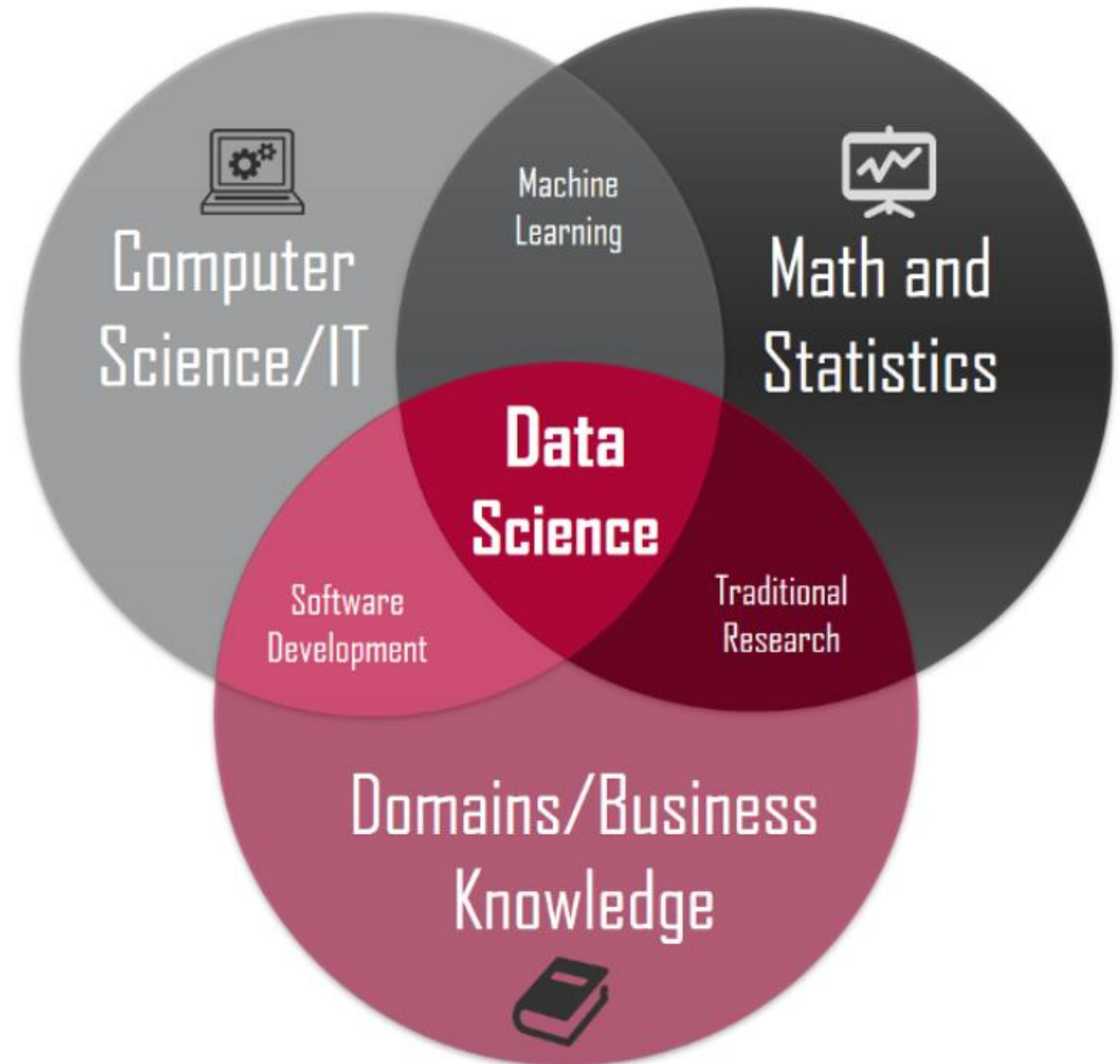


# DATA SCIENCE

- Aprendizagem de Máquina Supervisionada
- Aprendizagem de Máquina Não Supervisionada

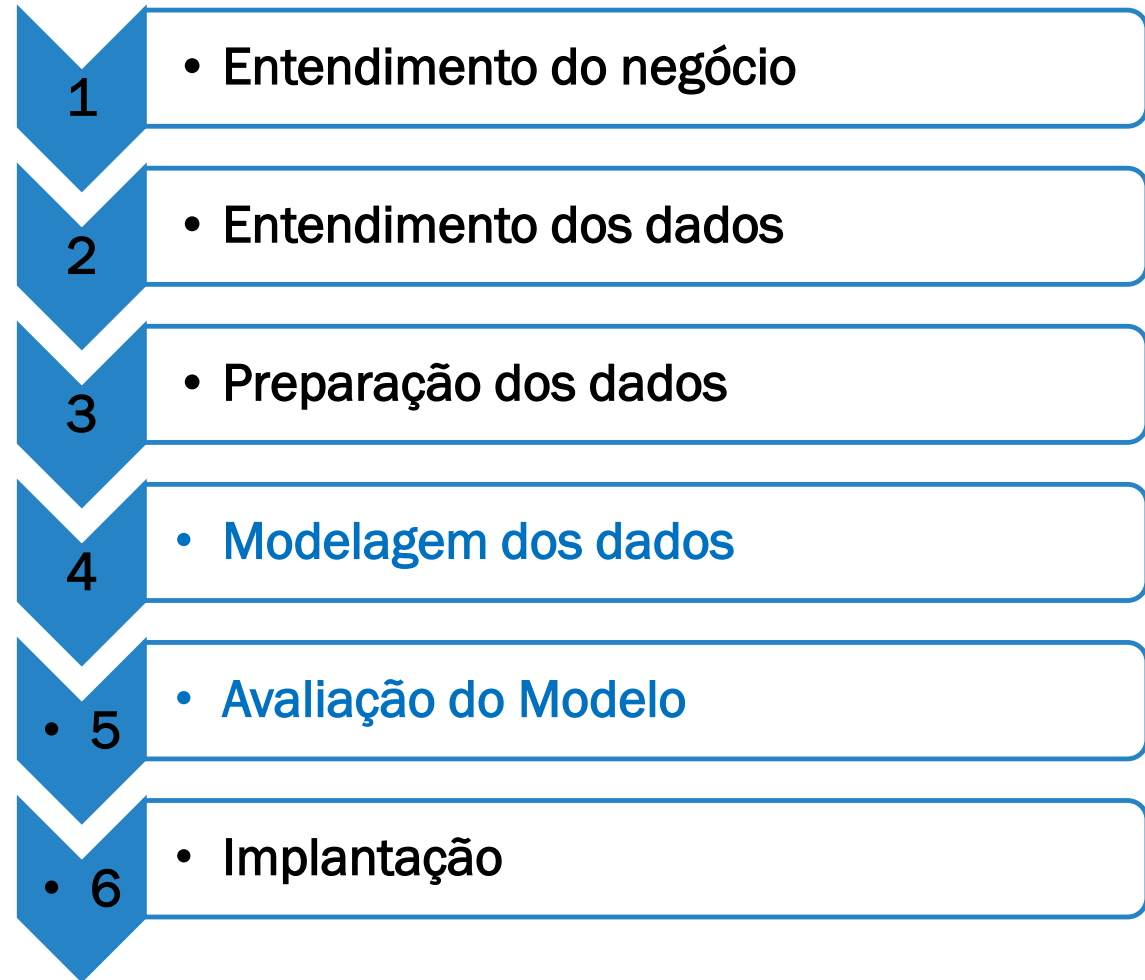
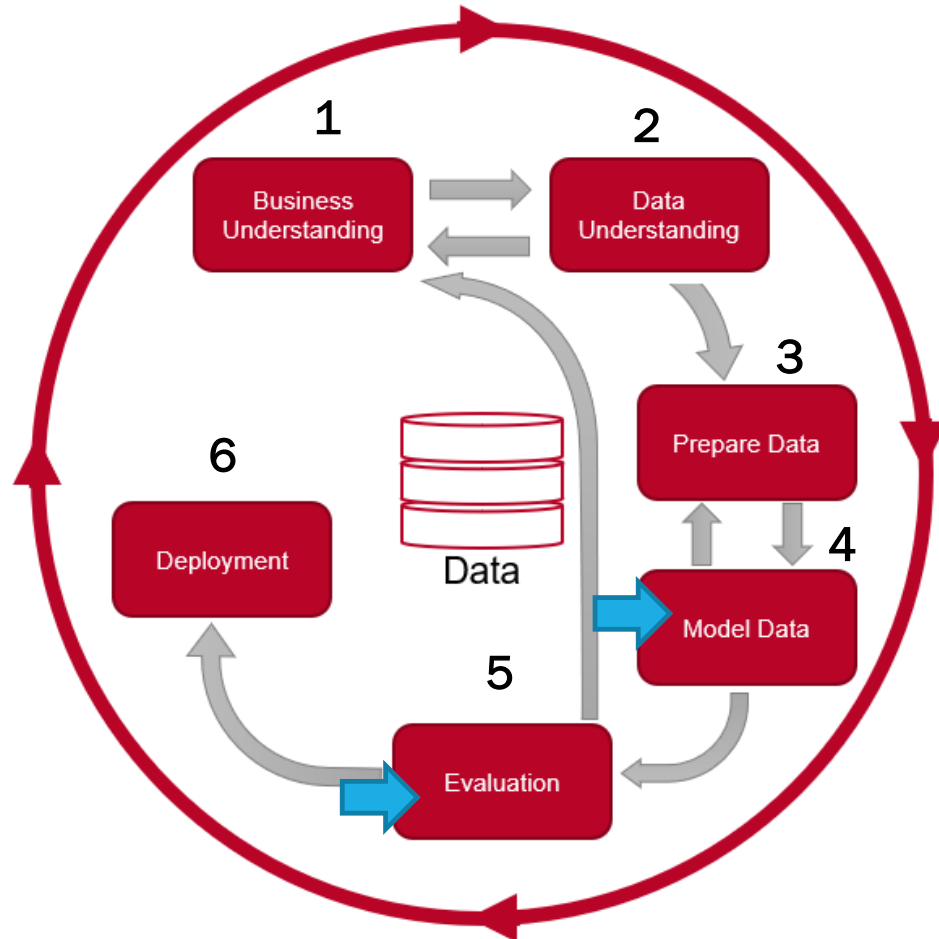
PROFESSORA:  
**CARLA OLIVEIRA SANTOS**



