FIAP

DATA SCIENCE & STATISTICAL COMPUTING

ANÁLISE EXPLORATÓRIA E ESTATÍSTICA DESCRITIVA

Prof. Dr. Carlos André M. Vieira
profcarlos.vieira@fiap.com.br

MANIPULAÇÃO DE DADOS COM PANDAS

- **O que é Pandas?**
  - ❑ Uma biblioteca do Python projetada para manipulação e análise de dados;
  - ❑ Oferece estruturas de dados rápidas, flexíveis e expressivas para trabalhar com dados estruturados.

MANIPULAÇÃO DE DADOS COM PANDAS

- **Por que Pandas?**
  - ❑ Facilita a limpeza, transformação e análise de dados;
  - ❑ Integração perfeita com outras bibliotecas como NumPy e Matplotlib;
  - ❑ Amplamente utilizado na ciência de dados, finanças, e em áreas que necessitam de análise de dados complexos.
- **Características Principais:**
  - ❑ DataFrames e Series para manipulação de tabelas e séries temporais;
  - ❑ Leitura e escrita de dados entre múltiplos formatos;
  - ❑ Funções poderosas para limpeza, filtragem, e agregação de dados.

MANIPULAÇÃO DE DADOS COM PANDAS

- **DataFrames:** são estruturas de dados bidimensionais, como uma planilha ou uma tabela SQL, que são fundamentais no trabalho com o Pandas. Eles nos permitem armazenar e manipular tabelas de dados com facilidade;
- Você pode realizar uma variedade de operações em DataFrames, como ordenação, subconjuntos, e criação de novas colunas. Isso oferece uma grande flexibilidade para preparar e analisar seus dados;
- Dominar DataFrames é essencial porque a maioria dos conjuntos de dados em análises reais é estruturada desta forma. Se você pode manipular DataFrames, você pode lidar com a maioria dos desafios de dados que encontrará.

CARACTERÍSTICAS DOS DATAFRAMES

- **.head()**
  - ❑ É usada para obter as primeiras N linhas de um DataFrame. Por padrão, ela retorna as primeiras 5 linhas. É uma maneira rápida de ter uma noção dos dados, incluindo o tipo de informação e os valores que você pode esperar;
- **.tail()**
  - ❑ Similar à função .head(), mas para as últimas N linhas. Esta função é útil para verificar rapidamente os últimos dados inseridos ou para confirmar se os dados estão completos e não foram truncados;
- **.info()**
  - ❑ Fornece um resumo conciso do DataFrame. Isso inclui o número de entradas, a quantidade de colunas, os tipos de dados de cada coluna, a quantidade de valores não nulos e o uso de memória.

CARACTERÍSTICAS DOS DATAFRAMES

- **.shape**
  - Esta não é uma função, mas um atributo que retorna uma tupla representando a dimensionalidade do DataFrame, ou seja, o número de linhas e colunas. Isso é essencial para confirmar se a manipulação de dados, como merges ou joins, resultou no número esperado de linhas e colunas;
- **.describe()**
  - Gera estatísticas descritivas que resumem a tendência central, dispersão e forma da distribuição de um dataset, excluindo valores NaN. Por padrão, fornece informações como contagem, média, desvio padrão, mínimo, quartis e máximo para colunas numéricas. É extremamente útil para uma análise exploratória inicial para entender melhor a distribuição dos dados numéricos. Pode ser transporta com um **.T**.

ATRIBUTOS DOS DATAFRAMES

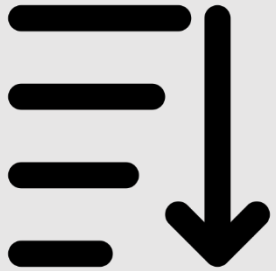
- **.values**
  - ❑ Este atributo contém os dados dentro do DataFrame como um array NumPy. É útil quando você precisa dos dados brutos para operações ou algoritmos que operam diretamente em arrays NumPy. Por exemplo, ao passar os dados para um modelo de machine learning, muitas vezes você utilizará o atributo;
- **.columns**
  - ❑ Este atributo contém os rótulos das colunas do DataFrame. É essencial para entender as variáveis que você tem em seu conjunto de dados e para manipular colunas específicas. Por exemplo, você pode querer renomear colunas ou selecionar um subconjunto delas para análise.

ATRIBUTOS DOS DATAFRAMES

- **.index**
  - Este atributo contém os rótulos das linhas. O índice pode ser uma simples sequência de números, mas também pode conter datas, strings, ou múltiplos níveis (em DataFrames hierárquicos). O índice é crítico para alinhar dados durante operações como junções, e também é útil para subconjunto de linhas baseadas em rótulos.

ORDENAÇÃO

- **.sort_values('col')**
  - ❑ Permite ordenar um DataFrame com base nos valores de uma ou mais colunas. A função ordenará o DataFrame com base nos valores da 'coluna' em ordem ascendente, que é o padrão;
  - ❑ É possível ordenar por mais de uma coluna. Isso é feito passando uma lista de nomes de colunas para .sort_values(). Por exemplo, df.sort_values(['coluna1', 'coluna2']) ordenará primeiro pela 'coluna1' e depois pela 'coluna2'.



ORDENAÇÃO

- **.sort_values('col', ascending=False)**
  - Você pode ordenar os dados em ordem decrescente passando o argumento ascending=False;
  - Ao ordenar por múltiplas colunas, você pode especificar uma ordem de classificação diferente para cada coluna usando uma lista de booleanos no argumento ascending. Por exemplo, df.sort_values(['coluna1', 'coluna2'], ascending=[True, False]) ordenará 'coluna1' em ordem ascendente e 'coluna2' em ordem decrescente.
- **.sort_values('col', inplace=True)**
  - Por padrão, .sort_values() retorna um novo DataFrame ordenado, mas você pode ordenar o DataFrame no local passando inplace=True.

OUTRAS FUNÇÕES

- **.copy()**
  - ❑ Cria uma cópia independente do DataFrame, garantindo que alterações feitas na cópia não afetem o original. Isso é útil para evitar modificações acidentais quando manipulamos subconjuntos de dados. Sem .copy(), a operação pode não ter sido aplicada corretamente;

OUTRAS FUNÇÕES

- **.sort_index()**
  - ❑ Ordena o DataFrame com base nos índices, e não nos valores de uma coluna específica.Por padrão, o Pandas ordena em ordem crescente (ascending=True), mas pode ser alterado para decrescente (ascending=False).
- **.reset_index(drop=True)**
  - ❑ Reinicia os índices do DataFrame, retornando um novo DataFrame com o índice padrão (0, 1, 2, ...).Caso o índice antigo seja relevante, ele é movido para uma nova coluna.Para descartar o índice antigo, use drop=True.

SELEÇÃO DE SUBCONJUNTOS

- **df['col']**
  - ❑ Esta sintaxe é utilizada para selecionar uma coluna específica e retorna uma Series;
- **df[['col1', 'col2']]**
  - ❑ Esta sintaxe é utilizada para selecionar uma coluna específica e retorna um novo DataFrame;
- **df[df['col'] > valor]**
  - ❑ Esta sintaxe é utilizada para selecionar linhas com base em uma condição e retorna um DataFrame com linhas que atendem à condição;
- **df[df['col'] == 'texto']**
  - ❑ Esta sintaxe é utilizada para filtrar linhas que tenham uma correspondência exata de texto em uma coluna.

