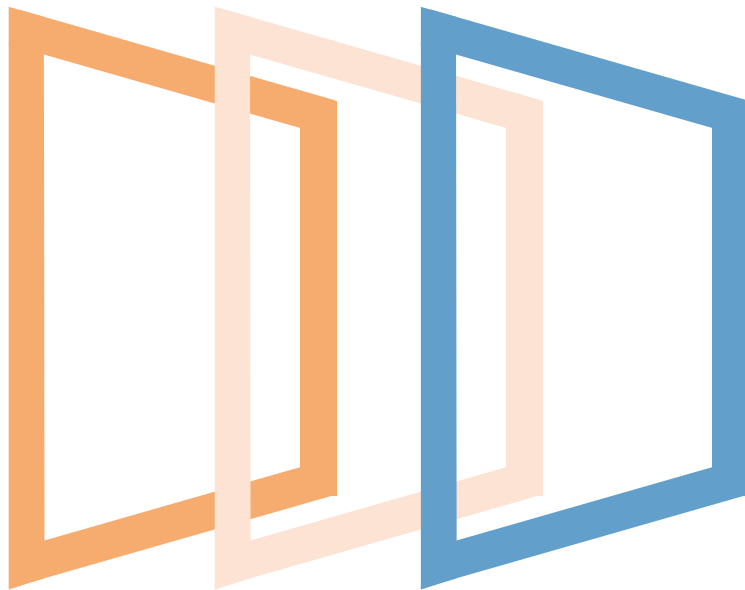


# Trilhando Caminhos em Ciência de Dados

Thaís Ratis

João Pessoa, 26.08.2023

minsoit



An Indra company

# Conteúdo programático

1. Crescimento dos Dados
2. Tipos de dados
3. Pré-processamento
4. Estatística

# Crescimento dos dados

01

# Crescimento dos dados

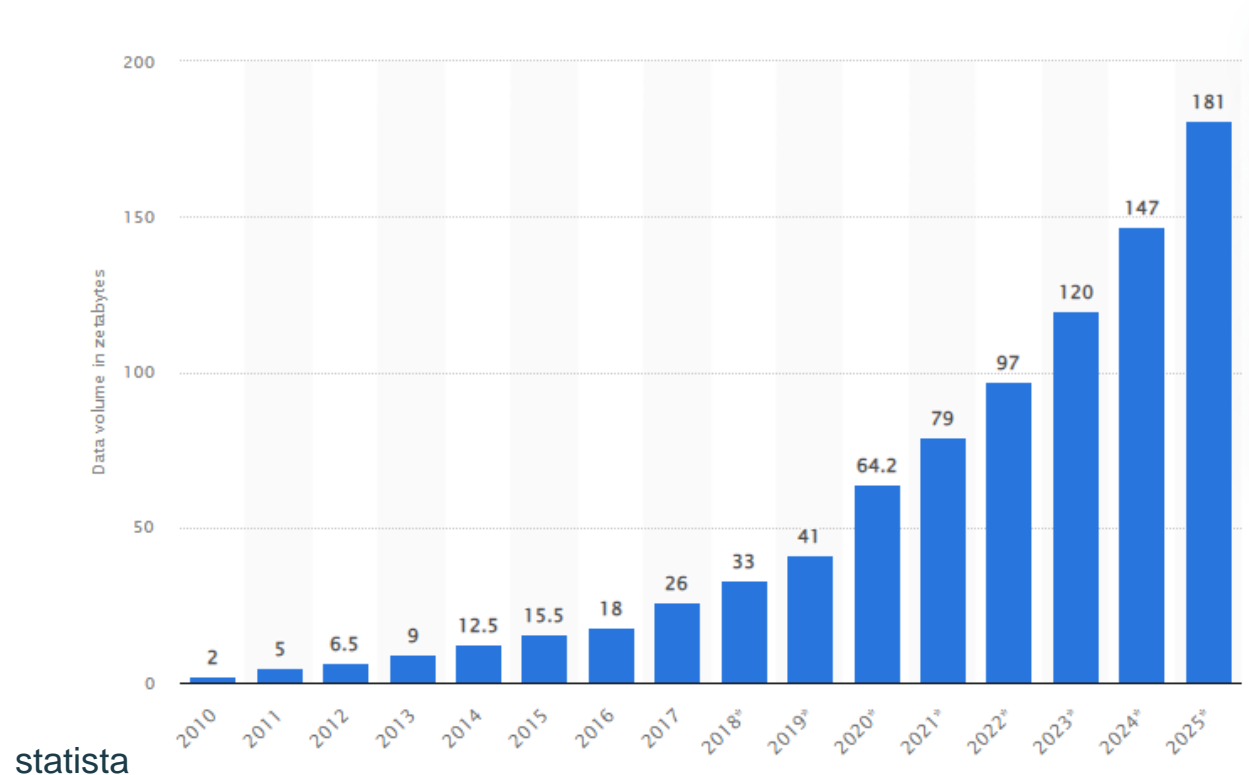


Fonte: Ouse tecnologia web

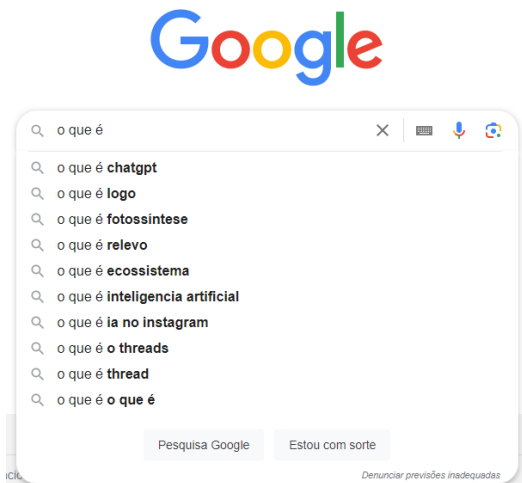
A quantidade de informações geradas, coletadas e armazenadas está aumentando exponencialmente, moldando o mundo em que vivemos.

O crescimento de dados é impulsionado por diversos fatores, como o aumento da conectividade, o avanço da tecnologia, a proliferação de dispositivos inteligentes e a digitalização de processos. Desde transações financeiras e registros médicos até interações nas redes sociais e dados coletados por sensores, cada vez mais aspectos da nossa vida cotidiana são registrados e transformados em dados.

# Volume de dados/informações criados, capturados, copiados e consumidos mundialmente de 2010 a 2020, com previsões de 2021 a 2025



