Existem várias técnicas para detectar outliers em um conjunto de dados. Algumas das mais comuns são: Z-score, regra dos 3 sigmas e a análise de dispersão.

1) **Z-score:**

O z-score é uma medida estatística que indica a quantos múltiplos do desvio-padrão um dado está distante da média. Para detectar outliers utilizando essa medida, basta calcular o z-score de cada dado e verificar se ele está muito distante da média. Se o z-score for maior do que um determinado limite, podemos considerar esse dado como um outlier.

O limite geralmente utilizado é de 3 desvios-padrões, ou seja, se o z-score de um dado for maior do que 3, podemos considerá-lo como um outlier.

import numpy as np

# Dados de exemplo

data = np.array([10, 20, 30, 40, 150, 50, 60, 70, 80, 90, 100, 350])

# Cálculo do z-score

z\_scores = (data - np.mean(data)) / np.std(data)

# Limite para considerar um dado como outlier

limite = 3

# Identificação dos outliers

outliers = data[np.abs(z\_scores) > limite]

print("Outliers encontrados:", outliers)

2) **Regra dos 3 sigmas:**

A regra dos 3 sigmas é uma técnica que utiliza a distribuição normal para identificar outliers. Ela diz que cerca de 99,7% dos dados estarão dentro de 3 desvios padrão da média. Portanto, se um dado estiver a mais de 3 desvios padrão da média, podemos considerá-lo como um outlier.

import numpy as np

# Criar um array com os dados

dados = np.array([0, 10, 12, 13, 15, 16, 18, 20, 22, 25, 30, 35, 40, 50, 350])

# Calcular a média e o desvio padrão do conjunto de dados

media = np.mean(dados)

desvio\_padrao = np.std(dados)

# Definir o limite superior e inferior para identificar os outliers

limite\_superior = media + (3 \* desvio\_padrao)

limite\_inferior = media - (3 \* desvio\_padrao)

# Identificar os outliers no conjunto de dados

outliers = dados[(dados > limite\_superior) | (dados < limite\_inferior)]

print("Outliers:", outliers)

3) **Análise de dispersão**

A análise de dispersão é uma técnica que utiliza gráficos para identificar outliers. Um gráfico comumente utilizado é o boxplot que mostra a distribuição dos dados em quartis. Os outliers são identificados como pontos fora dos limites superior e inferior do boxplot.

import matplotlib.pyplot as plt

import numpy as np

# Dados de exemplo

data = np.array([10, 20, 30, 40, 50, 60, 70, 80, 90, 100])

# Criação do boxplot

fig, ax = plt.subplots()

ax.boxplot(data)

# Identificação dos outliers

outliers = data[(data < np.percentile(data, 25) - (1.5 \* (np.percentile(data, 75) - np.percentile(data, 25)))) |

(data > np.percentile(data, 75) + (1.5 \* (np.percentile(data, 75) - np.percentile(data, 25))))]

print("Outliers encontrados:", outliers)