

CÓDIGO PYTHON COMPLETO UTILIZADO PARA ANÁLISE DE GORJETAS

Arquivo: main.py

Total de linhas: 279

BIBLIOTECAS UTILIZADAS:

- pandas: Manipulação e análise de dados
- matplotlib.pyplot: Criação de gráficos
- seaborn: Visualizações estatísticas
- matplotlib.backends.backend_pdf: Geração de PDF

=====

CÓDIGO FONTE COMPLETO:

=====

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from matplotlib.backends.backend_pdf import PdfPages
```

```
def analise_principal():
    """
    FUNÇÃO 1: ANÁLISE PRINCIPAL - GERA PDF COM GRÁFICOS
    Código do arquivo: analise_gorjetas.py
    """
    print("="*60)
    print("EXECUTANDO ANÁLISE PRINCIPAL - GERAÇÃO DE GRÁFICOS PDF")
    print("="*60)

    plt.rcParams['font.size'] = 12

    # Carregando os dados
    df = pd.read_csv('gorjetas.csv')

    # Verificando os dados
    print("Informações sobre o dataset:")
    print(f"Total de registros: {len(df)}")
    print(f"Colunas: {list(df.columns)}")
    print(f"\nPrimeiras linhas:")
    print(df.head())

    # Criando o PDF com os gráficos e análises
    with PdfPages('analise_gorjetas_restaurante.pdf') as pdf:

        # Página inicial: Código utilizado
        fig, ax = plt.subplots(figsize=(8, 11))
        ax.axis('off')

        # Lendo o código do próprio arquivo main.py
        try:
            with open('main.py', 'r', encoding='utf-8') as file:
                codigo_completo = file.read()

        # Adicionando cabeçalho
        codigo_com_cabecalho = f"""CÓDIGO PYTHON COMPLETO UTILIZADO PARA ANÁLISE DE GORJETAS
Arquivo: main.py
Total de linhas: {len(codigo_completo.splitlines())}

BIBLIOTECAS UTILIZADAS:
• pandas: Manipulação e análise de dados
• matplotlib.pyplot: Criação de gráficos
• seaborn: Visualizações estatísticas
• matplotlib.backends.backend_pdf: Geração de PDF

{' '*80}
CÓDIGO FONTE COMPLETO:
{' '*80}

{codigo_completo}"""

        except FileNotFoundError:
            codigo_com_cabecalho = "Erro: Não foi possível ler o arquivo main.py"

        ax.text(0.05, 0.95, codigo_com_cabecalho, transform=ax.transAxes, fontsize=8,
               verticalalignment='top', fontfamily='monospace')

        pdf.savefig(fig, bbox_inches='tight')
        plt.close()

        # Página 1: Gráfico de densidade do total de gorjetas por sexo
        fig, ax = plt.subplots(figsize=(8, 5))

        # Criando o gráfico de densidade simples
        sns.kdeplot(data=df, x='gorjeta', hue='sexo', fill=True, ax=ax)

        ax.set_title('Densidade das Gorjetas por Sexo')
        ax.set_xlabel('Gorjeta ($)')
        ax.set_ylabel('Densidade')

        plt.tight_layout()
        pdf.savefig(fig, bbox_inches='tight')
        plt.close()

        # Página 2: Gráfico de densidade do total de gorjetas pela quantidade de pessoas
        fig, ax = plt.subplots(figsize=(8, 5))

        # Filtrando apenas as quantidades mais comuns (2, 3, 4 pessoas)
        df_filtrado = df[df['quantidade'].isin([2, 3, 4])]

        # Criando o gráfico de densidade simples
        sns.kdeplot(data=df_filtrado, x='gorjeta', hue='quantidade', fill=True, ax=ax)

        ax.set_title('Densidade das Gorjetas por Quantidade de Pessoas')
        ax.set_xlabel('Gorjeta ($)')
        ax.set_ylabel('Densidade')

        plt.tight_layout()
        pdf.savefig(fig, bbox_inches='tight')
        plt.close()

        # Página 3: Gráfico de densidade do total de gorjetas por dia da semana
        fig, ax = plt.subplots(figsize=(8, 5))

        # Criando o gráfico de densidade simples
        sns.kdeplot(data=df, x='gorjeta', hue='dia', fill=True, ax=ax)

        ax.set_title('Densidade das Gorjetas por Dia da Semana')
        ax.set_xlabel('Gorjeta ($)')
        ax.set_ylabel('Densidade')

        plt.tight_layout()
        pdf.savefig(fig, bbox_inches='tight')
        plt.close()

