Relatório da Exploração de Dados

Primeiro, carregamos os quatro arquivos
CSV(CampaignQueue, Campaign, Customer, Order) em DataFrames do pandas. Isso é o primeiro passo para poder trabalhar com os dados.

Célula 1 (CampaignQueue - FILA/ENVIOS): Nesta célula, trabalhamos no DataFrame df1, que representa as filas de envio de campanhas.

- Primeiro, convertemos as colunas que pareciam ser datas (scheduledAt, sendAt, createdAt, updatedAt) para o formato datetime. Isso é super importante para fazer análises temporais. Usamos errors='coerce' para garantir que se alguma data estivesse "estranha", ela não quebrasse tudo, virando um valor nulo (NaT).
- Depois, demos uma olhada geral nas informações do DataFrame (df1.info()) e na quantidade de valores únicos em cada coluna (df1.nunique()). Isso nos ajuda a ter uma ideia do que cada coluna representa e se há IDs duplicados onde não deveria.
- Verificamos também quantos valores estavam faltando (df1.isnull().sum()), o que é crucial para saber se precisamos tratar dados ausentes.
- Fizemos uma análise mais a fundo nas colunas que pareciam ser IDs (id, jobId, campaignId, etc.) para ver se eram únicas e qual a distribuição dos valores mais frequentes. Por exemplo, vimos quais campaignId apareciam mais vezes na fila de envios.
- Analisamos a distribuição temporal dos eventos em relação às colunas de data que convertemos. Vimos a contagem diária e mensal de agendamentos, envios, criações e atualizações.
- Para visualizar melhor essa distribuição temporal, geramos gráficos de linha mostrando a contagem mensal de createdAt e updatedAt. Isso nos permite ver picos ou tendências ao longo do tempo na criação e atualização das filas.
- Fizemos o mesmo para as colunas scheduledAt e sendAt, visualizando a distribuição mensal de agendamentos e envios. Deu para notar que a coluna sendAt tinha muitos valores nulos, o que afeta o gráfico.
- Por fim, calculamos a taxa de resposta na coluna response. Contamos quantos registros tinham algum valor (não nulo) nessa coluna e comparamos com o total de registros para saber o percentual de envios que tiveram alguma resposta registrada.

Célula 2 (Campaign - Campanha): Aqui, focamos no DataFrame df2, que contém informações sobre as campanhas em si.

- Começamos convertendo as colunas de data (createdAt, updatedAt) para datetime, assim como fizemos com o df1. Tentamos alguns formatos comuns e usamos a inferência para lidar com possíveis variações nos dados.
- Visualizamos as informações gerais (df2.info()) e a contagem de valores únicos (df2.nunique()).
- Checamos os valores nulos (df2.isnull().sum()) para entender a completude dos dados das campanhas.
- Exploramos a distribuição das campanhas por segmentId, vendo quais segmentos de clientes foram mais alvo de campanhas.
- Contamos o número de campanhas por type (tipo) e por status. Isso nos deu uma visão rápida de quais tipos de campanha são mais comuns e qual o status atual da maioria delas.
- Criamos um gráfico de barras para visualizar a distribuição de campanhas por status, tornando mais fácil comparar a quantidade de campanhas em cada estado (ativo, inativo, etc.).
- Para ir mais a fundo, separamos a distribuição dos tipos de campanha por status. Geramos gráficos de pizza para cada status, mostrando a proporção de cada tipo de campanha dentro daquele status específico. Isso ajuda a entender, por exemplo, se certos tipos de campanha tendem a ficar em um status particular.
- Finalmente, visualizamos a distribuição temporal (mensal) da criação (createdAt) e atualização (updatedAt) das campanhas com gráficos de linha, similar ao que fizemos com o df1, para identificar padrões ao longo do tempo na gestão das campanhas.

Célula 3 (Customer - Cliente): Nesta célula, exploramos o DataFrame df3, com informações sobre os clientes.

- Começamos verificando a contagem de valores únicos e os valores nulos nas colunas, como de costume.
- Convertemos as colunas de data (dateOfBirth, enrichedAt, createdAt, updatedAt) para datetime. Foi um pouco mais desafiador aqui, pois tentamos diferentes formatos para garantir que a conversão funcionasse para a coluna dateOfBirth.

- Olhamos a distribuição de gender (gênero) dos clientes e criamos um gráfico de barras para visualizar essa distribuição.
- Fizemos uma análise interessante com a coluna phone. Tentamos extrair o DDD (os dois primeiros dígitos após o código do país, assumindo um formato específico) para ver a distribuição geográfica dos clientes e mostramos um gráfico com os 10 DDDs mais frequentes.

Célula 4 (Order - Pedido): Por último, exploramos o DataFrame df4, contendo os detalhes dos pedidos.

- Começamos com a contagem de valores únicos e a verificação de valores nulos.
- Convertemos as colunas de data (createdAt, updatedAt) para datetime, lidando com diferentes formatos possíveis.
- Analisamos a distribuição dos status dos pedidos e geramos um gráfico de barras para visualizar quantos pedidos estão em cada status (pendente, concluído, cancelado, etc.).
- Calculamos a média do totalAmount (valor total) dos pedidos agrupados por status. Isso nos dá uma ideia do valor médio dos pedidos em cada fase.
- Identificamos e mostramos os 10 clientes que fizeram mais pedidos, olhando a contagem de ocorrências de cada customer na coluna.
- Visualizamos a distribuição temporal (mensal) da criação (createdAt) e atualização (updatedAt) dos pedidos com gráficos de linha, para entender a frequência de pedidos e atualizações ao longo do tempo.

Basicamente, em cada DataFrame, a ideia foi entender a estrutura dos dados, identificar valores importantes, verificar a qualidade (nulos, unicidade) e visualizar as distribuições de atributos chave e temporais para ter uma visão geral do que cada conjunto de dados representa.