Maria Eduarda Villéla Silva - RA: 212974
31 de agosto de 2025

Desafio 03 - ME315

Arquivo parquet

É um formato de arquivo colunar, isto é, armazena dados por coluna (não por linha). Tal formato foi criado para guardar grandes volumes de dados de forma eficiente, possuindo compressão automática, o que ocupa menos espaço em disco. O parquet é muito usado em big data e ciência de dados para armazenar tabelas grandes.

Arquivo JSON (JavaScript Object Notation)

É um formato de arquivo baseado em texto, usado para representar dados estruturados, isto é, ele usa pares de chave-valor e listas. Esse tipo de arquivo pode armazenar desde objetos simples até estruturas aninhadas (listas dentro de listas). O JSON é muito usado em APIs, troca de dados entre sistemas e armazenamento de configurações.

Leitura dos arquivos

Os bancos de dados presentes neste desafio podem ser encontrados na plataforma Kaggle. Também é possível acessá-los diretamente pelos links:

- all_weather_data (parquet)
- Musical_Instruments_5 (JSON)

Para ler o arquivo parquet, foi utilizado o pacote *arrow*. Já para o arquivo JSON, foi usado o pacote *jsonlite.*

```
library(arrow) #pacote para ler parquet
library(jsonlite) #pacote para ler json e trabalhar com arquivos json em geral
```

Para a leitura do **arquivo parquet** foi realizado o uso da função *read_parquet* para ler o arquivo *all_weather_data.parquet* do diretório atual.

Por padrão, o pacote *arrow* converte os dados diretamente em um data frame.

Além disso, foi usada a função *class()* para mostrar que o banco de dados foi carregado como data frame, já que essa função serve para identificar a classe de um objeto.

```
arq_parquet <- read_parquet("all_weather_data.parquet")
class(arq_parquet)
```

```
[1] "tbl_df"     "tbl"        "data.frame"
```

Para a leitura do **arquivo JSON**, sabe-se que esse arquivo em específico é do tipo JSON Lines (cada linha é um objeto JSON separado). Assim, o R lê cada linha separadamente e depois junta tudo em um data frame.

A função *file(),* do pacote *jsonlite,* cria uma conexão R para o arquivo, o que permite que ele seja lido linha a linha, sem tentar carregar tudo como um único JSON grande.

Além disso, a função *stream_in()* consome a conexão e decodifica cada linha JSON, empilhando em um data frame, se as chaves forem consistentes.

Isso resolve o problema de arquivos de reviews (como Amazon, que é o caso usado aqui) que não estão em um array, e sim como objetos soltos por linha.

Aqui também foi utilizada a função *class()* para ver que o carregamento do banco de dados resultou diretamente em um data frame.

```
arq_json <- stream_in(file("Musical_Instruments_5.json"))
class(arq_json)
[1] "data.frame"
```