**UNIVERSIDADE PRESBITERIANA MACKENZIE**

**MBA EM ENGENHARIA DE DADOS**

**DATA SCIENCE EXPERIENCE**

**LISTA DE EXERCÍCIOS**

Professor: Matheus Pavani

Aluno: Fábio Fumio Wada (10741479)

Lista de Exercícios

**Parte 1: Distribuições de Probabilidade**

**Exercício 1**:

Gere uma amostra de 1000 valores que seguem uma distribuição normal com média μ=5 e desvio padrão σ=2. Plote o histograma da amostra e sobreponha a curva da função densidade de probabilidade (PDF) correspondente. Comente sobre a forma do gráfico e a distribuição dos dados.

**Exercício 2**:

Crie um simulador para modelar o número de caras obtidas ao lançar uma moeda 20 vezes, repetindo o experimento 1000 vezes. Utilize a distribuição binomial para isso. Plote o histograma dos resultados e calcule a probabilidade de obter exatamente 10 caras.

**Exercício 3**:

Um banco recebe em média 3 ligações por minuto. Utilize a distribuição de Poisson para simular o número de ligações recebidas em 1000 minutos. Plote o histograma dos resultados e calcule a probabilidade de receber exatamente 5 ligações em um minuto.

**Parte 2: Testes de Hipótese**

**Exercício 4**:

Dois grupos de estudantes foram submetidos a métodos de ensino diferentes. As notas de 20 alunos do Grupo A e 20 alunos do Grupo B estão disponíveis. Use o teste t de Student para determinar se há uma diferença significativa entre as médias das notas dos dois grupos. Considere α=0,05. Mostre o código em Python para a realização do teste e interprete o resultado.

**Exercício 5**:

Uma pesquisa foi realizada em duas cidades para verificar a preferência dos moradores por dois tipos de bebidas (A e B). A tabela de contingência a seguir mostra os resultados:

Padrão do plano de fundo, Tabela

O conteúdo gerado por IA pode estar incorreto.

Realize o teste Qui-Quadrado de independência para verificar se a preferência pela bebida é independente da cidade. Utilize Python para calcular a estatística do teste e o valor-p. Considere α=0,05 e interprete os resultados.

**Parte 1: Distribuições de Probabilidade**

**Exercício 1**:

Gere uma amostra de 1000 valores que seguem uma distribuição normal com média μ=5 e desvio padrão σ=2. Plote o histograma da amostra e sobreponha a curva da função densidade de probabilidade (PDF) correspondente. Comente sobre a forma do gráfico e a distribuição dos dados.

**Exercício 1 - Resultados**:

Requisitos Técnicos:

Para este exercício foram utilizadas as bibliotecas:

import os

import pandas as pd

import numpy as np

import seaborn as sns

import matplotlib.pyplot as plt

from scipy import stats

O objeto central para este estudo foi a exploração da função “**random.seed**” do Numpy.

Resultado da Análise:

Analisando os 1000 valores gerados com média μ=5 e desvio padrão σ=2, é possível visualizar e concluir como os dados se comportam e se encaixam no modelo teórico.

Tanto no histograma quanto na curva da PDF os dados seguem a curvatura de um sino O histograma mostra mais valores perto da média, e vai diminuindo conforme a gente se afasta do centro. A curva vermelha da PDF teórica se encaixa perfeitamente no histograma, o que sugere que os dados gerados se assemelham.

Os valores calculados confirmam o que a gente está vendo. A média da amostra calculada foi de 5,0387, ficando próximo ao valor 5. O desvio padrão ficou em 1,9575, próximo ao valor de 2.

A amostra gerada segue um fluxo padrão com resultados próximos a média e o desvio padrão mencionado.

Gráfico, Gráfico de linhas, Histograma

O conteúdo gerado por IA pode estar incorreto.

Gráfico, Gráfico de linhas, Histograma

O conteúdo gerado por IA pode estar incorreto.

**Parte 1: Distribuições de Probabilidade**

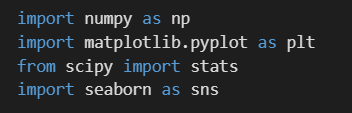
**Exercício 2**:

Crie um simulador para modelar o número de caras obtidas ao lançar uma moeda 20 vezes, repetindo o experimento 1000 vezes. Utilize a distribuição binomial para isso. Plote o histograma dos resultados e calcule a probabilidade de obter exatamente 10 caras.

**Exercício 2 - Resultados**:

Requisitos Técnicos:

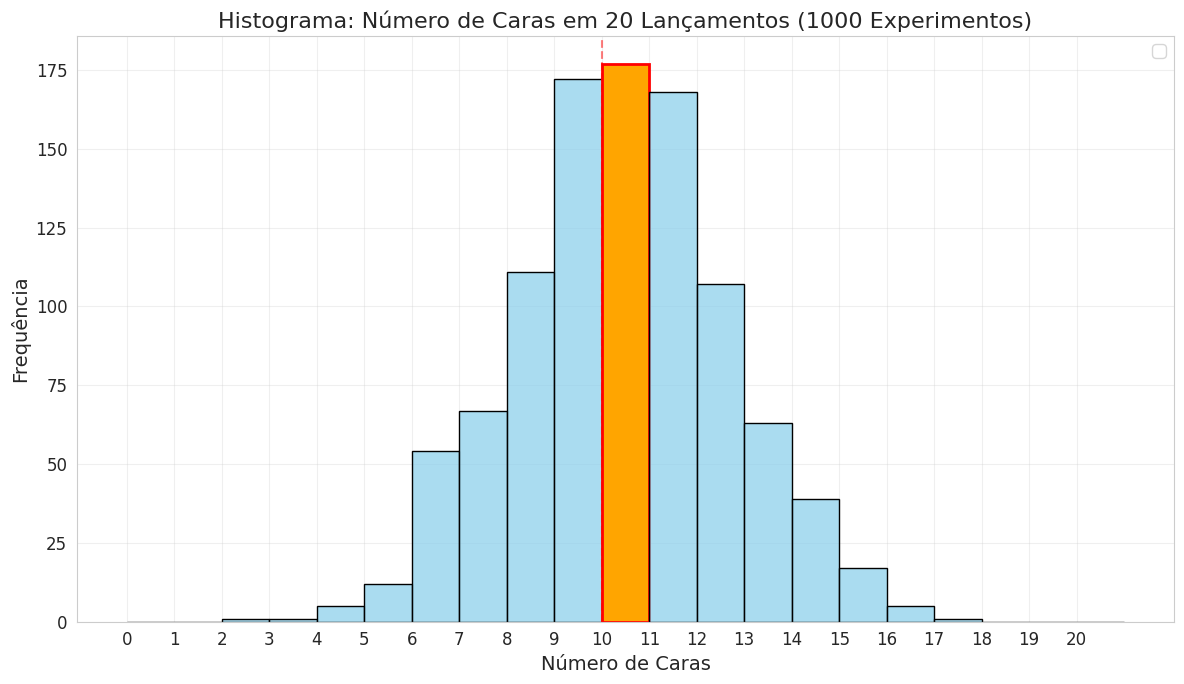
Para este exercício foram utilizadas as bibliotecas:



O objeto central para este estudo foi a exploração da função “**random.binomial**” do Numpy.

Resultado da Análise:

A probabilidade de obter "cara" foi de 0.177000, sendo 177 ocorrências em 1000 experimentos.  
A média de "cara" obtida por experimento foi de 10.  
O desvio padrão apresentado foi de 2.24.



**Parte 1: Distribuições de Probabilidade**

**Exercício 3**:

Um banco recebe em média 3 ligações por minuto. Utilize a distribuição de Poisson para simular o número de ligações recebidas em 1000 minutos. Plote o histograma dos resultados e calcule a probabilidade de receber exatamente 5 ligações em um minuto.

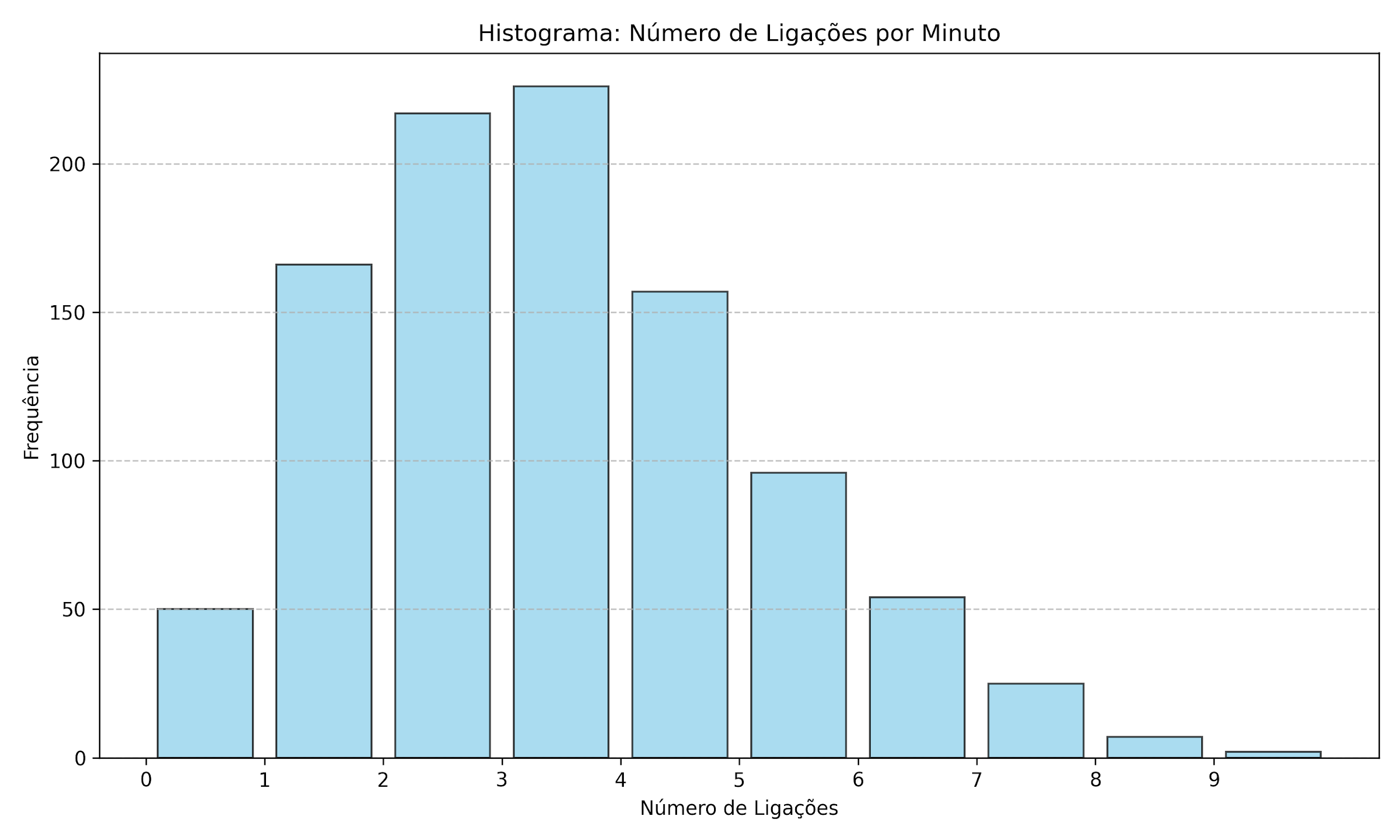
**Exercício 3 - Resultados**:

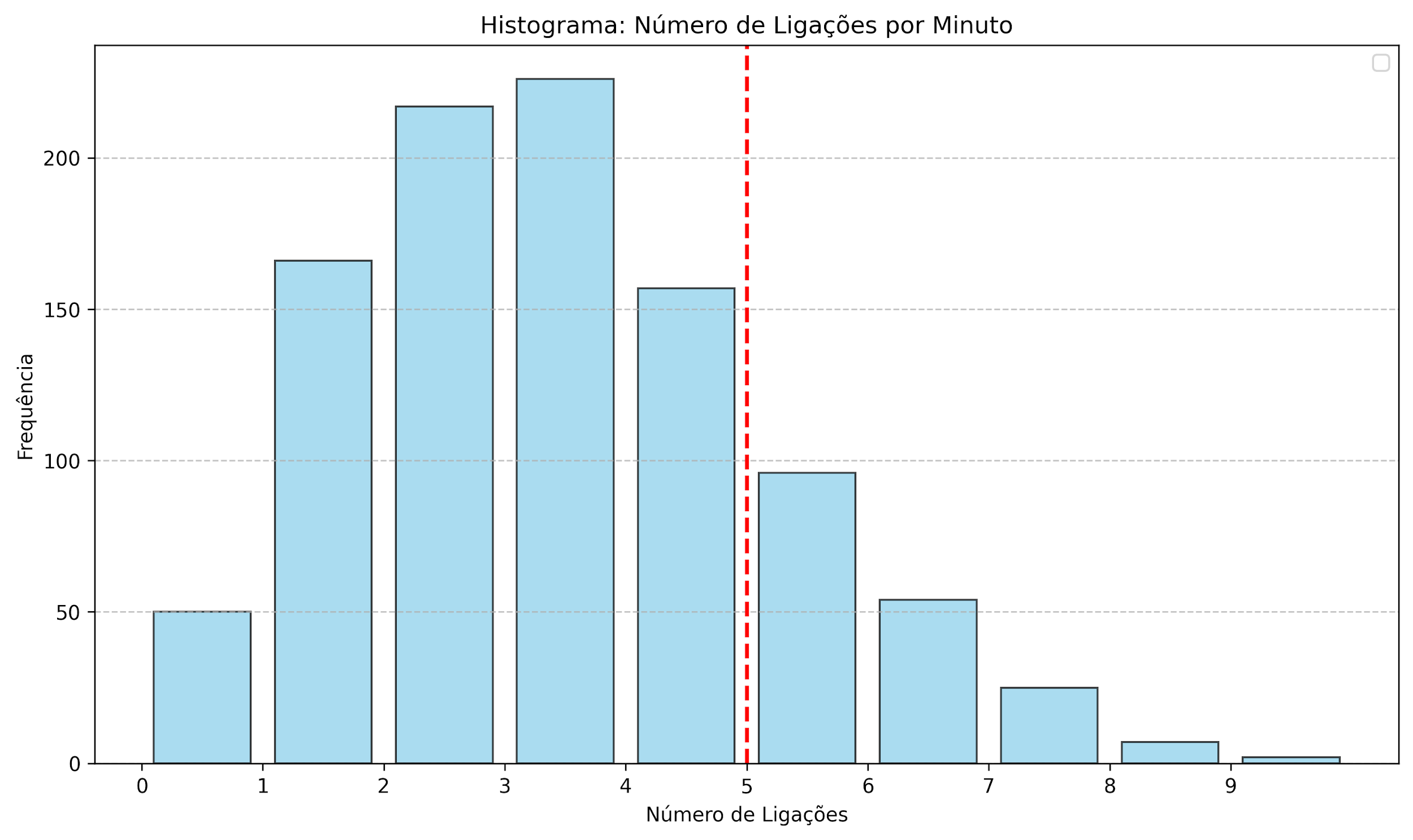
Requisitos Técnicos:

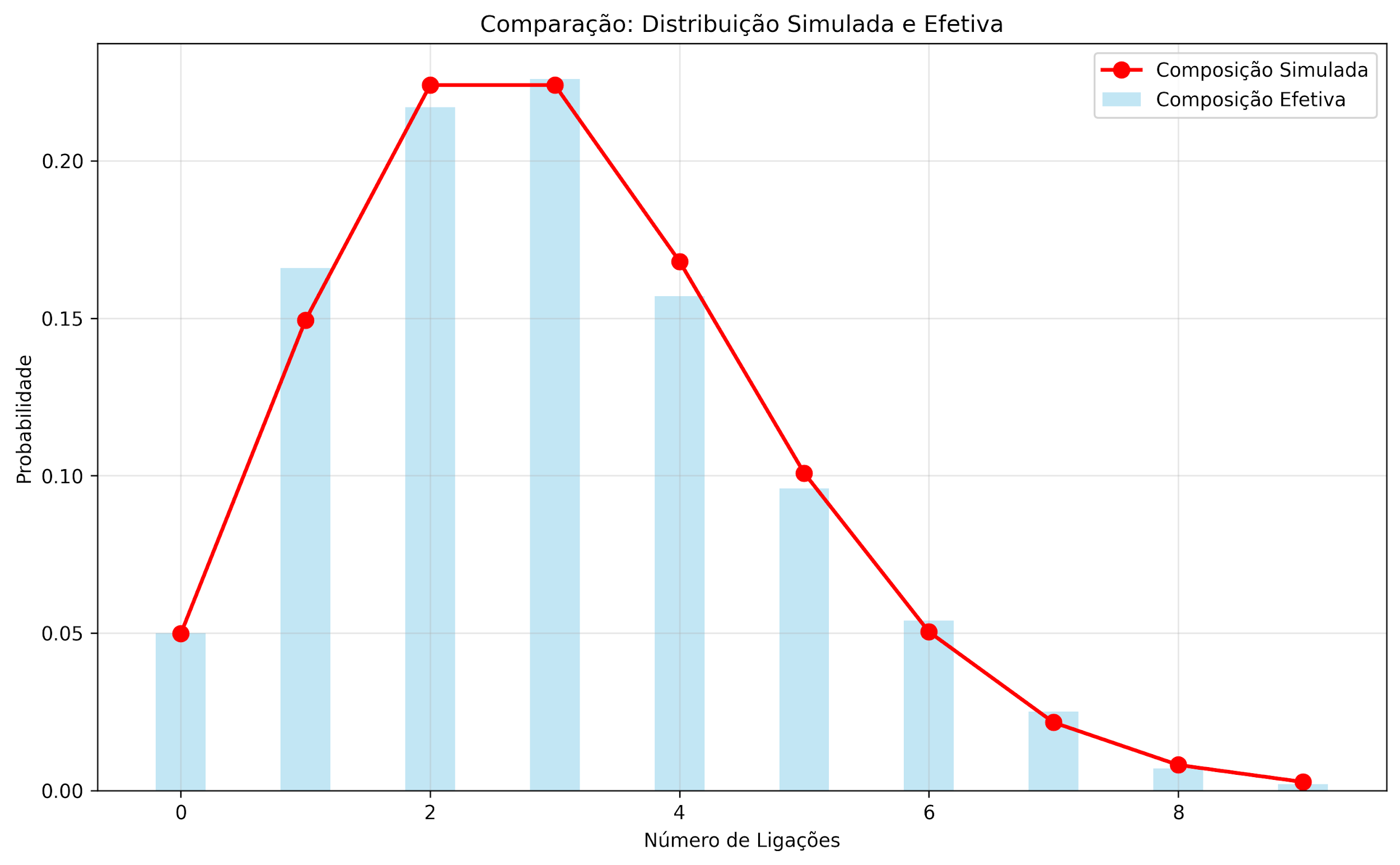
O objeto central para este estudo foi a exploração da função “**random.poisson**” do Numpy

Resultado da Análise:

A probabilidade calculada para o recebimento de 5 ligações em um minuto foi de 0.100819.







**Parte 2: Teste de Hipótese**

**Exercício 4**:

Dois grupos de estudantes foram submetidos a métodos de ensino diferentes. As notas de 20 alunos do Grupo A e 20 alunos do Grupo B estão disponíveis. Use o teste t de Student para determinar se há uma diferença significativa entre as médias das notas dos dois grupos. Considere α=0,05. Mostre o código em Python para a realização do teste e interprete o resultado.

**Exercício 4 - Resultados**:

Requisitos Técnicos:

O objeto central para este estudo foi a exploração da função “s**tats.shapiro / stats.levene / funções diversas Numpy**.

Resultado da Análise:

Estatísticas:

Grupo A: Média = 73.62, Desvio Padrão = 7.68

Grupo B: Média = 78.15, Desvio Padrão = 6.79

Distribuição Normal (Stats.Shapiro):

Grupo A: estatística=0.9744, p-valor=0.8438

Grupo B: estatística=0.9703, p-valor=0.7618

Dispersão dos Dados (Stats.Levene):

Estatística=0.2463, p-valor=0.6225

Resultado do Teste t de Student:

Resultado do Teste t (Estatístico): -1.9771

Resultado do Teste t (Valor): 0.0553

Disponibilidade: 38

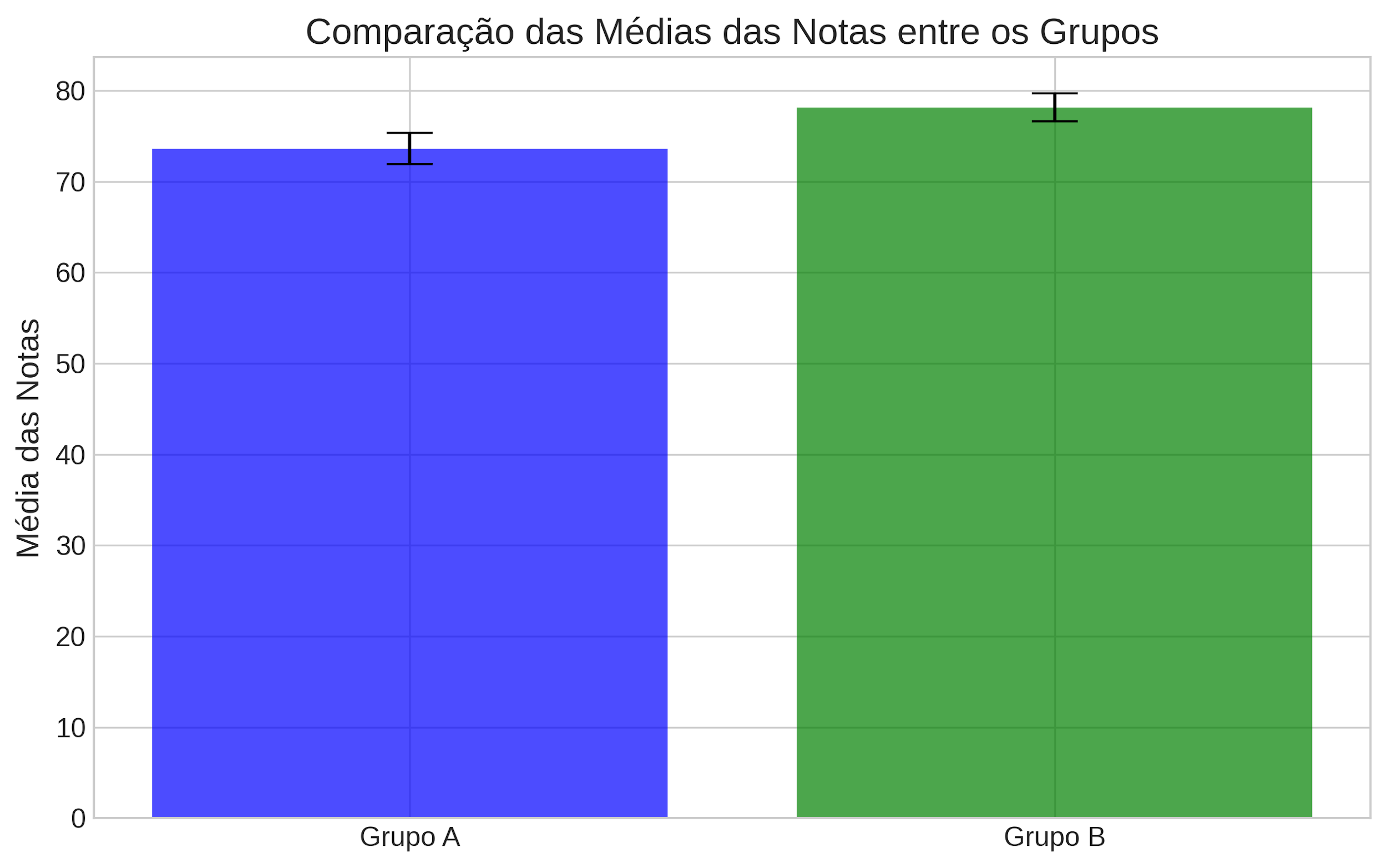
Alpha (α): 0.05

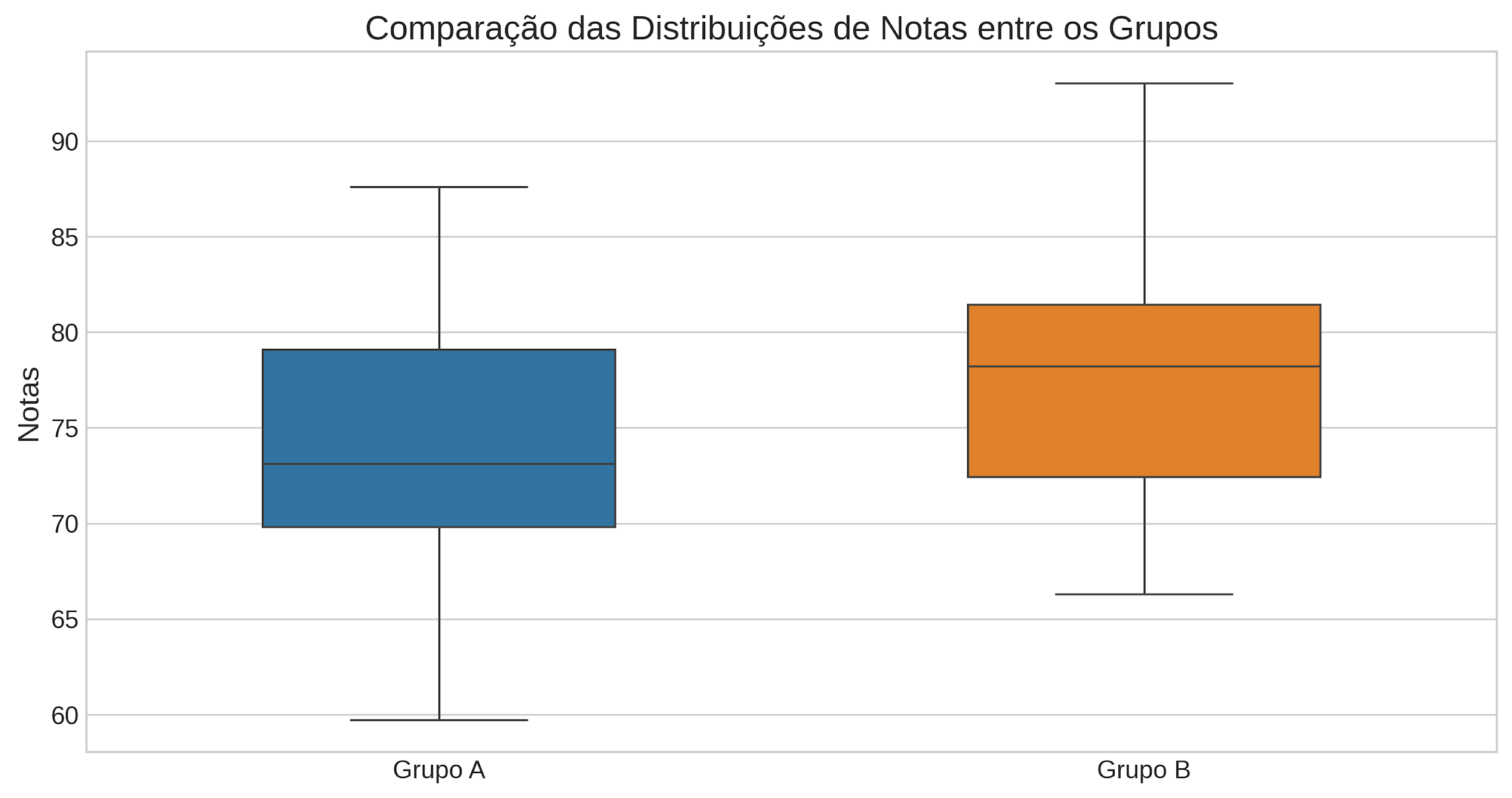
**Não foram apresentadas probabilidades estatísticas de diferenças entre as médias dos dois grupos.**

