

# Análise do Histograma: Quantidade de Infrações por Tipo de Veículo

## Objetivo:

A análise do histograma visa fornecer uma representação visual da distribuição da quantidade de infrações de trânsito de acordo com o tipo de veículo. O gráfico permitirá uma compreensão rápida e clara das diferenças nas quantidades de infrações entre os tipos de veículos.

## Coleta de Dados:

Os dados foram coletados no site do Detran/DF, utilizando a base de dados disponível para consulta pública a respeito das infrações ocorridas entre os anos 2018 a 2021.

## Importação de Bibliotecas:

Foi necessário importar as bibliotecas essenciais para análise de dados e visualização, incluindo pandas, matplotlib.pyplot e seaborn.

pythonCopy code

```
import matplotlib.pyplot as plt
```

```
import seaborn as sns
```

```
import pandas as pd
```

## Seleção dos Dados:

Os dados sobre a quantidade de infrações para diferentes tipos de veículos foram fornecidos em um dicionário.

pythonCopy code

```
dados_veiculo = {  
    'tipo_veiculo': ['Automóvel', 'Caminhonete', 'Motocicleta', 'Utilitário'],  
    'quantidade': [1665637, 222287, 114936, 77653]  
}  
  
df_veiculo = pd.DataFrame(dados_veiculo)
```

### **Criação do Histograma de Barra:**

Um gráfico de barra foi gerado usando a biblioteca seaborn, onde o eixo x representa os tipos de veículos e o eixo y representa a quantidade de infrações.

pythonCopy code

```
plt.figure(figsize=(10, 6))  
sns.barplot(x='tipo_veiculo', y='quantidade', data=df_veiculo, palette='Blues')
```

### **Adição de Rótulos e Título:**

Rótulos foram adicionados aos eixos x e y para fornecer informações contextuais. Além disso, um título foi incluído para resumir o propósito do gráfico.

pythonCopy code

```
plt.title("Quantidade de Infrações por Tipo de Veículo", color='black')  
plt.xlabel('Tipo de Veículo')  
plt.ylabel('Quantidade de Infrações')
```

### **Formatação dos Valores no Eixo Y:**

Os valores no eixo y foram formatados para serem exibidos como números inteiros com separadores de milhar (vírgulas).

pythonCopy code

```
plt.gca().get_yaxis().set_major_formatter(plt.FuncFormatter(lambda x, loc: "{:,}".format(int(x))))
```

### **Exibição do Gráfico:**

O gráfico final foi exibido para proporcionar uma representação visual das informações.

pythonCopy code

```
plt.show()
```

### **Codificação:**

```
import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
%matplotlib inline  
import seaborn as sns
```

```
infracao = pd.read_csv('infracaostransito.csv', sep = ';', encoding='utf-8', engine= 'python')
veiculo = infracao.tipo_veiculo.value_counts()
```

```
# Dados fornecidos
```

```
dados_veiculo = {
    'tipo_veiculo': ['Automóvel', 'Caminhonete', 'Motocicleta', 'Utilitário'],
    'quantidade': [1665637, 222287, 114936, 77653]
}
```

```
# Criar DataFrame
```

```
df_veiculo = pd.DataFrame(dados_veiculo)
```

```
# Criar o gráfico de barra
```

```
plt.figure(figsize=(10, 6))
```

```
sns.barplot(x='tipo_veiculo', y='quantidade', data=df_veiculo, palette='Blues')
```

```
# Adicionar rótulos e título
```

```
plt.title("Quantidade de Infrações por Tipo de Veículo", color='black')
```

```
plt.xlabel('Tipo de Veículo')
```

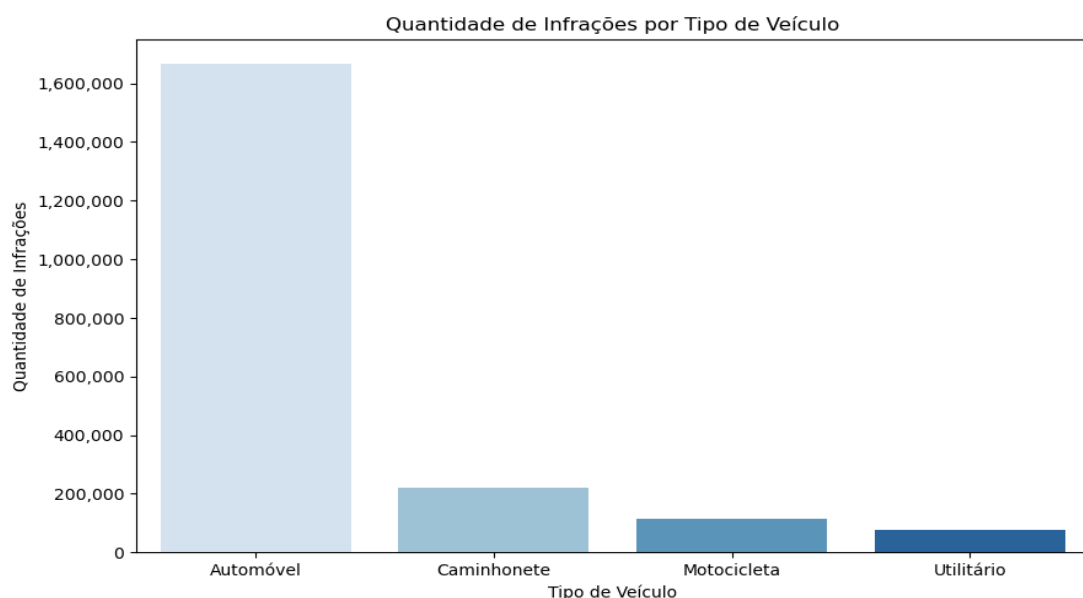
```
plt.ylabel('Quantidade de Infrações')
```

```
# Formatando os valores do eixo y como inteiros
```

```
plt.gca().get_yaxis().set_major_formatter(plt.FuncFormatter(lambda x, loc: "{:,}".format(int(x))))
```

```
# Mostrar o gráfico
```

```
plt.show()
```



**Conclusões:**

O histograma fornece uma visão clara da distribuição da quantidade de infrações para cada tipo de veículo. Através da análise do gráfico, podemos identificar facilmente quais tipos de veículos têm maior ou menor incidência de infrações, contribuindo para uma compreensão visual dos padrões de comportamento de trânsito.