



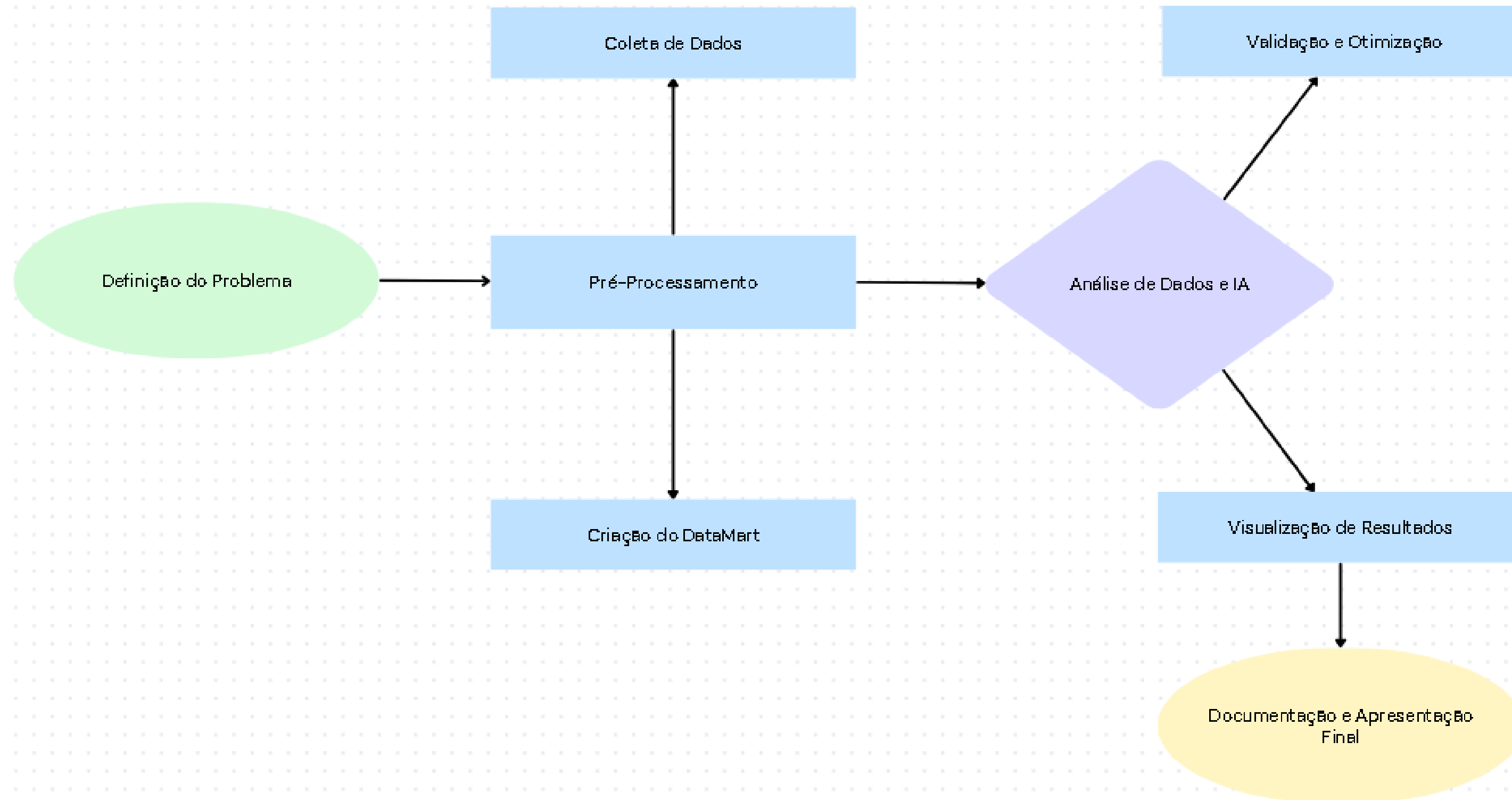
Inteligência Artificial

Projeto ELTA Upcycling

Docente: Antonio Ferreira