SELEÇÃO DE SUBCONJUNTOS

- **df[df['data'] < 'YYYY-MM-DD']**
  - ❑ Esta sintaxe é utilizada para filtrar linhas com base em datas, retornando linhas onde as datas são anteriores à data especificada;
- **df[(df['col1'] == valor1) & (df['col2'] < valor2)]**
  - ❑ A combinação de condições lógicas pode ser utilizada para filtrar linhas;
- **df[(df['col1'] == valor1) | (df['col2'] < valor2)]**
  - ❑ | é o operador "ou" e seleciona as linhas onde pelo menos uma das condições anteriores é verdadeira;
- **df[df['col1'].isin([valor1, valor2])]**
  - ❑ Este método é útil para filtrar linhas onde a coluna contém um dos valores especificados na lista.

CRIAÇÃO E MODIFICAÇÃO DE NOVAS COLUNAS

- **df['col1'] = df['col2'] - df['col3']**
  - ❑ Novas colunas podem ser criadas a partir de operações aritméticas com colunas existentes, como a soma, subtração, multiplicação, divisão e derivadas;
- **df['col1'] = df['col2'] * 1.60934**
  - ❑ Somar, subtrair, multiplicar, dividir ou realizar outra operação com uma coluna por uma constante pode criar uma nova coluna;
- **df['E'] = (df['B'] - df['B'].mean()) / df['B'].std()**
  - ❑ Também podemos aplicar funções para mudar a escala de uma coluna (normalização, padronização, etc.)

ESTATÍSTICA

- **Estatística:**
  - ❑ O campo da estatística é a prática e o estudo de coletar e analisar dados. Ele se concentra em métodos para representar, resumir e interpretar variáveis e relações encontradas em conjuntos de dados;
- **Estatísticas Descritivas:**
  - ❑ São um subconjunto do campo da estatística que se dedica a descrever e resumir dados. Fornecem informações que capturam a essência dos conjuntos de dados com métricas simples, como média, mediana, moda, variância e desvio padrão;
  - ❑ Ao contrário da estatística inferencial, que visa fazer previsões ou inferências sobre uma população a partir de uma amostra, a estatística descritiva foca apenas na descrição do que é observado nos dados coletados.

ESTATÍSTICA

- **Usos da Estatística:**
  - ❑ Qual a probabilidade de um cliente adquirir um serviço? Esse interesse aumenta se oferecermos métodos de pagamento alternativos?
  - ❑ Quantos hóspedes seu estabelecimento pode esperar? Como maximizar a taxa de ocupação?
  - ❑ Quantas variações de tamanhos de camisetas são necessárias para atender a 95% da população? É necessário produzir quantidades iguais de cada tamanho?
  - ❑ Comparação direta (Testes A/B): Entre duas campanhas publicitárias, qual é mais eficiente em incentivar a compra de um serviço?

ESTATÍSTICA

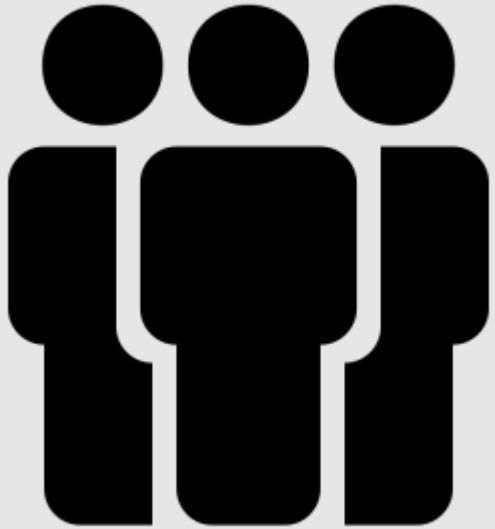
- **Estatística Descritiva:**
  - São um conjunto de procedimentos que visam a descrição e a sumarização de um conjunto de dados. Elas não tentam fazer inferências ou previsões, mas sim fornecer uma visão clara e concisa dos dados coletados;
  - Ex.: Média, mediana, moda, percentis, desvio-padrão, etc.
- **Estatística Inferencial:**
  - Utilizam uma amostra de dados para fazer generalizações ou inferências sobre uma população maior. Elas são usadas para testar hipóteses ou fazer previsões com base em dados;
  - Ex.: Testes de hipótese, intervalos de confiança, regressões.

ESTATÍSTICA

- **Dados Numéricos:** São valores quantitativos que representam uma quantidade mensurável e podem ser manipulados matematicamente:
  - ❑ **Contínuos:** Representam medidas e podem assumir qualquer valor dentro de um intervalo. Eles são muitas vezes o resultado de medições;
  - ❑ **Discretos:** Representam contagens e só podem assumir valores específicos e separados.
- **Dados Categóricos:** São valores qualitativos que representam características ou rótulos que não têm uma ordem matemática intrínseca:
  - ❑ **Nominais:** São dados que incluem categorias sem qualquer ordem ou ranking natural;
  - ❑ **Ordinais:** São categorias com uma ordem ou ranking natural, mas os intervalos entre as categorias não são necessariamente iguais.

ESTATÍSTICA

- **Dados Categóricos:** Podem ser convertidos em números:
  - ❑ **Nominais:** Solteiro/Casado (1,0);
  - ❑ **Ordinais:** Concordo Fortemente (5), Concordo (4), Neutro (3), Discordo (2), Discordo Fortemente (1).



ESTATÍSTICAS DESCRITIVAS

- **Medidas de Tendência Central:** As medidas de tendência central são estatísticas que resumem um conjunto de dados, apontando para o ponto central em torno do qual os dados estão agrupados. Elas são fundamentais para entender a distribuição dos dados e incluem a média, a mediana e a moda.
  - **Média:** Representa a soma de todos os valores dividida pelo número total de valores.
  - É a medida de tendência central mais comum e dá uma noção geral do valor médio dos dados.
  - Sensível a valores extremos (outliers).

ESTATÍSTICAS DESCRITIVAS

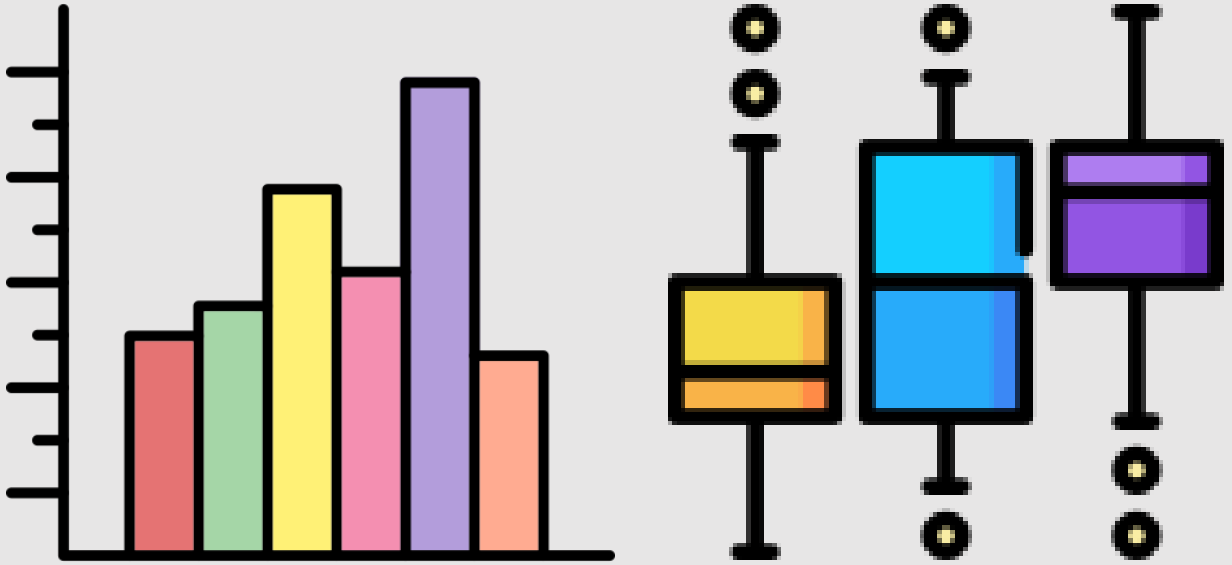
- **Medidas de Tendência Central:** As medidas de tendência central são estatísticas que resumem um conjunto de dados, apontando para o ponto central em torno do qual os dados estão agrupados. Elas são fundamentais para entender a distribuição dos dados e incluem a média, a mediana e a moda.
  - ❑ **Mediana:** O valor central em um conjunto de dados ordenados.
  - ❑ Divide os dados em duas partes iguais: 50% dos valores estão abaixo da mediana e 50% acima.
  - ❑ Menos sensível a outliers do que a média, sendo preferível em distribuições assimétricas.