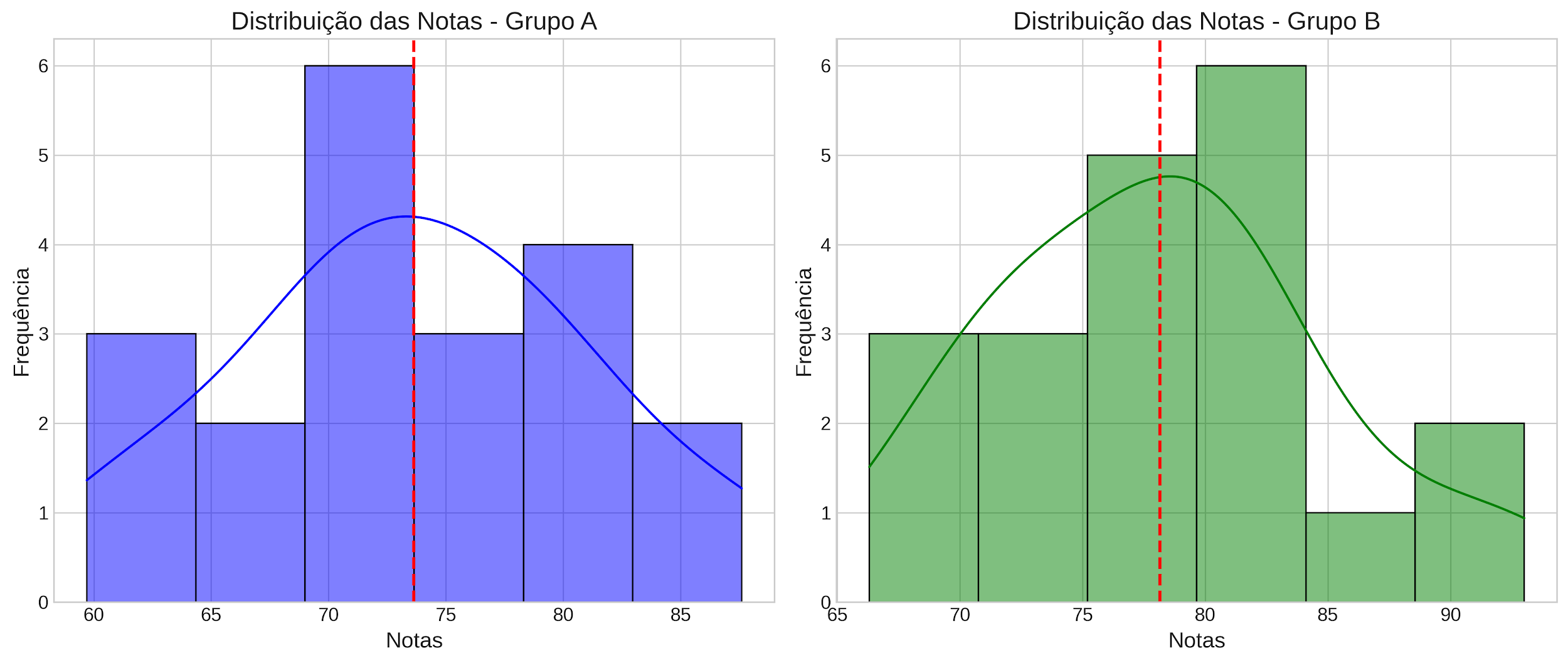
**Notas dos Grupos:**

| Grupo A | Grupo B |  | Grupo A | Grupo B |
| --- | --- | --- | --- | --- |
| 79 | 90.3 |  | 71.3 | 75.8 |
| 73.9 | 78.4 |  | 71.3 | 93 |
| 80.2 | 80.5 |  | 76.9 | 79.9 |
| 87.2 | 70 |  | 59.7 | 72.6 |
| 73.1 | 76.2 |  | 61.2 | 85.8 |
| 73.1 | 80.8 |  | 70.5 | 71.5 |
| 87.6 | 71.9 |  | 66.9 | 81.5 |
| 81.1 | 82.6 |  | 77.5 | 66.3 |
| 71.2 | 75.8 |  | 67.7 | 70.7 |
| 79.3 | 78 |  | 63.7 | 81.4 |

**Gráficos**:







**Python - Código Fonte:**

*'''*

*Parte 2: Testes de Hipótese / Exercício 4*

*Dois grupos de estudantes foram submetidos a métodos de ensino diferentes. As notas de 20 alunos*

*do Grupo A e 20 alunos do Grupo B estão disponíveis. Use o teste t de Student para determinar se*

*há uma diferença significativa entre as médias das notas dos dois grupos. Considere α=0,05. Mostre*

*o código em Python para a realização do teste e interprete o resultado.*

*Objeto de Estudo: stats.shapiro / stats.levene / funções diversas Numpy*

*'''*

*#Globais*

*#!/usr/bin/env python3*

*# -\*- coding: utf-8 -\*-*

*#Bibliotecas*

*import os*

*import pandas as pd*

*import numpy as np*

*import seaborn as sns*

*import matplotlib.pyplot as plt*

*from scipy import stats*

*#Conectar ao Google Drive*

*from google.colab import drive*

*drive.mount('/content/drive')*

*v\_caminho\_google\_drive = '/content/drive/MyDrive/[MBA]/[Data Science Experience]/[Listas]/'*