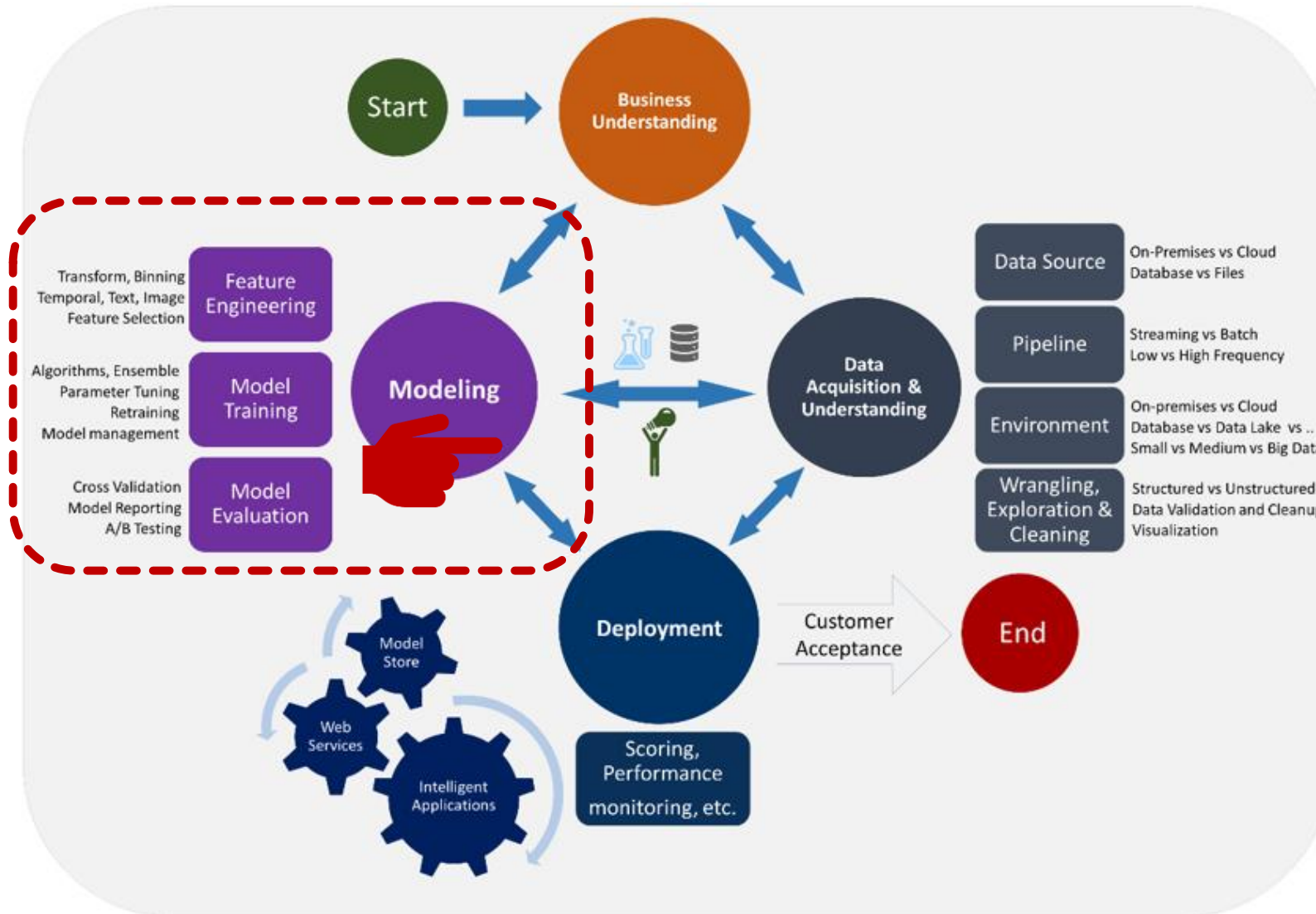
# AGENDA



# CRISP-DM: A METODOLOGIA IDEAL PARA CIÊNCIA DE DADOS

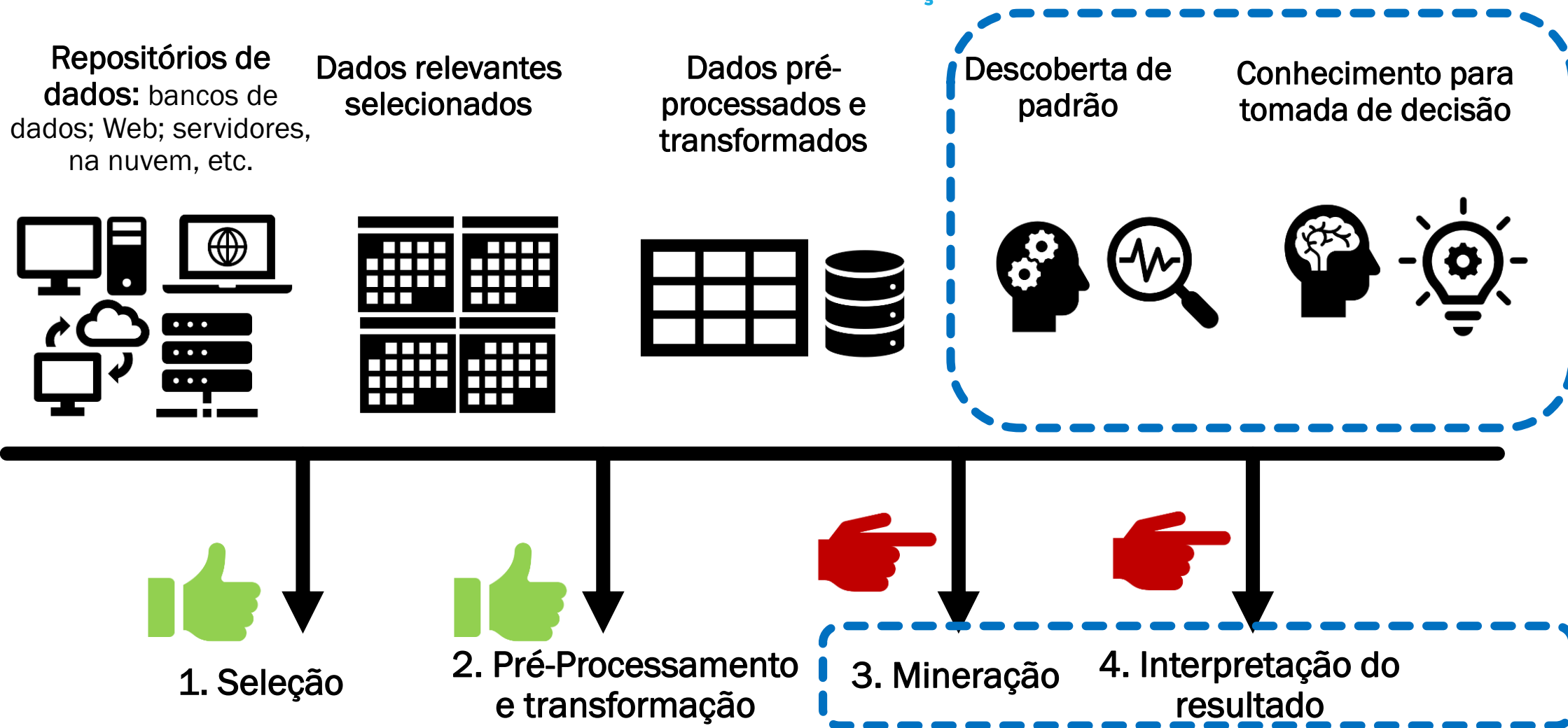


# CICLO DE VIDA DO DATA SCIENCE



# DATA MINING (MINERAÇÃO DE DADOS)

## ETAPAS DO PROCESSO DE MINERAÇÃO DE DADOS



# OVERFITTING E UNDERFITTING

## Overfitting

O modelo tem um desempenho excelente, porém quando utilizado com os dados de teste o resultado é ruim.

## Underfitting

O desempenho do modelo já é ruim no próprio treinamento. O modelo não consegue encontrar relações entre as variáveis e o teste nem precisa acontecer.