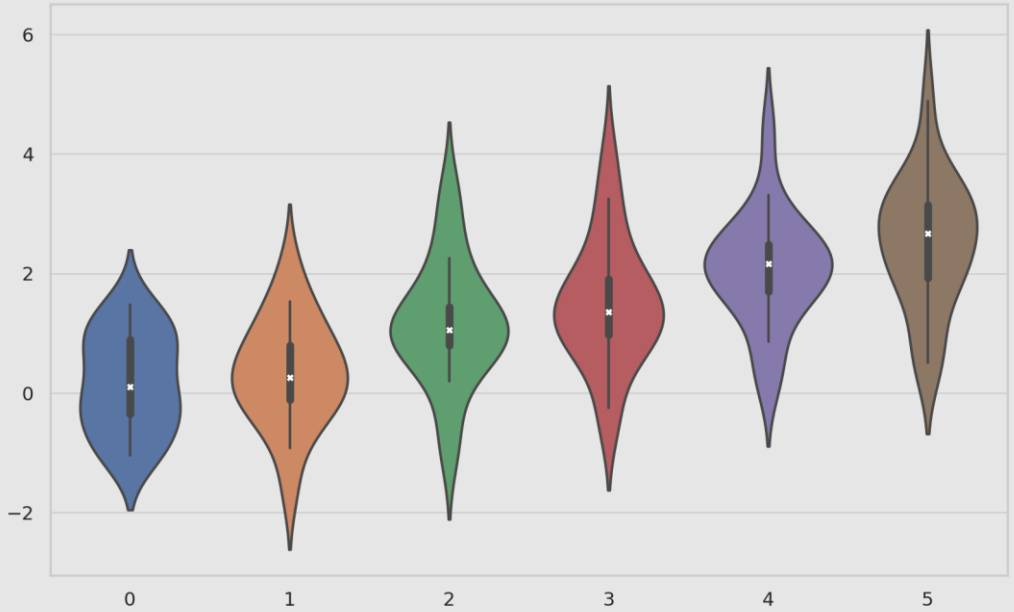
ESTATÍSTICAS DESCRITIVAS

- **Medidas de Tendência Central:** As medidas de tendência central são estatísticas que resumem um conjunto de dados, apontando para o ponto central em torno do qual os dados estão agrupados. Elas são fundamentais para entender a distribuição dos dados e incluem a média, a mediana e a moda.
  - **Moda:** Valor que aparece com mais frequência em um conjunto de dados.
  - Pode haver mais de uma moda em um conjunto de dados (bimodal, trimodal, etc.).
  - Útil para dados categóricos e para identificar categorias mais comuns.

ESTATÍSTICAS DESCRITIVAS

- **Gráficos Associados:** Histogramas e gráficos de caixa (boxplots) são frequentemente usados para visualizar a tendência central e a dispersão dos dados.





ESTATÍSTICAS DESCRITIVAS

- **.mean()**
  - Esta função calcula a média aritmética dos valores em uma coluna. A média é a soma de todos os valores dividida pelo número de valores;
- **.median()**
  - A mediana é o valor que separa a metade superior da metade inferior de um conjunto de dados. Se houver um número par de observações, a mediana é a média dos dois valores centrais;
- **.mode()**
  - A moda é o valor que aparece com maior frequência em um conjunto de dados. Um conjunto de dados pode ter uma única moda, várias modas ou nenhuma moda (se todos os valores aparecerem com a mesma frequência).

