Aula 09: Assimetria e Curtose

O Formato da Distribuição

Até agora, aprendemos sobre:

- **Tendência Central (Média, Mediana, Moda):** Onde os dados se concentram.
- **Dispersão (Variância, Desvio Padrão):** O quão espalhados os dados estão em torno da média.

Agora, vamos entender a **forma** da curva de distribuição:

1. **Assimetria:** Indica se a distribuição é simétrica ou se há uma "cauda" mais longa para a esquerda ou para a direita.
2. **Curtose:** Indica o quão "achatada" ou "pontiaguda" é a distribuição, e a concentração de dados nas caudas e no centro.

IMPORTANTE! Assimetria trata da lateralidade da amostra; Curtose trata da cauda/concentração dos dados.

Assimetria (*Skewness*)

A Assimetria é uma medida que identifica o quanto uma distribuição tende para os valores maiores ou menores.

Utilizamos o método `skew()` do Pandas para calcular essa medida. Usaremos novamente os dados de `valor_total` dos pedidos que analisamos na aula anterior.

Interpretação da Assimetria

Valor da Assimetria	Nome da Distribuição	Relação Média vs. Mediana	Característica
Assimetria >= -0.5 e <= 0.5	Simétrica (ou Quase Simétrica)	Média == Mediana	Os dados se distribuem de forma equilibrada em torno da média.
Assimetria > 0.5	Assimétrica Positiva (à Direita)	Média > Mediana	A "cauda" da distribuição se estende para os valores maiores.

Assimetria < -0.5	Assimétrica Negativa (à Esquerda)	Média < Mediana	A "cauda" da distribuição se estende para os valores menores.

Vamos calcular a Assimetria para a coluna `valor_total` e integrá-la às medidas já calculadas:

```
# Carregar o DataFrame
pedidos_df = pd.read_csv("vendas_pedidos.csv")
dados_valor_total = pedidos_df['valor_total']

# 1. Calcular Assimetria
assimetria = dados_valor_total.skew()
print(f"Assimetria dos Valores Totais: {assimetria:.4f}")

# 2. Relembrar Média e Mediana para contextualizar
media = dados_valor_total.mean()
mediana = dados_valor_total.median()
print(f"Média: {media:.2f}")
print(f"Mediana: {mediana:.2f}")

# 3. Análise da Assimetria
if assimetria >= -0.5 and assimetria <= 0.5:
    analise_assimetria = "Simétrica (ou Quase Simétrica). Média e Mediana são próximas."
elif assimetria > 0.5:
    analise_assimetria = "Positiva. A cauda se estende para a direita (valores maiores). Média > Mediana."
else:
    analise_assimetria = "Negativa. A cauda se estende para a esquerda (valores menores). Média < Mediana."

print(f"\nConclusão da Assimetria: {analise_assimetria}")
```

O que o valor encontrado para a Assimetria em `valor_total` nos diz sobre o comportamento de vendas? Se for uma Assimetria Positiva, por exemplo, significa que a maioria dos pedidos tem um valor baixo, mas há uma cauda de pedidos de valores muito altos (que "puxam" a média para cima, ficando maior que a mediana).

Curtose (*Kurtosis*)

A Curtose é uma medida que apresenta o quão concentrados os dados de uma distribuição se aproximam da média, ou seja, ela mede o "peso" das caudas e o pico da distribuição.