# OVERFITTING

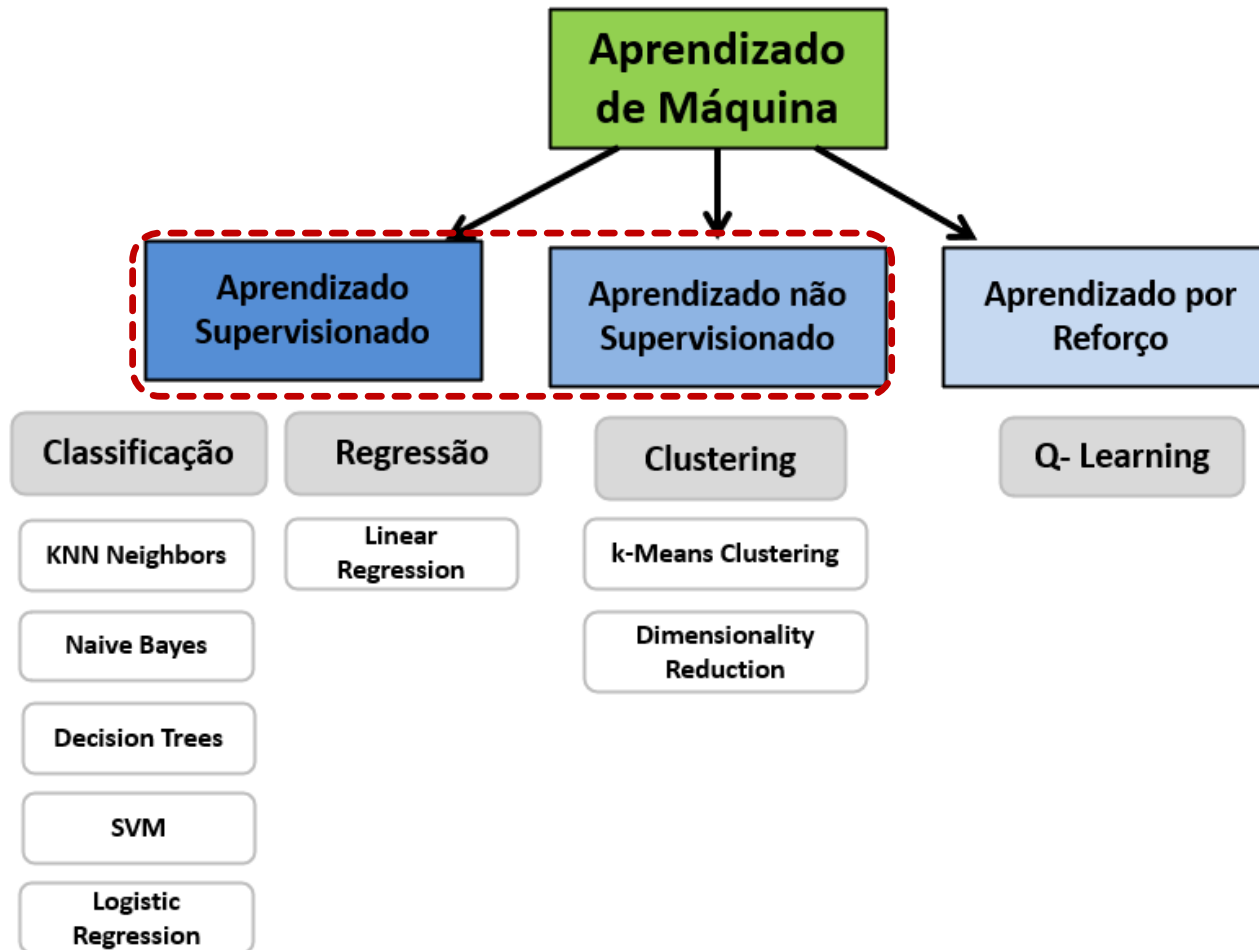
- ❖ Um cenário de overfitting ocorre quando, nos dados de treinamento, o modelo tem um desempenho excelente, porém quando utilizado com os dados de teste o resultado é ruim.
- ❖ Neste cenário, o modelo aprendeu tão bem as relações existentes no treino, que acabou “*apenas decorando o que deveria ser feito, e ao receber as informações das variáveis preditoras nos dados de teste, o modelo tenta aplicar as mesmas regras decoradas, porém com dados diferentes e esta regra não tem validade, e o desempenho é afetado.*”
- ❖ É comum ouvirmos que neste cenário o modelo treinado não tem capacidade de generalização (Didática Tech Inteligência Artificial & Data Science, 2020, Online).

# UNDERFITTING

- ❖ “Neste cenário o desempenho do modelo já é ruim no próprio treinamento.
- ❖ O modelo não consegue encontrar relações entre as variáveis e o teste nem precisa acontecer.
- ❖ *Este modelo já pode ser descartado, pois não terá utilidade”* (Didática Tech Inteligência Artificial & Data Science, 2020, Online).

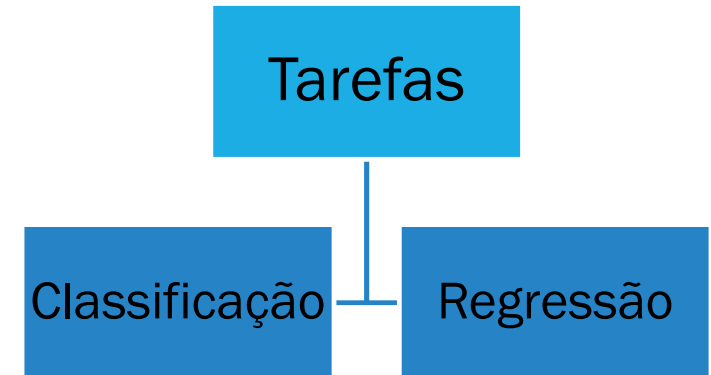
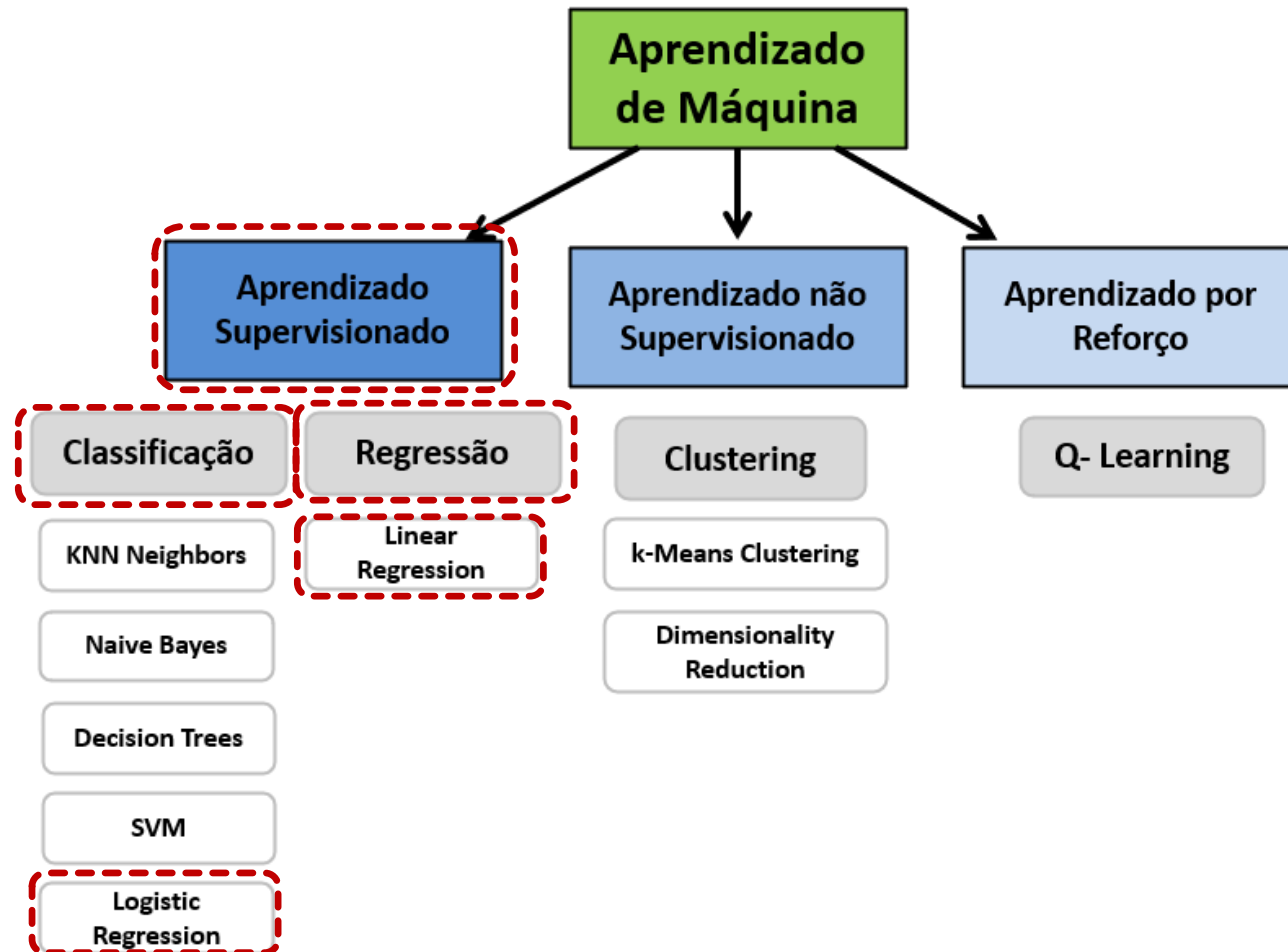