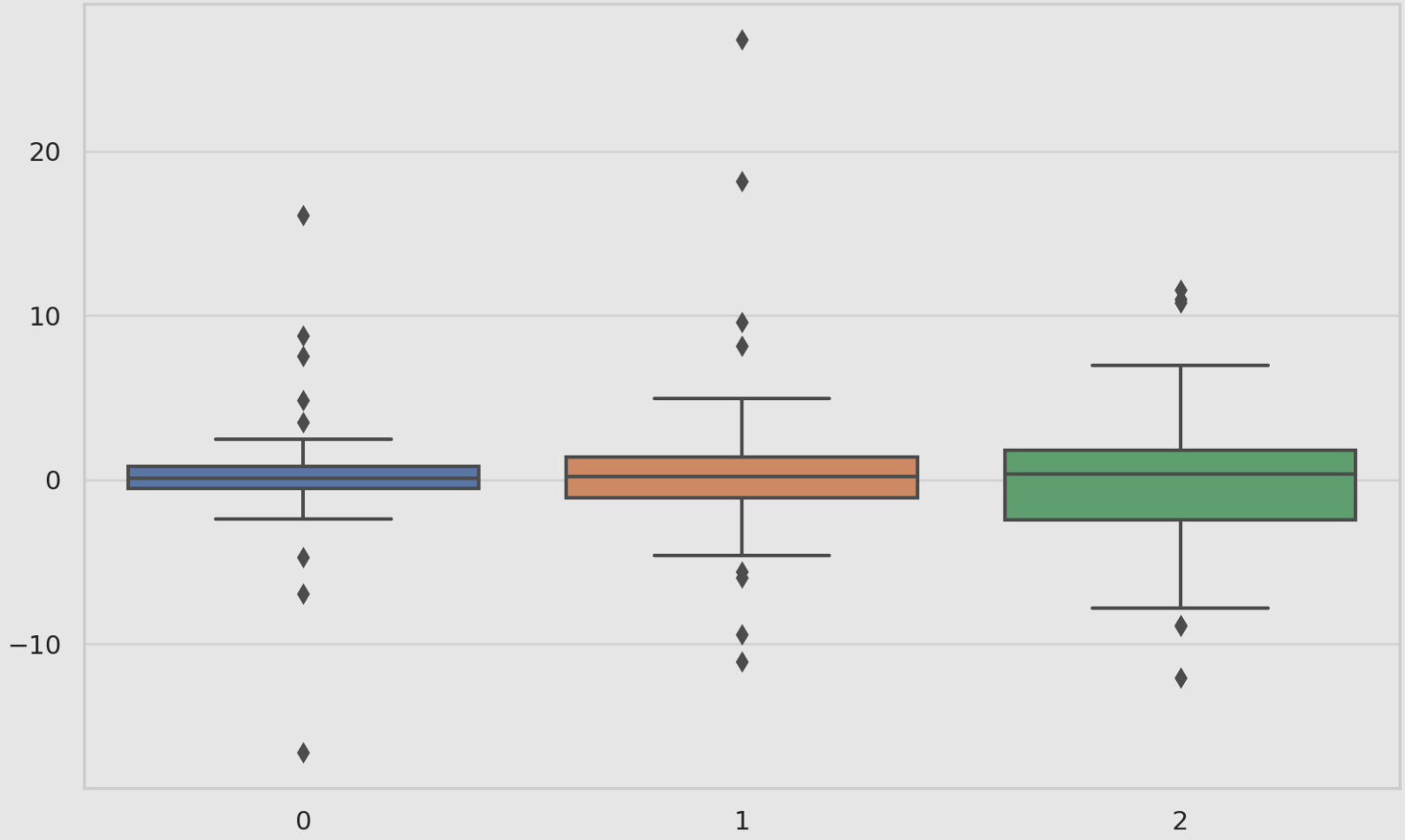
ESTATÍSTICAS DESCRITIVAS

- **.min()**
  - Esta função retorna o menor valor em uma coluna;
- **.max()**
  - Similarmente, .max() retorna o maior valor em uma coluna;
- **.quantile(x)**
  - A função .quantile() é usada para calcular quantis, que são pontos em uma distribuição que correspondem a certos percentis. Por exemplo, o quantil de 0.5 é a mediana (50º percentil), enquanto os quantis de 0.25 e 0.75 são, respectivamente, o primeiro e o terceiro quartis. Os quantis são úteis para entender a distribuição dos dados além da média e da mediana, particularmente em termos de variabilidade e caudas da distribuição. O **x** da função é o percentil em que os dados são divididos.

ESTATÍSTICAS DESCRITIVAS

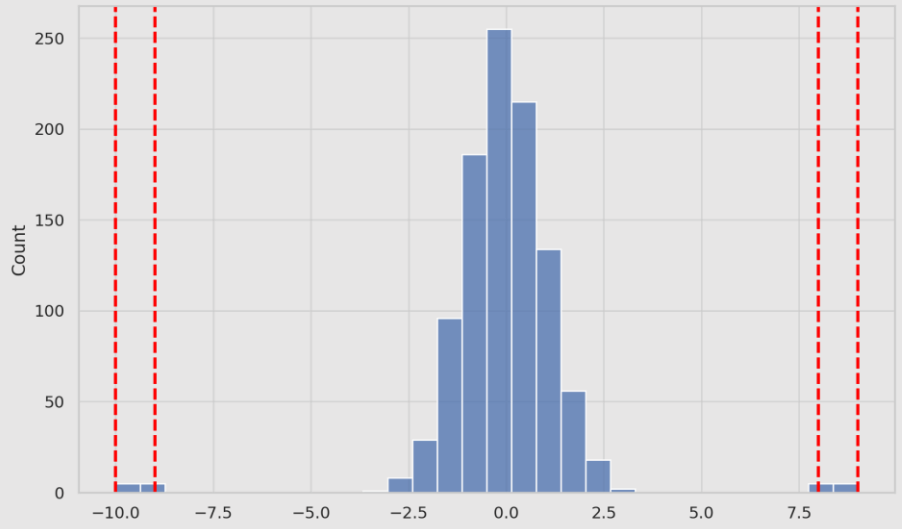
- **Outliers:** São pontos de dados que diferem significativamente do restante do conjunto de dados.
- Eles podem indicar uma variação experimental, um erro de entrada de dados, ou uma distribuição de dados com caudas pesadas.
- A identificação de outliers é uma parte crucial da análise de dados, pois podem distorcer resultados estatísticos, como médias e variâncias.
- Um ponto de dado é considerado um outlier se estiver abaixo de Q1 - 1.5 × IQR ou acima de Q3 + 1.5 × IQR, onde Q1 (percentil 25º) e Q3 (percentil 75º) são o primeiro e terceiro quartil, e IQR é o intervalo interquartil.
- **IQR:** Mede a variação dentro do meio 50% dos dados, subtraindo o primeiro quartil (percentil 25º) do terceiro quartil (percentil 75º).

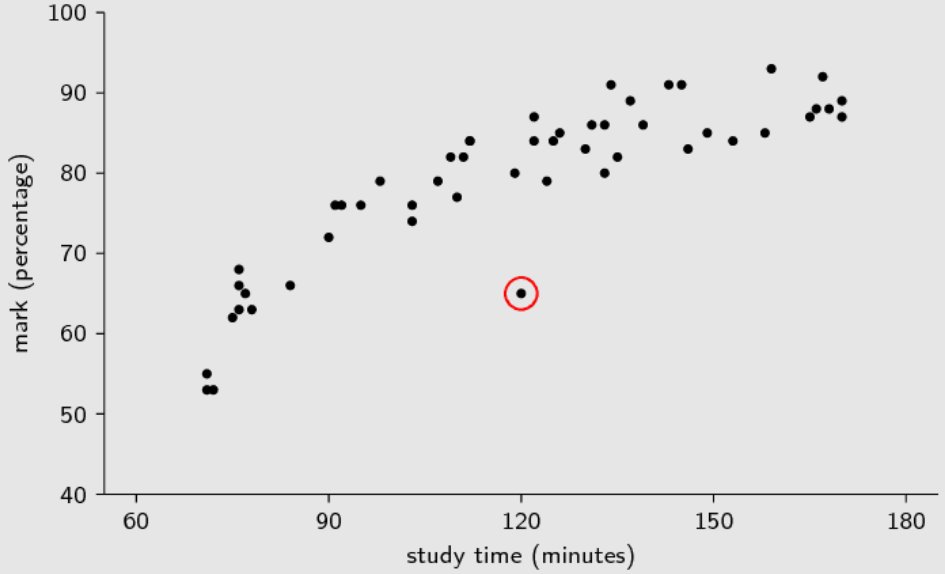
ESTATÍSTICAS DESCRITIVAS



ESTATÍSTICAS DESCRITIVAS

- **Gráficos Associados:** Boxplots são uma ferramenta comum para visualizar outliers, onde pontos fora dos "bigodes" do gráfico são tipicamente considerados outliers.
- Histogramas e scatter plots também podem ajudar a identificar valores atípicos.





ESTATÍSTICAS DESCRITIVAS

- **Medidas de Dispersão:** Medidas de dispersão quantificam a extensão pela qual um conjunto de dados se espalha em torno de um centro ou valor médio. Essas medidas incluem variância, desvio padrão e amplitude interquartil (IQR).
  - ❑ **Variância:** Representa a média das distâncias quadradas de cada ponto de dados até a média do conjunto de dados;
  - ❑ Calcula-se subtraindo a média de cada ponto de dados, elevando o resultado ao quadrado, e então calculando a média desses valores quadrados.