        # Página 4: Análise textual simplificada
        fig, ax = plt.subplots(figsize=(8, 6))
        ax.axis('off')

        # Calculando estatísticas básicas
        stats_sexo = df.groupby('sexo')['gorjeta'].mean()
        stats_quantidade = df.groupby('quantidade')['gorjeta'].mean()
        stats_dia = df.groupby('dia')['gorjeta'].mean()

        analise_texto = f"""
ANÁLISE TEXTUAL DOS GRÁFICOS DE GORJETAS

Análise de {len(df)} clientes usando Python com Pandas, Seaborn e Matplotlib.

PRINCIPAIS RESULTADOS:

1. GORJETAS POR SEXO:
Homens: gorjeta média de ${stats_sexo['Homem']:.2f}
Mulheres: gorjeta média de ${stats_sexo['Mulher']:.2f}
Diferença pequena entre os sexos

2. GORJETAS POR QUANTIDADE DE PESSOAS:
2 pessoas: ${stats_quantidade[2]:.2f} (mais comum)
3 pessoas: ${stats_quantidade[3]:.2f}
4 pessoas: ${stats_quantidade[4]:.2f}
Mesas maiores deixam gorjetas maiores

3. GORJETAS POR DIA DA SEMANA:
Domingo: ${stats_dia['Dom']:.2f} (maior gorjeta)
Sábado: ${stats_dia['Sab']:.2f} (mais movimento)
Quinta: ${stats_dia['Qui']:.2f}
Sexta: ${stats_dia['Sex']:.2f}

CONCLUSÕES:
0 restaurante tem bom desempenho em gorjetas
Fins de semana são mais movimentados
Grupos maiores são mais generosos
Oportunidade de melhorar quinta e sexta-feira
"""

        ax.text(0, 0, analise_texto, transform=ax.transAxes, fontsize=11, verticalalignment='center')

        plt.tight_layout()
        pdf.savefig(fig)
        plt.close()

        print("Análise concluída! Arquivo 'analise_gorjetas_restaurante.pdf' gerado com sucesso.")
        print("\nResumo dos dados:")
        print(f"Total de registros analisados: {len(df)}")
        print(f"Gorjeta média geral: ${df['gorjeta'].mean():.2f}")
        print(f"Gorjeta mediana: ${df['gorjeta'].median():.2f}")
        print(f"Faixa de gorjetas:  $df[gorjeta].min():.2f - {df[gorjeta].max():.2f}$ ")

def estatisticas_complementares():
    """
    FUNÇÃO 2: ESTATÍSTICAS COMPLEMENTARES DETALHADAS
    Código do arquivo: estatisticas_complementares.py
    """
    print("\n" + "="*60)
    print("EXECUTANDO ESTATÍSTICAS COMPLEMENTARES")
    print("="*60)

    # Carregando os dados
    df = pd.read_csv('gorjetas.csv')

    print("="*60)
    print("ESTATÍSTICAS COMPLEMENTARES - ANÁLISE DE GORJETAS")
    print("="*60)

    # Estatísticas gerais
    print("\n1. ESTATÍSTICAS GERAIS:")
    print(f"• Total de clientes analisados: {len(df)}")
    print(f"• Gorjeta média: ${df['gorjeta'].mean():.2f}")
    print(f"• Gorjeta mediana: ${df['gorjeta'].median():.2f}")
    print(f"• Desvio padrão: ${df['gorjeta'].std():.2f}")
    print(f"• Gorjeta mínima: ${df['gorjeta'].min():.2f}")
    print(f"• Gorjeta máxima: ${df['gorjeta'].max():.2f}")

    # Análise por sexo
    print("\n2. ANÁLISE DETALHADA POR SEXO:")
    stats_sexo = df.groupby('sexo')['gorjeta'].agg(['count', 'mean', 'median', 'std', 'min', 'max'])
    for sexo in stats_sexo.index:
        print(f"\n {sexo.upper():}")
        print(f"• Quantidade: {stats_sexo.loc[sexo, 'count']} clientes ({stats_sexo.loc[sexo, 'count']/len(df)*100:.1f}%)")
        print(f"• Gorjeta média: ${stats_sexo.loc[sexo, 'mean']:.2f}")
        print(f"• Gorjeta mediana: ${stats_sexo.loc[sexo, 'median']:.2f}")
        print(f"• Desvio padrão: ${stats_sexo.loc[sexo, 'std']:.2f}")
        print(f"• Faixa:  $stats\_sexo.loc[sexo, 'min']:.2f - {stats\_sexo.loc[sexo, 'max']:.2f}$ ")