# Captura de dados



# Áreas impactadas



# Tipos de Dados

02



# Tipos de Dados

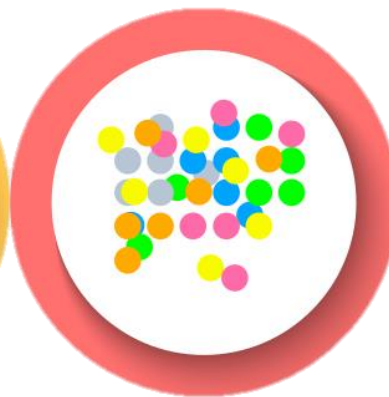
**Estruturados**



**Semi-Estruturados**

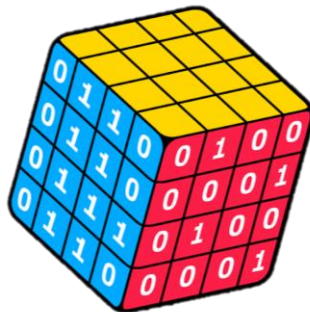


**Não Estruturados**

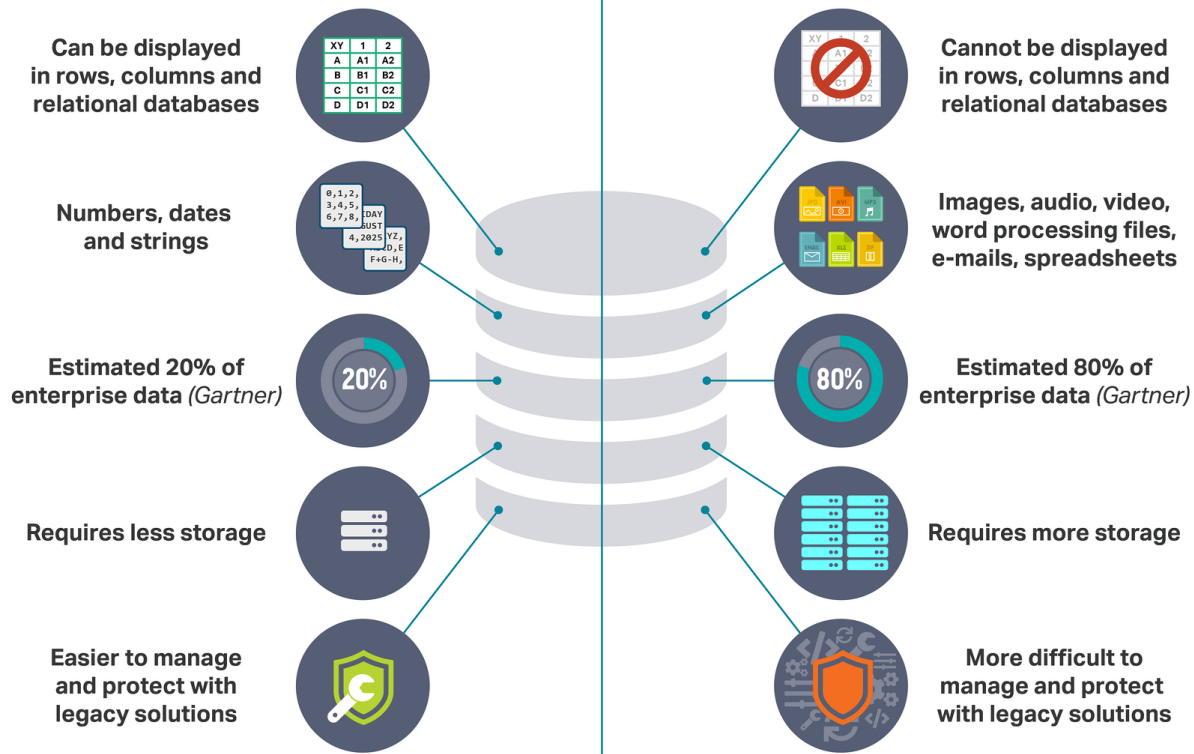


# Estruturado

Dados estruturados são aqueles organizados e representados com uma **estrutura rígida**, a qual foi previamente planejada para armazená-los. Possuem um esquema claramente definido para as informações que contêm. Simplificando a definição, quaisquer dados que possam ser apresentados em um programa de planilha, como o Planilhas Google ou Microsoft Excel, são dados estruturados.



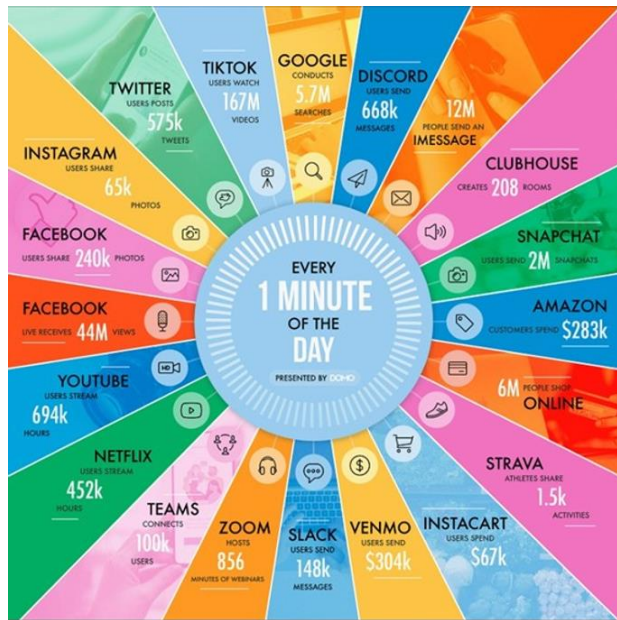
# Structured Data vs Unstructured Data



hevodata.com

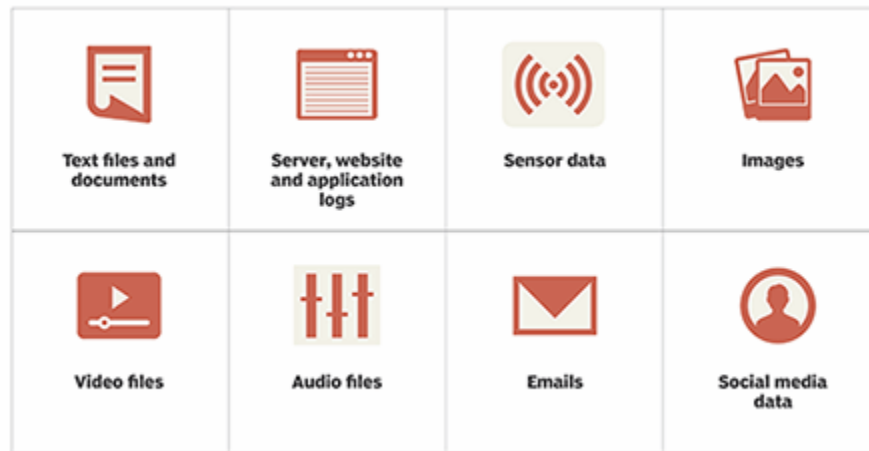
# Não Estruturado

Não possuem qualquer estrutura pré-definida, a estrutura é flexível e dinâmica ou **sem estrutura**. Constituem a maioria dos dados corporativos e são a maior parte das informações da Web (cerca de 90%). São exemplos relatórios, documentos, imagens, áudios e vídeos.



Agile Data  
Science 2.0  
(2017)

minsait



hevodata.com

An Indra company

# Semi Estruturado

Apresentam uma representação heterogênea, ou seja, possuem estrutura, mas ela é flexível. Não podemos considerar completamente desestruturados, nem fortemente tipados. São auto-descritivos possuem um *schema* de representação associado ao dado. Ex.: XML, RDF, OWL.



# Diferenças

## Dados estruturados

**Ex.:** Banco de dados

**Estrutura rígida**  
Projetada previamente  
Representação homogêna

Cada campo de dados  
tem um formato bem  
definido.

Formato é um padrão  
aceito pelo campo.

Dados de um mesmo  
registro possuem  
relação entre eles.

Registros possuem  
valores diferentes, mas  
mesmos atributos.

Atributos ou campos  
são definidos por um  
esquema.

## Dados semi estruturados

**Ex.:** XML, JSON, RDF,  
OWL.

**Estrutura flexível**  
Representação heterogêna

Cada campo de dados  
tem uma estrutura,  
mas não existe uma  
imposição de formato

O esquema é criado  
com a definição de  
elementos internos dos  
arquivos (nós), legíveis  
para seres humanos

## Dados não estruturados

**Ex.:** Textos, arquivos,  
documentos, imagens,  
vídeos, áudios, redes  
sociais etc.

**Sem estrutura**  
(ou com estrutura mínima  
de arquivo)

Mais de 80% dos  
dados gerados no  
mundo é deste tipo

# Pré-processamento

03

O que é Pré-processamento  
e qual seu objetivo?

3.1



# O que é pré-processamento?

O pré-processamento de dados é uma etapa crítica na análise de dados, que ajuda a garantir que os dados estejam limpos, padronizados e prontos para serem usados em modelos de análise. Quando realizado corretamente, o pré-processamento de dados pode melhorar significativamente a tomada de decisões informadas.

# Objetivo do pré-processamento

Uma das principais razões pelas quais o pré-processamento de dados é importante é porque os dados brutos frequentemente contêm erros e inconsistências. Por exemplo, os dados podem conter valores ausentes, informações duplicadas ou formatos de dados inconsistentes. Esses problemas podem dificultar a análise dos dados e prejudicar a precisão das conclusões que se pode tirar deles. Portanto, transformar os dados brutos em uma forma que possa ser facilmente analisada e interpretada pelos algoritmos de aprendizado de máquina ou pelos especialistas em dados. Esse processo pode melhorar significativamente a qualidade das informações disponíveis para as decisões, resultando em decisões mais precisas e informativas.

# Objetivo do pré-processamento

De forma resumida, o objetivo do pré-processamento de dados é garantir que os dados estejam prontos para serem usados em modelos de análise e para que os resultados da análise sejam confiáveis e precisos.

Quais as principais etapas  
de um pré-processamento?

3.2

# Quais as principais etapas de um pré-processamento?

**Limpeza de dados**

**Transformação de dados**

**Redução de dimensionalidade**

**Balanceamento de dados**

**Separação de dados**

# Limpeza dos dados

A limpeza de dados é um processo fundamental na análise de dados, que consiste em identificar e corrigir erros, inconsistências e valores ausentes nos dados. Esses erros podem ser causados por diversos fatores, como problemas na coleta, armazenamento ou processamento dos dados. Além disso, dados sujos e inconsistentes podem levar a conclusões equivocadas e a tomada de decisões incorretas, devido a qualidade dos dados estar ligada diretamente à eficácia de modelos de aprendizado de máquina e outras técnicas de análise de dados.

# Principais etapas no processo de limpeza dos dados

**Eliminação  
manual do  
dados**

**Identificação de  
erros e  
inconsistências**

**Remoção de  
duplicatas**

**Tratamento de  
valores  
ausentes**

**Verificação de  
integridade**

**Validação de  
dados**

# Limpeza dos dados - Eliminação manual do dados

Quando um atributo não contribui para a estimativa do valor do atributo alvo, ou seja, ele é considerado irrelevante.

