**Análise Estatística e Testes de Hipóteses**

Este repositório contém uma análise de dados focada em testes de hipóteses, utilizando datasets para responder a questões de negócio relacionadas com testes A/B e estatísticas de pacientes de um hospital. O projeto demonstra a aplicação de vários métodos estatísticos para extrair conclusões significativas a partir dos dados.

**Contexto do Projeto**

A análise está dividida em duas partes principais:

1. **Teste A/B:** Análise do ficheiro experimento\_teste\_ab.csv para determinar qual de duas versões de uma página (A ou B) tem a maior taxa de conversão e para calcular o tamanho de amostra necessário para futuros testes.
2. **Análise de Pacientes:** Utilização do ficheiro pacientes.csv para aplicar testes de hipóteses (Teste t, ANOVA, Qui-quadrado, etc.) e responder a questões sobre a relação entre variáveis como pressão arterial, idade, género e etnia.

**Questões Investigadas**

Este projeto responde às seguintes perguntas com base nos dados fornecidos:

* Qual cenário (A ou B) possui a maior taxa de conversão?
* Qual é a diferença na pressão arterial média entre fumadores e não fumadores?
* Existe uma diferença significativa na pressão arterial entre diferentes grupos étnicos?
* Há uma relação entre o género dos pacientes e a presença de condições de saúde adicionais?
* A distribuição da pressão arterial na população segue uma distribuição normal? *(Entre outras questões detalhadas no notebook e no relatório.)*

**Estrutura do Repositório**

O projeto está organizado da seguinte forma:

* **/data**: Contém os datasets originais (experimento\_teste\_ab.csv e pacientes.csv).
* **/notebook**: Inclui o Jupyter Notebook Testes\_de\_Hipoteses.ipynb com todo o código para a limpeza de dados, análise exploratória e aplicação dos testes estatísticos.
* **/relatorio**: Contém o relatório final Estatística - Teste de Hipóteses.pdf que resume as questões e conclusões do estudo.
* **/modelo**: Armazena um ficheiro analise\_estatistica.pkl simbólico, indicando que este projeto não envolve modelagem preditiva.
* requirements.txt: Lista de todas as bibliotecas Python necessárias para reproduzir o ambiente e executar a análise.
* .gitignore: Ficheiro de configuração para que o Git ignore ficheiros e pastas desnecessários.
* README.md: Este guia, com a documentação completa do projeto.

**Como Instalar e Executar o Projeto**

Siga os passos abaixo para configurar o ambiente e executar a análise no seu computador.

**Pré-requisitos**

* Python 3.8 ou superior
* pip (gestor de pacotes do Python)

**1. Clonar o Repositório**

Clone este repositório para a sua máquina local usando o seguinte comando no seu terminal:

git clone <URL-DO-SEU-REPOSITÓRIO-AQUI>

cd <NOME-DA-PASTA-DO-REPOSITÓRIO>

**2. Criar um Ambiente Virtual (Recomendado)**

Para isolar as dependências do projeto, crie e ative um ambiente virtual:

# Para Windows

python -m venv venv

venv\Scripts\activate

# Para macOS/Linux

python3 -m venv venv

source venv/bin/activate

**3. Instalar as Dependências**

Com o ambiente virtual ativado, instale todas as bibliotecas necessárias:

pip install -r requirements.txt

**4. Executar a Análise**

Para visualizar e reexecutar a análise, inicie o Jupyter Notebook:

jupyter notebook notebook/Testes\_de\_Hipoteses.ipynb

**Ferramentas Utilizadas**

* **Linguagem:** Python 3
* **Bibliotecas Principais:**
  + **Pandas & Numpy:** para manipulação e análise de dados.
  + **Matplotlib & Seaborn:** para visualização de dados.
  + **Scipy.stats:** para a aplicação dos testes estatísticos.
  + **Jupyter Notebook:** como ambiente de desenvolvimento interativo.