# MACHINE LEARNING (APRENDIZADO DE MÁQUINA)



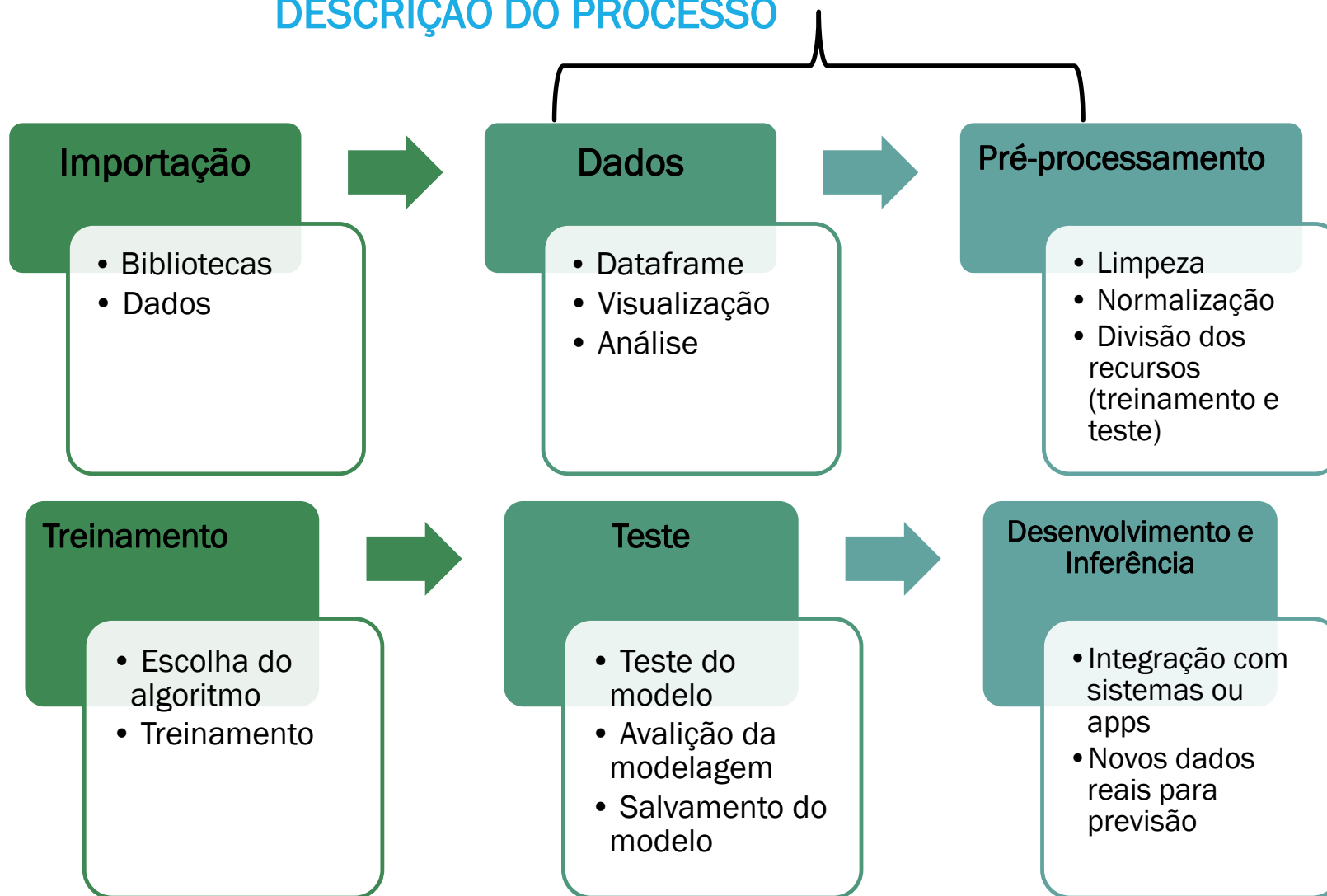
# MACHINE LEARNING (APRENDIZADO DE MÁQUINA)

## APRENDIZADO SUPERVISIONADO



# UTILIZANDO O APRENDIZADO SUPERVISIONADO

## DESCRIÇÃO DO PROCESSO



# UTILIZANDO O APRENDIZADO SUPERVISIONADO

## ESCOLHA DOS ATRIBUTOS E PRÉ-PROCESSAMENTO DOS DADOS

### *Feature Selection*

(seleção dos recursos ou atributos)

### *Feature Engineering*

(engenharia de recursos) ou atributos

❖ Consiste em analisar e executar as operações de pré-processamento que podem ser realizadas nos conjuntos de dados antes da utilização do algoritmos de Aprendizado de Máquina. Essas operações consideram:

- ✓ Eliminação manual de atributos;
- ✓ Integração de dados;
- ✓ Amostragem de dados;
- ✓ Dados desbalanceados;
- ✓ Limpeza de dados;
- ✓ Transformação de dados;
- ✓ Redução de dimensionalidade.

# UTILIZANDO O APRENDIZADO SUPERVISIONADO

## SEPARAÇÃO DOS DADOS EM BASE DE TREINAMENTO E BASE TESTE

- ❖ No aprendizado supervisionado a base de dados deve ser separada em base de treinamento e base de teste. Há formas de separação da base de dados em bases de treinamento e de testes conforme descrito abaixo:

### *Percentage Split / Hold out*

Particiona a base por amostragem. Tipos de amostragem interferem no resultado. Costuma ser utilizado quando a base de dados é grande. Geralmente a divisão considera 80% dos dados para treinamento e 20% para teste.

### *Cross Validation*

Particiona em K-partes. Por exemplo: separa a base em 10 partes. Em cada rodada usa 9 blocos para treinamento e 1 bloco para teste. Costuma ser utilizado quando a base de dados é pequena.

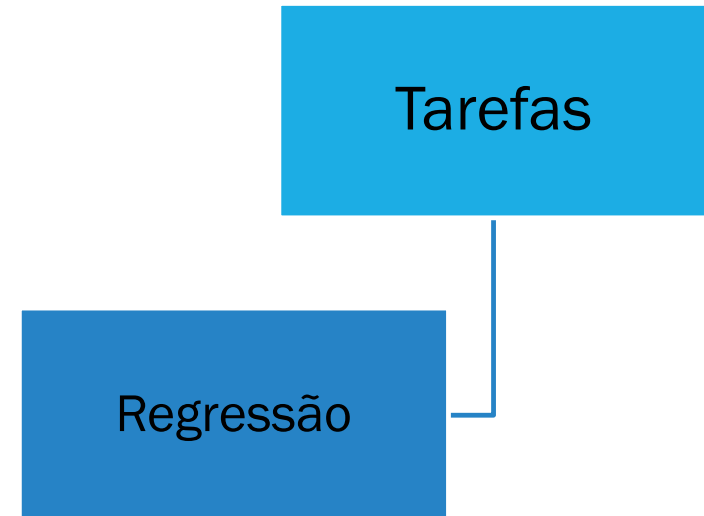
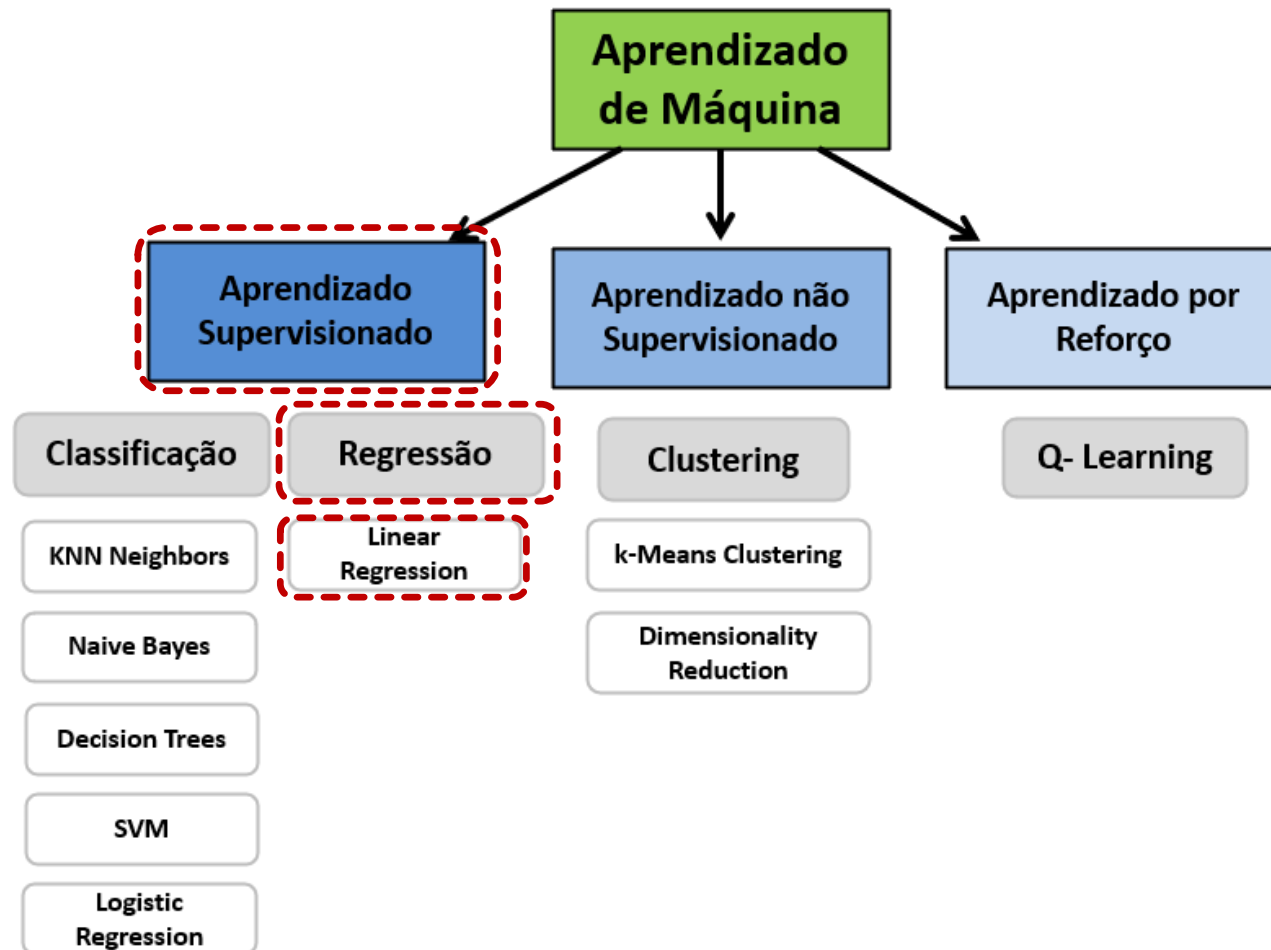
# UTILIZANDO APRENDIZADO SUPERVISIONADO