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

df = pd.read_csv("hospital.csv", sep = ';')

df = df.drop(columns=['identificador', 'nome'])

df
```

	identificador	nome	idade	sexo	peso	manchas	temperatura	internacoes	estado	diagnostico
0	4201	Joao	28	M	72.0	Concentradas	38.0	2	SP	Doente
1	3217	Maria	18	F	67.0	Inexistentes	39.5	4	MG	Saudavel
2	4039	Luiz	49	M	92.0	Espalhadas	38.0	2	RS	Doente
3	1920	Jose	18	M	43.0	Inexistentes	38.5	8	MG	Saudavel
5	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
6	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
7	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
8	1322	Marta	19	F	87.0	Espalhadas	39.0	6	AM	Saudavel
9	3027	Paulo	34	M	67.0	Uniformes	38.4	2	GO	Saudavel



# Limpeza dos dados - Identificação de erros e inconsistências

Nesta etapa, os dados são analisados para identificar valores ausentes, erros de digitação, duplicatas, inconsistências e outros problemas.

	identificador	nome	idade	sexo	peso	manchas	temperatura	internacoes	estado	diagnostico
0	4201	Joao	28	M	NaN	Concentradas	38.0	2	SP	Doente
1	3217	Maria	18	F	67.0	Inexistentes	39.5	4	MG	Saudavel
2	4039	Luiz	49	M	92.0	Espalhadas	38.0	2	RS	Doente
3	1920	Jose	18	M	43.0	Inexistentes	38.5	8	MG	Saudavel
4	4340	Claudia	21	F	52.0	Uniformes	NaN	1	PE	Doente
5	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
6	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
7	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
8	1322	Marta	19	F	87.0	Espalhadas	39.0	6	AM	Saudavel
9	3027	Paulo	34	M	67.0	Uniformes	38.4	2	GO	Saudavel

# Limpeza dos dados - Remoção de duplicatas

Processo de pré-processamento de dados que envolve a identificação e exclusão de observações ou registros que são duplicados em um conjunto de dados. Em outras palavras, a remoção de duplicatas é uma técnica utilizada para garantir que cada registro ou observação em um conjunto de dados seja único. Isso é importante porque as duplicatas podem distorcer a análise de dados e prejudicar a qualidade dos resultados.

## ANTES

	identificador	nome	idade	sexo	peso	manchas	temperatura	internacoes	estado	diagnostico
0	4201	Joao	28	M	NaN	Concentradas	38.0	2	SP	Doente
1	3217	Maria	18	F	67.0	Inexistentes	39.5	4	MG	Saudavel
2	4039	Luiz	49	M	92.0	Espalhadas	38.0	2	RS	Doente
3	1920	Jose	18	M	43.0	Inexistentes	38.5	8	MG	Saudavel
4	4340	Claudia	21	F	52.0	Uniformes	NaN	1	PE	Doente
5	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
6	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
7	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
8	1322	Marta	19	F	87.0	Espalhadas	39.0	6	AM	Saudavel
9	3027	Paulo	34	M	67.0	Uniformes	38.4	2	GO	Saudavel

## DEPOIS

```
# Elimina todas as linhas com dados duplicados  
df.drop_duplicates()
```

	identificador	nome	idade	sexo	peso	manchas	temperatura	internacoes	estado	diagnostico
0	4201	Joao	28	M	NaN	Concentradas	38.0	2	SP	Doente
1	3217	Maria	18	F	67.0	Inexistentes	39.5	4	MG	Saudavel
2	4039	Luiz	49	M	92.0	Espalhadas	38.0	2	RS	Doente
3	1920	Jose	18	M	43.0	Inexistentes	38.5	8	MG	Saudavel
4	4340	Claudia	21	F	52.0	Uniformes	NaN	1	PE	Doente
5	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
8	1322	Marta	19	F	87.0	Espalhadas	39.0	6	AM	Saudavel
9	3027	Paulo	34	M	67.0	Uniformes	38.4	2	GO	Saudavel

# Limpeza dos dados - Tratamento de valores ausentes

Valores ausentes devem ser tratados de forma adequada, seja através de imputação (preenchimento com valores substitutos, por ex: média, moda ou mediana) ou remoção de linhas ou colunas inteiras.

## ANTES

	identificador	nome	idade	sexo	peso	manchas	temperatura	internacoes	estado	diagnostico
0	4201	Joao	28	M	NaN	Concentradas	38.0	2	SP	Doente
1	3217	Maria	18	F	67.0	Inexistentes	39.5	4	MG	Saudavel
2	4039	Luiz	49	M	92.0	Espalhadas	38.0	2	RS	Doente
3	1920	Jose	18	M	43.0	Inexistentes	38.5	8	MG	Saudavel
4	4340	Claudia	21	F	52.0	Uniformes	NaN	1	PE	Doente
5	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
6	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
7	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
8	1322	Marta	19	F	87.0	Espalhadas	39.0	6	AM	Saudavel
9	3027	Paulo	34	M	67.0	Uniformes	38.4	2	GO	Saudavel

## DEPOIS

```
#Substituir o valor faltante pela mediana|
median = df['peso'].median()
df['peso'].fillna(median, inplace=True)
```

df

```
Valores faltantes: identificador    0
nome                               0
idade                             0
sexo                              0
peso                              1
manchas                           0
temperatura                       1
internacoes                       0
estado                            0
diagnostico                       0
dtype: int64
```

	identificador	nome	idade	sexo	peso	manchas	temperatura	internacoes	estado	diagnostico
0	4201	Joao	28	M	72.0	Concentradas	38.0	2	SP	Doente
1	3217	Maria	18	F	67.0	Inexistentes	39.5	4	MG	Saudavel

# Limpeza dos dados - Tratamento de valores ausentes

Valores ausentes devem ser tratados de forma adequada, seja através de imputação (preenchimento com valores substitutos, por ex: média, moda ou mediana) ou remoção de linhas ou colunas inteiras.

## ANTES

	identificador	nome	idade	sexo	peso	manchas	temperatura	internacoes	estado	diagnostico
0	4201	Joao	28	M	NaN	Concentradas	38.0	2	SP	Doente
1	3217	Maria	18	F	67.0	Inexistentes	39.5	4	MG	Saudavel
2	4039	Luiz	49	M	92.0	Espalhadas	38.0	2	RS	Doente
3	1920	Jose	18	M	43.0	Inexistentes	38.5	8	MG	Saudavel
4	4340	Claudia	21	F	52.0	Uniformes	NaN	1	PE	Doente
5	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
6	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
7	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
8	1322	Marta	19	F	87.0	Espalhadas	39.0	6	AM	Saudavel
9	3027	Paulo	34	M	67.0	Uniformes	38.4	2	GO	Saudavel

## DEPOIS

```
# Elimina todas as linhas com dados ausentes  
df = df.dropna(how='any')
```

df

	identificador	nome	idade	sexo	peso	manchas	temperatura	internacoes	estado	diagnostico
1	3217	Maria	18	F	67.0	Inexistentes	39.5	4	MG	Saudavel
2	4039	Luiz	49	M	92.0	Espalhadas	38.0	2	RS	Doente
3	1920	Jose	18	M	43.0	Inexistentes	38.5	8	MG	Saudavel
5	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
6	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
7	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
8	1322	Marta	19	F	87.0	Espalhadas	39.0	6	AM	Saudavel
9	3027	Paulo	34	M	67.0	Uniformes	38.4	2	GO	Saudavel

# Limpeza dos dados - Verificação de integridade

A verificação de integridade é um processo de pré-processamento de dados que tem como objetivo garantir que os dados estejam completos e não contenham erros ou valores inconsistentes que possam comprometer a qualidade da análise ou do modelo de aprendizado de máquina construído a partir desses dados. Assim, os dados são verificados para garantir que as relações estão corretas e que não há valores discrepantes.

	identificador	nome	idade	sexo	peso	manchas	temperatura	internacoes	estado	diagnostico
0	4201	Joao	28	M	67	Concentradas	38.0	2	SP	Doente
1	3217	Maria	18	F	67	Inexistentes	39.5	4	MG	Saudavel
2	4039	Luiz	49	M	92	Espalhadas	38.0	2	RS	Doente
3	1920	Jose	18	M	43	Inexistentes	38.5	8	MG	Saudavel
4	4340	Claudia	21	F	52	Uniformes	38.5	1	PE	Doente
5	4340	Claudia	21	F	52	Uniformes	38.5	1	PE	Saudavel
6	2301	Ana	22	F	72	Inexistentes	58.0	3	RJ	Doente
7	2301	Ana	22	F	72	Inexistentes	58.0	3	RJ	Doente
8	2301	Ana	22	F	72	Inexistentes	58.0	3	RJ	Doente
9	1322	Marta	19	F	87	Espalhadas	39.0	6	AM	Saudavel
10	3027	Paulo	34	M	67	Uniformes	38.4	2	GO	Saudavel

# Limpeza dos dados - Validação de dado

A validação de dados é uma etapa importante do processo de pré-processamento de dados que tem como objetivo verificar se os dados são adequados e relevantes para a análise ou modelo de aprendizado de máquina em questão. Assim, os dados limpos são validados para garantir que atendam aos requisitos de qualidade necessários para a análise, visto que dados inadequados podem levar a resultados incorretos e conclusões inválidas.

	identificador	nome	idade	sexo	peso	manchas	temperatura	internacoes	estado	diagnostico
0	4201	Joao	28	M	72.0	Concentradas	38.0	2	SP	Doente
1	3217	Maria	18	F	67.0	Inexistentes	39.5	4	MG	Saudavel
2	4039	Luiz	49	M	92.0	Espalhadas	38.0	2	RS	Doente
3	1920	Jose	18	M	43.0	Inexistentes	38.5	8	MG	Saudavel
5	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
6	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
7	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
8	1322	Marta	19	F	87.0	Espalhadas	39.0	6	AM	Saudavel
9	3027	Paulo	34	M	67.0	Uniformes	38.4	2	GO	Saudavel



# Transformação de dados

Inclui a normalização (transformar as variáveis em uma escala comum), padronização (conversão de valores em uma forma consistente e uniforme, por exemplo, a padronização de datas em um único formato), conversão de valores e outros tipos de transformações para garantir que os dados sejam comparáveis e que o modelo de análise possa funcionar corretamente.

	name	gender	age	city		name	gender	age	city	
a	Abby	F	33	Berlin	→	a	Abby	0	33	0
b	Ben	M	16	Tokyo		b	Ben	1	16	2
c	Charlie	M	22	Sydney		c	Charlie	1	22	1
d	Dave	M	65	York		d	Dave	1	65	3
e	Ella	F	18	Sydney		e	Ella	0	18	1

# Transformação de dados

	identificador	nome	idade	sexo	peso	manchas	temperatura	internacoes	estado	diagnostico
0	4201	Joao	28	M	72.0	Concentradas	38.0	2	SP	Doente
1	3217	Maria	18	F	67.0	Inexistentes	39.5	4	MG	Saudavel
2	4039	Luiz	49	M	92.0	Espalhadas	38.0	2	RS	Doente
3	1920	Jose	18	M	43.0	Inexistentes	38.5	8	MG	Saudavel
5	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
6	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
7	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
8	1322	Marta	19	F	87.0	Espalhadas	39.0	6	AM	Saudavel
9	3027	Paulo	34	M	67.0	Uniformes	38.4	2	GO	Saudavel