Aluno: Gabriel S. da Silveira
RA:13522110172

Workflow



Coleta de Dados

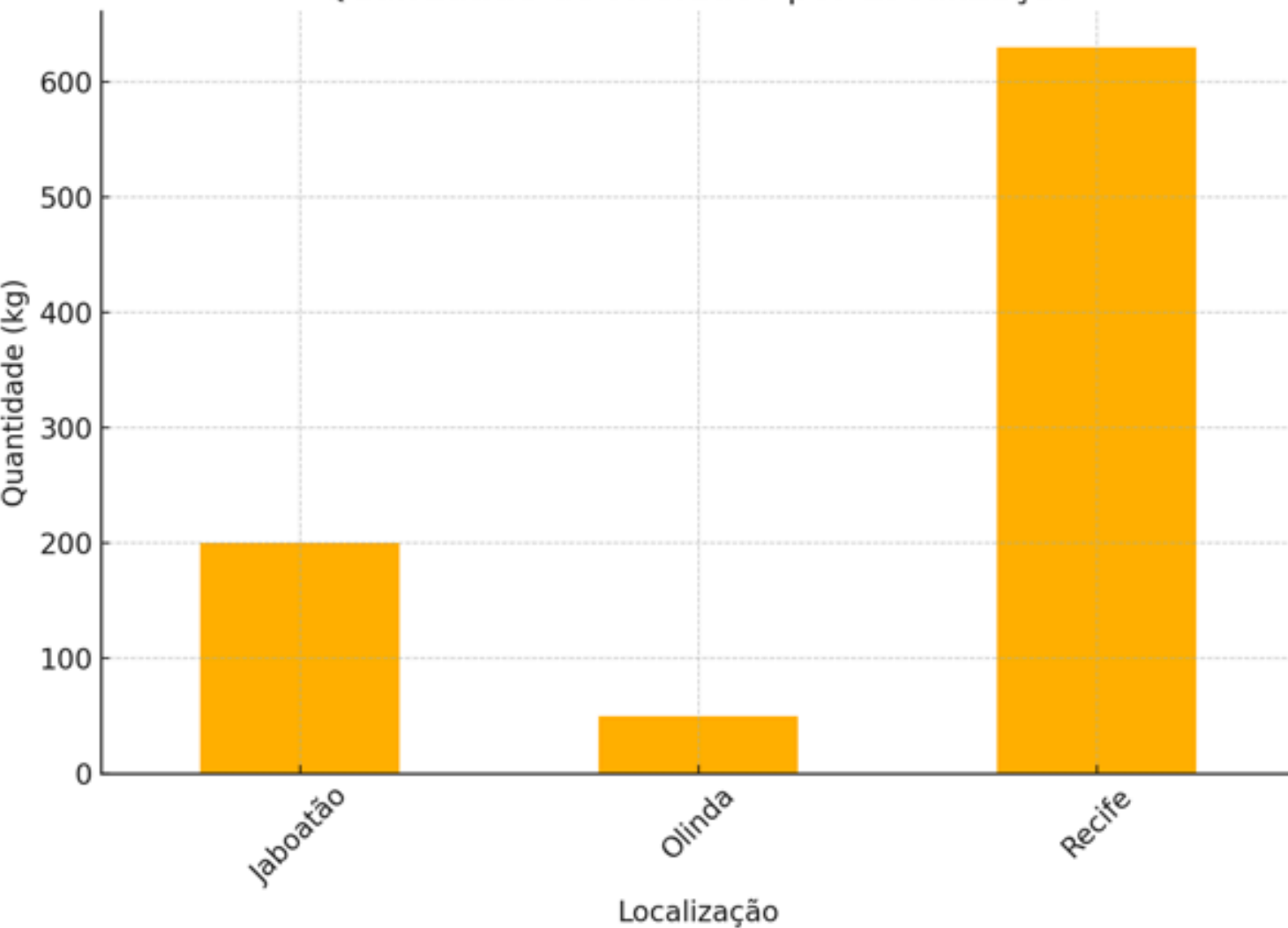


sum 3....