    # Análise por quantidade de pessoas
    print("\n3. ANÁLISE POR QUANTIDADE DE PESSOAS NA MESA:")
    stats_qtd = df.groupby('quantidade')['gorjeta'].agg(['count', 'mean', 'median', 'std'])
    for qtd in sorted(stats_qtd.index):
        print(f"\n {qtd} PESSOAS:")
        print(f"• Frequência: {stats_qtd.loc[qtd, 'count']} mesas ({stats_qtd.loc[qtd, 'count']/len(df)*100:.1f}%)")
        print(f"• Gorjeta média: ${stats_qtd.loc[qtd, 'mean']:.2f}")
        print(f"• Gorjeta mediana: ${stats_qtd.loc[qtd, 'median']:.2f}")

    # Análise por dia da semana
    print("\n4. ANÁLISE POR DIA DA SEMANA:")
    stats_dia = df.groupby('dia')['gorjeta'].agg(['count', 'mean', 'median', 'std'])
    ordem_dias = ['Dom', 'Seg', 'Ter', 'Qua', 'Qui', 'Sex', 'Sab']
    dias_presentes = [dia for dia in ordem_dias if dia in stats_dia.index]

    for dia in dias_presentes:
        print(f"\n {dia.upper():}")
        print(f"• Movimento: {stats_dia.loc[dia, 'count']} clientes ({stats_dia.loc[dia, 'count']/len(df)*100:.1f}%)")
        print(f"• Gorjeta média: ${stats_dia.loc[dia, 'mean']:.2f}")
        print(f"• Gorjeta mediana: ${stats_dia.loc[dia, 'median']:.2f}")

    # Análise por período (Almoço vs Jantar)
    print("\n5. ANÁLISE POR PERÍODO:")
    stats_tempo = df.groupby('tempo')['gorjeta'].agg(['count', 'mean', 'median', 'std'])
    for tempo in stats_tempo.index:
        print(f"\n {tempo.upper():}")
        print(f"• Movimento: {stats_tempo.loc[tempo, 'count']} clientes ({stats_tempo.loc[tempo, 'count']/len(df)*100:.1f}%)")
        print(f"• Gorjeta média: ${stats_tempo.loc[tempo, 'mean']:.2f}")
        print(f"• Gorjeta mediana: ${stats_tempo.loc[tempo, 'median']:.2f}")

    # Correlação entre conta total e gorjeta
    correlacao = df['total_conta'].corr(df['gorjeta'])
    print("\n6. CORRELAÇÃO CONTA TOTAL vs GORJETA:")
    print(f"• Coeficiente de correlação: {correlacao:.3f}")
    if correlacao > 0.7:
        print("• Correlação FORTE positiva")
    elif correlacao > 0.3:
        print("• Correlação MODERADA positiva")
    else:
        print("• Correlação FRACA")

    # Top 5 maiores gorjetas
    print("\n7. TOP 5 MAIORES GORJETAS:")
    top_gorjetas = df.nlargest(5, 'gorjeta')[['total_conta', 'gorjeta', 'sexo', 'quantidade', 'dia', 'tempo']]
    for i, (idx, row) in enumerate(top_gorjetas.iterrows(), 1):
        print(f" {i}º.  $row[gorjeta]:.2f$  - Conta:  ${row[total\_conta]:.2f}$ , {row[sexo]}, {row[quantidade]} pessoas, {row[dia]} {row[tempo]}")

    print("\n" + "="*60)
    print("ANÁLISE CONCLUÍDA")
    print("="*60)

def main():
    """
    FUNÇÃO PRINCIPAL - EXECUTA TODAS AS ANÁLISES
    """
    print("INICIANDO ANÁLISE COMPLETA DE GORJETAS DE RESTAURANTE")
    print("="*60)
    print("Este script consolida todos os códigos Python do projeto:")
    print("1. analise_gorjetas.py - Gera PDF com gráficos")
    print("2. estatisticas_complementares.py - Estatísticas detalhadas")
    print("="*60)