```
from sklearn import preprocessing

le = preprocessing.LabelEncoder()
for column in df.columns:
    if df[column].dtypes == 'object':
        df[column] = le.fit_transform(df[column])

print(df)
```

	identificador	nome	idade	sexo	peso	manchas	temperatura	internacoes
0	4201	2	28	1	67	0	38.0	2
1	3217	5	18	0	67	2	39.5	4
2	4039	4	49	1	500	1	38.0	2
3	1920	3	18	1	43	2	38.5	8
4	4340	1	21	0	52	3	38.5	1
5	4340	1	21	0	52	3	38.5	1
6	2301	0	22	0	72	2	58.0	3
7	2301	0	22	0	72	2	58.0	3
8	2301	0	22	0	72	2	58.0	3
9	1322	6	19	0	87	1	39.0	6
10	3027	7	34	1	67	3	38.4	2



# Redução de dimensionalidade

Reduzir a dimensão dos dados, geralmente por meio de técnicas de redução de dados como a Análise de Componentes Principais (PCA) ou seleção de características, para simplificar a análise e evitar problemas de alta dimensionalidade.

	sepal length	sepal width	petal length	petal width
0	-0.900681	1.032057	-1.341272	-1.312977
1	-1.143017	-0.124958	-1.341272	-1.312977
2	-1.385353	0.337848	-1.398138	-1.312977
3	-1.506521	0.106445	-1.284407	-1.312977
4	-1.021849	1.263460	-1.341272	-1.312977

PCA  
(2 components)



	principal component 1	principal component 2
0	-2.264542	0.505704
1	-2.086426	-0.655405
2	-2.367950	-0.318477
3	-2.304197	-0.575368
4	-2.388777	0.674767

# Redução de dimensionalidade

```
from sklearn.decomposition import PCA
from sklearn.preprocessing import StandardScaler

features = ['sepal length', 'sepal width', 'petal length', 'petal width']

# Separação características
x = df.loc[:, features].values

# Separação alvo
y = df.loc[:, ['target']].values

# Normaliza as características
x = StandardScaler().fit_transform(x)

pca = PCA(n_components=2)

principalComponents = pca.fit_transform(x)

principalDf = pd.DataFrame(data = principalComponents ,
                           columns = ['principal component 1', 'principal component 2'])
```

	principal component 1	princial component 2
0	-2.264542	0.505704
1	-2.086426	-0.655405
2	-2.367950	-0.318477
3	-2.304197	-0.575368
4	-2.388777	0.674767

# Redução de dimensionalidade

```
principalDf = pd.DataFrame(data = principalComponents ,  
                           columns = ['principal component 1', 'principal component 2'])  
  