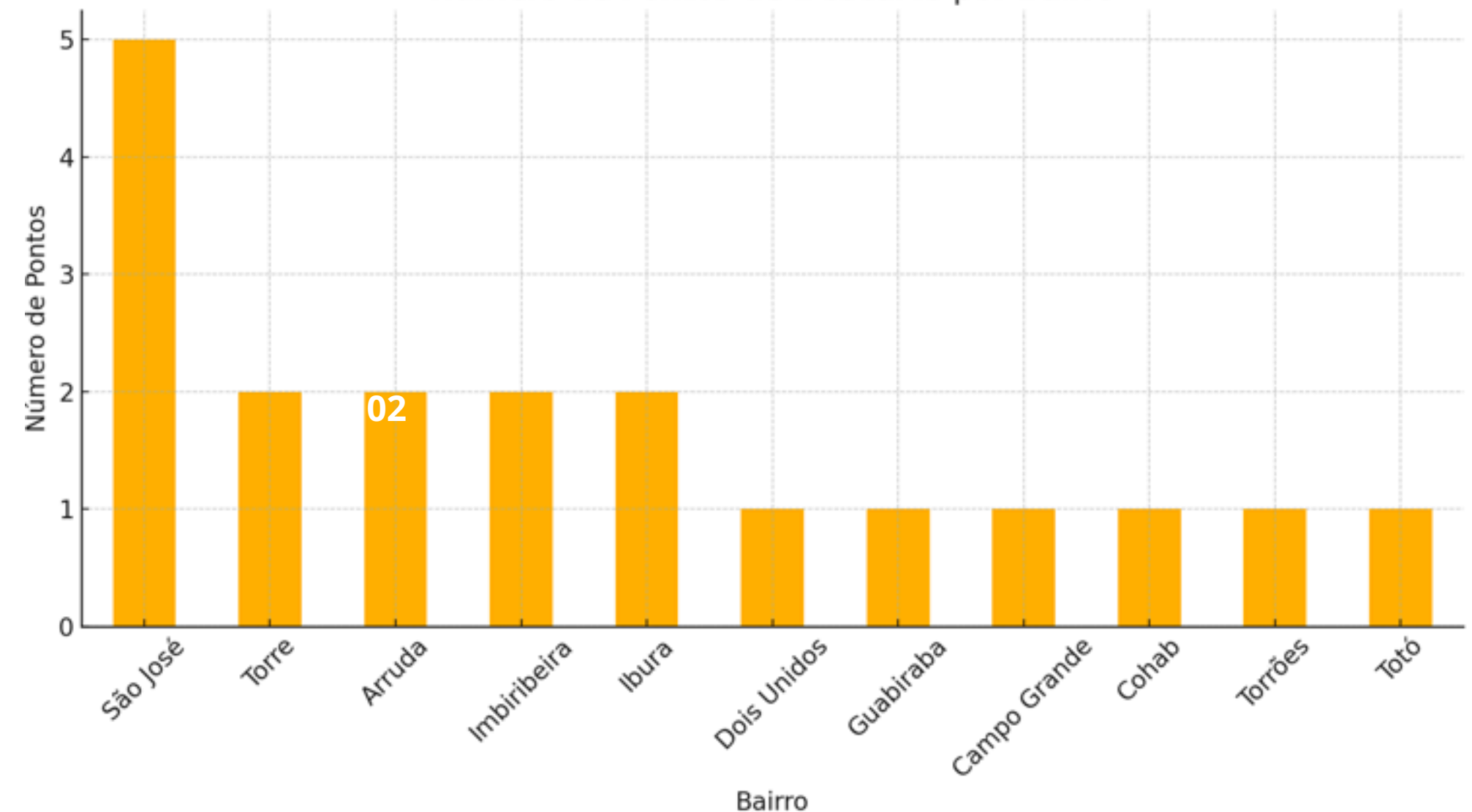
1	Materiais Recicláveis	Cooperativa	São José	Avenida Sul
2	Materiais Recicláveis	Cooperativa	São José	Avenida Sul
3	Materiais Recicláveis	Cooperativa	Dois Unidos	Av. Chagas Ferr
4	Materiais Recicláveis	Cooperativa	Torre	Rua Eliezer Olín
5	Materiais Recicláveis	Cooperativa	São José	Travessa do Gu
6	Materiais Recicláveis	Cooperativa	Guabiraba	Av. Padre Mosc
7	Materiais Recicláveis	Cooperativa	Arruda	Avenida Profes
8	Materiais Recicláveis	Cooperativa	São José	Rua Pacatuba
9	Materiais Recicláveis	Cooperativa	São José	Rua Oscar de M