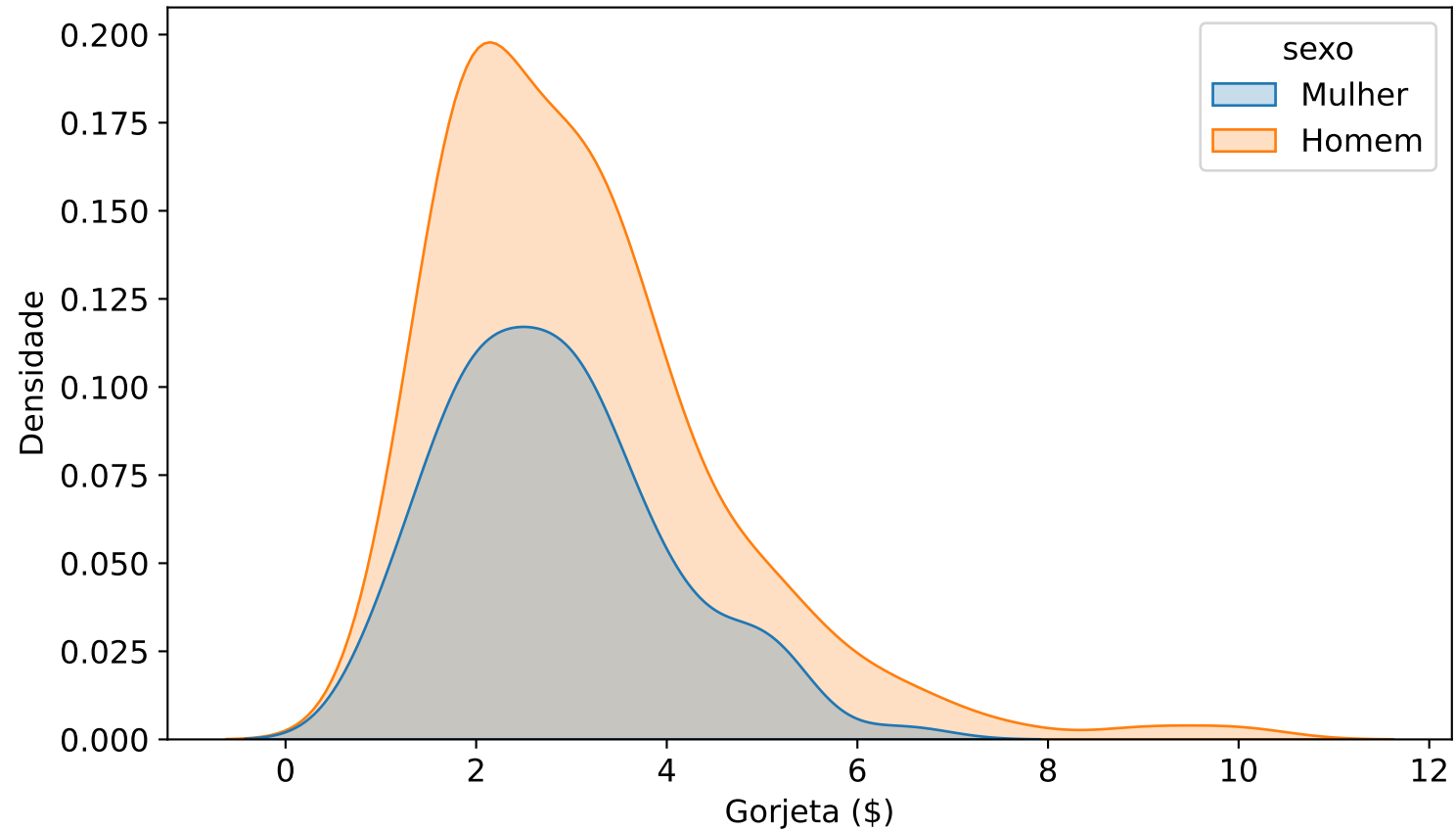
    # Executar análise principal (gera PDF)
    analise_principal()