*v\_exercicio = '[Exercicio 04]/'*

*# Gerando a amostra aleatória.*

*np.random.seed(42)*

*# Gerando dados aleatórios/simulados para os dois grupos*

*v\_grupo\_a = np.random.normal(75, 8, 20) # Grupo A: média 75, desvio padrão 8*

*v\_grupo\_b = np.random.normal(80, 7, 20) # Grupo B: média 80, desvio padrão 7*

*# Limitando as notas entre 0 e 100*

*v\_grupo\_a = np.clip(v\_grupo\_a, 0, 100)*

*v\_grupo\_b = np.clip(v\_grupo\_b, 0, 100)*

*# Arredondando para uma casa decimal*

*v\_grupo\_a = np.round(v\_grupo\_a, 1)*

*v\_grupo\_b = np.round(v\_grupo\_b, 1)*

*# Criando um DataFrame para facilitar a visualização e análise*

*df\_grupos = pd.DataFrame({*

*'Grupo A': v\_grupo\_a,*

*'Grupo B': v\_grupo\_b*

*})*

*# Exportando os dados*

*v\_arq\_notas\_grupos = 'notas\_grupos.csv';*

*df\_grupos.to\_csv((v\_caminho\_google\_drive + v\_exercicio + v\_arq\_notas\_grupos), index=False)*

*#Funções Estatísticas*

*print("Estatísticas:")*

*print("\nGrupo A:")*

*print(f"Média: {np.mean(v\_grupo\_a):.2f}")*

*print(f"Desvio Padrão: {np.std(v\_grupo\_a, ddof=1):.2f}")*

*print(f"Mínimo: {np.min(v\_grupo\_a):.2f}")*

*print(f"Máximo: {np.max(v\_grupo\_a):.2f}")*

*print("\nGrupo B:")*

*print(f"Média: {np.mean(v\_grupo\_b):.2f}")*

*print(f"Desvio Padrão: {np.std(v\_grupo\_b, ddof=1):.2f}")*

*print(f"Mínimo: {np.min(v\_grupo\_b):.2f}")*

*print(f"Máximo: {np.max(v\_grupo\_b):.2f}")*

*# Verificando a distribuição normal dos valores (pressuposto do teste t)*

*#O pressuposto de normalidade no teste t refere-se à ideia de que os dados de cada grupo devem vir de uma distribuição normal (ou próxima de*

*#uma distribuição normal).*

*#Isso significa que os valores da variável dependente devem estar distribuídos simetricamente em torno da média, com a maioria dos valores*

*#concentrados perto do centro e poucos valores extremos.*

*print("\nDistribuição normal dos valores (Stats.Shapiro):")*

*print(f"Grupo A: estatística={stats.shapiro(v\_grupo\_a)[0]:.4f}, p-valor={stats.shapiro(v\_grupo\_a)[1]:.4f}")*

*print(f"Grupo B: estatística={stats.shapiro(v\_grupo\_b)[0]:.4f}, p-valor={stats.shapiro(v\_grupo\_b)[1]:.4f}")*

*# Verificando a homogeneidade das variâncias (pressuposto do teste t)*

*#A homogeneidade das variâncias, é um pressuposto fundamental em muitos testes estatísticos*

*#paramétricos, como o teste t de Student e a análise de variância (ANOVA). Significa que as variâncias (medida da dispersão dos dados)*

*#entre diferentes grupos ou amostras são aproximadamente iguais. Quando esse pressuposto é violado, a interpretação dos resultados do*

*#teste t pode ser comprometida, pois o erro padrão das médias pode ser distorcido.*

*print("\nValidando a Dispersão dos Dados (Stats.Levene):")*

*print(f"Estatística={stats.levene(v\_grupo\_a, v\_grupo\_b)[0]:.4f}, p-valor={stats.levene(v\_grupo\_a, v\_grupo\_b)[1]:.4f}")*

*# Realizando o teste t de Student para dados aleatórios*

*v\_alpha = 0.05*

*v\_resultado\_teste\_t = stats.ttest\_ind(v\_grupo\_a, v\_grupo\_b, equal\_var=True)*

*print("\nResultado do Teste t de Student:")*

*print(f"Resultado do Teste t (Estatístico): {v\_resultado\_teste\_t.statistic:.4f}")*

*print(f"Resultado do Teste t (Valor): {v\_resultado\_teste\_t.pvalue:.4f}")*

*print(f"Disponibilidade: {len(v\_grupo\_a) + len(v\_grupo\_b) - 2}")*

*print(f"Alpha (α): {v\_alpha}")*

*# Analisando o valor calculado*

*if v\_resultado\_teste\_t.pvalue < v\_alpha:*

*print("Foram apresentados probabilidades estatísticas de diferenças entre as médias dos dois grupos.")*

*else:*

*print("Não Foram apresentados probabilidades estatísticas de diferenças entre as médias dos dois grupos.")*

*# Criando visualizações*

*# Configuração para gráficos*

*plt.style.use('seaborn-v0\_8-whitegrid')*

*plt.rcParams['figure.figsize'] = (10, 6)*

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

*plt.rcParams['axes.labelsize'] = 14*

*plt.rcParams['axes.titlesize'] = 16*

*plt.rcParams['xtick.labelsize'] = 12*

*plt.rcParams['ytick.labelsize'] = 12*

*# Boxplot para comparar as distribuições*

*v\_arq\_comparacao\_notas = 'boxplot\_grupos'*

*plt.figure(figsize=(12, 6))*

*sns.boxplot(data=[v\_grupo\_a, v\_grupo\_b], width=0.5)*

*plt.xticks([0, 1], ['Grupo A', 'Grupo B'])*

*plt.ylabel('Notas')*

*plt.title('Comparação das Distribuições de Notas entre os Grupos')*

*plt.savefig((v\_caminho\_google\_drive + v\_exercicio + v\_arq\_comparacao\_notas + '.png'), dpi=300, bbox\_inches='tight')*

*plt.savefig((v\_caminho\_google\_drive + v\_exercicio + v\_arq\_comparacao\_notas + '.pdf'))*

