Introdução à Ciência de Dados e Aprendizado de Máquina em Python

Sobre mim

- Físico;
- Cosmologia;
- Aluno (5º semestre) ADS – FATEC;
- Curso: Análise de Dados Complexos – UNICAMP;
- Python/Django;

Sumário

1. O que é Ciência de Dados?
2. Pacotes disponíveis em Python;
3. Carregamento e limpeza dos dados;
4. Pré-processamento;
5. Aprendizado de Máquina;
    1. ~~Classificação não-supervisionada;~~
    2. Classificação supervisionada;
    3. ~~Regressão;~~
    4. Métricas de avaliação;

O que é Ciência de Dados?

NIST - National Institute of Standards and Technology

Data science is the extraction of actionable knowledge directly from data through a process of discovery, or hypothesis formulation and hypothesis testing.*

**NIST Big Data Interoperability Framework: Volume 1, Definitions*

Pacotes disponíveis em Python

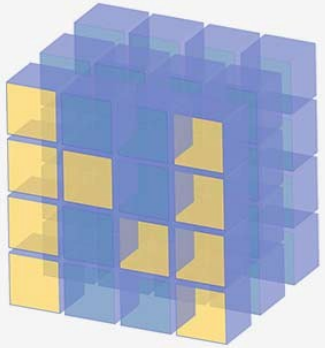
Pandas: Análise e manipulação de dados

- Objeto DataFrame para a manipulação de dados;
- Ferramentas para leitura de dados de arquivos para memória;
- Ferramentas para tratar dados faltantes;
- Fatiamento de dados, indexação e particionamento de conjuntos de dados;
- Inserção de dados (colunas e linhas);
- Agrupamento, join, merge;
- Funcionalidades para séries temporais;

Pacotes disponíveis em Python

Numpy: suporte a manipulação de arrays multidimensionais

- objeto ndarray: de tipo homogêneo;
- fatiamento e reformatação de vetores e matrizes;
- mudar dimensões, joins, splits;
- álgebra linear;
- números aleatórios;



Pacotes disponíveis em Python

Scikit-learn: algoritmos de aprendizado de máquina (*machine learning*)

- classificação: identificação de categorias;
- regressão: predição de valores contínuos;
- agrupamento em conjuntos por similaridade;
- pré-processamento: normalização e extração de características;
- redução de dimensionalidade: reduzir o número de variáveis;
- seleção de modelos: comparação e validação;

Pacotes disponíveis em Python

Matplotlib: vizualização de dados;

Seaborn: gráficos mais bonitos que o Matplotlib, focado em estatística;

Além dos pacotes

Jupyter Notebook: ambiente computacional web interativo para criação de documentos

Carregamento dos dados

***Iris flower data set*: conjunto apresentado por Ronald Fisher em 1936 em *The use of multiple measurements in taxonomic problems*, onde são apresentadas variações morfológicas em três espécies**

https://archive.ics.uci.edu/ml/datasets/iris

Carregamento dos dados

Import pandas as pd

iris = pd.read_csv('.\data\iris.data')

Algumas opções:

sep (ou **delimiter**)→ string: separador

names → list, array, ... : rótulos das colunas

Também é possível ler Excel, JSON, HTML, SQL, etc...

Carregamento dos dados

Explorando os dados:

DataFrame.head(n) → exibe as n (default = 5) primeiras linhas;

DataFrame.tail(n) → exibe as n (default = 5) últimas linhas;

DataFrame.sample(n) → exibe as n (default = 5) linhas aleatórias;

Carregamento dos dados

Fatiando os dados:

DataFrame[indexer] → retorna a fatia correspondente ao indexador;

DataFrame.loc[linhas, colunas] → retorna a fatia correspondente aos indexadores;

DataFrame.iloc[linhas, colunas] → retorna a fatia correspondente aos indexadores **inteiros (posições)**;

Carregamento dos dados

Informações sobre o conjunto de dados:

DataFrame.info() → retorna informações básicas: # linhas, # colunas, tipo de cada coluna, **espaço na memória**, **# de valores válidos por coluna**

DataFrame.describe() → estatística descritiva por coluna: contagem, média, desvio padrão, max., min., quartis;

Limpeza dos dados

Processo com objetivo de encontrar e corrigir (ou eliminar) valores corrompidos ou incorretos;

Garbage in → Garbage out

Pode envolver:

- ajustar o tipo de dados;
- **substituir (ou remover) registros incompletos;**
- resolver conflitos;
- *outlier*.

Limpeza dos dados

DataFrame.isna() → retorna um objeto booleano de mesmo tamanho que indica valores NA (None ou numpy.NaN);

DataFrame.dropna(axis) → axis (0 ou ‘index’, 1 ou ‘columns’), default 0: remove linhas (colunas) com pelo menos um NA;

DataFrame.fillna(valor) → valor (escalar, dict., *Series*): substitui NA’s pelos valores passados;

Pré-processamento

Transformar os dados em representações mais apropriadas para os estimadores. Pode envolver:

- escalonar (média, max.←→min., max. Abs., etc);
- codificar (categorias → números);
- criar *features;*
- extrair *features;*

Pré-processamento: PCA

PCA (*Principal Component Analysis*):

- converte um conjunto de valores possivelmente correlacionadas num conjunto de valores de linearmente não correlacionadas ;

- é definida de forma que o primeiro componente principal tem a maior variância;

- projeta os dados em um espaço de menor dimensão;

APRENDIZADO DE MÁQUINA!!!

Aprendizado de máquina

Algoritmos que montam um modelo a partir de amostras de dados a fim de fazer previsões ou decisões guiadas pelos dados ao invés de simplesmente seguindo instruções programadas

- Aprendizado supervisionado: são apresentadas entradas e saídas;

- Aprendizado não-supervisionado: apenas entradas;

- Aprendizado por reforço: é fornecido feedback;

Aprendizado de máquina

Tipo de problemas:

- Classificação: saídas discretas;

- Regressão: saídas contínuas;

- Agrupamento: dividir em grupos, sem conhecê-los previamente;

Aprendizado de máquina

Receita:

- **Passo 0:** Limpeza e tratamento dos dados;

  (**NaN**, **Scalers**, **Encoders**, **PCA**, etc...)
- **Passo 1:** Dividir os dados em **treino**, **validação** e **testes**;

  (**train_test_split**)
- **Passo 2:** Treinar o modelo;
- **Passo 3:** Avaliar o modelo;
- **Passo 4:** Recomeçar