**PDF do notebook**

**Estatística - "Testes de Hipóteses"**

**Bibliotecas:**

**--------------------------------------------------**

**Bibliotecas-base**

import pandas as pd

import numpy as np

**Bibliotecas para visualização de dados**

import seaborn as sns

import matplotlib as plt

import matplotlib.pyplot as plt

**Biblioteca para estatística**

import scipy.stats as stats

from scipy.stats import ttest\_1samp

from scipy.stats import ttest\_ind

from scipy.stats import f\_oneway

from scipy.stats import chi2\_contingency

from scipy.stats import kstest

from statsmodels.stats.power import TTestIndPower

**--------------------------------------------------**

**1. Qual dos cenários tem a maior taxa de conversão?**

**--------------------------------------------------**

teste\_ab = pd.read\_csv('experimento\_teste\_ab.csv')

**Calcular a taxa de conversão para cada grupo**

taxas\_conversao = teste\_ab.groupby('Versão\_Página')['Conversões'].mean()

print(taxas\_conversao)

**Conclusão:**

**O cenário B possui a maior taxa de conversão (0.12 ou 12%), enquanto o cenário A tem uma taxa de 0.08 (ou 8%).**

**--------------------------------------------------**

**2. Calcule qual o tamanho da amostra necessária para o desenvolvimento de um teste A/B**

**--------------------------------------------------**

**Parâmetros**

alpha = 0.05 # Nível de significância

power = 0.80 # Poder estatístico

p1 = 0.08 # Taxa de conversão atual (Cenário A)

p2 = 0.10 # Taxa de conversão desejada (aumento para 10%)

**Z-scores**

z\_alpha = stats.norm.ppf(1 - alpha / 2)

z\_beta = stats.norm.ppf(power)

**Cálculo do tamanho da amostra por grupo**

n = ((z\_alpha \* np.sqrt(2 \* p1 \* (1 - p1)) + z\_beta \* np.sqrt(p1 \* (1 - p1) + p2 \* (1 - p2)))\*\*2) / ((p1 - p2)\*\*2)

print(f"Tamanho da amostra necessário por grupo: {int(np.ceil(n))}")

**Conclusão:**

**Para detetar um aumento na taxa de conversão de 8% para 10% com 95% de confiança e 80% de poder,**

**seriam necessários aproximadamente 2125 visitantes por grupo (A e B).**

**--------------------------------------------------**

**Base de Dados de Pacientes**

**--------------------------------------------------**

pacientes = pd.read\_csv('pacientes.csv')

print(pacientes.head())

**--------------------------------------------------**

**3. Considerando uma amostra de 45 números... a pressão arterial média da amostra é significativamente diferente da população?**

**--------------------------------------------------**

**Hipótese Nula (H0): A pressão arterial média da amostra é igual à da população.**

**Hipótese Alternativa (H1): A pressão arterial média da amostra é diferente da da população.**

indices\_amostra = [909, 751, 402, 400, 726, 39, 184, 269, 255, 769, 209, 715, 677, 381, 793, 697, 89, 47, 233, 206, 27, 442, 126, 401, 788, 882, 17, 723, 534, 513, 50, 911, 42, 638, 595, 56, 362, 395, 866, 281, 196, 26, 722, 107, 71]

amostra\_pacientes = pacientes.loc[indices\_amostra]

media\_populacao = pacientes['Pressao\_Arterial'].mean()

media\_amostra = amostra\_pacientes['Pressao\_Arterial'].mean()

t\_statistic, p\_value = ttest\_1samp(amostra\_pacientes['Pressao\_Arterial'], media\_populacao)

print(f"Média da População: {media\_populacao}")

print(f"Média da Amostra: {media\_amostra}")

print(f"Estatística t: {t\_statistic}")

print(f"Valor-p: {p\_value}")

alpha = 0.05

if p\_value < alpha:

print("Rejeitamos a hipótese nula.")

else:

print("Não rejeitamos a hipótese nula.")