*# Histograma para visualizar a distribuição das notas*

*plt.figure(figsize=(14, 6))*

*plt.subplot(1, 2, 1)*

*sns.histplot(v\_grupo\_a, kde=True, color='blue')*

*plt.axvline(np.mean(v\_grupo\_a), color='red', linestyle='dashed', linewidth=2)*

*plt.title('Distribuição das Notas - Grupo A')*

*plt.xlabel('Notas')*

*plt.ylabel('Frequência')*

*plt.subplot(1, 2, 2)*

*sns.histplot(v\_grupo\_b, kde=True, color='green')*

*plt.axvline(np.mean(v\_grupo\_b), color='red', linestyle='dashed', linewidth=2)*

*plt.title('Distribuição das Notas - Grupo B')*

*plt.xlabel('Notas')*

*plt.ylabel('Frequência')*

*plt.tight\_layout()*

*v\_arq\_histograma\_notas = 'histograma\_grupos'*

*plt.savefig((v\_caminho\_google\_drive + v\_exercicio + v\_arq\_histograma\_notas + '.png'), dpi=300, bbox\_inches='tight')*

*plt.savefig((v\_caminho\_google\_drive + v\_exercicio + v\_arq\_histograma\_notas + '.pdf'))*

*# Gráfico de barras para comparar as médias*

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

*v\_medias = [np.mean(v\_grupo\_a), np.mean(v\_grupo\_b)]*

*v\_erros = [stats.sem(v\_grupo\_a), stats.sem(v\_grupo\_b)] # Erro padrão da média*

*plt.bar([0, 1], v\_medias, yerr=v\_erros, capsize=10, color=['blue', 'green'], alpha=0.7)*

*plt.xticks([0, 1], ['Grupo A', 'Grupo B'])*

*plt.ylabel('Média das Notas')*

*plt.title('Comparação das Médias das Notas entre os Grupos')*

*v\_arq\_barras\_medias = 'barras\_medias'*

*plt.savefig((v\_caminho\_google\_drive + v\_exercicio + v\_arq\_barras\_medias + '.png'), dpi=300, bbox\_inches='tight')*

*plt.savefig((v\_caminho\_google\_drive + v\_exercicio + v\_arq\_barras\_medias + '.pdf'))*

*# Salvando os resultados em um arquivo de texto*

*v\_arq\_resultado\_final = 'resultados\_teste\_t.txt'*

*with open((v\_caminho\_google\_drive + v\_exercicio + v\_arq\_resultado\_final), 'w') as f:*

*f.write("Resultados do Teste t de Student\n")*

*f.write("===============================\n\n")*

*f.write("Estatísticas:\n")*

*f.write(f"Grupo A: Média = {np.mean(v\_grupo\_a):.2f}, Desvio Padrão = {np.std(v\_grupo\_a, ddof=1):.2f}\n")*

*f.write(f"Grupo B: Média = {np.mean(v\_grupo\_b):.2f}, Desvio Padrão = {np.std(v\_grupo\_b, ddof=1):.2f}\n\n")*

*f.write("Distribuição Normal (Stats.Shapiro):\n")*

*f.write(f"Grupo A: estatística={stats.shapiro(v\_grupo\_a)[0]:.4f}, p-valor={stats.shapiro(v\_grupo\_a)[1]:.4f}\n")*

*f.write(f"Grupo B: estatística={stats.shapiro(v\_grupo\_b)[0]:.4f}, p-valor={stats.shapiro(v\_grupo\_b)[1]:.4f}\n\n")*

*f.write("Dispersão dos Dados (Stats.Levene):\n")*

*f.write(f"Estatística={stats.levene(v\_grupo\_a, v\_grupo\_b)[0]:.4f}, p-valor={stats.levene(v\_grupo\_a, v\_grupo\_b)[1]:.4f}\n\n")*

*f.write("Resultado do Teste t de Student:\n")*

*f.write(f"Resultado do Teste t (Estatístico):: {v\_resultado\_teste\_t.statistic:.4f}\n")*

*f.write(f"Resultado do Teste t (Valor): {v\_resultado\_teste\_t.pvalue:.4f}\n")*

*f.write(f"Disponibilidade: {len(v\_grupo\_a) + len(v\_grupo\_b) - 2}\n")*

*f.write(f"Alpha (α): {v\_alpha}\n\n")*

*if v\_resultado\_teste\_t.pvalue < v\_alpha:*

*f.write("Foram apresentados probabilidades estatísticas de diferenças entre as médias dos dois grupos.\n")*

*else:*

*f.write("Não foram apresentados probabilidades estatísticas de diferenças entre as médias dos dois grupos.\n")*

*print("\nFim da Análise. Os resultados foram salvos em arquivos.")*

**Parte 2: Teste de Hipótese**

**Exercício 5**:

Uma pesquisa foi realizada em duas cidades para verificar a preferência dos moradores por dois tipos de bebidas (A e B). A tabela de contingência a seguir mostra os resultados:

Padrão do plano de fundo, Tabela

O conteúdo gerado por IA pode estar incorreto.

Realize o teste Qui-Quadrado de independência para verificar se a preferência pela bebida é independente da cidade. Utilize Python para calcular a estatística do teste e o valor-p. Considere α=0,05 e interprete os resultados.

**Exercício 5 - Resultados**:

Requisitos Técnicos:

O objeto central para este estudo foi a exploração da função “**stats.chi2\_contingency”.**

Resultado da Análise:

Estatística do teste Qui-Quadrado: 16.9899

Teste Qui-Quadrado (Valor) 0.0000

Graus de liberdade: 1

**Conclusão: Como o "Teste Qui-Quadrado (valor)” é (0.0000) ficando abaixo do nível de significância (0.05), conclui-se que a preferência pela bebida depende da cidade.**

