# Aula 08: Variância, Desvio Padrão e Coeficiente de Variação

## A Variabilidade e a Média

**Questão para reflexão:** Se a média de vendas em duas lojas é a mesma (por exemplo, R$500,00), as vendas são igualmente estáveis?

- **Loja A:** Vendas diárias de R$490, R$500, R$510.
- **Loja B:** Vendas diárias de R$100, R$500, R$900.

Em ambos os casos a média é R$500,00, mas a **Loja B** tem uma variabilidade (ou dispersão) muito maior. As **medidas de dispersão** são usadas para quantificar essa variabilidade em torno da média.

### Variância

Dado um conjunto de dados, a **Variância** é uma medida de dispersão que mostra ***o quão distante***, em média e ao quadrado, ***cada valor desse conjunto está do valor central*** (a média). Na prática:

- Quanto **menor** a variância, mais próximos os valores estão da média.
- Quanto **maior** a variância, mais os valores estão distantes da média (mais dispersos).

### Desvio Padrão

O **Desvio Padrão** é a medida estatística mais utilizada para expressar o quanto um conjunto de dados se dispersa em função da média. Ele é a **raiz quadrada da variância**. Ao tirar a raiz quadrada da variância, o Desvio Padrão retorna a dispersão para a mesma unidade de medida dos dados originais, o que facilita a interpretação. Na prática:

- Quanto **menor** o desvio padrão, mais homogêneo (uniforme) é o conjunto de dados.
- Quanto **maior** o desvio padrão, mais heterogêneo (espalhado) é o conjunto de dados.

**Exemplo:** em um conjunto de dados simples, como 1, 2, 3, 4, 5:

- **Média:** (1+2+3+4+5) / 5 = 3
- A Variância e o Desvio Padrão irão nos dizer o quão "longe" os números 1, 2, 4 e 5 estão do valor central 3.

Em Python, temos:

```
dados = [2, 4, 6, 8, 10]
```

```
# Passo 1: Calcular a média
media = sum(dados) / len(dados)
print(f"Dados: {dados}")
print(f"Média: {media}")

# Passo 2: Calcular as diferenças entre cada valor e a média
diferencas = [x - media for x in dados]
print(f"Diferenças em relação à média: {diferencas}")
```

Elevamos ao quadrado para:

1. Eliminar sinais negativos (para que desvios positivos e negativos não se anulem).
2. Dar mais peso aos valores mais distantes da média (os *outliers*).

```
# Passo 3: Elevar as diferenças ao quadrado
quadrados_diferencas = [x ** 2 for x in diferencas]
print(f"Quadrados das diferenças: {quadrados_diferencas}")

# Passo 4: Calcular a média dos quadrados das diferenças
variancia = sum(quadrados_diferencas) / len(quadrados_diferencas)
print(f"Variância: {variancia:.2f}")


# Importa a função sqrt (raiz quadrada) do módulo math, que é mais preciso que **0.5
import math

# Passo 5: Calcular a raiz quadrada da variância
desvio_padrao = math.sqrt(variancia)
print(f"Desvio Padrão: {desvio_padrao:.2f}")
```

## Aplicação com Bibliotecas

Na prática da análise de dados, usamos **NumPy** e **Pandas** para calcular essas medidas de forma mais prática:

- A função `.var()` do Pandas e `np.var()` do NumPy calculam a variância.
- A função `.std()` do Pandas e `np.std()` do NumPy calculam o desvio padrão.

**Importante:** por padrão, as funções de `pandas` (como `.var()`, `.std()`) usam **N-1** no denominador (para amostras). Para obter o resultado exato do nosso cálculo de **população** (N no denominador), usaremos `np.var()` sem parâmetros extras:

```
import pandas as pd
import numpy as np

dados_serie = pd.Series([2, 4, 6, 8, 10])

print("--- Usando Pandas e NumPy ---")
```

```
print(f"\n- Média (usando Pandas .mean()): {dados_serie.mean():.2f}")

# NumPy por padrão usa N no denominador (população)
print(f"- Variância (usando NumPy np.var()): {np.var(dados_serie):.2f}")

# Pandas por padrão usa N-1 (amostra), mas podemos forçar o cálculo da população:
# print(f"- Variância (usando Pandas .var(ddof=0)): {dados_serie.var(ddof=0):.2f}")

# NumPy por padrão usa N no denominador (população)
print(f"- Desvio Padrão (usando NumPy np.std()): {np.std(dados_serie):.2f}")

# Pandas por padrão usa N-1 (amostra)
# print(f"- Desvio Padrão (usando Pandas .std(ddof=0)): {dados_serie.std(ddof=0):.2f}")
```

