Avaliação

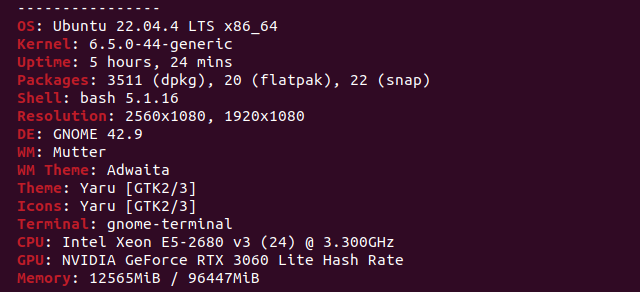
Premissas:

Utilizando os seguintes instruções e requisitos:

Instruções:  
1. Implemente uma solução para ler o arquivo 'vendas\_grandes.csv' de forma eficiente, considerando o grande volume de dados.  
2. Identifique o produto mais vendido em termos de quantidade e canal.  
3. Determine qual pais e região teve o maior volume de vendas (em valor).  
4. Calcule a média de vendas mensais por produto, considerando o período dos dados disponíveis.  
  
Requisitos:  
- Sua solução deve ser capaz de rodar em uma máquina com memória limitada, não assuma que o arquivo inteiro pode ser carregado na memória de uma vez.  
- Use técnicas como leitura em partes (chunking).  
- Priorize a eficiência do processamento.  
- O uso de bibliotecas como Pandas é permitido, especialmente com seu recurso de leitura em chunks.  
- Documente seu código adequadamente e inclua comentários explicativos sobre suas escolhas de implementação.

Foram usados mais 2 framework para comparação com Pandas: Duckdb e Polars. Cada com seu script python diferente.

Foi executado em ambiente linux com a seguint configuração:



Para execução foi criado um script que já faz as instalação dos pacotes necessários para execução e monitoramento.

Esse mesmo arquivo irá gerar um arquivo de teste com +/-5GB baseado na concatenação do arquivo de teste (624mb)

./start.sh

Após o termino será gerado 3 arquivos de imagem para efeito de comparação:

plot\_cogna\_duckdb.png

plot\_cogna\_pandas.png

plot\_cogna\_polars.png

Comparativos:

|  |  |  |  |
| --- | --- | --- | --- |
| **Framework** | **%CPU** | **Memoria** | **Tempo** |
| Panda | 400 | 6500 | 500 |
| Ducdb | 2000 | 8000 | 9 |
| Polaris | 2500 | 14000 | 9 |

|  |  |  |
| --- | --- | --- |
| **Framework** | **Pros** | **Contra** |
| Panda | Mais popular Mais dinamismo | Lento Não indicado para grandes volumes |
| Duckdb | Rápido Baixo consumo de memória Baixo processamento Usa SQL como linguagem | Consumo médio de memória |
| Polaris | Pouco consumo de memoria Rápido | Alto consumo de memória Ainda é pouco conhecido |

Conclusão:

Apesar das otimizações para melhor uso de memória, o pandas é muito lento em comparação aos demais frameworks.

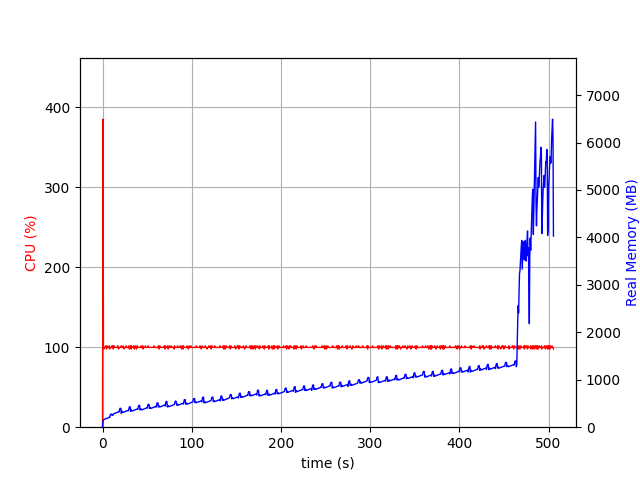
DuckDb se mostrou mais rápido e com consumo de memoria baixo.

Polars teve um desempenho um pouco melhor no uso de CPU, mas com elevado consumo de memória.

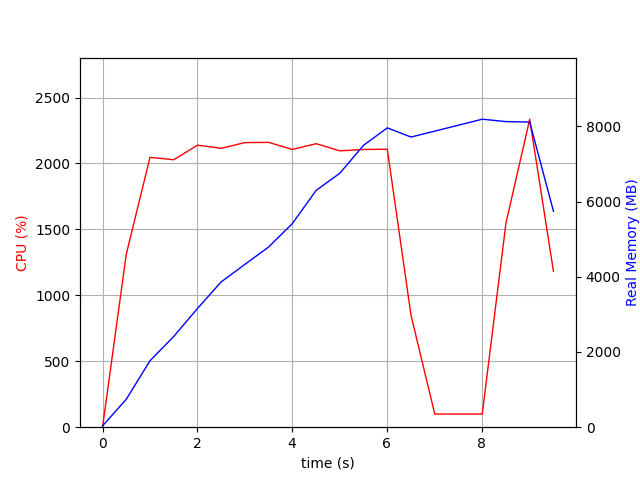
Dentre os 3 frameworks, o Duckdb é mais indicados para maquinas com pouca memória, e com bom desempenho no processamento de um arquivo de 6GB.

Graficos comparativos:

Pandas:



Duckdb:



Polars