Resíduos / Localizações

Quantidade de Resíduos por Localização



Número de Pontos de Descarte por Bairro



Pré-Processamento

Codificação de variáveis categóricas

```
from sklearn.preprocessing import LabelEncoder
encoder = LabelEncoder()
dataset["tipo_residuo_encoded"] = encoder.fit_transform(dataset["tipo_residuo"])
dataset["localizacao_encoded"] = encoder.fit_transform(dataset["localizacao"])
```

Padronização dos dados

```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
features_scaled = scaler.fit_transform(features)
```

Clustering com KMeans

```
from sklearn.cluster import KMeans
kmeans = KMeans(n_clusters=3, random_state=42)
dataset["cluster"] = kmeans.fit_predict(features_scaled)
```

Análise Clustering

Seleção de colunas relevantes

```
coords = data[["latitude", "longitude"]].dropna()
```

Agrupamento com KMeans

```
kmeans = KMeans(n_clusters=3, random_state=42)  
data["cluster"] = kmeans.fit_predict(coords)
```

Pré-Processamento Geral

Limpeza de valores ausentes

```
data = data.dropna()
```

Conversão de variáveis categóricas

```
data["bairro_encoded"] = LabelEncoder().fit_transform(data["bairro"])
```

Normalização ou padronização

```
data_normalized = scaler.fit_transform(data[["coluna1", "coluna2"]])
```

Criação do DataMart

- Latitude e longitude.
- Tipo de resíduo aceito.
 - Cluster atribuído. -
- Quantidade total de resíduos (quando disponível).

Ferramentas Utilizadas:

Python (bibliotecas como Pandas e NumPy)
para o fluxo foi utilizado matplotlib e networkx
Scripts de pré-processamento e clustering.

```
("1. Coleta de Dados", "2. Pré-Processamento"),  
("2. Pré-Processamento", "3. Clustering (KMeans)"),  
("2. Pré-Processamento", "4. Predição (Random Forest)"),  
("3. Clustering (KMeans)", "5. Criação do DataMart"),  
("4. Predição (Random Forest)", "5. Criação do DataMart"),  
("5. Criação do DataMart", "6. Visualização de Resultados")
```


Resultados Finais

1. Análises Realizadas

Clustering (KMeans):

Agrupamento dos pontos de descarte em 3 clusters principais.

Identificação de padrões geográficos e lacunas.

Predição (Random Forest):

Previsão de bairros prioritários para novos pontos de descarte.

Precisão geral do modelo: 83%.

2. Descobertas Principais

Pontos de Descarte Existentes:

79 pontos aceitam resíduos recicláveis.

Apenas 8 pontos para resíduos de construção civil.

Nenhum ponto para resíduos volumosos ou orgânicos.

Geografia dos Pontos:

Concentração em bairros como São José e Torre.

Áreas periféricas com pouca infraestrutura.

3. Impacto e Propostas

Melhoria da logística com base nos clusters.

Necessidade de criar novos pontos em bairros periféricos.

Uso de IA para priorizar investimentos em infraestrutura.