## ESCOLHA DO ALGORITMO

- ❖ Critérios que deve ser considerados para efetuar a escolha:
  - ✓ **Tarefa:** classificação ou previsão?
  - ✓ **Tipos de dados.**
  - ✓ **Distribuição das classes:** verificar se elas estão balanceadas.
  - ✓ **Interpretabilidade dos resultados.**

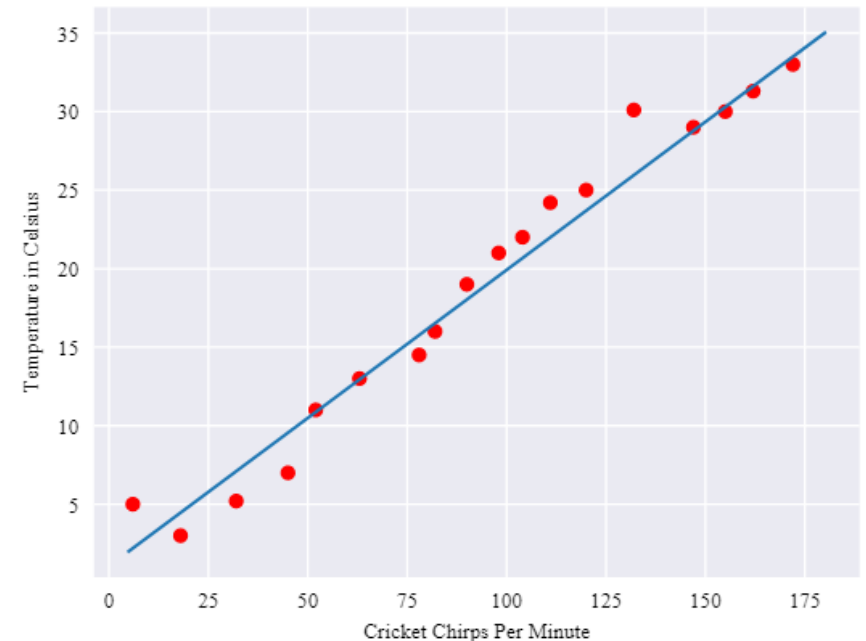
# MACHINE LEARNING (APRENDIZADO DE MÁQUINA)

## PREVISÃO - REGRESSÃO



# PREVISÃO - REGRESSÃO LINEAR

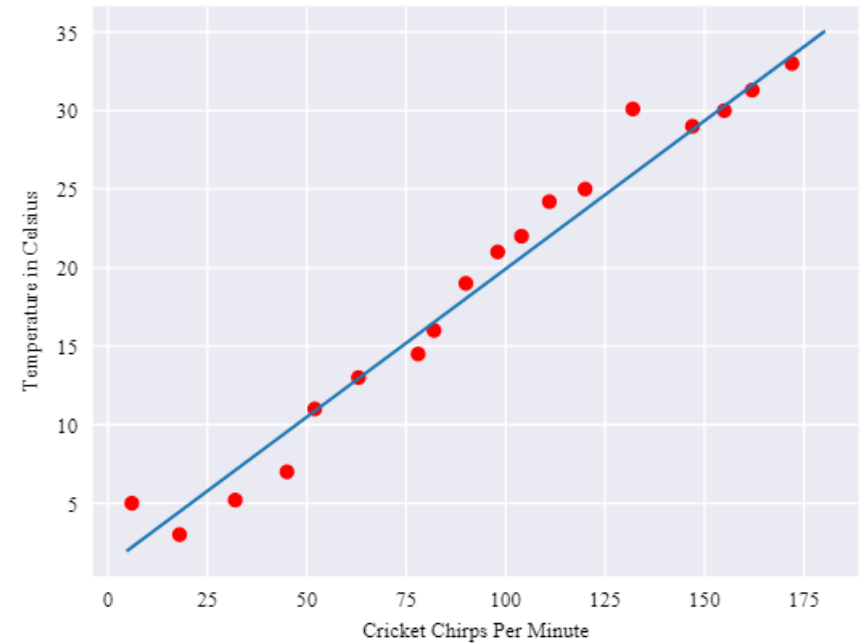
- ❖ Modelo que descreve a relação linear (reta) entre duas ou mais variáveis.
- ❖ Utiliza valores contínuos para efetuar a previsão.
- ❖ Considera dois tipos de variáveis:
  - ✓ **dependente (y)** => "alvo" ou "rótulo" da previsão;
  - ✓ **uma ou mais variáveis independentes (x)** => variáveis preditoras.





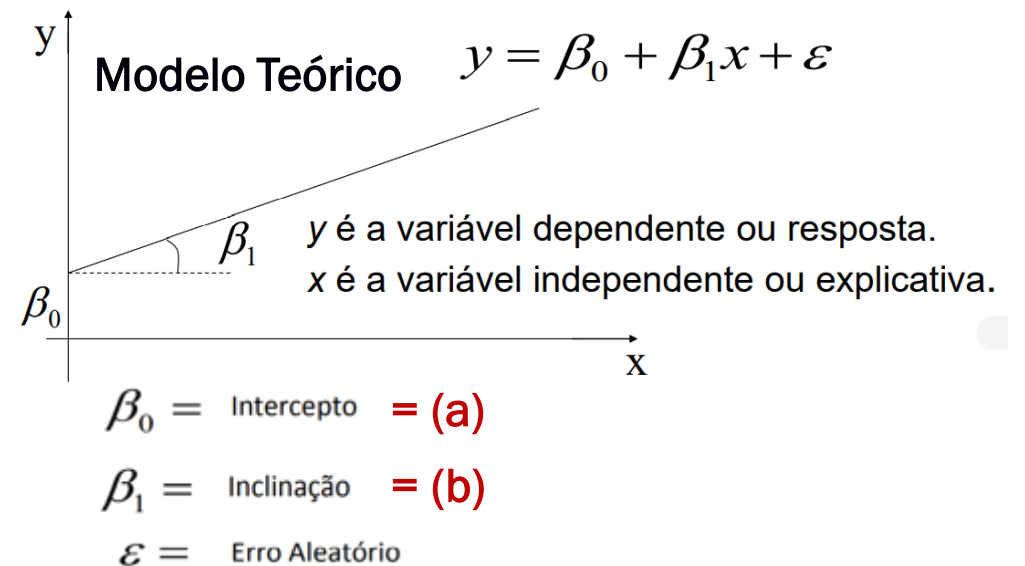
# PREVISÃO - REGRESSÃO LINEAR

- ❖ Para estabelecer a relação entre as variáveis é utilizada a equação da reta:  $\hat{y} = a + bx$ .
- ❖ Onde:
  - $\hat{y}$  é a variável dependente ou o valor a ser previsto;
  - **a** e **b** são os parâmetros da reta;
  - **a** é conhecido como "interceptação" (*intercept*);
  - **b** é conhecido como "declive ou inclinação" (*slope*);
  - **x** é a variável independente ou preditora.



# PREVISÃO - REGRESSÃO LINEAR SIMPLES

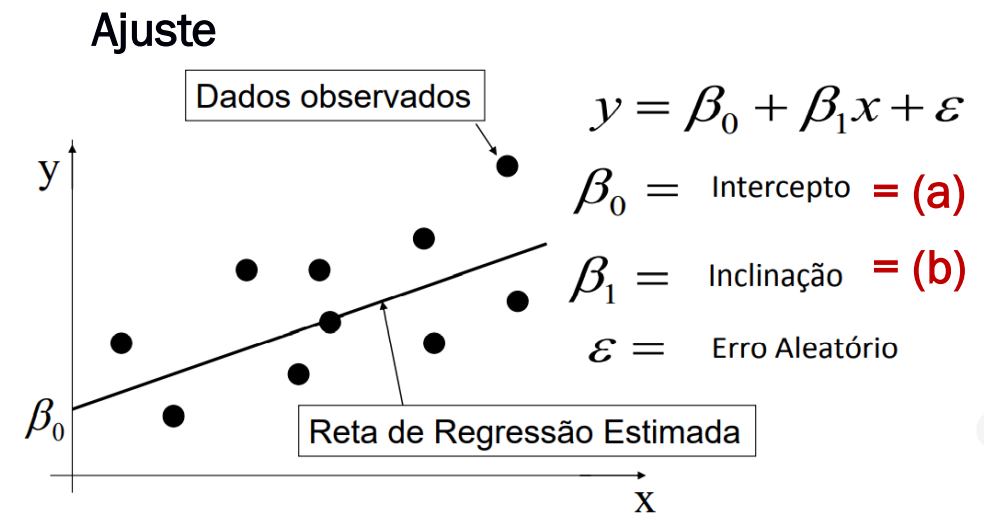
- ❖ Prever o valor de  $y$  com base em  $x$ , na regressão linear, implica encontrar uma reta que seja a mais adequada e melhor descreva os dados.
- ❖ O resultado do modelo vai ser uma reta, com equação:
  - ✓  $f(x) = \hat{y} = a + bx$  que vai passar por esses pontos.



Fonte: PUC-Minas

# PREVISÃO - REGRESSÃO LINEAR SIMPLES

- ❖ O objetivo é encontrar os coeficientes angular e linear que tornam essa reta ideal.
- ❖  $\hat{y}$  é a predição do modelo, e **Erro** o quão longe está da previsão correta.
- ❖ Para cada combinação dos coeficientes ( $a$ ,  $b_1$ ,  $b_2$ , ...,  $b_n$ ), com os dados de treino, temos um Erro, ou seja, uma função que determina o quão erradas as previsões do modelo estão.
- ❖ A modelagem da regressão linear consiste em encontrar os valores dos coeficientes da função que minimizam o erro.