finalDf = pd.concat([principalDf, df[['target']], axis = 1)
```

principal component 1		principal component 2
0	-2.264542	0.505704
1	-2.086426	-0.655405
2	-2.367950	-0.318477
3	-2.304197	-0.575368
4	-2.388777	0.674767

principalDf

target
0 Iris-setosa
1 Iris-setosa
2 Iris-setosa
3 Iris-setosa
4 Iris-setosa

df[['target']]

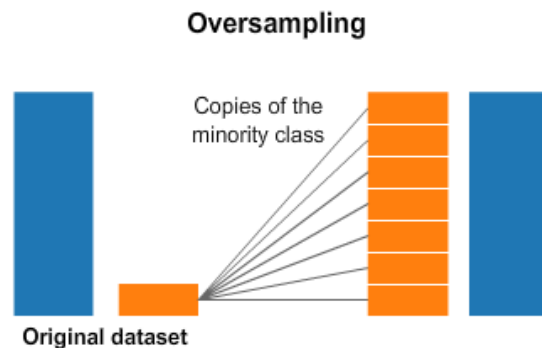
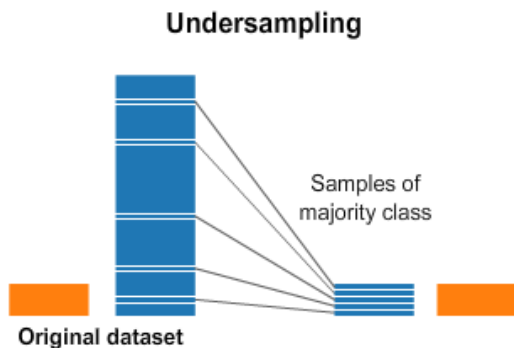
pd.concat(axis = 1)

principal component 1	principal component 2	target
0	-2.264542	0.505704 Iris-setosa
1	-2.086426	-0.655405 Iris-setosa
2	-2.367950	-0.318477 Iris-setosa
3	-2.304197	-0.575368 Iris-setosa
4	-2.388777	0.674767 Iris-setosa

finalDf

# Balanceamento dos dados

Refere-se ao processo de equilibrar a distribuição de classes nas amostras de dados. Em muitos casos, os dados podem estar desbalanceados, ou seja, uma classe pode estar sub-representada em comparação com outras. Isso pode levar a modelos de aprendizado de máquina tendenciosos, que favorecem a classe majoritária. Para evitar esse problema, é necessário equilibrar a distribuição de classes, o que pode ser feito por meio de técnicas como *oversampling*, *undersampling* e geração sintética de dados.



# Balanceamento dos dados - Exemplo

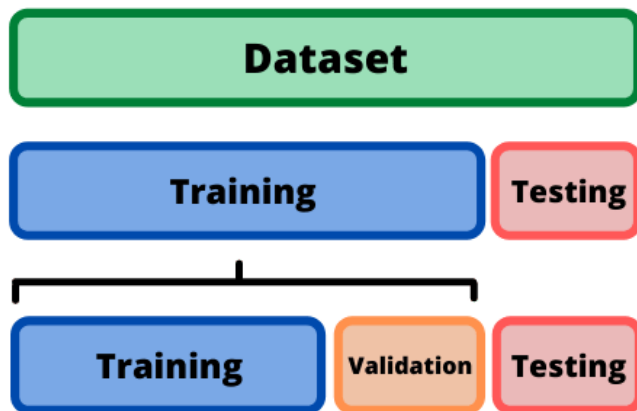
	identificador	nome	idade	sexo	peso	manchas	temperatura	internacoes	estado	diagnostico
4	4340	Claudia	21	F	52	Uniformes	37.6	1	PE	Doente
5	2301	Ana	22	F	72	Inexistentes	58.0	3	RJ	Doente
2	4039	Luiz	49	M	92	Espalhadas	38.0	2	RS	Doente
0	4201	Joao	28	M	79	Concentradas	38.0	2	SP	Doente
8	1322	Marta	19	F	87.0	Espalhadas	39.0	6	AM	Saudavel



	identificador	nome	idade	sexo	peso	manchas	temperatura	internacoes	estado	diagnostico
0	4201	Joao	28	M	72.0	Concentradas	38.0	2	SP	Doente
1	3217	Maria	18	F	67.0	Inexistentes	39.5	4	MG	Saudavel
2	4039	Luiz	49	M	92.0	Espalhadas	38.0	2	RS	Doente
3	1920	Jose	18	M	43.0	Inexistentes	38.5	8	MG	Saudavel
5	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
6	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
7	2301	Ana	22	F	72.0	Inexistentes	58.0	3	RJ	Doente
8	1322	Marta	19	F	87.0	Espalhadas	39.0	6	AM	Saudavel
9	3027	Paulo	34	M	67.0	Uniformes	38.4	2	GO	Saudavel

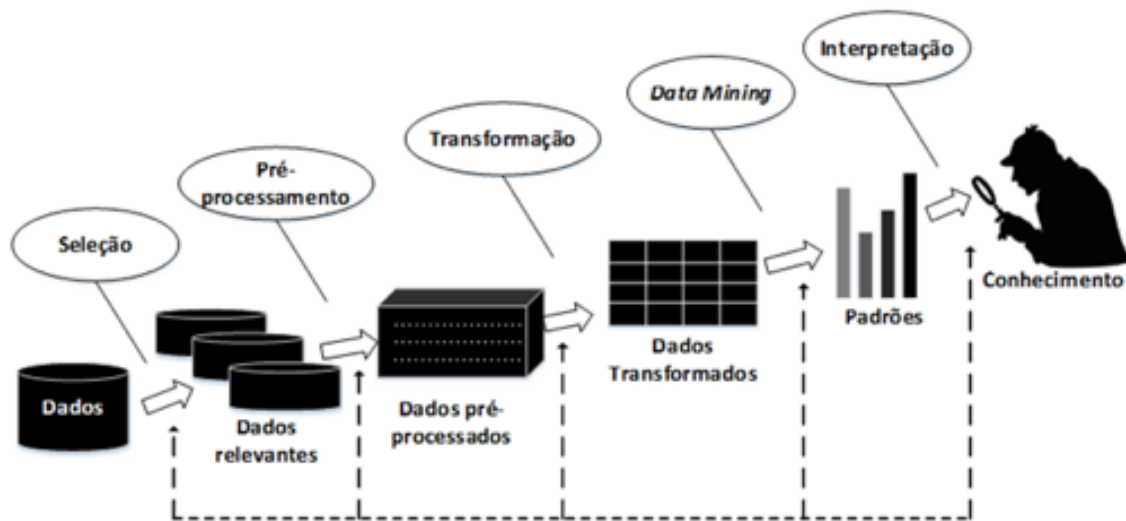
# Separação de dados

Refere-se ao processo de dividir os dados em conjuntos de treinamento, validação e teste. O conjunto de treinamento é usado para treinar o modelo, o conjunto de validação é usado para ajustar os hiperparâmetros do modelo e o conjunto de teste é usado para avaliar o desempenho do modelo em dados nunca antes vistos. É importante garantir que os dados sejam divididos de forma aleatória e estratificada, a fim de evitar a introdução de viés nos modelos.



# Etapas em processos de aprendizado de máquina

As etapas de citadas de pré-processamento e as mostradas abaixo são, geralmente, realizadas em uma ordem específica, ou seja, cada etapa é derivada da anterior.



Como o pré-processamento  
melhora as tomadas de  
decisões informadas?

3.3



# Como o pré-processamento melhora as tomadas de decisões informadas?

**Melhora a  
qualidade dos  
dados**

**Simplifica a  
análise de  
dados**

**Ajuda a  
identificar  
tendências e  
padrões**

**Reduz o tempo  
de análise**

# Como o pré-processamento melhora a qualidade dos seus dados?

O pré-processamento de dados envolve a limpeza dos dados, identificação e correção de erros e preenchimento de lacunas. Quando os dados são limpos e de alta qualidade, os insights gerados a partir da análise de dados são mais precisos e confiáveis, permitindo a tomada de decisões informadas.

# Como o pré-processamento simplifica a análise de dados?

O pré-processamento de dados pode ajudar a reduzir a dimensão dos dados e simplificar a análise. A análise de dados em conjuntos de dados de alta dimensão pode ser complicada e difícil de interpretar. A redução de dimensionalidade ajuda a simplificar a análise e torná-la mais compreensível, permitindo uma melhor tomada de decisões informadas.

# Como o pré-processamento ajuda a identificar tendências e padrões?

O pré-processamento de dados ajuda a identificar padrões e tendências nos dados, que podem ser usados para a tomada de decisões informadas. Quando os dados são limpos e transformados, padrões que antes eram difíceis de identificar podem se tornar mais evidentes, permitindo que os usuários obtenham insights importantes.

# Como o pré-processamento reduz o tempo de análise?

O pré-processamento de dados pode ajudar a reduzir o tempo de análise, permitindo que os usuários obtenham *insights* mais rapidamente. Quando os dados são limpos e preparados para análise, o tempo necessário para analisar e interpretar os dados é significativamente reduzido, permitindo que as decisões sejam tomadas rapidamente.

# Como o pré-processamento melhora as tomadas de decisões informadas?

Em resumo, o pré-processamento de dados é uma etapa crucial na análise de dados. Ajuda a garantir que os dados sejam de alta qualidade, simplifica a análise de dados, identifica tendências e padrões nos dados e reduz o tempo de análise. Realizar o pré-processamento de dados adequadamente é uma parte importante do processo de análise de dados, que pode ajudar a maximizar o valor dos dados disponíveis, levar a resultados mais precisos e obter insights valiosos para informar as decisões empresariais.

# JUPYTER NOTEBOOK + ATIVIDADE PRÁTICA

# Estatística

04



# Estatística

A estatística é um conjunto de técnicas que permite de forma sistemática organizar, descrever, analisar e interpretar dados advindos de diversas origens, a fim de extrair deles conclusões. Pode ser dividida em 4 sub-áreas:

- 1) Estatística descritiva;
- 2) Probabilidade;
- 3) Amostragem;
- 4) Estatística inferencial.

# Sub-áreas da Estatística

## 4.1

# Estatística descritiva

A Estatística Descritiva é a área da estatística que busca conhecer e sintetizar as informações a partir de um conjunto de dados quaisquer. Existem diversas ferramentas estatísticas que podem ser utilizadas na interpretação dos dados, entre elas métricas, tabelas e gráficos que irão auxiliar no entendimento e ajudar a resumir as informações deste conjunto de dados. Um ponto importante é que para que essa análise estatística seja feita de forma clara e direta, deve-se entender os **tipos de variáveis** e suas características para escolher as melhores abordagens.

# Estatística descritiva – Tipos de variáveis

As variáveis são valores, numéricos ou não, que representam características de interesse a respeito do conjunto de dados. Na Estatística Descritiva, as variáveis mais utilizadas são separadas em duas categorias, sendo elas **qualitativas** e **quantitativas**, onde, dentro destas categorias, pode-se dividir essas variáveis em dois grupos cada.