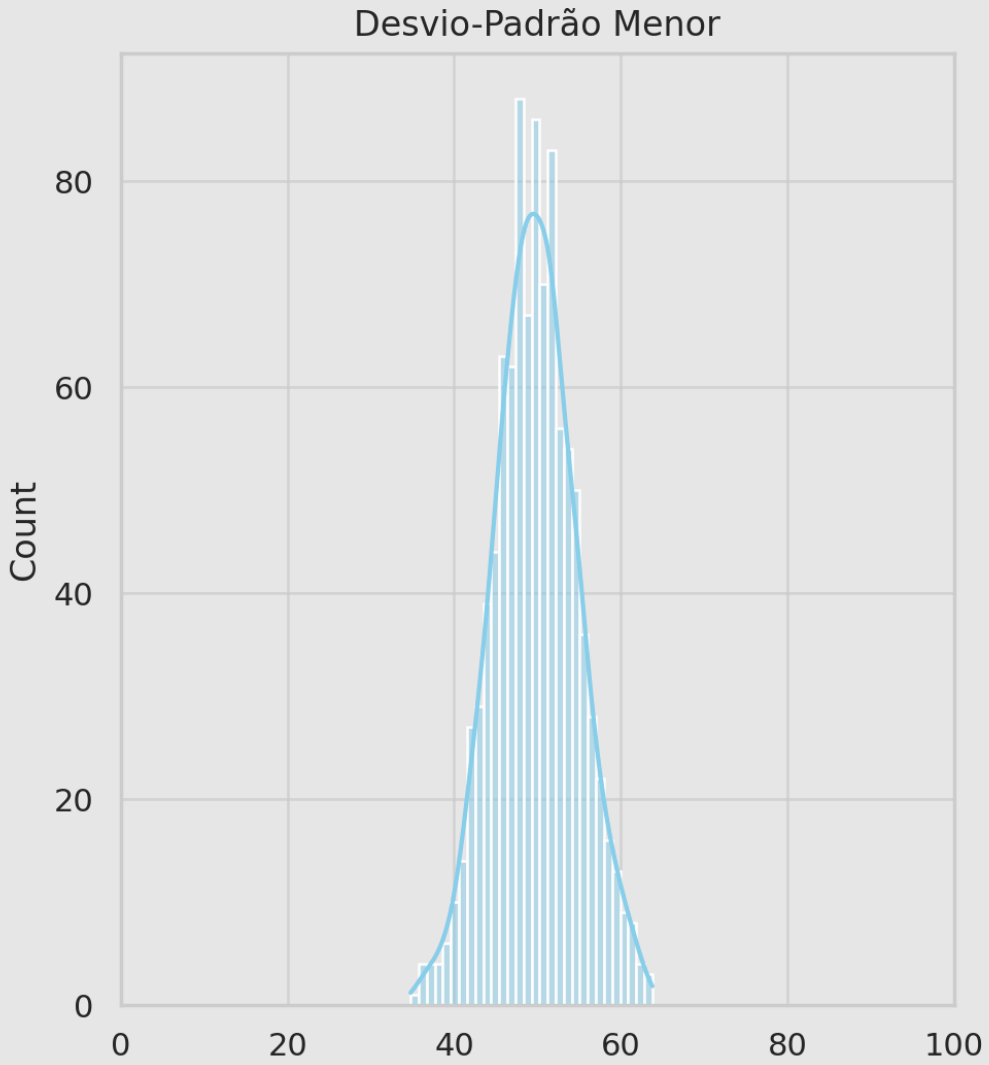
$$\sigma^2 = \frac{\sum_{i=1}^{n}(x_i - \overline{x})^2}{n-1}$$

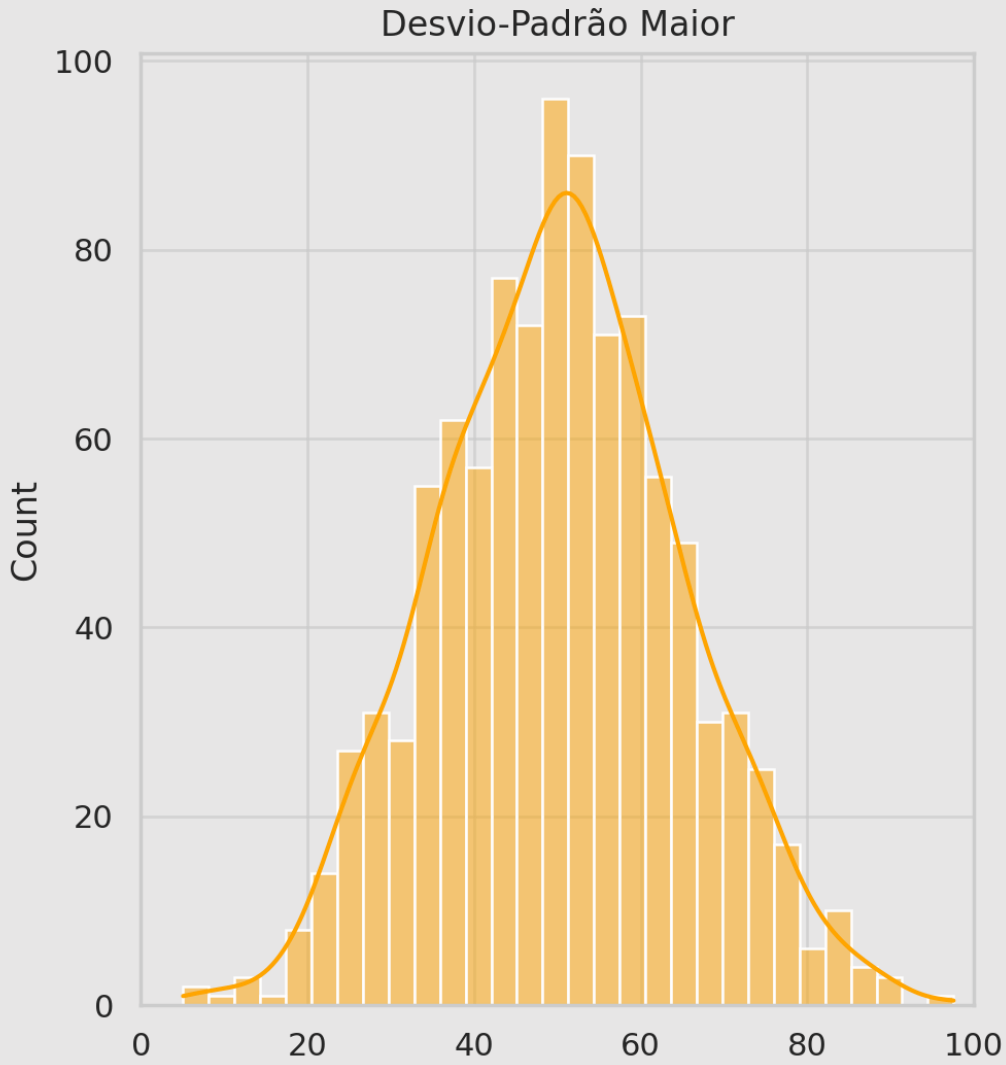
ESTATÍSTICAS DESCRITIVAS

- **Medidas de Dispersão:** Medidas de dispersão quantificam a extensão pela qual um conjunto de dados se espalha em torno de um centro ou valor médio. Essas medidas incluem variância, desvio padrão e amplitude interquartil (IQR).
  - ❑ **Desvio-padrão:** É a raiz quadrada da variância, oferecendo uma medida de dispersão que está na mesma unidade dos dados.
  - ❑ Fornece uma noção de quão "espalhados" estão os dados em relação à média.