## Coeficiente de Variação (CV) e Análise de Dispersão

O **Coeficiente de Variação** é uma medida de dispersão relativa, que expressa ***o desvio padrão como uma porcentagem da média***. Ele é útil para comparar a variabilidade de conjuntos de dados que têm médias muito diferentes.

### Cálculo do Coeficiente de Variação

CV = ((Desvio-Padrão) / Média) x 100%

### Cálculo da Distância da Variância em Relação à Média

Essa **distância** é uma forma de medir a ***magnitude*** da dispersão:

Distância = Variância / Média ao Quadrado

### Interpretando da Dispersão (Magnitude da Variabilidade):

| Critério | Interpretação |
|---|---|
| 0.10 | Dispersão baixa |
| 0.10 < Distância < 0.25 | Dispersão moderada dos dados em relação à média |
| 0.25 | Dispersão mais elevada |

Vamos aplicar esses conceitos aos resultados dos nossos dados iniciais ($[2, 4, 6, 8, 10]$), onde $\text{Média} = 6.0$, $\text{Variância} = 8.0$ e $\text{Desvio Padrão} \approx 2.83$:

```
# Coeficiente de Variação (CV)
cv = (desvio_padrao / media) * 100
print(f"Coeficiente de Variação (CV): {cv:.2f}%")

# Distância da Variância em relação à Média
distancia = variancia / (media ** 2)
print(f"Distância da Variância em relação à Média: {distancia:.2f}")

# Análise de Dispersão
if distancia <= 0.10:
    analise = "Baixa dispersão dos dados em relação à média."
elif distancia < 0.25:
    analise = "Dispersão moderada dos dados em relação à média."
else:
    analise = "Alta dispersão dos dados em relação à média."

print(f"Análise de Dispersão: {analise}")
```

## Atividade Prática: Análise de Dados de Vendas

Agora, vamos aplicar todos os conceitos aprendidos utilizando o arquivo de vendas (`vendas_pedidos.csv`) aplicado em aulas anteriores.

***Cenário:*** *queremos entender a variabilidade dos* ***Valores Totais dos Pedidos*** *(`valor_total`) para avaliar o quão estáveis são os preços dos produtos vendidos.*

```
# Carregar o DataFrame
pedidos_df = pd.read_csv("vendas_pedidos.csv")

# Selecionar os dados de interesse
dados_valor_total = pedidos_df['valor_total']

# Cálculo das medidas
media_vendas = dados_valor_total.mean()
# Usando np.var com ddof=0 para calcular a variância populacional
variancia_vendas = np.var(dados_valor_total, ddof=0)
# Usando np.std com ddof=0 para calcular o desvio padrão populacional
desvio_padrao_vendas = np.std(dados_valor_total, ddof=0)

# Coeficiente de Variação (CV)
cv_vendas = (desvio_padrao_vendas / media_vendas) * 100

# Distância da Variância em relação à Média
distancia_vendas = variancia_vendas / (media_vendas ** 2)
```

```python
print(f"--- Análise dos Valores Totais dos Pedidos ---")
print(f"Média dos Valores Totais: R$ {media_vendas:.2f}")
print(f"Variância dos Valores Totais: {variancia_vendas:.2f}")
print(f"Desvio Padrão dos Valores Totais: R$ {desvio_padrao_vendas:.2f}")
print("-" * 40)
print(f"Coeficiente de Variação (CV): {cv_vendas:.2f}%")
print(f"Distância da Variância / Média²: {distancia_vendas:.2f}")

# Análise de Dispersão
if distancia_vendas <= 0.10:
    analise_vendas = "Baixa dispersão dos dados em relação à média."
elif distancia_vendas < 0.25:
    analise_vendas = "Dispersão moderada dos dados em relação à média."
else:
    analise_vendas = "Alta dispersão dos dados em relação à média."

print(f"Conclusão da Dispersão: {analise_vendas}")
```