Relatório 4 - Principais Bibliotecas e Ferramentas Python para Aprendizado de Máquina

Edryck Freitas Nascimento

1. Introdução

O objetivo desta aula foi aprender sobre as principais bibliotecas (Numpy e Pandas) e ferramentas (Jupyter Notebook) da linguagem Python, para análise de dados e aprendizado de máquina. A tarefa consistia em fazer as seguintes seções do curso Python para Data Science e Machine Learning, de forma mais específica: Seção 3 – Jupyter Notebook, Seção 5 – Numpy, Seção 6 – Pandas e Seção 7, que são os exercícios.

2. Desenvolvimento

2.1. Jupyter Notebook

O Jupyter Notebook é um aplicativo para a criação de notebooks, pertencente ao Projeto Jupyter. Ele oferece novas maneiras rápidas e interativas de prototipar e explicar o código, explorar e visualizar os dados e compartilhar suas ideias com outras pessoas.[1]

Ele permitiu a criação de documentos que combina células de código e texto explicativo (células Markdown). Foi utilizado para realizar a execução do código passo a passo, permitindo a visualização de forma imediata de saída de dados e erros, o que facilitou o processo de documentação da lógica e análise.

2.2. Numpy

O *Numpy* é uma biblioteca do Python usada para trabalhar com *Arrays*, também usada para trabalhar com álgebra linear. Ela é o bloco de construção de todas as outras bibliotecas usadas para análise de dados, ela é extremamente rápida, devido seus principais métodos são compilados em C.

[1] https://jupyter-notebook.readthedocs.io/en/stable/notebook.html

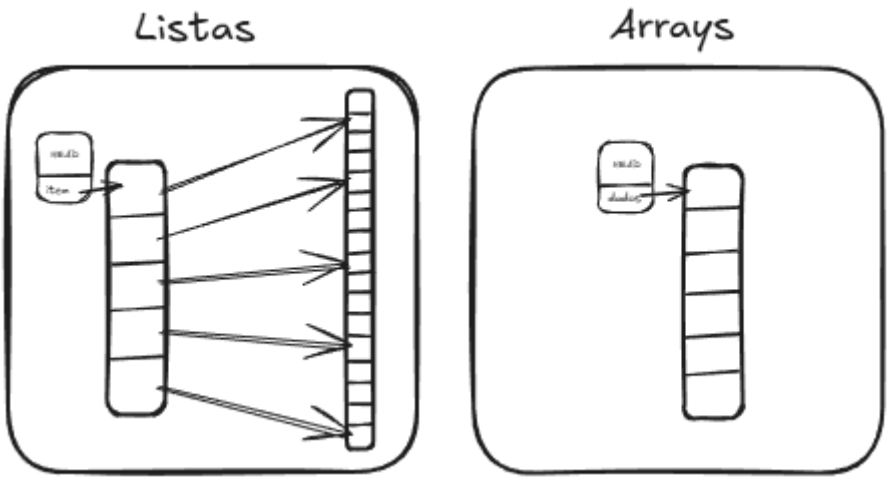

Figura 1. NumPy é rápido porque os dados moram juntos na memória (autoria própria).

Durante o curso, foi mostrado a diferença entre listas nativas do Python e *Arrays NumPy*, focando na criação de vetores e matrizes. Foram utilizados métodos para gerar sequências numéricas (como *arange*, *linspace*) e matrizes especiais (matriz de zeros, uns e identidade, que são essenciais para trabalhar com álgebra linear).

Além disso, foi explorado o uso do módulo *random* para criar amostras de dados, baseadas em distribuições uniformes ou normais. A manipulação de elementos dentro das *arrays*, assim ajudando selecionar trechos específicos dos dados ou alterar vários valores de uma vez.

Também foi visto a aplicação de funções básicas, como raiz quadrada, exponencial, seno, e operações estatísticas (média, desvio padrão, mínimo e máximo) de forma direta sobre as *arrays*, sem criar laços de repetição.

2.3. Pandas

O *Pandas* é uma biblioteca de código aberto escrita sobre o *Numpy*. Ela é a biblioteca para análise e manipulação de dados para Python, ela permite a visualização e limpeza de dados, podendo trabalhar com dados de diversos tipos diferentes. Possui métodos próprios para a visualização de dados e tem uma estrutura parecida com o Excel.

O conteúdo do curso auxiliou no entendimento e manipulação de *Series* (parecido com vetores com rótulos) e dos *DataFrames* (conjunto de *Series*).

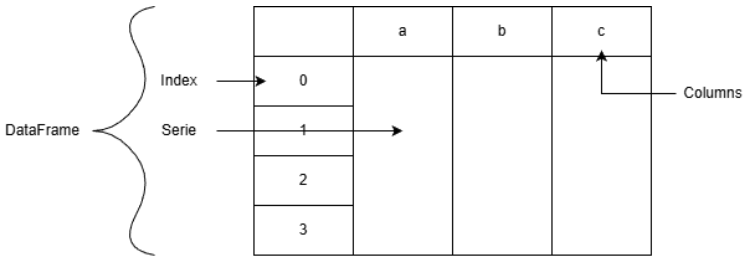

Figura 2. “Anatomia” do DataFrame (autoria própria).

Foi analisada a diferença entre a seleção por rótulos (*loc*) e a seleção por posição numérica (*iloc*), além da seleção condicional baseada em operações lógicas.

O curso incluiu também técnicas para lidar com valores nulos (NaN), incluindo a remoção de registros incompletos ou o preenchimento com médias ou propagação de valores vizinhos e o uso do método *groupby* para agregar dados por categorias e extrair estatísticas resumidas.

Também foram mostradas formas de combinar diferentes DataFrames (Concatenação, *Merge* e *Join*), utilizando lógica parecida com a de bancos de dados SQL. Por fim, a capacidade do Pandas de ler e exportar dados de diversas fontes, como arquivos CSV, planilhas do Excel e tabelas HTML da web.

3. Conclusão

A integração entre NumPy e Pandas no Jupyter fortalece a base para a Ciência de Dados, essa integração junta a performance da computação para cálculos com a flexibilidade da manipulação tabular. Substituindo planilhas manuais por esses *frameworks*, se mostrou muito melhor que o uso manual de planilhas, dessa forma, garantindo velocidade e organização para o desenvolvimento de projetos.

Referências:

Jupyter Team, “The Jupyter Notebook”, https://jupyter-notebook.readthedocs.io/en/stable/notebook.html.