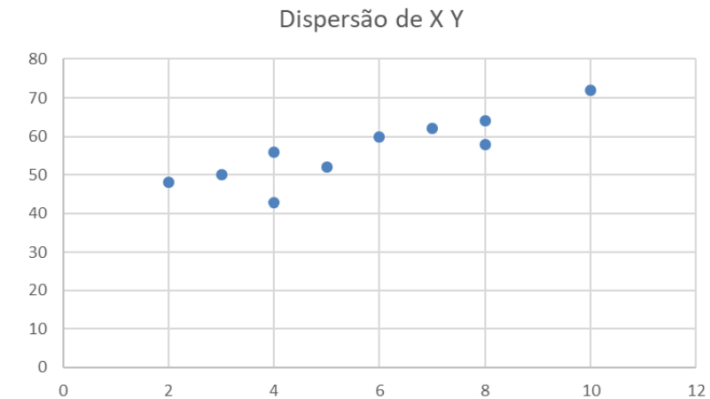
Fonte: PUC-Minas

# PREVISÃO REGRESSÃO LINEAR SIMPLES - EXEMPLO

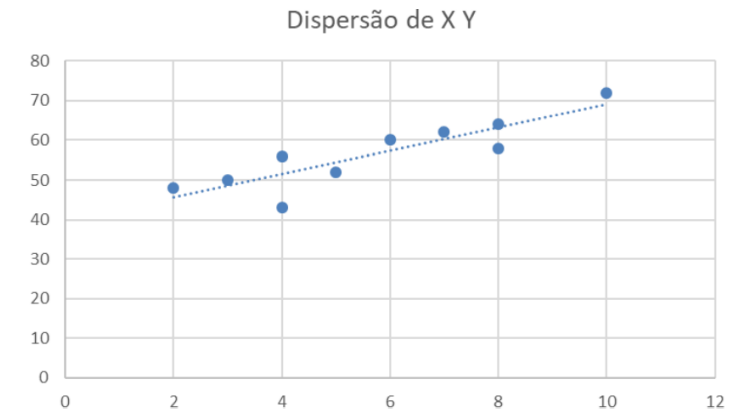
- ❖  $X$  = dados do tempo de serviço em anos, dos funcionários de uma seguradora (variável independente ou preditora).
- ❖  $Y$  = clientes que cada funcionário possui (variável dependente ou rótulo).

ID	A	B	C	D	E	F	G	H	I	J
X	2	3	4	5	4	6	7	8	8	10
Y	48	50	56	52	43	60	62	58	64	72

Gráfico de Dispersão



Equação da Reta de Dispersão



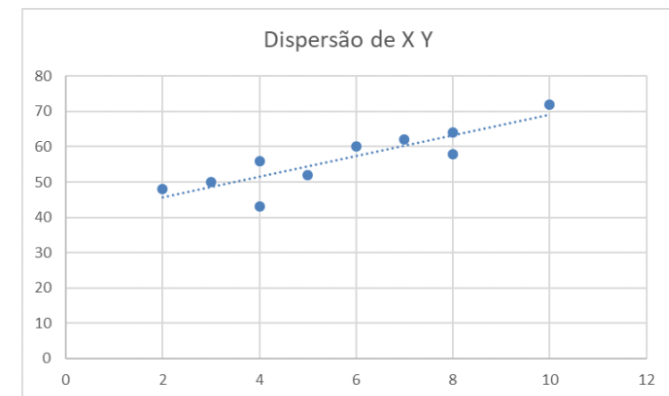
# PREVISÃO - REGRESSÃO LINEAR SIMPLES - EXEMPLO

## MODELO DE REGRESSÃO

### Ajuste

$$\hat{y} = 39,67 + 2,95x + \varepsilon$$

*Intercept      Slope*



### Modelo de Previsão

$$\hat{y} = 39,67 + 2,95 * 8 = 63,286 \cong 63 \text{ clientes}$$

- ❖ X = dados do tempo de serviço em anos, dos funcionários de uma seguradora (variável independente ou preditora).
- ❖ Y = clientes que cada funcionário possui (variável dependente ou rótulo).

ID	A	B	C	D	E	F	G	H	I	J
X	2	3	4	5	4	6	7	8	8	10
Y	48	50	56	52	43	60	62	58	64	72

# PREVISÃO - REGRESSÃO LINEAR MÚLTIPLA

- ❖ A **Regressão Linear Múltipla** é muito semelhante à **Regressão Linear Simples**, mas este método é utilizado para explicar a relação entre uma variável de resposta (dependente) e duas ou mais variáveis preditoras (independentes).

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + \varepsilon$$

$\beta_0$  = Intercepto = **(a)**

$\beta_1$  = Inclinação = **(b)**

$\varepsilon$  = Erro Aleatório

## Matriz de Regressão

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \cdot \\ \cdot \\ \cdot \\ y_n \end{bmatrix} \quad X = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1k} \\ 1 & x_{21} & x_{22} & \cdots & x_{2k} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nk} \end{bmatrix} \quad \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \cdot \\ \cdot \\ \cdot \\ \beta_k \end{bmatrix} \quad \varepsilon = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \cdot \\ \cdot \\ \cdot \\ \varepsilon_n \end{bmatrix}$$

# PREVISÃO - REGRESSÃO LINEAR MÚLTIPLA - EXEMPLO

- ❖ Esses dados representam a resistência à tração (y) de uma ligação de fio em um processo de fabricação de semicondutores, comprimento de fio (x1) e altura da matriz (x2) para ilustrar a construção de um modelo empírico. p

## Modelo de Previsão

$$\hat{y} = 2,26 + 2,744x_1 + 0,013x_2 + \varepsilon$$

*Intercept      Slope              Slope*

$$y^{\wedge} = 2,26 + 2,744 * 2 + 0,013 * 50 =$$
$$y^{\wedge} = 2,26 + 5,488 + 0,65 = \mathbf{8,398}$$

ID	y	x1	x2
1	9,95	2	50
2	24,45	8	110
3	31,75	11	120
.	.	.	.
.	.	.	.
.	.	.	.
24	22,13	6	100
25	21,15	5	400

# PREVISÃO - REGRESSÃO LINEAR SIMPLES E MÚLTIPLA

## MÉTRICAS PARA AVALIAÇÃO DO MODELO

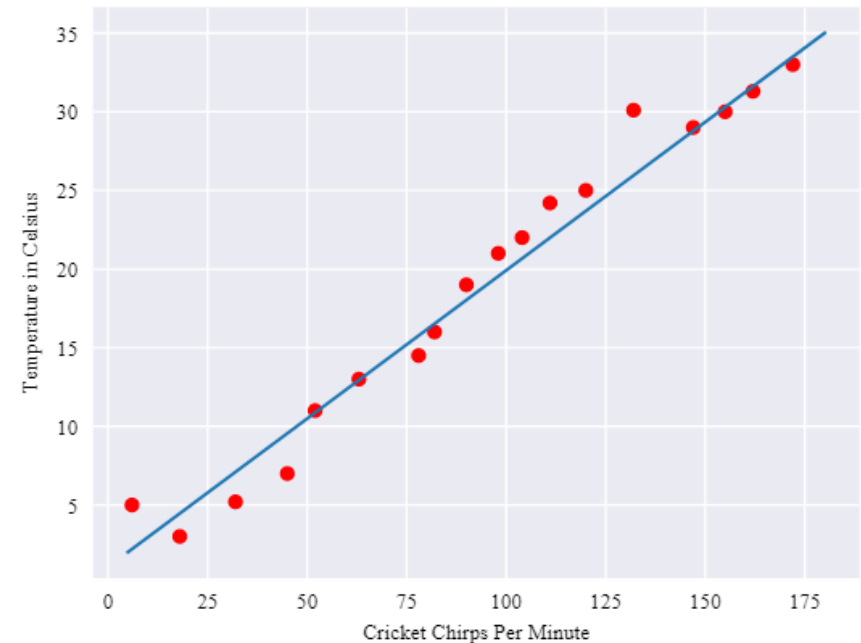
- ❖ **Mean Error (ME):** média da diferença entre o realizado e o previsto. Esta sujeita a valores positivos e negativos.
- ❖ **Mean Absolute Error (MAE):** media da diferença absoluta entre o realizado e o previsto.
- ❖ **Root Mean Squared Error (RMSE):** desvio padrão da amostra da diferença entre o previsto e o realizado.
- ❖ **Mean Percentage Error (MPE):** diferença percentual de erro.
- ❖ **Mean Absolute Percentage Error (MAPE):** diferença absoluta percentual de erro;
- ❖ **Mean Squared Error (MSE):** erro quadrático médio;
- ❖ **Mean Absolute Distance (MAD):** distância absoluta média.
  - ✓ **Analizando o resultado:** quanto menor o valor melhor.
- ❖ **R2:** coeficiente de determinação é uma medida de ajuste de um modelo linear que mede a quantidade da variância dos dados. Varia de 0 a 1.
  - ✓ **Analizando o resultado:** quanto maior o valor e próximo de 1 melhor.