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \overline{x})^2}{n-1}}$$

ESTATÍSTICAS DESCRITIVAS





ESTATÍSTICAS DESCRITIVAS

- **.var()**
  - A variância é o quadrado do desvio padrão e é uma medida da variabilidade ou dispersão dos dados. Ela representa o quanto, em média, cada valor no conjunto de dados se desvia do valor médio (a média);
- **.std()**
  - O desvio padrão é a raiz quadrada da variância e fornece uma medida da dispersão dos dados em relação à média. Valores de desvio padrão maiores indicam maior dispersão.

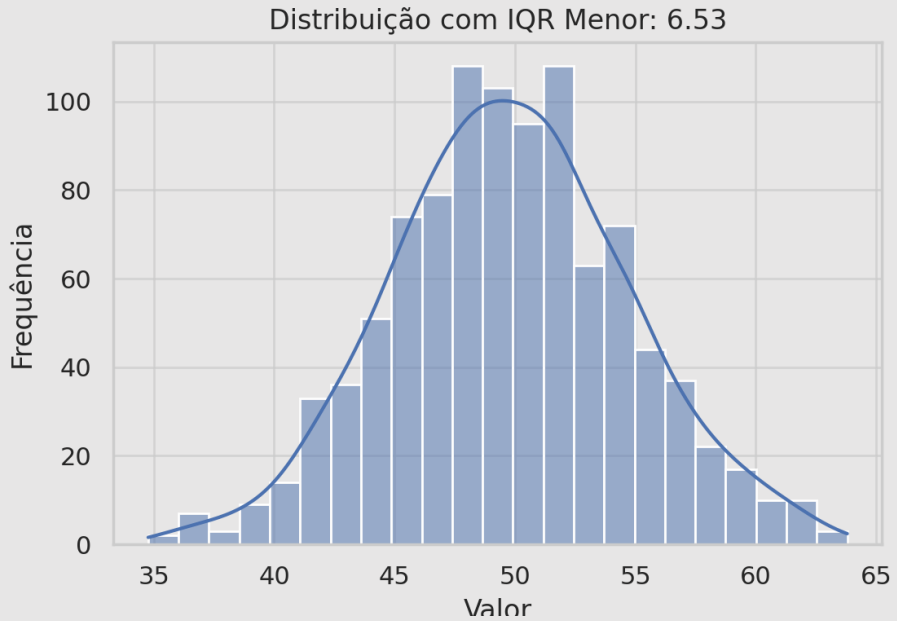
ESTATÍSTICAS DESCRITIVAS

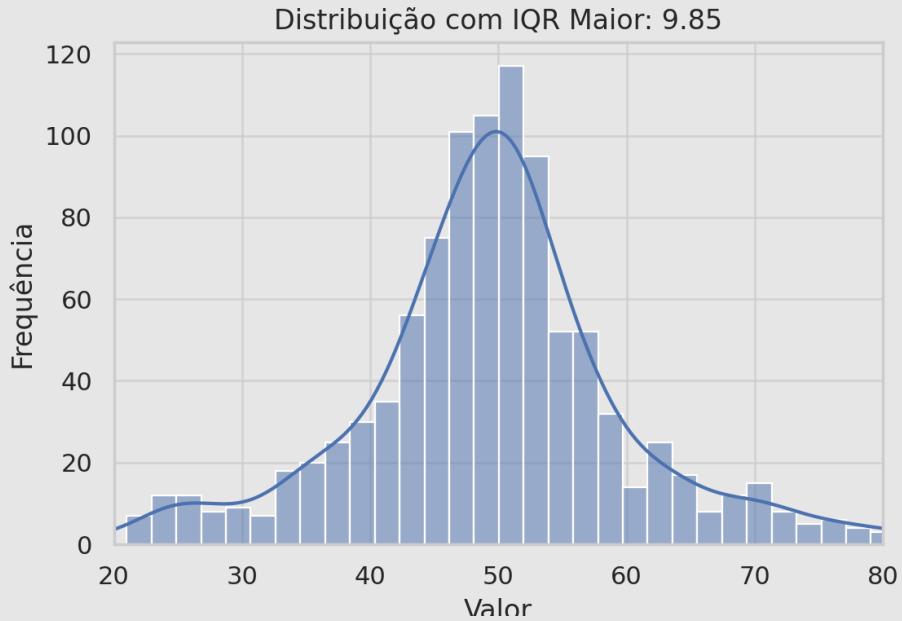
- **Medidas de Dispersão:** Medidas de dispersão quantificam a extensão pela qual um conjunto de dados se espalha em torno de um centro ou valor médio. Essas medidas incluem variância, desvio padrão e amplitude interquartil (IQR).
  - ❑ **IQR:** Mede a variação dentro do meio 50% dos dados, subtraindo o primeiro quartil (25%) do terceiro quartil (75%).
  - ❑ Útil para entender a dispersão dos dados centrais e identificar outliers.

$$\mathbf{IQR} = \mathbf{Q3} - \mathbf{Q1}$$

ESTATÍSTICAS DESCRITIVAS

- **IQR**
  - ❑ Q1 = df['col1'].quantile(0.25)
  - ❑ Q3 = df['col1'].quantile(0.75)
  - ❑ IQR = Q3 - Q1





ESTATÍSTICAS DESCRITIVAS

- **.agg(['fun1', 'fun2', 'fun3'])**
  - ❑ É uma função poderosa que permite aplicar várias funções de agregação a um DataFrame ou Series simultaneamente;
  - ❑ series.agg(['sum', 'mean', 'std'])
  - ❑ df.agg({'col1': ['mean', 'min', 'max'],
    'col2': ['std', 'var'],
    'col3': ['count', 'nunique']})
  - ❑ df['col1'].agg(['mean', 'min', 'max'])

PANDAS vs. NUMPY

Estatística	Pandas	Numpy
Média	DataFrame.mean() ou Series.mean()	numpy.mean(array)
Mediana	DataFrame.median() ou Series.median()	numpy.median(array)
Moda	DataFrame.mode() ou Series.mode()	
Variância	DataFrame.var() ou Series.var()	numpy.var(array, ddof=1)
Desvio-padrão	DataFrame.std() ou Series.std()	numpy.std(array, ddof=1)
Soma	DataFrame.sum() ou Series.sum()	numpy.sum(array)
Contagem	DataFrame.count() ou Series.count()	numpy.count_nonzero(array)
Quantis/Percentis	DataFrame.quantile(q) ou Series.quantile(q)	numpy.quantile(array, q)

PANDAS vs. NUMPY

- **Tratamento de Valores Faltantes**
  - **Pandas:** Ignora automaticamente os valores NaN ao calcular estatísticas descritivas, o que pode ser muito conveniente ao lidar com dados reais que muitas vezes têm dados faltantes.
  - **NumPy:** Não ignora NaN por padrão ao calcular estatísticas descritivas. Se o array contiver um ou mais NaN, o resultado da maioria das funções estatísticas será NaN, a menos que você trate ou remova esses valores previamente.
- **Estrutura de Dados**
  - **Pandas:** Calcula estatísticas descritivas em Series e DataFrames, que podem conter dados de tipos mistos e têm rótulos de linha e coluna, facilitando o rastreamento dos resultados com o contexto dos dados.
  - **NumPy:** Opera em arrays n-dimensionais (ndarray), que são coleções homogêneas de números e não têm rótulos integrados para linhas ou colunas.