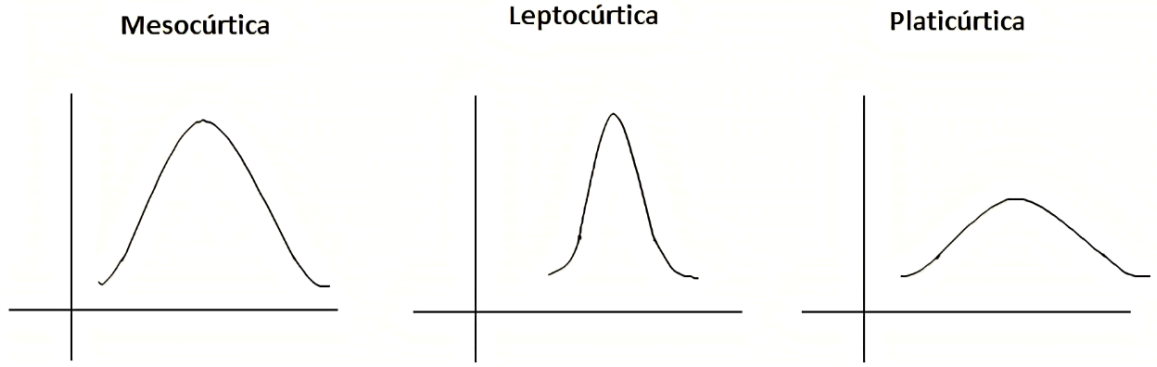
Aqui, utilizamos o método `kurtosis()` do Pandas.

Interpretação da Curtose

Ela é frequentemente comparada com a curtose da **distribuição normal**, que é igual a 3. O Pandas, por padrão, calcula a **Curtose em Excesso**, subtraindo 3 do valor real, então a referência para a normal passa a ser 0.

Para fins de análise, usaremos a referência de 3.0, onde o cálculo real é Curtose = Curtose do Pandas + 3:

Valor da Curtose	**Nome da Distribuição**	**Característica**
Curtose >= 2.5 e <= 3.5	Mesocúrtica	A distribuição é próxima da curva normal (dados uniformemente distribuídos no entorno da média). Outliers são menos comuns.
Curtose < 2.5	Platicúrtica	O pico é mais baixo e as caudas são mais finas. Os dados estão mais dispersos em relação à média. Outliers são comuns.
Curtose > 3.5	Leptocúrtica	O pico é mais alto e as caudas são mais grossas. Os dados estão extremamente concentrados no entorno da média. **Outliers são muito comuns** nas caudas grossas.



Continuando em Python:

```
# 1. Calcular Curtose
curtose_excesso = dados_valor_total.kurtosis()

# 2. Calcular a Curtose Real
curtose_real = curtose_excesso + 3
print(f"Curtose em Excesso (Pandas): {curtose_excesso:.4f}")
print(f"Curtose Real (Referência 3.0): {curtose_real:.4f}")

# 3. Análise da Curtose
if curtose_real >= 2.5 and curtose_real <= 3.5:
    analise_curtose = "Mesocúrtica. Distribuição próxima da normal (dados uniformes no entorno da média)."
elif curtose_real < 2.5:
    analise_curtose = "Platicúrtica. Dados mais dispersos em relação à média. Caudas finas e Outliers comuns."
else: # curtose_real > 3.5
    analise_curtose = "Leptocúrtica. Dados extremamente concentrados no centro e caudas pesadas. Outliers muito comuns."

print(f"\nConclusão da Curtose: {analise_curtose}")
```

A Curtose nos informa se a maioria dos pedidos se concentra muito perto do valor médio ou se há uma grande variação de valores, incluindo muitos valores extremos (outliers).

Atividade Prática

Cenário: A Diretoria de Vendas precisa de uma análise completa sobre a distribuição dos preços dos produtos no estoque para planejar a próxima campanha promocional e identificar produtos com preços fora da curva.

Você deve realizar um estudo sobre a coluna **específica** do seu arquivo escolhido em aulas anteriores.

Como exemplo de análise em vendas_produtos.csv::

- O que a **Assimetria** revela sobre a faixa de preços dos produtos? (A maioria é barata com alguns caros, ou vice-versa?)
- O que a **Curtose** e o **Boxplot** indicam sobre a concentração de preços e a presença de *outliers* (produtos com preços muito acima ou abaixo da maioria)?
- Qual seria uma sugestão de ação de marketing baseada na distribuição dos preços (ex: focar nos produtos de cauda, ou nos concentrados na média)?