**--------------------------------------------------**

**4. Qual é a diferença na pressão arterial média entre fumadores e não fumadores?**

**--------------------------------------------------**

**Hipótese Nula (H0): A pressão arterial média é a mesma para fumadores e não fumadores.**

**Hipótese Alternativa (H1): A pressão arterial média é diferente entre fumadores e não fumadores.**

**Visualização com Boxplot**

plt.figure(figsize=(8, 6))

sns.boxplot(x='Fumante', y='Pressao\_Arterial', data=pacientes)

plt.title('Pressão Arterial por Status de Fumador')

plt.show()

**Teste t para amostras independentes**

fumadores = pacientes[pacientes['Fumante'] == 1]['Pressao\_Arterial']

nao\_fumadores = pacientes[pacientes['Fumante'] == 0]['Pressao\_Arterial']

t\_statistic\_fum, p\_value\_fum = ttest\_ind(fumadores, nao\_fumadores)

print(f"Estatística t: {t\_statistic\_fum}")

print(f"Valor-p: {p\_value\_fum}")

if p\_value\_fum < 0.05:

print("Rejeitamos a hipótese nula.")

else:

print("Não rejeitamos a hipótese nula.")

**--------------------------------------------------**

**5. A pressão arterial média para pacientes com condições de saúde adicionais é igual à de pacientes sem?**

**--------------------------------------------------**

**Hipótese Nula (H0): As médias são iguais.**

**Hipótese Alternativa (H1): As médias são diferentes.**

com\_condicoes = pacientes[pacientes['Estado\_Saude'] == 1]['Pressao\_Arterial']

sem\_condicoes = pacientes[pacientes['Estado\_Saude'] == 0]['Pressao\_Arterial']

t\_statistic\_saude, p\_value\_saude = ttest\_ind(com\_condicoes, sem\_condicoes)

print(f"Estatística t: {t\_statistic\_saude}")

print(f"Valor-p: {p\_value\_saude}")

alpha = 0.06

if p\_value\_saude < alpha:

print("Rejeitamos a hipótese nula.")

else:

print("Não rejeitamos a hipótese nula.")

**--------------------------------------------------**

**6. Existe uma diferença significativa na pressão arterial média entre diferentes grupos étnicos? (ANOVA)**

**--------------------------------------------------**

**Hipótese Nula (H0): As médias são as mesmas em todos os grupos.**

**Hipótese Alternativa (H1): Há diferença em pelo menos dois grupos.**

**Visualização com Boxplot**

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

sns.boxplot(x='Nome\_Etnia', y='Pressao\_Arterial', data=pacientes)

plt.title('Pressão Arterial por Etnia')

plt.show()

**Teste ANOVA**

grupos\_etnia = [pacientes[pacientes['Nome\_Etnia'] == etnia]['Pressao\_Arterial'] for etnia in pacientes['Nome\_Etnia'].unique()]

f\_statistic, p\_value\_anova = f\_oneway(\*grupos\_etnia)

print(f"Estatística F: {f\_statistic}")

print(f"Valor-p: {p\_value\_anova}")

if p\_value\_anova < 0.05:

print("Rejeitamos a hipótese nula.")

else:

print("Não rejeitamos a hipótese nula.")

**--------------------------------------------------**

**7. Há uma relação entre o género dos pacientes e a presença de condições de saúde adicionais? (Qui-quadrado)**

**--------------------------------------------------**

tabela\_contingencia = pd.crosstab(pacientes['Nome\_Genero'], pacientes['Nome\_Estado\_Saude'])

print(tabela\_contingencia)

chi2, p\_value\_chi, \_, \_ = chi2\_contingency(tabela\_contingencia)

print(f"Estatística Qui-quadrado: {chi2}")

print(f"Valor-p: {p\_value\_chi}")

if p\_value\_chi < 0.05:

print("Rejeitamos a hipótese nula (há uma relação).")

else:

print("Não rejeitamos a hipótese nula (não há evidência de relação).")

**--------------------------------------------------**

**8. Existe uma associação entre a idade dos pacientes e sua pressão arterial? (Correlação)**

**--------------------------------------------------**

correlacao, p\_value\_corr = stats.pearsonr(pacientes['Idade'], pacientes['Pressao\_Arterial'])

print(f"Coeficiente de Correlação de Pearson: {correlacao}")

print(f"Valor-p: {p\_value\_corr}")

if p\_value\_corr < 0.05:

print("Rejeitamos a hipótese nula (há uma associação).")

else:

print("Não rejeitamos a hipótese nula (não há associação linear).")