PANDAS vs. NUMPY

- **Funcionalidades Específicas**
  - **Pandas:** Fornece um método describe() que retorna rapidamente um resumo estatístico de colunas selecionadas em um DataFrame, incluindo contagem, média, desvio padrão, mínimo, máximo e quartis.
  - **NumPy:** Possui funções para calcular estatísticas individuais, mas não um método único para gerar um resumo estatístico completo. Você precisaria calcular cada estatística separadamente.
- **Rótulos e Agrupamento**
  - **Pandas:** Tem recursos integrados para agrupamento e resumo de estatísticas (groupby), o que é muito poderoso para análise de dados.
  - **NumPy:** Não possui recursos de agrupamento inerentes, embora você possa realizar operações similares com funções adicionais ou com indexação sofisticada.

ESTATÍSTICAS DESCRITIVAS

- **.sum()**
  - Retorna a soma total de todos os valores em uma coluna.
- **.cusum()**
  - É usada para calcular a soma cumulativa de uma coluna ou Series, onde cada elemento é a soma de si mesmo com todos os elementos anteriores na sequência;
  - Pode ser visualizada como uma linha de tendência ou usado para detectar mudanças nos dados ao longo do tempo.
- **Funções derivadas**
  - .cumprod()
  - .cummax()
  - .cummin()

DADOS FALTANTES

- Em análise de dados, valores faltantes (ou NAs - Not Available) são entradas ausentes em um conjunto de dados. Eles podem ocorrer por diversos motivos, como erros de coleta, falhas no armazenamento ou ausência intencional de informações.
- No Pandas, valores faltantes são representados como NaN (Not a Number) para colunas numéricas e None para colunas categóricas.
- Para identificar e tratar esses valores, utilizamos funções específicas como .isna(), .isnull(), .dropna() e suas variações.

DADOS FALTANTES

- **df.isna() e df.isnull() ou df['col'].isna() e df['col'].isnull()**
  - ❑ Retorna uma série booleana, onde True indica que o valor está faltando naquela linha..isnull() é um sinônimo de .isna(), funcionando da mesma forma.
- **.notna() e .notnull()**
  - ❑ Retorna uma série booleana, onde True indica que o valor não está faltando naquela linha. .notnull() é um sinônimo de .notna().
- **df.isna().sum() e df['col'].isna().sum()**
  - ❑ Contar os valores faltantes em uma coluna ou em todas as colunas.
- **df.isna().any(axis=1) e df['col'].isna().any(axis=1)**
  - ❑ Retorna todas as linhas que possuem pelo menos um valor faltante. axis=1 significa que a verificação é feita por linha.

DADOS FALTANTES

- **df.dropna()**
  - ❑ Remove todas as linhas que possuem pelo menos um valor faltante.Pode ser muito restritivo, pois remove toda a linha mesmo que apenas um valor esteja ausente.
- **df.dropna(subset=['col'])**
  - ❑ Apenas as linhas onde a coluna for NaN serão removidas.
- **df.dropna(axis=1)**
  - ❑ axis=1 significa que a remoção será aplicada às colunas.
- **df.dropna(how='all')**
  - ❑ how='all' remove apenas linhas onde todas as colunas são NaN.

GRUPOS

- **.count()**
  - ❑ Na biblioteca Pandas, que é amplamente utilizada para manipulação de dados, count é usado para contar o número de elementos não nulos em um DataFrame ou Series;
- **.nunique()**
  - ❑ Retorna o número de elementos distintos em cada coluna ou linha, ou seja, o número de valores únicos. Essa função não conta os valores nulos como um valor distinto.

GRUPOS

- **.value_counts()**
  - Usada para contar a frequência de valores únicos em uma Series ou em colunas específicas de DataFrames. Ela retorna uma Series contendo as contagens de valores únicos de forma decrescente, ou seja, do valor mais frequente para o menos frequente;
  - **normalize=True:** a função retorna a proporção (ou percentual) de frequências de cada valor único, em vez de contagens absolutas (Padrão = False);
  - **sort=False:** Este argumento controla se o resultado será ordenado pelo número de contagens ou não (Padrão = True).

GRUPOS

- Calcular estatísticas descritivas entre vários grupos de um DataFrame é uma prática fundamental na análise de dados, pois fornece insights significativos sobre a distribuição, tendências e diferenças entre os grupos. Essa abordagem permite compreender melhor a variabilidade dos dados e identificar padrões ou anomalias que podem não ser evidentes à primeira vista.
  - **df[df['col1'] == 'type1'] ['col2'].fun()**
  - **df[df['col1'] == 'type2'] ['col2'].fun()**
  - **df[df['col1'] == 'type3'] ['col2'].fun()**
  - **df[df['col1'] == 'type4'] ['col2'].fun()**
  - **df[df['col1'] == 'type5'] ['col2'].fun()**

GRUPOS

- **.groupby('col1') ['col2'].fun()**
  - ❑ Quando utilizamos o groupby com uma única chave (ou coluna), estamos dividindo o DataFrame com base nos valores únicos dessa coluna. Isso resulta em grupos nos quais cada grupo consiste em todas as linhas do DataFrame que compartilham o mesmo valor para a coluna especificada;
- **.groupby(['col1', 'col3']) ['col2'].fun()**
  - ❑ O groupby também pode ser usado com múltiplas chaves, permitindo agrupar dados com base em uma combinação de valores em várias colunas. Isso é particularmente útil quando se deseja analisar subgrupos dentro dos dados que são definidos por mais de uma dimensão.

GRUPOS

- **Tabelas Dinâmicas (Pivot Tables):** É uma ferramenta que reorganiza e resume dados selecionados de um DataFrame. É utilizada para transformar ou reestruturar dados, agregando-os de forma a fornecer uma perspectiva diferente sobre os dados.
- No Pandas, as tabelas dinâmicas são criadas com o método **pivot_table**, permitindo resumir grandes conjuntos de dados e realizar análises complexas com facilidade.
- **df.pivot_table(values='col1', index='col2', aggfunc=['fun1' , 'fun2'])**
  - ❑ **values:** colunas para agregar
  - ❑ **index:** grupos para se indexar as colunas
  - ❑ **aggfunc:** lista de funções para se calcular (a média é a padrão)
  - ❑ As funções agregadas do NumPy, como np.median, np.mean, np.sum, np.max, e np.min, são comumente usadas com pivot_table para realizar vários cálculos estatísticos.

GRUPOS

- **Cross-Table:** É usada para criar uma tabela de contingência, que resume a frequência de ocorrência de variáveis categóricas em um DataFrame. Ela é útil para análises estatísticas e comparações entre categorias.
- **pd.crosstab(index= df['col1'], columns=['col2'])**
  - ❑ **index:** A variável que será usada como linhas da tabela.
  - ❑ **columns:** A variável que será usada como colunas da tabela.
  - ❑ **values:** Valores a serem agregados nas categorias (Deve-se especificar o aggfunc)
- **normalize=True**
  - ❑ **True:** Obter percentuais em relação ao total geral.
  - ❑ **index:** Agora cada linha somará 100% (útil para comparar fraudes dentro de cada categoria)
  - ❑ **columns:** Cada coluna somará 100%.

GRÁFICOS

- **Matplotlib**
  - ❑ Biblioteca de plotagem de baixo nível que oferece grande flexibilidade na criação de gráficos, mas pode requerer mais código para gráficos complexos.
- **Seaborn**
  - ❑ Biblioteca de alto nível baseada no Matplotlib, focada em estatísticas, que simplifica a criação de gráficos mais complexos e esteticamente agradáveis com menos código.
- **Principais Diferenças**
  - ❑ O Matplotlib é mais versátil e poderoso para personalização detalhada, enquanto o Seaborn é mais eficiente e fácil de usar para análises estatísticas e exploratórias de dados.

