Tipos de Machine Learning

- Existem três principais tipos de aprendizado em Machine Learning:
- **2.1. Aprendizado Supervisionado:**
- **Definição**: O algoritmo aprende a partir de um conjunto de dados rotulados, ou seja, onde o **input** (entrada) e o **output** (saída) são conhecidos.
- **Exemplos**:
    - **Classificação**: Prever se um e-mail é "spam" ou "não spam".
    - **Regressão**: Prever o preço de uma casa com base no tamanho e localização.

2.2. Aprendizado Não Supervisionado:

- **Definição**: O algoritmo recebe dados sem rótulos, e deve descobrir padrões ocultos ou estruturas nesses dados.
- **Exemplos**:
  - **Agrupamento (Clustering)**: Agrupar clientes com base em padrões de comportamento.
  - **Redução de Dimensionalidade**: Reduzir a quantidade de variáveis em um conjunto de dados, mantendo a maior parte da informação (ex: PCA - Análise de Componentes Principais).

2.3. Aprendizado por Reforço:

- **Definição**: O algoritmo toma decisões em um ambiente dinâmico, e aprende a partir de tentativa e erro, recebendo **recompensas** ou **punições**.
- **Exemplo**: Treinamento de robôs para jogar videogames ou controlar veículos autônomos.

2. Modelagem em Aprendizado de Máquina

- O processo envolve a criação de modelos matemáticos baseados em dados de treino e a validação deles com novos dados (dados de teste).
- Modelos em aprendizado de máquina são baseados na premissa de ajuste aos dados, balanceando complexidade e generalização.

3. Sobreajuste (Overfitting) e Subajuste (Underfitting)

- **Sobreajuste:** Quando o modelo é muito complexo, ele se ajusta demais aos dados de treino, capturando o ruído. Isso leva a um desempenho ruim com novos dados.
- **Subajuste:** Quando o modelo é muito simples, ele não captura a complexidade dos dados, resultando em previsões imprecisas, mesmo com os dados de treino.

. Compromisso entre Polarização e Variância (Bias-Variance Tradeoff)

Polarização (bias): A tendência do modelo de fazer simplificações excessivas que causam erros sistemáticos.

- Variância: O modelo captura o ruído dos dados, resultando em erros em novas previsões.
- O desafio é encontrar o equilíbrio entre polarização e variância para minimizar o erro total.

6. Extração e Seleção de Características (Feature Extraction and Selection)

- Identificar as variáveis mais relevantes (características) que ajudam a construir o modelo.
- É crucial evitar características irrelevantes que possam aumentar a complexidade e o risco de sobreajuste.

7. Validação Cruzada (Cross-Validation)

- Técnica usada para garantir que o modelo funciona bem com novos dados, dividindo o conjunto de dados em várias partes para treino e teste múltiplas vezes.
- A validação cruzada é uma ferramenta importante para avaliar a capacidade de generalização do modelo.

8. Técnicas e Algoritmos Comuns

- **Regressão Linear:** Estima a relação entre variáveis dependentes e independentes para fazer previsões.
- **Árvores de Decisão:** Usa uma estrutura hierárquica de decisões para fazer previsões.
- **K-Nearest Neighbors (KNN):** Classifica novas amostras com base nas amostras mais próximas em termos de distância.
- **Redes Neurais:** Modelos inspirados no funcionamento do cérebro humano, amplamente usados para problemas complexos como reconhecimento de imagem e fala.

Exemplo Prático:

- **Problema:** Prever o preço de casas com base em dados de características como área, número de quartos, localização, etc.
- **Solução:** Usar uma combinação de regressão linear e validação cruzada para construir e validar o modelo, evitando sobreajuste.

REFERÊNCIAS

- HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. **The Elements of Statistical Learning: Data Mining, Inference, and Prediction.** New York: Springer, 2009.
- KIMBALL, R.; CASERTA, J. **The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleaning, Conforming, and Delivering Data.** Indianapolis: Wiley, 2004.
- GERON, A. **Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow.** 2. ed. Sebastopol: O'Reilly Media, 2019.
- DASU, T.; JOHNSON, T. **Exploratory Data Mining and Data Cleaning.** Hoboken: Wiley-Interscience, 2003.
- JOLLIFFE, I. T. Principal Component Analysis. 2. ed. New York: Springer, 2002.

REFERÊNCIAS

- Christopher., Chatfield, (1995). Problem solving : a statistician's guide 2nd ed. London: Chapman & Hall. ISBN 0412606305. OCLC 32881624
- Tukey, John W. (1962). «The Future of Data Analysis». The Annals of Mathematical Statistics (em inglês). 33 (1): 1–67. ISSN 0003-4851. doi:10.1214/aoms/1177704711