Documentação de Instruções

Este manual orienta como executar o código que realiza a análise de um conjunto de dados de tweets, permitindo explorar, limpar, visualizar e interpretar as informações contidas no dataset. As etapas abaixo descrevem como carregar o dataset, verificar a qualidade dos dados, explorar suas características e visualizar diferentes distribuições.

Etapas de Execução

1. Preparar o Ambiente

Antes de iniciar a análise, é necessário configurar o ambiente de desenvolvimento corretamente.

1. **Instalar o Python:** Certifique-se de ter o Python instalado na sua máquina. Baixe-o em python.org.
2. **Instalar as Bibliotecas Necessárias:** O código utiliza bibliotecas como pandas, numpy, matplotlib, seaborn, scipy e sklearn. Instale-as com o comando:
   pip install pandas numpy matplotlib seaborn scipy scikit-learn
3. **Escolher o Ambiente de Desenvolvimento:** Utilize um editor de código de sua preferência, como:
   - **PyCharm**: Download
   - **VS Code**: Download
   - **Jupyter Notebook** ou **Google Colab**, para uma experiência interativa.

---

2. Carregar o Dataset

O primeiro passo é carregar o dataset.

- Certifique-se de que o arquivo tweets.csv esteja no caminho especificado no código. Caso esteja em outro local, altere o caminho no comando:
  df = pd.read_csv('sample_data/tweets.csv')

---

3. Verificar a Qualidade dos Dados

- Execute a função verificar_dados() para obter informações sobre:
  - Tipos de dados.
  - Resumo estatístico com médias, desvios, mínimos e máximos.
  - Percentual de valores nulos e não nulos.
  - Presença de dados duplicados.
  - Intervalos de valores para colunas numéricas.

---

4. Limpeza dos Dados

- A função limpar_dados(df):
    - Remove colunas irrelevantes como country, latitude, longitude e id.
    - Converte variáveis categóricas em variáveis dummy, por exemplo, a coluna language.
    - Retorna um DataFrame limpo e pronto para análise.

---

5. Visualizações Iniciais

- Use a função visualizar_dados(df) para:
    - Gerar histogramas das variáveis numéricas.
    - Exibir boxplots para identificar outliers e distribuições.

---

6. Análise de Correlação

- A função plotCorrelationMatrix(df, graphWidth) exibe a matriz de correlação para variáveis numéricas:
    - Remove colunas com valores nulos ou constantes.
    - Gera um mapa de calor visualizando as correlações.

---

7. Identificar e Remover Outliers

- Utilize a função remover_outliers(df):
    - Identifica outliers com base no método **Z-Score**.
    - Filtra o DataFrame para remover os valores atípicos.

---

8. Normalizar os Dados

- A função normalizar_dados(df) utiliza StandardScaler para normalizar as variáveis numéricas:
    - Torna os dados uniformes com média 0 e desvio padrão 1.

---

9. Clusterização dos Dados

9.1 Determinar o Número de Clusters

- A função determinar_numero_clusters(df, features) aplica o método **Elbow**:
    - Calcula a inércia para diferentes valores de .
    - Gera um gráfico para ajudar na escolha do ideal.

9.2 Aplicar o Algoritmo K-Means

- A função aplicar_kmeans(df, features, k):
    - Realiza a clusterização com o escolhido.
    - Avalia a qualidade da clusterização com:
        - **Coeficiente de Silhouette.**
        - **Índice Davies-Bouldin.**
        - **Índice Calinski-Harabasz.**

9.3 Visualizar os Clusters

- A função visualizar_clusters(df, features) gera um scatter plot para visualizar os grupos formados.

---

10. Resumo dos Clusters

- A função resumo_clusters(df) exibe a média das variáveis para cada cluster, permitindo interpretar os grupos formados.

Considerações Finais

Este pipeline analítico oferece um processo estruturado para explorar, limpar, visualizar e interpretar dados de tweets, além de realizar clusterização para segmentação. Altere o dataset ou as variáveis utilizadas conforme suas necessidades e objetivos analíticos.