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```python
# 1. Carregar os dados de produtos
produtos_df = pd.read_csv("vendas_produtos.csv")
dados_preco = produtos_df['preco']

# 2. Calcular Medidas
media_preco = dados_preco.mean()
mediana_preco = dados_preco.median()
std_preco = dados_preco.std(ddof=0) # Desvio padrão populacional

assimetria_preco = dados_preco.skew()
curtose_excesso_preco = dados_preco.kurtosis()
curtose_real_preco = curtose_excesso_preco + 3 # Para a nossa referência de 3.0

# 3. Painel de Resultados

print("-" * 50)
print("PAINEL: DISTRIBUIÇÃO DOS PREÇOS DOS PRODUTOS")
print("-" * 50)
print(f"Média do Preço: R$ {media_preco:.2f}")
print(f"Mediana do Preço: R$ {mediana_preco:.2f}")
print(f"Desvio Padrão: R$ {std_preco:.2f}")
print("-" * 50)
print(f"Assimetria: {assimetria_preco:.4f}")
# Interpretação da Assimetria
if assimetria_preco > 0.5:
    print(" -> Assimétrica Positiva (Cauda à Direita). Média > Mediana. Produtos mais caros
'puxam' a cauda.")
elif assimetria_preco < -0.5:
    print(" -> Assimétrica Negativa (Cauda à Esquerda). Média < Mediana. Produtos mais baratos
'puxam' a cauda.")
else:
    print(" -> Simétrica. Média ≈ Mediana. Distribuição equilibrada.")

print(f"Curtose (Real, Ref. 3.0): {curtose_real_preco:.4f}")
# Interpretação da Curtose
if curtose_real_preco > 3.5:
    print(" -> Leptocúrtica. Pico alto e caudas pesadas. Preços extremamente concentrados no
centro (muitos outliers de preço).")
elif curtose_real_preco < 2.5:
    print(" -> Platicúrtica. Pico baixo. Preços mais dispersos em relação à média.")
else:
    print(" -> Mesocúrtica. Próximo à Normal. Distribuição uniforme dos preços.")
print("-" * 50)

# 4. Geração do Boxplot para visualização de Outliers
plt.figure(figsize=(8, 4))
sns.boxplot(x=dados_preco)
plt.title('Boxplot da Distribuição dos Preços dos Produtos')
plt.xlabel('Preço (R$)')
```

```
plt.show()

# 5. Geração do Histograma com KDE

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

sns.histplot(dados_preco, kde=True, bins=30, color='skyblue', edgecolor='black', stat='density')


# Adicionar a Média e Mediana para referência

plt.axvline(media_preco, color='red', linestyle='--', label=f'Média: {media_preco:.2f}')

plt.axvline(mediana_preco, color='green', linestyle=':', label=f'Mediana: {mediana_preco:.2f}')


# Adicionar textos para Assimetria e Curtose (apenas para referência visual, não desenha uma linha específica)

plt.text(0.95, 0.95, f'Assimetria: {assimetria_preco:.2f}', transform=plt.gca().transAxes,

        fontsize=10, verticalalignment='top', horizontalalignment='right',

        bbox=dict(boxstyle='round,pad=0.5', fc='yellow', alpha=0.5))

plt.text(0.95, 0.88, f'Curtose (Real): {curtose_real_preco:.2f}', transform=plt.gca().transAxes,

        fontsize=10, verticalalignment='top', horizontalalignment='right',

        bbox=dict(boxstyle='round,pad=0.5', fc='lightgreen', alpha=0.5))


plt.title('Distribuição dos Preços dos Produtos (Histograma com KDE)', fontsize=14)

plt.xlabel('Preço (R$)', fontsize=12)

plt.ylabel('Densidade', fontsize=12)

plt.legend()

plt.grid(axis='y', alpha=0.75)

plt.show()
```