# PREVISÃO - REGRESSÃO LINEAR

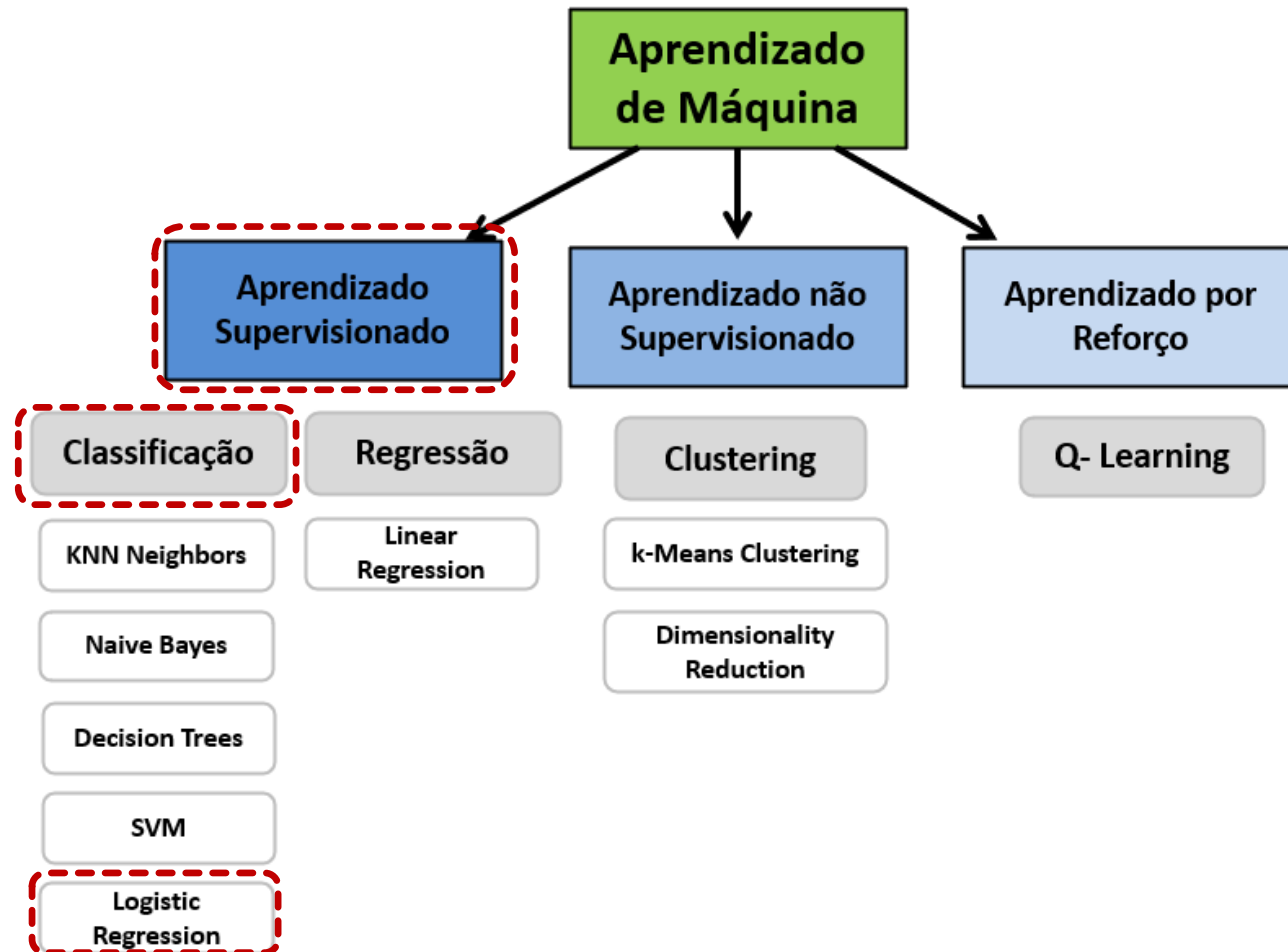
## ❖ Aplicações:

- ✓ previsão de vendas anuais a partir de variáveis independentes, como idade, educação e anos de experiência;
- ✓ na psicologia, para determinar a satisfação individual com base em fatores demográficos e psicológicos;
- ✓ prever o preço de uma casa, com base em seu tamanho, número de quartos, etc.



# MACHINE LEARNING (APRENDIZADO DE MÁQUINA)

## APRENDIZADO SUPERVISIONADO

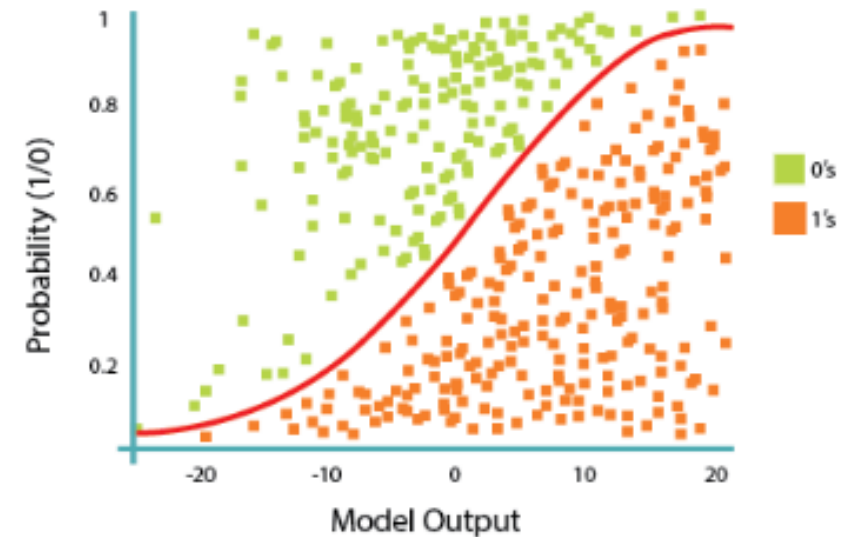


Tarefas

Classificação

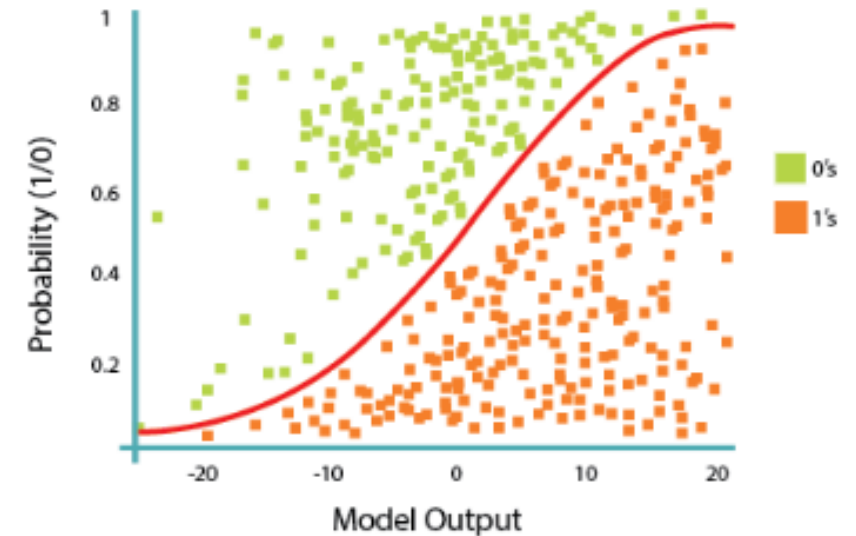
# CLASSIFICAÇÃO - REGRESSÃO LOGÍSTICA

- ❖ **Regressão logística** é uma variação da **regressão linear**, útil quando a variável dependente observada,  $y$ , é qualitativa (categórica).
- ❖ Produz uma fórmula que prediz a probabilidade do rótulo da classe em função das variáveis independentes.
- ❖ **REFORÇANDO:** a regressão logística é análoga à regressão linear, basicamente a diferença está no campo utilizado como rótulo que na regressão logística é qualitativo (categórico) e na regressão linear é quantitativo (numérico).



# CLASSIFICAÇÃO - REGRESSÃO LOGÍSTICA

- ❖ A **regressão logística** se ajusta a uma curva especial em forma de s.
- ❖ Toma a regressão linear como base e transforma a estimativa numérica em uma probabilidade.
- ❖ Esta função s é chamada sigmóide.
- ❖ **REFORÇANDO:** a regressão logística é uma junção de regressão linear + sigmóide + probabilidade.

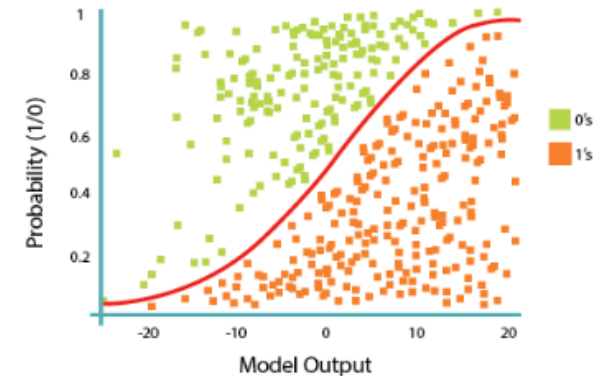


# CLASSIFICAÇÃO - REGRESSÃO LOGÍSTICA

❖ Na regressão logística a variável dependente ou alvo/rótulo (Y) é qualitativa. Exemplo:

- ✓ (sim/não);
- ✓ (sucesso / fracasso);
- ✓ doente / saudável;
- ✓ (0 ou 1);
- ✓ Nestes exemplos temos a quantidade de duas classes.

❖ Já, as variáveis independentes ou preditoras (X) podem ser quantitativas ou qualitativas.



# CLASSIFICAÇÃO – REGRESSÃO LOGÍSTICA

## MÉTRICAS PARA AVALIAÇÃO DO MODELO

- ❖ Mostra o número de classificações corretas versus as classificações preditas para cada classe, sobre um conjunto de exemplos.
- ❖ **Analisando o resultado:** a matriz de confusão de um classificador ideal possui valores não nulos apenas na diagonal.