GRÁFICOS

Estatística	Matplotlib	Seaborn
Estilização e Aparência	Oferece controle preciso sobre a aparência dos gráficos, incluindo cores, tipos de linha, e marcadores. Ideal para ajustes finos e personalização detalhada.	Vem com várias paletas de cores e estilos predefinidos que tornam os gráficos visualmente atraentes por padrão. Facilita a aplicação de estilos consistentes aos gráficos com menos esforço.
Funcionalidades Estatísticas	Não inclui funcionalidades estatísticas embutidas. A análise estatística dos dados deve ser feita antes da plotagem ou integrada manualmente ao processo de criação do gráfico.	Projetado especificamente para visualizações estatísticas, incorporando automaticamente cálculos estatísticos nos gráficos, como distribuições, correlações e médias.
Uso Recomendado	Recomendado para usuários que precisam de controle total sobre a aparência dos gráficos, e para a criação de gráficos personalizados ou para publicação.	Ideal para análise exploratória de dados e para usuários que desejam criar visualizações estatísticas avançadas de maneira rápida e com mínima codificação.

GRÁFICOS

- **Gráfico de Linhas**
  - ❑ Um gráfico de linhas é um tipo de gráfico que mostra informações como uma série de pontos de dados chamados 'marcadores' conectados por segmentos de linha reta. É ideal para visualizar dados em uma sequência temporal, comparar várias séries de dados ao longo do tempo, ou destacar tendências e padrões;
  - ❑ **sns.lineplot(data=df, x="eixo_x", y="eixo_y")**
  - ❑ **plt.show()**

GRÁFICOS

- **Customizações Adicionais**
  - ❑ plt.title('Título Central', fontsize=20)
  - ❑ plt.xlabel('Rótulo do Eixo X', fontsize=15)
  - ❑ plt.ylabel('Rótulo do Eixo Y', fontsize=15)
  - ❑ plt.xticks(fontsize=12, rotation=45)
  - ❑ plt.yticks(fontsize=12, rotation=45)
  - ❑ plt.xlim([0, 5]) # Define os limites do eixo X de 0 a 5
  - ❑ plt.ylim([0, 20]) # Define os limites do eixo Y de 0 a 20
  - ❑ sns.set_style("whitegrid") # Opções incluem: "darkgrid", "whitegrid", "dark", "white", e "ticks“

GRÁFICOS

- **Histogramas**
  - Um histograma é uma representação gráfica de dados que usa barras para mostrar a frequência de valores numéricos dentro de intervalos, conhecidos como bins. É uma ferramenta fundamental para entender a distribuição dos dados e usada para analisar a forma da distribuição dos dados (por exemplo, normal, binomial, skewness), identificar outliers, e avaliar a centralidade e dispersão dos dados;
  - **sns.histplot(data=df, x="variavel_x", bins=20, kde=False, color="blue")**

GRÁFICOS

- **Gráfico de Barras**
  - ❑ O gráfico de barras é uma forma de representar dados utilizando retângulos horizontais ou verticais, onde o comprimento de cada barra é proporcional ao valor que ela representa. É utilizado para comparar quantidades de diferentes categorias, visualizar distribuições de dados categóricos, e destacar diferenças entre grupos;
  - ❑ **sns.barplot(data=df, x="variavel_x", y="variavel_y", color="yellow")**

GRÁFICOS

- **Gráfico de Dispersão**
  - ❑ Gráficos de dispersão são representações visuais de dados onde cada ponto no gráfico mostra o valor de duas variáveis. É usado para observar a relação entre essas duas variáveis. É utilizado para identificar correlações ou padrões entre duas variáveis, detectar agrupamentos ou outliers, e examinar tendências;
  - ❑ **sns.scatterplot(data=df, x="variavel_x", y="variavel_y", hue="variavel_y ")**

GRÁFICOS

- **Gráfico de Pizza**
  - ❑ Gráficos de pizza ou torta são representações circulares que são divididas em fatias para ilustrar proporções numéricas. São utilizados para mostrar comparações de partes com o todo, como a distribuição de categorias em um conjunto de dados;
  - ❑ **plt.pie(x="contagem", labels="legenda", colors=["red", "green", "blue"], startangle = 0, explode = (0,0,0.1)**

GRÁFICOS

- **Gráfico de Contagem**
  - Gráficos de contagem são utilizados para mostrar a frequência de cada categoria dentro de uma variável categórica. Eles são uma forma visual de resumir dados categóricos, semelhante aos histogramas usados para dados numéricos. São úteis para análise exploratória de dados, permitindo identificar rapidamente as categorias mais frequentes ou para comparar as frequências entre grupos;
  - **sns.countplot(data=df, x="variavel_x", hue="variavel_x ")**

GRÁFICOS

- **Boxplot**
  - Um boxplot, ou diagrama de caixa, é um método gráfico para representar a distribuição de dados numéricos através de quartis, destacando a mediana, os quartis superior e inferior, e potenciais valores discrepantes (outliers). É usado para avaliar a centralidade, dispersão, assimetria e outliers dos dados. Comparar distribuições entre diferentes grupos;
  - **sns.boxplot(data=df, x="variavel_x", y="variavel_y_opcional")**

GRÁFICOS

- **Violinplot**
  - ❑ Gráficos de violino combinam a representação de um boxplot com a densidade de probabilidade dos dados, mostrando tanto a distribuição dos dados quanto sua probabilidade de ocorrência. São ideais para visualizar a distribuição dos dados e identificar tanto a centralidade quanto a dispersão dos dados, além de possíveis outliers;
  - ❑ **sns.violinplot(data=df, x="variavel_x", y="variavel_y_opcional")**

GRÁFICOS

- **Boxenplot**
  - O gráfico de boxenplot, também conhecido como gráfico de caixa de letra (Letter-Value Plot), é uma variação do boxplot tradicional, desenhado para representar melhor a distribuição de dados com grandes conjuntos de dados. Serve para destacar diferenças entre distribuições de grupos em dados com um grande número de observações, facilitando a identificação de medianas, quartis e valores extremos;
  - **sns.boxenplot(data=df, x="variavel_x", y="variavel_y_opcional")**

GRÁFICOS

- **Mapa de Densidade**
  - ❑ O Density Map, ou mapa de calor geográfico, é um tipo de visualização usado para representar a distribuição espacial de eventos. Ele destaca áreas com maior concentração de ocorrências através de uma escala de cores, onde tons mais intensos indicam maior densidade.
  - ❑ Este gráfico é útil para identificar padrões de distribuição geográfica em grandes conjuntos de dados, como transações financeiras, crimes, atividades sísmicas, entre outros;
  - ❑ **sns.boxenplot(data=df, x="variavel_x", y="variavel_y_opcional")**

GRÁFICOS

Parâmetro	Descrição
lat e long	Nome das colunas contendo a latitude e a longitude dos dados.
z	Coluna cujos valores definem a intensidade da cor no mapa
radius	Determina o tamanho da área de influência de cada ponto. Valores maiores suavizam o mapa de calor.
center	Define o ponto central do mapa (lat=0, lon=0 centraliza no Equador).
zoom	Controla o nível de zoom do mapa (0 = visão global, 10+ = zoom em cidades).
map_stype	Define o estilo do mapa de fundo ('open-street-map', 'carto-positron', 'satellite', etc.).
title	Define o título do gráfico.