

# Resumo de Estatística Descritiva

Otacílio de Araújo Ramos Neto

IFPB

Maio 5, 2022

# Tópicos

- 1 Descrevendo um conjunto de dados
- 2 Dispersão
- 3 Correlação
- 4 Referências

# Descrevendo um conjunto de dados

- A estatística é usada para sintetizar e comunicar o aspecto mais relevante dos dados (GRUS, 2021).

# Tendências centrais

## Média

- Soma dos dados dividida pela sua contagem.  $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$
- A média depende do valor de todos os elementos da lista e, por isso, se move quando qualquer um deles aumenta ou diminui.

## Exemplo de implementação:

```
from typing import List

def media(xs: List[float]) -> float:
    return sum(xs) / len(xs)
```

# Tendências centrais

## Mediana

- É o número central em uma lista de tamanho ímpar em um conjunto de dados ordenado ou a média dos dois valores do meio em uma lista de tamanho par;
- A mediana não depende do valor de todos os elementos (apenas dos centrais), mas depende do número de elementos.

# Mediana

## Exemplo de implementação:

```
from typing import List

def _med_impar(xs: List[float]) -> float:
    return sorted(xs)[len(xs) // 2]

def _med_par(xs: List[float]) -> float:
    sorted_xs = sorted(xs)
    meio_hi = len(xs) // 2
    return (sorted_xs[meio_hi-1] + sorted_xs[meio_hi]) / 2

def mediana(v: List[float]) -> float:
    return _med_par(v) if len(v) % 2 == 0 else _med_impar(v)
```

# Tendências centrais

## Quantil

- É uma generalização da mediana;
- Valor que separa uma determinada porcentagem dos elementos;
- A mediana é um quantil de 50%.

## Exemplo de implementação:

```
from typing import List

def quantil(xs: List[float], p:float) -> float:
    p_index = int(p * len(xs))
    return sorted(xs)[p_index]
```

# Tendências centrais

## Moda

- Valores que ocorrem com mais frequência;
- Não é tão utilizado.

## Exemplo de implementação:

```
from typing import List

def moda(x: List[float]) -> List[float]:
    counts = Counter(x)
    max_count = max(counts.values())
    return [x_i for x_i, count in counts.items()
            if count == max_count]
```



# Tópicos

- 1 Descrevendo um conjunto de dados
- 2 Dispersão**
- 3 Correlação
- 4 Referências

# Dispersão

## Características da dispersão

- A *dispersão* expressa a medida da distribuição dos dados (GRUS, 2021);
- Valores próximos de zero indicam que os dados *não estão espalhados*;
- Valores distantes de zero indicam que os dados *estão muito espalhados*;
- A *amplitude* é um exemplo simples de medida de dispersão.

## Exemplo de implementação:

```
from typing import List

def amplitude(xs: List[float]) -> float:
    return max(xs) - min(xs)
```

# Dispersão

## Variância

- Uma medida da dispersão que indica “quão longe” os valores estão do valor esperado

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

# Variação

## Exemplo de implementação:

```
from typing import List
Vector = List[float]

def dot(v: Vector, w: Vector) -> float:
    """Calcula o produto escalar dos vetores"""
    assert len(v) == len(w)
    return sum(v_i * w_i for v_i, w_i in zip(v, w))

def sum_of_squares(v: Vector) -> float:
    """Retorna v_1*v_1 + ... v_n*v_n"""
    return dot(v, v)

def de_media(xs: List[float]) -> List[float]:
    x_bar = mean(xs)
    return [x - x_bar for x in xs]

def variancia(xs: List[float]) -> float:
    assert len(xs) >= 2
    n = len(xs)
    desvios = de_media(xs)
    return sum_of_squares(desvios) / (n - 1)
```

# Dispersão

## Desvio padrão

- Uma medida da dispersão que indica “quão longe” os valores estão do valor esperado sem o quadrado da unidade,

- $s = \sqrt{\text{var}(x)} = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x - \bar{x})^2}$

# Desvio padrão

## Exemplo de implementação:

```
import math
from typing import List

def desvio_padrao(xs: List[float])>float:
    """Desvio padrão é a raiz quadrada da variância"""

    return math.sqrt(variancia(xs))
```

# Tópicos

- 1 Descrevendo um conjunto de dados
- 2 Dispersão
- 3 Correlação**
- 4 Referências

# Correlação

## Covariância

- É um tipo de variância aplicada a pares;
- Se a variância mede o desvio de uma variável da média, a covariância mede a variação simultânea de duas variáveis e relação às suas médias.

## Exemplo de implementação:

```
from variancia import dot
from typing import List

def covariancia(xs: List[float], ys: List[float]) -> float:
    assert len(xs) == len(ys), "Tamanhos de xs e ys precisam"\
                                "ser iguais"

    return dot(de_media(xs), de_media(ys)) / (len(xs)-1)
```



# Correlação

## Covariância

- Covariância positiva alta indica que  $x$  tende a ser alto com  $y$  alto e baixo quando  $y$  baixo;
- Covariância negativa alta indica que  $x$  tende a ser alto com  $y$  baixo e baixo quando  $y$  alto;
- Covariância próxima de zero indica que não existe correlação (GRUS, 2021).

# Correlação

## Sobre a *correlação*

- Diferente da covariância, não tem unidade;
- Fica sempre entre -1 (anticorrelação perfeito) e 1 (correlação perfeita).

# Correlação

## Exemplo de implementação:

```
from typing import List
from desvio import *
from covariancia import *

def correlacao(xs: List[float], ys: List[float]) -> float:
    """Mede a variação simultânea de xs e ys a partir
    de duas médias"""

    stdev_x = desvio_padrao(xs)
    stdev_y = desvio_padrao(ys)

    if(stdev_x>0 and stdev_y>0):
        return covariancia(xs, ys) / stdev_x / stdev_y
    else:
        return 0
```


# Tópicos

- 1 Descrevendo um conjunto de dados
- 2 Dispersão
- 3 Correlação
- 4 Referências**

# Livro

Todo o material apresentado nesta apresentação foi obtido do livro  
“Data Science do Zero” (GRUS, 2021).



 GRUS, J. *Data Science do Zero*. [s.n.], 2021. ISBN 9788550803876. Disponível em: [⟨https://books.google.com.br/books?id=2LZwDwAAQBAJ⟩](https://books.google.com.br/books?id=2LZwDwAAQBAJ).