**--------------------------------------------------**

**9. Qual é o intervalo de confiança para a média da pressão arterial entre os pacientes com condições de saúde adicionais?**

**--------------------------------------------------**

pacientes\_com\_condicoes = pacientes[pacientes['Nome\_Estado\_Saude'] == 'Com condições de saúde adicionais']['Pressao\_Arterial']

intervalo\_confianca = stats.t.interval(0.95, len(pacientes\_com\_condicoes)-1, loc=np.mean(pacientes\_com\_condicoes), scale=stats.sem(pacientes\_com\_condicoes))

print(f"Intervalo de Confiança de 95%: {intervalo\_confianca}")

**--------------------------------------------------**

**10. A distribuição da pressão arterial na população segue uma distribuição normal? (Kolmogorov-Smirnov)**

**--------------------------------------------------**

**Hipótese Nula (H0): A distribuição segue uma normal.**

**Hipótese Alternativa (H1): A distribuição não segue uma normal.**

pressao\_arterial\_populacao = pacientes['Pressao\_Arterial']

statistic, p\_value\_ks = kstest(pressao\_arterial\_populacao, 'norm')

print(f"Estatística do teste KS: {statistic}")

print(f"Valor-p: {p\_value\_ks}")

if p\_value\_ks < 0.05:

print("Rejeitamos a hipótese nula.")

else:

print("Não rejeitamos a hipótese nula.")

Com certeza! Analisei os novos ficheiros e entendi perfeitamente o que precisa. O seu projeto de **Testes de Hipóteses** está muito bem estruturado, com análises estatísticas claras no notebook e um relatório em PDF que resume as questões.

Para "publicá-lo" no GitHub como um portfólio profissional, o mais importante é a organização e a documentação, para que qualquer pessoa (como um recrutador) consiga entender o seu trabalho e, se quiser, reproduzir a sua análise.

Notei que, ao contrário do projeto anterior, este notebook foca-se em análises estatísticas e testes de hipóteses, e não na criação de um modelo de machine learning. Por isso, **não há um modelo para ser salvo num ficheiro .pkl**. A entrega principal aqui é o próprio notebook com as análises.

A pensar nisso, preparei quatro ficheiros essenciais que vão estruturar o seu repositório de forma limpa e profissional, cumprindo todos os seus requisitos:

1. **README.md**: O guia principal do seu projeto. Explica o contexto da análise, as perguntas que respondeu, a estrutura das pastas e como qualquer pessoa pode instalar as dependências e executar o seu notebook.
2. **analise\_estatistica.pkl**: Para cumprir o requisito de entrega de um ficheiro .pkl, criei um ficheiro simbólico que contém uma mensagem a explicar que o projeto é de análise estatística, não de modelagem preditiva. Isto demonstra que sabe como criar o ficheiro, mesmo que não seja aplicável aqui.
3. **requirements.txt**: Um ficheiro essencial que lista as bibliotecas Python que utilizou.
4. **.gitignore**: Um ficheiro de boas práticas que informa o Git para ignorar pastas e ficheiros desnecessários.

Aqui estão os ficheiros prontos para adicionar à pasta do seu projeto.

**O que fazer agora:**

1. **Organize suas pastas:** No seu computador, crie as pastas data, notebook, modelo e relatorio.
2. **Mova os ficheiros:**
   * experimento\_teste\_ab.csv e pacientes.csv para a pasta /data.
   * Testes\_de\_Hipoteses.ipynb para a pasta /notebook.
   * Estatística - Teste de Hipóteses.pdf para a pasta /relatorio.
3. **Adicione os ficheiros gerados:** Coloque o README.md, o .pkl (dentro da pasta /modelo), o requirements.txt e o .gitignore que criei na estrutura de pastas correta.
4. **Envie para o GitHub:** Siga os passos de git que discutimos anteriormente para criar o repositório e enviar todos os ficheiros.

Com esta estrutura, o seu projeto de testes de hipóteses estará apresentado de forma muito profissional e completa no seu portfólio. Parabéns pela análise!