# Estatística descritiva – Tipos de variáveis

- 1) **qualitativa nominal**: as variáveis do tipo qualitativas não apresentam valores mensuráveis. No caso das variáveis **qualitativas** e **nominais**, as variáveis **não apresentam uma ordenação ou hierarquia** entre as categorias. Exemplo: Sexo, País, estado civil e etc;
- 2) **qualitativa ordinal**: Já para as variáveis **qualitativas** e **ordinais**, as variáveis **apresentam uma ordenação ou hierarquia** entre as categorias. Exemplo: escolaridade, faixa salarial, período do dia e etc;



# Estatística descritiva – Tipos de variáveis

- 1) **quantitativa discreta**: as variáveis do tipo quantitativas apresentam valores mensuráveis, e para as variáveis **quantitativas** e **discretas**, as variáveis são representadas por **quantidades enumeráveis** (isto é, que podemos contar). Exemplos: Quantidade de filhos, quantidade de TVs, número de carros que trafegam por dia em determinada rua, entre outros;
- 2) **quantitativa contínua**: as variáveis **quantitativas** e **contínuas** podem apresentar valores contínuos dentro da escala real, podendo apresentar valores fracionários, decimais, etc. Exemplo: Salário, fração de *Bitcoin*, altura e etc.



# Probabilidade

Nos permite descrever os fenômenos aleatórios, ou seja, aqueles em que está presente a incerteza.

Por exemplo, em um estudo onde o objetivo é avaliar os resultados do lançamento de um dado, ou simétrico e homogêneo, a definição do espaço amostral  $\Omega$  é dado por:  $\Omega = \{1, 2, 3, 4, 5, 6\}$ .

$$P(E) = \frac{n(E)}{n(\Omega)} = \frac{\text{nº de casos favoráveis}}{\text{nº total de casos possíveis}}$$

Onde  $E$  é o evento e  $\Omega$  o espaço amostral

# Amostragem

A amostra deve ser representativa do conjunto de dados original. Diferentes amostras de uma mesma população podem gerar modelos distintos. Os dados devem obedecer a mesma distribuição estatística que gerou o conjunto de dados original.

- 1) Amostragem aleatória simples;
- 2) Amostragem estratificada;
- 3) Amostragem progressiva.

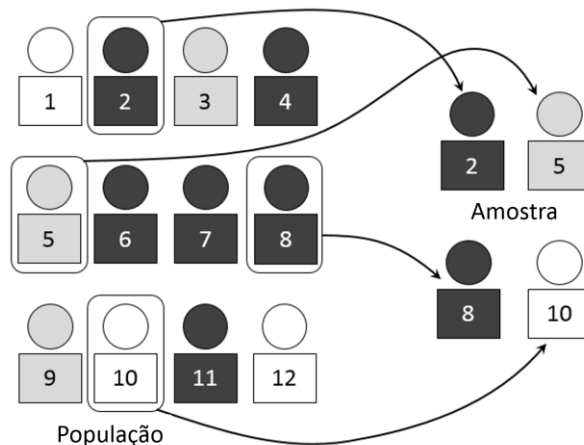


# Amostragem - Amostragem aleatória simples

Os exemplos são extraídos do conjunto original para a amostra ser utilizada.

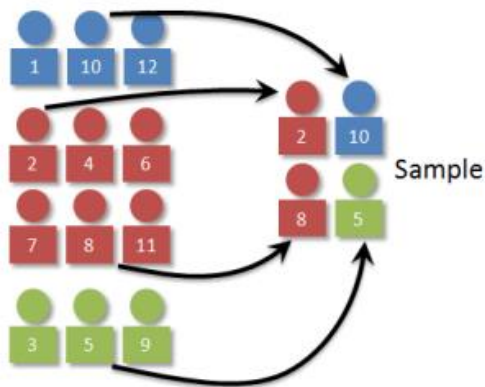
Sem reposição: cada exemplo só pode ser utilizado uma única vez.

Com reposição: a probabilidade de escolher qualquer objeto se mantém constante.



# Amostragem - Amostragem estratificada

Usada quando as classes apresentam propriedades distintas (Ex. número de objetos diferentes).  
Esta amostra, mantém o mesmo número de objetos para cada classe, proporcional ao conjunto original.



# Amostragem - Amostragem progressiva

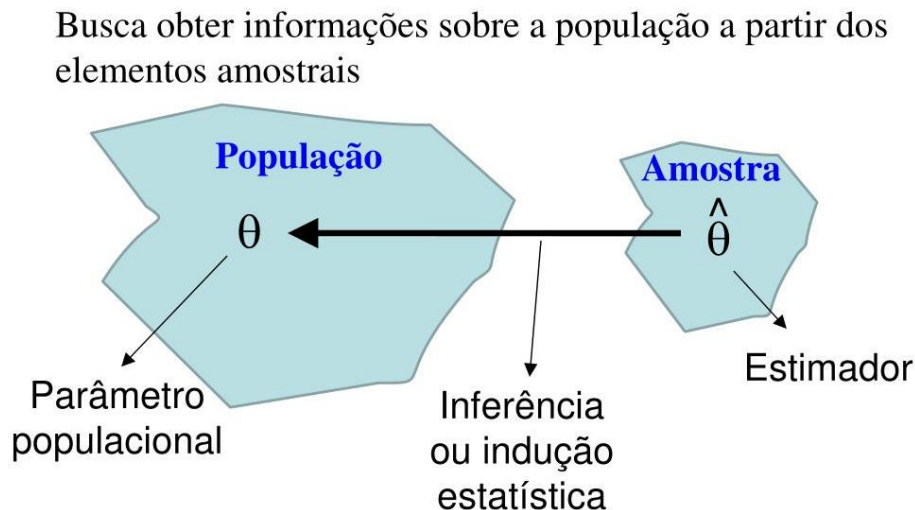
Começa com uma amostra pequena e aumenta progressivamente o tamanho da amostra extraída.

A amostra vai aumentando enquanto a acurácia preditiva continua a aumentar.

Define-se a menor quantidade de dados necessária.

# Amostragem - Estatística inferencial

É o estudo de técnicas que possibilitam a extrapolação, a um grande conjunto de dados, das informações e conclusões obtidas a partir da amostra.



# Medidas de tendência central

## 4.2

# Medidas de tendência central

Possibilitam saber o grau de concentração dos dados, uma forma de resumir os seus dados por meio de valores, representativos do conjunto de dados. Se sub dividem em 4 principais medidas:

- 1) Média;
- 2) Mediana;
- 3) Moda;
- 4) Quartis.

# Média

É a soma de todos os elementos do conjunto, divididos pelo número de elementos que compõe o conjunto, essa nós estamos acostumados, sempre usamos para auferir nossos resultados no colégio.

No	Employee ID	First Name	Last Name	Age	Worked years	Salary	Status	Grade
1	1000001	John	Denver	23	1	\$500	Single	Elementary
2	1000002	Peter	Hank	30	3	\$900	Married	High School
3	1000003	Jack	Sullivan	27	2	\$900	Married	High School
4	1000004	Marco	Aurelio	40	8	\$1,500	Married	Master Degree
5	1000005	Claudia	Perez	35	5	\$1,300	Single	Master Degree

$$\bar{x} = \frac{\sum_{i=1}^n X_i}{n}$$

Average salary =  $\frac{500 + 900 + 900 + 1,500 + 1,300}{5} = 1,020$

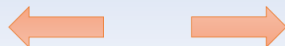
# Mediana

Métrica que indica a tendência central dos dados, representando o valor central que separa os **dados ordenados** de uma determinada distribuição. O valor da mediana varia de acordo com o número de elementos que têm na amostra, portanto pode-se definir a mediana como:

$$M = \frac{n + 1}{2}$$

Employee ID	First Name	Last Name	Age	Worked years	Salary	Status	Grade
1000001	John	Denver	23	1	\$500	Single	Elementary
1000002	Peter	Hank	30	3	\$900	Married	High School
1000003	Jack	Sullivan	27	2	\$900	Married	High School
1000004	Marco	Aurelio	40	8	\$1,500	Married	Master Degree
1000005	Claudia	Perez	35	5	\$1,300	Single	Master Degree

\$500	\$900	\$900	\$1,500	\$1,300
1	2	3	4	5
\$500	\$900	\$900	\$1,300	\$1,500



$$M = \frac{\frac{(n)}{2} + (\frac{(n)}{2} + 1)}{2}$$

CustomerID	Type	Payments	Purchases	Sales	Refunds	Country	Continent
10000	Person	Cash	120,000	150,000	240	Canada	America
10001	Company	Cash	521,400	651,750	1,043	Japón	Asia
10002	Company	Credit Card	451,000	563,750	902	Mexico	America
10003	Company	Transference	565,000	706,250	1,130	España	Europe
10004	Person	Transference	512,300	640,375	1,024	Argentina	America
10005	Person	Transference	415,500	519,375	0	Canada	America
10006	Company	Credit Card	696,300	870,375	1,392	EEUU	America
10007	Person	Cash	741,000	926,250	1,482	Chile	America

150000	651750	563750	1206250	640375	519375	1120375	926250
1	2	3	4	5	6	7	8
150000	519375	563750	640375	651750	926250	1120375	1206250

$$\text{Median} = \frac{640,375 + 651,750}{2} = 646,063$$



# Moda

Métrica de posição que indica o valor de **maior ocorrência** em um conjunto de dados. Para o caso da moda, dependendo do conjunto de dados, ele pode ser definida das seguintes formas:

- **Sem Moda/Amodal:** Todos os valores da amostra são distintos, ou seja nenhum valor se repete;
- **Unimodal:** Apenas um valores se repete com maior frequência no conjunto de dados;
- **Multimodal:** 2 ou mais valores se repetem com maior frequência no conjunto de dados.

a) 12, 18, 20, 15, 12, 19, 15, 12. >>> **Mo = 12**

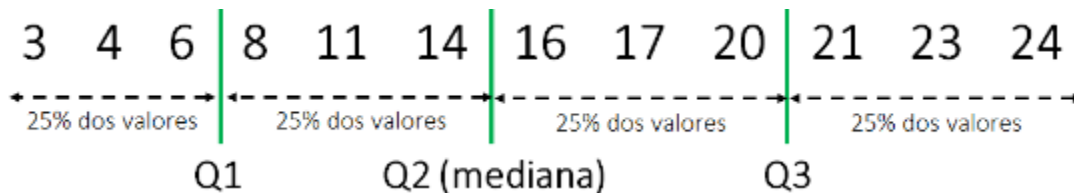
b) 15, 19, 21, 12, 15, 21, 17, 14. >>> **Mo = 15 e Mo = 21**

c) 12, 16, 13, 18, 20, 14, 25, 11 >>> amodal.

# Quartis

Os quartis são valores dados a partir do conjunto de observações ordenado em ordem crescente, que dividem os dados em quatro partes iguais. Dessa forma define-se 3 métricas:

- O primeiro quartil (Q1), sendo o número que separa 25% das observações abaixo deste valor e 75% acima;
- O segundo quartil (Q2) equivale a mediana, ou seja é o número que separa as observações em duas partes iguais (50%);
- O terceiro quartil (Q3), sendo o número que separa 75% das observações abaixo deste valor e 25% acima.



# Exemplo python

