Proposta de Comparação da Eficiência e Escalabilidade de Bibliotecas Python para Manipulação e Análise de Dados

Gabriel R. B. Cirillo[1], **Guilherme Galante**[1]

[1] Ciências da Computação - Universidade Estadual do Oeste do Paraná (UNIOESTE)
Caixa Postal – 85.819-110 – Cascavel – PR– Brazil

gabriel.cirillo@unioeste.br, guilherme.galante@unioeste.br

Resumo. *Esta proposta de pesquisa visa analisar a eficiência e escalabilidade de bibliotecas Python para manipulação e análise de dados. O objetivo central consiste em identificar as soluções mais adequadas para lidar com grandes volumes de dados, considerando o desempenho em termos de tempo de execução, uso de memória e capacidade de escalabilidade. O estudo se propõe a comparar bibliotecas como Pandas, Polars, Dask, Modin e PySpark, contribuindo para a criação de diretrizes mais eficazes no uso dessas bibliotecas.*

1. Introdução

A crescente geração de dados e a necessidade de sua análise eficiente têm impulsionado o uso de ferramentas computacionais especializadas. Entre as soluções amplamente adotadas, a biblioteca Pandas [Mckinney 2011] destaca-se como uma das principais bibliotecas Python para este fim. No entanto, seu desempenho é limitado por operar em apenas um núcleo do processador [Petersohn 2018], tornando-a inadequada para grandes volumes de dados.

No cenário atual, surgem alternativas como Polars [Pola-rs 2024], Modin [Modin 2024], PySpark [Foundation 2024] e Dask [Dask Development Team 2024], entre outras, que prometem superar as limitações do Pandas ao explorar a escalabilidade e o paralelismo. Contudo, a falta de estudos comparativos abrangentes dificulta a escolha da biblioteca mais eficiente para diferentes contextos. Dessa forma, esta pesquisa objetiva comparar bibliotecas *Python* para análise de dados, com ênfase na eficiência e escalabilidade, abordando aspectos como consumo de recursos computacionais e desempenho em operações de grande escala. Para a escalabilidade, serão realizados testes em diferentes configurações de *hardware*, variando o número de núcleos do processador e o tamanho da memória disponível.

2. Objetivos

O objetivo geral desta proposta é investigar e comparar a eficiência e escalabilidade de bibliotecas Python para manipulação e análise de dados. Os objetivos específicos incluem: Avaliar o desempenho das bibliotecas em termos de tempo de execução, uso de memória e processador. Analisar a escalabilidade das ferramentas em cenários com diferentes volumes de dados e recursos computacionais. Identificar os contextos em que cada biblioteca apresenta maior eficiência.

3. Metodologia

A metodologia adotada neste trabalho compreende quatro etapas. A primeira etapa consiste na seleção das bibliotecas. Na sequência, definem-se as cargas de trabalho e o *benchmark* a ser utilizado. Até o momento, optou-se por utilizar o *benchmark* TPC-H, que foi desenvolvido para simular uma carga de trabalho de consultas *ad hoc*, representando um cenário em que usuários conectados ao banco de dados enviam consultas individuais [Pöss and Floyd 2000]. Esse *benchmark* é amplamente adotado para comparar bancos de dados, permitindo avaliar seu desempenho e eficiência. Para os testes, serão utilizadas cargas de trabalho com fatores de escala de 1 GB e 10 GB de dados, variando ainda o número de núcleos do processador e a memória alocada para a instância de máquina virtual que será usada como ambiente computacional.

A terceira etapa consiste na execução dos testes. Serão executadas 7 das 22 consultas do TPC-H, devido a algumas limitações técnicas das bibliotecas e ao esforço necessário para adaptar todas as consultas, medindo o tempo de execução, o consumo de memória e a utilização do processador. Analisa-se, também, a eficiência das bibliotecas em cada cenário e sua escalabilidade, medida por meio do valor de *speedup*, conforme se altera o tamanho da entrada de dados e a quantidade de recursos. Por fim, avaliam-se os resultados obtidos, levando em consideração, para cada biblioteca, o desempenho, o uso de paralelismo da arquitetura e a capacidade de escalar nesse ambiente.

4. Resultados e Contribuições Esperados

Espera-se que esta pesquisa identifique as bibliotecas mais adequadas para diferentes cenários de análise de dados, destacando suas vantagens e limitações em situações que demandam escalabilidade. O estudo também pretende fornecer orientações práticas para a escolha e aplicação dessas ferramentas, contribuindo com uma análise detalhada e prática do uso de bibliotecas Python para esse fim. Além disso, os resultados podem influenciar o desenvolvimento futuro dessas ferramentas, auxiliando pesquisadores e profissionais na escolha de soluções mais eficientes e escaláveis.

References

Dask Development Team (2024). Dask: Scale the python tools you love. `https://www.dask.org`. Acesso em: 23 Nov 2024.

Foundation, A. S. (2024). Apache spark: A unified analytics engine for large-scale data processing. Acesso em: 23 Nov 2024.

Mckinney, W. (2011). pandas: a foundational python library for data analysis and statistics. *Python High Performance Science Computer*.

Modin (2024). Modin: Scale your pandas workflows by changing a single line of code. Acesso em: 23 Nov 2024.

Petersohn, D. (2018). Scaling interactive data science transparently with modin. Master's thesis, EECS Department, University of California, Berkeley.

Pola-rs (2024). Polars: Lightning-fast dataframe library for rust and python. Acesso em: 23 Nov 2024.

Pöss, M. and Floyd, C. (2000). New tpc benchmarks for decision support and web commerce. *ACM SIGMOD Record*, 29(4):64–71.