    # Executar estatísticas complementares
    estatisticas_complementares()

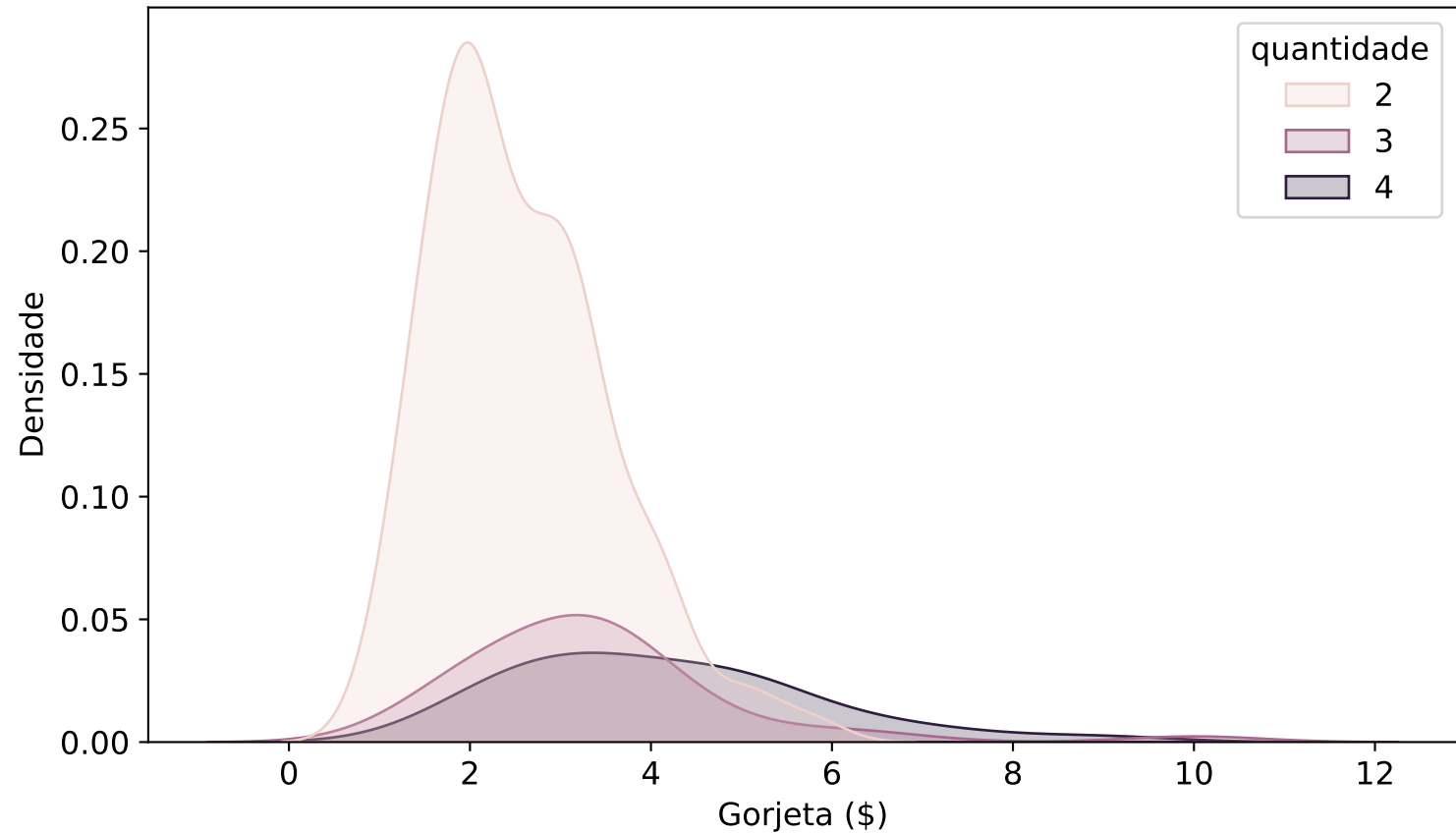
    print("\n" + "="*60)
    print("TODAS AS ANÁLISES FORAM CONCLUÍDAS COM SUCESSO!")
    print("="*60)
    print("Arquivos gerados:")
    print("• analise_gorjetas_restaurante.pdf - Relatório com gráficos")
    print("• Estatísticas detalhadas exibidas no terminal")
    print("="*60)