```
df = pd.read_csv('hospital1.csv', delimiter=";")
```

```
df
```

	Identificador	Nome	Idade	Sexo	Peso	Manchas	Temperatura	Internacoes	Estado	Diagnostico
0	4201	Joao	28	M	79	Concentradas	38.0	2	SP	Doente
1	3217	Maria	18	F	67	Inexistentes	39.5	4	MG	Saudavel
2	4039	Luiz	49	M	92	Espalhadas	38.0	2	RS	Doente
3	1920	Jose	18	M	43	Inexistentes	38.5	8	MG	Saudavel
4	4340	Claudia	21	F	52	Uniformes	37.6	1	PE	Doente
5	2301	Ana	22	F	72	Inexistentes	38.0	3	RJ	Doente
6	1322	Marta	19	F	87	Espalhadas	39.0	6	AM	Saudavel
7	3027	Paulo	34	M	67	Uniformes	38.4	2	GO	Saudavel

```
df.mean()
```

```
identificador    3045.875
idade            26.125
peso             69.875
temperatura      38.375
internacoes      3.500
dtype: float64
```

```
df['temperatura'].mean()
```

```
38.375
```

```
x = df['sexo'].value_counts()
x/len(df)
```

```
M    0.5
F    0.5
```

# Exemplo python

```
df = pd.read_csv('hospitall.csv', delimiter=";")  
df
```

	Identificador	Nome	Idade	Sexo	Peso	Manchas	Temperatura	Internacoes	Estado	Diagnostico
0	4201	Joao	28	M	79	Concentradas	38.0	2	SP	Doente
1	3217	Maria	18	F	67	Inexistentes	39.5	4	MG	Saudavel
2	4039	Luiz	49	M	92	Espalhadas	38.0	2	RS	Doente
3	1920	Jose	18	M	43	Inexistentes	38.5	8	MG	Saudavel
4	4340	Claudia	21	F	52	Uniformes	37.6	1	PE	Doente
5	2301	Ana	22	F	72	Inexistentes	38.0	3	RJ	Doente
6	1322	Marta	19	F	87	Espalhadas	39.0	6	AM	Saudavel
7	3027	Paulo	34	M	67	Uniformes	38.4	2	GO	Saudavel

```
df['manchas'].mode()
```

0      Inexistentes

```
df['temperatura'].median()
```

38.45

```
df['temperatura'].mean()
```

40.875

# Medidas de dispersão

# 4.3

# Medidas de dispersão

As **métricas de dispersão** são medidas de **variabilidade**, que indicam o quanto as observações de um conjunto de dados variam ao redor de alguma medida de centralidade (média, mediana, etc.). Nos permitem saber o grau de dispersão dos dados, em relação a uma medida de tendência central, geralmente a média. Algumas das principais métricas de são:

- Amplitude;
- Variância;
- Desvio Padrão.

# Amplitude

A **amplitude** identifica justamente a diferença entre o valor **máximo** e **mínimo** de uma determinada amostra aleatória, indicando o tamanho da sequência de valores possíveis que a amostra de dados possa assumir:

$$\text{Amplitude Total} = \max(n) - \min(n)$$

# Variância

A variância representa o quão distantes os dados estão em relação a média das observações, definida pela fórmula a seguir:

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

$$Variância = \frac{608000}{5 - 1} = 152000$$

$x_i$	$\bar{x}$	$x_i - \bar{x}$	$(x_i - \bar{x})^2$
Salary	Average	Difference	Square Difference
500	1,020	-520	270,400
900	1,020	-120	14,400
900	1,020	-120	14,400
1,500	1,020	480	230,400
1,300	1,020	280	78,400
		0	608,000



# Desvio Padrão

O desvio padrão, de forma análoga a variância, mede o quão distantes os dados estão em relação a média. Mas, como a variância trabalha com os valores quadráticos, no desvio padrão isto é corrigido aplicando a raiz quadrada da variância (assim, as unidades serão as mesmas dos dados).

$$\sigma(X) \equiv \sqrt{\text{Var}(X)} = \sqrt{\sum_{i=1}^n \frac{(x_i - \bar{x})^2}{n-1}}$$

$$\text{desvio} = \sqrt{152000} \approx 390$$

$x_i$	$\bar{x}$	$x_i - \bar{x}$	$(x_i - \bar{x})^2$
Salary	Average	Difference	Square Difference
500	1,020	-520	270,400
900	1,020	-120	14,400
900	1,020	-120	14,400
1,500	1,020	480	230,400
1,300	1,020	280	78,400
		0	608,000

# Exemplo Python

```
df = pd.read_csv('hospital.csv', delimiter=";")  
df
```

	Identificador	Nome	Idade	Sexo	Peso	Manchas	Temperatura	Internacoes	Estado	Diagnostico
0	4201	Joao	28	M	79	Concentradas	38.0	2	SP	Doente
1	3217	Maria	18	F	67	Inexistentes	39.5	4	MG	Saudavel
2	4039	Luiz	49	M	92	Espalhadas	38.0	2	RS	Doente
3	1920	Jose	18	M	43	Inexistentes	38.5	8	MG	Saudavel
4	4340	Claudia	21	F	52	Uniformes	37.6	1	PE	Doente
5	2301	Ana	22	F	72	Inexistentes	38.0	3	RJ	Doente
6	1322	Marta	19	F	87	Espalhadas	39.0	6	AM	Saudavel
7	3027	Paulo	34	M	67	Uniformes	38.4	2	GO	Saudavel

```
v = df["temperatura"].var() #variância do atributo "temperatura"  
d = df["temperatura"].std() #desvio padrão do atributo "temperatura"  
  
print(v)  
print(d)
```

```
48.24214285714286  
6.9456564021799165
```

# Associação de variáveis quantitativas

# 4.4

# Associação de variáveis quantitativas

Dados  $X$  e  $Y$  amostras aleatórias e suas respectivas observações  $x_1 \dots x_n$  e  $y_1 \dots y_n$ , respectivamente, pode se calcular algumas métricas de maneira análoga desenvolvido para as métricas de dispersão, mas que avalia interação entre diferentes amostras. Algumas dessas métricas são:

- Covariância;
- Correlação.

# Covariância

A Covariância seria o caso geral para a variância, onde faz-se o comparativo de quão distantes duas amostras aleatórias X e Y estão das suas respectivas médias. Mede o grau com que os atributos variam juntos.

$$S_{XY} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n - 1}$$

# Correlação

A correlação é uma métrica utilizada para avaliar a dependência entre duas variáveis, onde é possível quantificar como diferentes variáveis interagem entre si. O valor da correlação varia entre -1 e 1, ou seja, para valores mais próximos dos extremos, as variáveis apresentam maior correlação entre si e quanto mais próximo de zero a correlação, diminui cada vez mais a dependência dessas variáveis. Essa progressão entre os valores extremos da correlação podem ser representados na figura a seguir:



datadeck.com

# Exemplo Python

$$r_{xy} = \frac{Cov(X, Y)}{S_x S_y} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}}$$

```
corr = df.corr()  
corr.style.background_gradient(cmap='coolwarm')
```

	identificador	idade	peso	temperatura	internacoes
identificador	1	0.48618	0.0620799	-0.319808	-0.817134
idade	0.48618	1	0.560835	-0.191917	-0.512349
peso	0.0620799	0.560835	1	0.0593191	-0.28261
temperatura	-0.319808	-0.191917	0.0593191	1	-0.035277
internacoes	-0.817134	-0.512349	-0.28261	-0.035277	1

# JUPYTER NOTEBOOK + ATIVIDADE PRÁTICA



# Teste de hipótese

4.5

# Teste de hipótese

O Teste de Hipóteses é a ferramenta base da inferência estatística, onde pode-se tirar informações a partir de uma dada amostra e extrapolar (ou inferir) conclusões sobre a população. Uma hipótese será uma declaração sobre um parâmetro da população, ou seja para dado média, variância e proporção da população usam-se estimadores como média, variância e proporção para as amostras. Os cálculos desses estimadores são dados da seguinte forma:

$$\bar{X} = \sum_{i=1}^n \frac{X_i}{n}$$

$$S^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2$$

$$\hat{p} = \frac{1}{n} \sum_{i=1}^n I_i$$

No caso para a proporção  $p$ , o valor de  $li$  é igual a 1 se a observação tem a característica de interesse, ou seja  $li$  entre  $\{0, 1\}$ .

# Teste de hipótese

Ao aplicar um teste de hipóteses o objetivo é testar alguma informação relevante sobre um atributo da população, de maneira a aceitar ou rejeitar esta informação. Para isto, é que formulam-se as chamadas hipótese nula ( $H_0$ ) e a hipótese alternativa ( $H_a$  ou  $H_1$ ), sendo a hipótese alternativa o complemento/negação da hipótese nula. Ao testar essas hipóteses pode-se cair nos seguintes casos:

Hipóteses	$H_0$ é verdadeira	$H_0$ é falsa
Rejeitar $H_0$	Erro tipo I ( $\alpha$ )	Sem erro
Aceitar $H_0$	Sem erro	Erro tipo II ( $\beta$ )

Aplica-se teste de hipótese para diminuir a chance de ocorrer o erro do Tipo I, ou seja, costuma-se trabalhar com o nível de significância  $\alpha$  (total máximo das probabilidades de uma determinada amostra ocorrer erro tipo 1, normalmente  $\alpha$  é 0.05 ou 0.01).

# Aplicações

Uma fábrica anuncia que o índice de nicotina dos cigarros de uma dada marca é igual ou menor a 20 mg por cigarro. Um laboratório realiza 20 análises do índice obtendo: 22, 19, 21, 22, 20, 18, 27, 20, 21, 19, 20, 22, 17, 20, 21, 18, 25, 16, 20, 21. Sabe-se que o índice de nicotina dos cigarros dessa marca se distribui normalmente com variância 4 mg. Pode-se aceitar a afirmação do fabricante, ao nível de 5%?

$$H_0 : \mu \leq 20$$
$$H_1 : \mu > 20$$

```
from scipy.stats import norm
import numpy as np

# Cálculo do Z crítico para a probabilidade de 95%
# Norm.ppf () calcula o multiplicador do desvio padrão
# para um intervalo de confiança específico
prob = norm.ppf(0.95)

