DOCUMENTAÇÃO

OPERAÇÕES EM PYSPARK/SPARKSQL: ‘CASOS COVID’

1- Instalação do Pyspark;

2- Importando diversas funções para possível utilização;

Fizemos a importação de várias funções para melhorar a fluidez das operações sem que precisássemos voltar ao início para instalar alguma função que tivesse ficado pendente

3- Integração com a GCP para futura importação dos Datasets;

Cumprindo o requisito de armazenar os datasets originais no ‘bucket’ da GCP, esse passo é essencial para que possamos conectar a sessão do Colab com a conta administradora da GCP, fornecendo, assim, permissão para que possamos importar para o ‘notebook’ o dataset desejado.

4- Copiando os arquivos do ‘Google Storage’ para a pasta ‘tmp’ no Colab;

5- Iniciando a sessão Spark;

6- Fazendo a leitura do dataset (já tratado) localizado na pasta tmp;

Fazendo a importação do Dataset previamente tratado em Pandas para que os insights e pesquisas sejam feitos em cima de uma base de dados de maior qualidade.

7- Mostrando o Schema da construção do dataframe;

Usar o comando ‘df.printSchema’ é de grande valia para analisarmos o tipo dos dados após a importação, garantindo que todos estejam na formatação adequada para as consultas e operações a seguir.

8- Removendo o índice por não apresentar utilidade para o trabalho;

O índice ‘c0’ não representa utilidade para o trabalho. Optamos por removê-lo nesta etapa.

9- Alterando o tipo de dado da ‘Data’ para o formato adequado;

Com o intuito de facilitar futuras operações neste dataset, optamos por converter o tipo de dado da coluna ‘Data’ para ‘date’ (o formato correto para manipulação de datas)

10- Verificando modificação;

11- Criando nova coluna para agrupar os estados por região do Brasil;

Visando o agrupamento dos estados por região do Brasil, usamos o comando 'withColumn' para criar a coluna das regiões, juntamente com o 'F.when' para quando o estado for da respectiva região, ele ser adicionado à nova coluna.

12- Agrupando dados por estado e região;

13- Criando nova coluna com somente o mês;

Seguindo a mesma lógica do passo 11, criamos uma coluna para o mês.

14- Criando nova coluna com somente o ano;

Seguindo a mesma lógica do passo 11, criamos uma coluna para o ano.

15- Agrupando total de dados por mês e ano;

16- Filtro:

Objetivo: identificarmos quantas vezes o número de mortes atingiu duzentas ou mais por cada estado. Agrupamento por estado e ordenado pelo número de mortes

17- Filtro:

Objetivo: identificarmos quantas vezes o número de mortes atingiu duzentas ou mais por cada região. Agrupamento por região e ordenado pelo número de mortes

18- Média de novas mortes por cada estado;

19- Média de novas mortes por mês entre 2020 e 2021;

Observação interessante: o uso da função 'agg' em conjunto com a função 'round' para que o número resultante da média feita pela 'agg(mean)' tivesse somente uma casa decimal. Paralelamente, usamos também a função 'alias' para renomear a coluna resultante das operações acima para melhor visualização no dataframe

20- Mostrando o maior valor de 'Casos Novos' agrupado por estado;

Observação interessante: usamos a função 'max' para nos mostrar o maior valor presente na coluna

21- Comparando média de novas mortes por estado em relação ao ano;

22- Ranqueando;

Com uso das 'window functions' realizamos um ranqueamento do número de novos casos por estado, mês e ano.

23- Ranqueando;

Com uso das 'window functions' realizamos um ranqueamento da quantidade de novas mortes por estado, mês e ano.

24- Salvando o arquivo normalizado e consultado em Pyspark na GCP;

25- Carregando o arquivo para operações em SparkSQL;

26- Soma de número de novos casos por mês de janeiro a setembro em 2020;

27- Total de novos casos por ano.