if __name__ == "__main__":
    main()
```

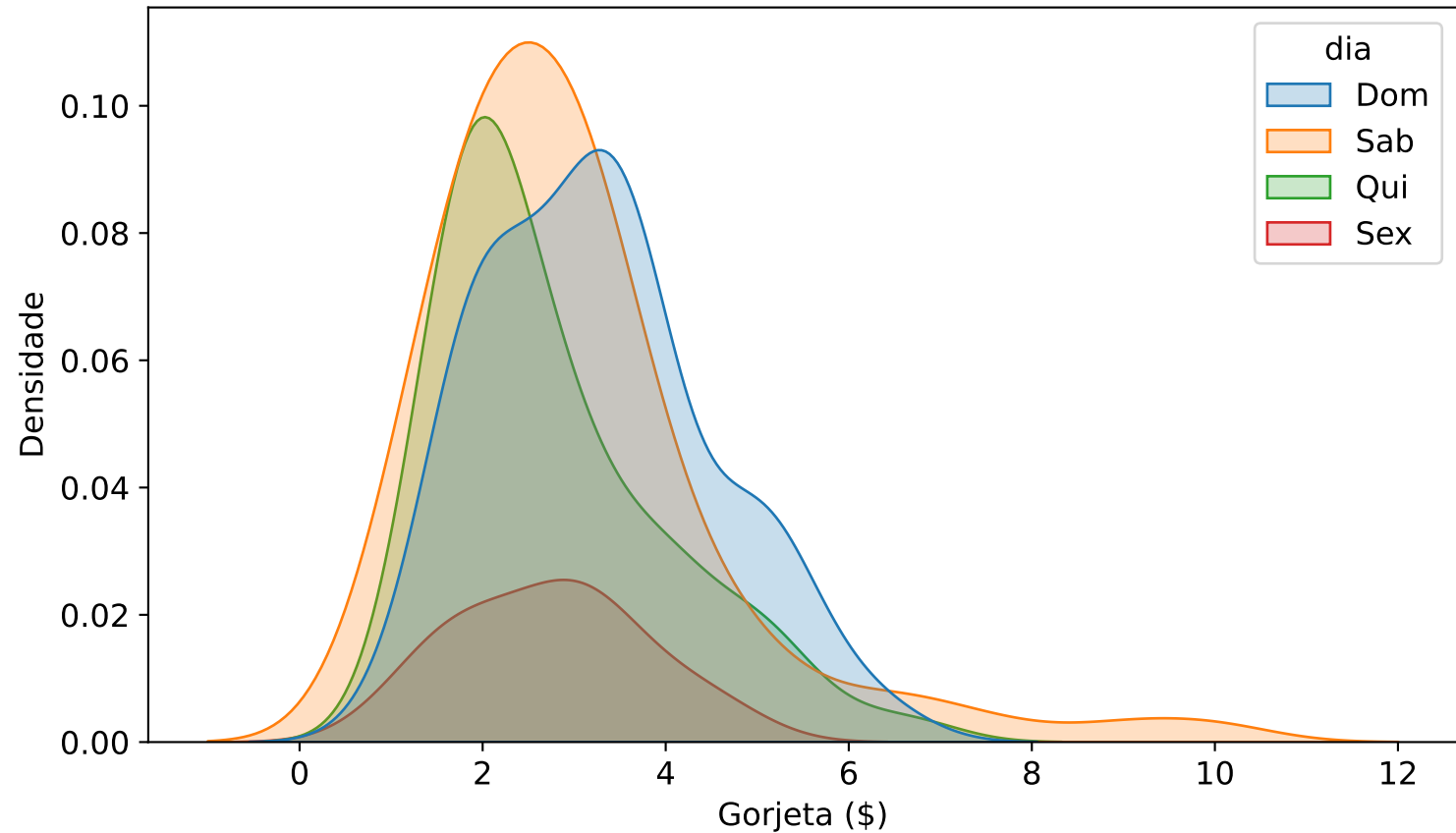
Densidade das Gorjetas por Sexo



Densidade das Gorjetas por Quantidade de Pessoas



Densidade das Gorjetas por Dia da Semana



ANÁLISE TEXTUAL DOS GRÁFICOS DE GORJETAS

Análise de 244 clientes usando Python com Pandas, Seaborn e Matplotlib.

PRINCIPAIS RESULTADOS:

1. GORJETAS POR SEXO:

Homens: gorjeta média de \$3.09

Mulheres: gorjeta média de \$2.83

Diferença pequena entre os sexos

2. GORJETAS POR QUANTIDADE DE PESSOAS:

2 pessoas: \$2.58 (mais comum)

3 pessoas: \$3.39

4 pessoas: \$4.14

Mesas maiores deixam gorjetas maiores

3. GORJETAS POR DIA DA SEMANA:

Domingo: \$3.26 (maior gorjeta)

Sábado: \$2.99 (mais movimento)

Quinta: \$2.77

Sexta: \$2.73

CONCLUSÕES:

O restaurante tem bom desempenho em gorjetas

Fins de semana são mais movimentados

Grupos maiores são mais generosos

Oportunidade de melhorar quinta e sexta-feira