# Mostra o valor da probabilidade para 95%
print("Z crítico para a probabilidade para 95%: ", prob)

# Parâmetros para o teste
mu = 20          # média
sigma = np.sqrt(4) # desvio padrão
n = 20          # tamanho da amostra

# Cálculo do valor crítico
Xc = mu + (sigma/np.sqrt(n))*prob

# Mostra o valor crítico
print("Valor crítico: ", Xc, " mg")
```

Z crítico para a probabilidade para 95%: 1.6448536269514722  
Valor crítico: 20.735600904580114 mg

```
# Aplicando as observações
X = [22, 19, 21, 22, 20, 18, 27, 20, 21, 19, 20, 22, 17, 20, 21, 18, 25, 16, 20, 21]

# Calculando a média das observações
Xobs = np.mean(X)

# Mostra o valor médio das observações
print("Valor médio observado: ", Xobs, " mg")

# Verificando se o valor médio ultrapassa o valor crítico
print("Teste de Hipótese: ")
if(Xobs < Xc):
    print("Aceitamos H0")
else:
    print("Rejeitamos H0")
```

Valor médio observado: 20.45 mg  
Teste de Hipótese:  
Aceitamos H0

# JUPYTER NOTEBOOK + ATIVIDADE PRÁTICA

# Regressão

# 4.6

# Regressão

Para o funcionamento do modelo de regressão serão fornecidas informações na forma de variáveis atributos e o modelo estimará o valor da variável resposta usando dados de referência durante o treinamento. Lembrando que, para o caso da Regressão, o tipo de resposta esperado na saída será um valor contínuo representativo de acordo com a variável resposta. Especificamente quando se trata de **Regressões Lineares**, a inferência feita sobre a relação entre as variáveis é que pode ser descrita por uma **equação de reta**.

# Regressão linear

Na regressão linear simples, tem-se um conjunto de dados formado por um único atributo  $X$  e a variável resposta  $Y$ . O modelo vai procurar estabelecer a melhor equação de reta que descreva o conjunto de dados, ou seja define a equação como:

$$Y \approx \beta_0 + \beta_1 X$$



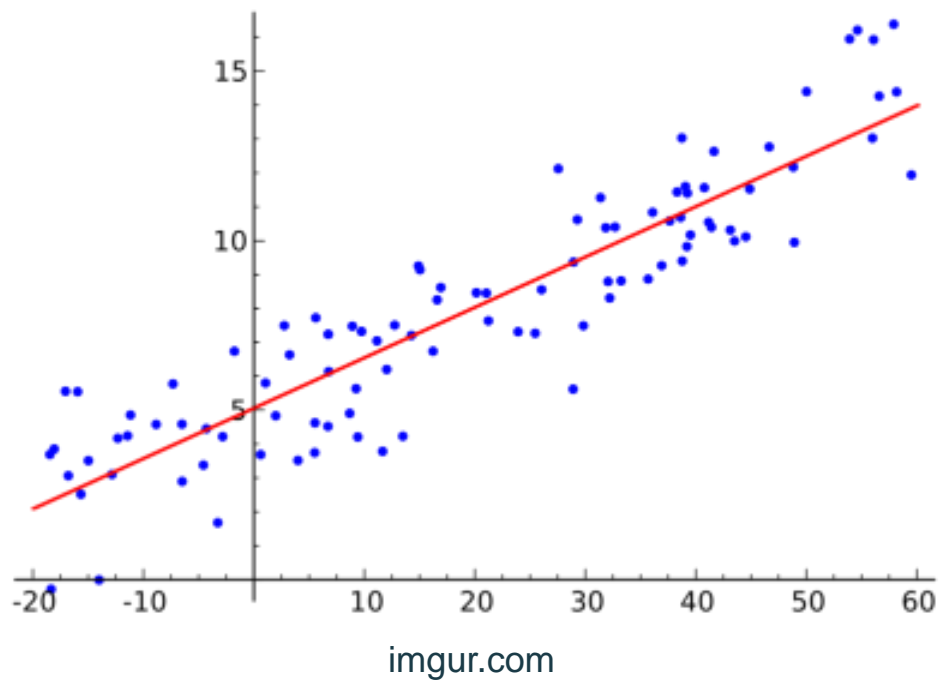
# Regressão linear

Uma forma para estimar os valores de  $\beta_0$  e  $\beta_1$  é a partir dos **valores médios** para X e Y, onde  $\beta_0$  é o **coeficiente independente** (onde a equação de reta vai cortar o eixo Y no gráfico) e o  $\beta_1$  é o **coeficiente angular** desta reta (indicando a inclinação da reta a ser ajustada). Assim os valores dos coeficientes também serão um valor médio da forma  $\beta_0$  e  $\beta_1$ , dados pelas seguintes equações:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} = \frac{\sigma_{xy}}{\sigma_{xx}} = \frac{Covar(x, y)}{Var(x)}$$
$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

- Covariância indica o grau de interdependência entre duas variáveis;
- Variância é uma medida de o quão disperso estão os dados, ou seja o quão distante está cada valor desse conjunto do valor médio.

# Regressão linear



# Implementação

```
def linear_regression(x, y):  
    # Definir as médias  
    mean_x, mean_y = np.mean(x), np.mean(y)  
  
    # Calcular a covariância e variância  
    S_xy = 0  
    S_xx = 0  
  
    # Laço para o somatório  
    for i in range(0, len(x)):  
        # termo covariância  
        S_xy += (x[i] - mean_x)*(y[i] - mean_y)  
        # termo variância  
        S_xx += (x[i] - mean_x)**2  
  
    # Calcular os coeficientes de regressão  
    beta_1 = S_xy / S_xx  
    beta_0 = mean_y - beta_1*mean_x  
  
    # Retorna os coeficientes  
    return beta_0, beta_1
```

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} = \frac{\sigma_{xy}}{\sigma_{xx}} = \frac{Covar(x, y)}{Var(x)}$$
$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

```
from sklearn.linear_model import LinearRegression  
  
# Instanciar o modelo  
model = LinearRegression()  
  
# Treinamento do Modelo  
model.fit(X_train, y_train)  
  
# Gerar novas previsões  
y_pred = model.predict(X_test)
```

# Referências

- [Dados Estruturados e Não Estruturados • Universidade da Tecnologia](#)
- [Understanding Structured Data: A Comprehensive Guide 101 \(hevodata.com\)](#)
- [Dados Estruturados x Semi x Não Estruturados - colete dados de todas as fontes parte II \(improova.com.br\)](#)
- [Inteligência Artificial - Aulas de Inteligência Artificial \(google.com\)](#)
- [PPT - Contagem e Probabilidade PowerPoint Presentation, free download - ID:5467135 \(slideserve.com\)](#)
- [Estatística\[1\] \(slideshare.net\)](#)
- [Data Analytics: Correlation vs. Causality - \(datadeck.com\)](#)
- [PCA using Python \(scikit-learn, pandas\) | Codementor](#)
- [Introdução a Estatística Prof Thiago Marques - Matemática Financeira \(passeidireto.com\)](#)

# Referências

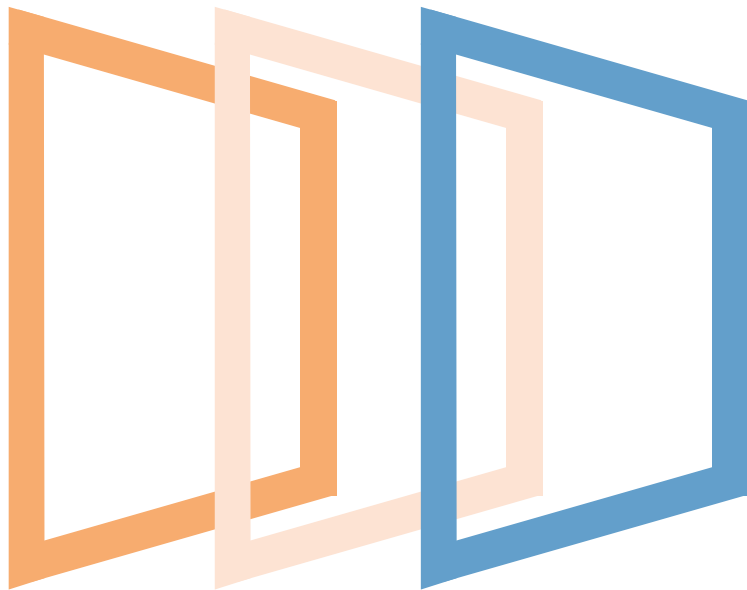
- [Data Analysis with Python \(udemy.com\)](https://www.udemy.com/course/data-analysis-with-python/)
- Pedro A. Morettin, Wilton O. Bussab, Estatística Básica, 8ª edição
- Peter Bruce, Andrew Bruce & Peter Gedeck, Practical Statistics for Data Scientists, 50+ Essential Concepts Using R and Python, 2ª edition
- Ron Larson & Betsy Farber, Estatística Aplicada, 6ª edição.
- James, Gareth, et al. An Introduction to Statistical Learning: With Applications in R. Alemanha, Springer New York, 2013;
- Bruce A., Bruce P. Estatística Prática para Cientistas de Dados. Segunda Edição, Alta books, 2019;

# Trilhando Caminhos em Ciência de Dados

Thaís Ratis

João Pessoa, 26.08.2023

minsoit



An Indra company