	CLASSE A	CLASSE B	PRECISÃO
PRED. CLASSE A	$T_P$	$F_P$	$T_P / (T_P + F_P)$
PRED. CLASSE B	$F_N$	$T_N$	
REVOCAÇÃO	$T_P / (T_P + F_N)$		
$T_P$ – True positive $F_P$ – False Positive			
$F_N$ – False Negative $T_N$ – True Negative			

# CLASSIFICAÇÃO – REGRESSÃO LÓGISTICA

## MÉTRICAS PARA AVALIAÇÃO DO MODELO

### ❖ Algumas métricas decorrentes da matriz de confusão:

**Accuracy (Acurácia):** diz quanto o modelo acertou das previsões possíveis. Ou seja, Porcentagem de elementos classificados corretamente (positivos ou negativos):

$$accuracy = \frac{TP + TN}{TP + FP + TN + FN} = \frac{\text{previsões corretas}}{\text{todas as previsões}}$$

### Exemplo: Detecção de SPAM

	PREV. SPAM	PREV. NÃO SPAM
SPAM	80	20
NÃO SPAM	5	195

### Acurária

$$A = \frac{80 + 195}{100 + 200} = 91,7\%$$

# CLASSIFICAÇÃO – REGRESSÃO LOGÍSTICA

## MÉTRICAS PARA AVALIAÇÃO DO MODELO

### ❖ Algumas métricas decorrentes da matriz de confusão:

**Precision (Precisão):** é uma métrica que indica, das classificações positivas do modelo, quantas foram acertadas. Dentre os exemplos classificados como verdadeiros, quantos eram realmente verdadeiros:

$$precision = \frac{TP}{TP + FP}$$

**Recall (Revoção/Sensitividade):** é uma métrica que indica, das amostras positivas existentes, quantas o modelo conseguiu classificar corretamente. Dentre o total de exemplos verdadeiros, quantos foram classificados como verdadeiros:

$$recall = \frac{TP}{TP + FN}$$

**Especificidade:** porcentagem de amostras com **negativos verdadeiros** identificadas corretamente sobre o total de amostras negativas:

$$S = T_N / (T_N + F_P)$$



# CLASSIFICAÇÃO – REGRESSÃO LOGÍSTICA

## MÉTRICAS PARA AVALIAÇÃO DO MODELO

### ❖ Algumas métricas decorrentes da matriz de confusão:

**F-measure ou F-score:** mostra o balanço entre *precision* e *recall*. Ou seja, é Média ponderada de precisão e revocação:

$$2 * \frac{precision * recall}{precision + recall}$$

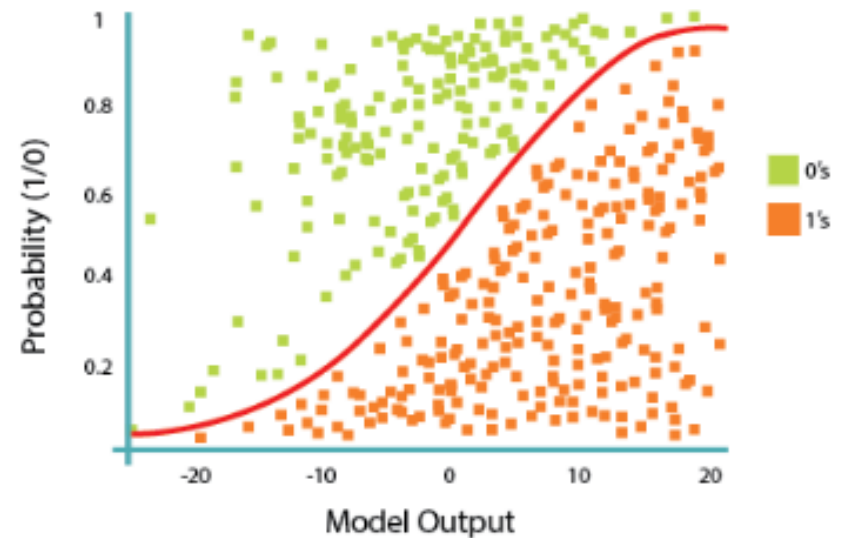
**Log-loss:** usado quando um classificado retorna uma probabilidade de classificação (“confiança”):

$$\log -loss = -\frac{1}{N} \sum_i^N y_i \log p_i + (1 - y_i) \log(1 - p_i)$$

# CLASSIFICAÇÃO - REGRESSÃO LOGÍSTICA

## ❖ Aplicações:

- ✓ prever se um paciente tem uma determinada doença, como diabetes, com base nas características observadas desse paciente, como peso, altura, pressão sanguínea e resultados de vários tipos de sangue e assim por diante.
- ✓ no marketing, prever a probabilidade de um cliente comprar um produto ou interromper uma assinatura (*churn*).
- ✓ prever a probabilidade de um proprietário deixar de pagar uma hipoteca.



# DIFERENÇA ENTRE REGRESSÃO LINEAR E REGRESSÃO LOGÍSTICA

- ❖ Embora a **regressão linear** seja adequada para estimar valores contínuos (por exemplo, estimar o preço da casa), não é a melhor ferramenta para prever a classe de um ponto de dados observado.
- ❖ Para estimar a classe de um ponto de dados, precisamos de algum tipo de orientação sobre qual seria a classe mais provável para aquele ponto de dados. Para isso, utilizamos **regressão logística**.
- ❖ A **Regressão linear** encontra uma função que relaciona uma variável dependente quantitativa,  $y$ , a alguns preditores (variáveis independentes  $x_1$ ,  $x_2$ , etc.). Por exemplo, a regressão linear assume uma função da forma:

$$y = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \dots$$

- ❖ A **Regressão linear** encontra os valores dos parâmetros  $\theta_0$ ,  $\theta_1$ ,  $\theta_2$ , etc, onde o termo  $\theta_0$  é a "interceptação" e os demais termos a "inclinação" da reta.

# DIFERENÇA ENTRE REGRESSÃO LINEAR E REGRESSÃO LOGÍSTICA

- ❖ A **Regressão Logística** é uma variação da Regressão Linear, utilizada quando a variável dependente  $y$  é categórica. A Regressão Logística produz uma fórmula que prevê a probabilidade do rótulo da classe em função das variáveis independentes  $x$ .
- ❖ A **Regressão Logística** se ajusta a uma curva em forma de  $s$ , tomando como base a regressão linear e transformando a estimativa numérica em uma probabilidade, utilizando uma função, chamada de função sigmóide  $\sigma$ :

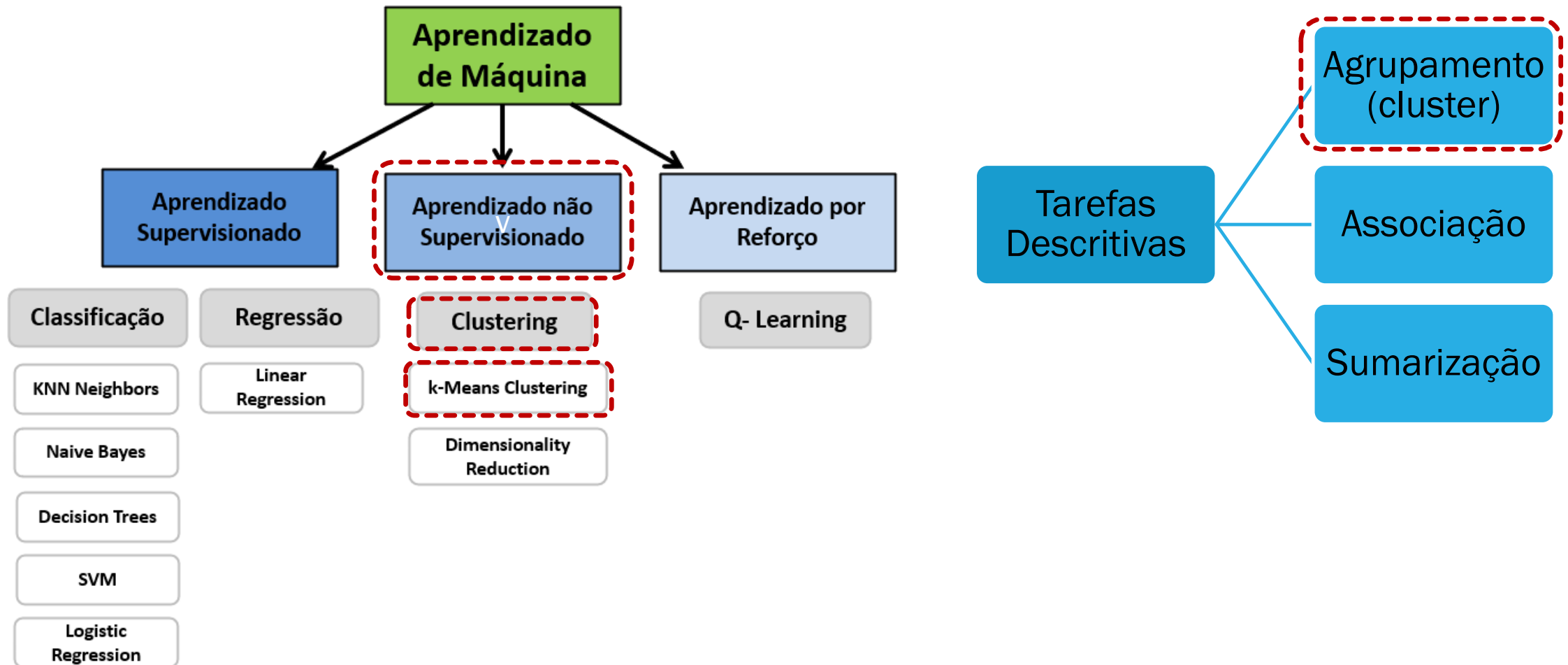
$$h_{\theta}(x) = \sigma(\theta^T X) = \frac{e^{(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \dots)}}{1 + e^{(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \dots)}}$$

OU

$$Probabilidade da Classe_1 = P(Y = 1|X) = \sigma(\theta^T X) = \frac{e^{\theta^T X}}{1 + e^{\theta^T X}}$$

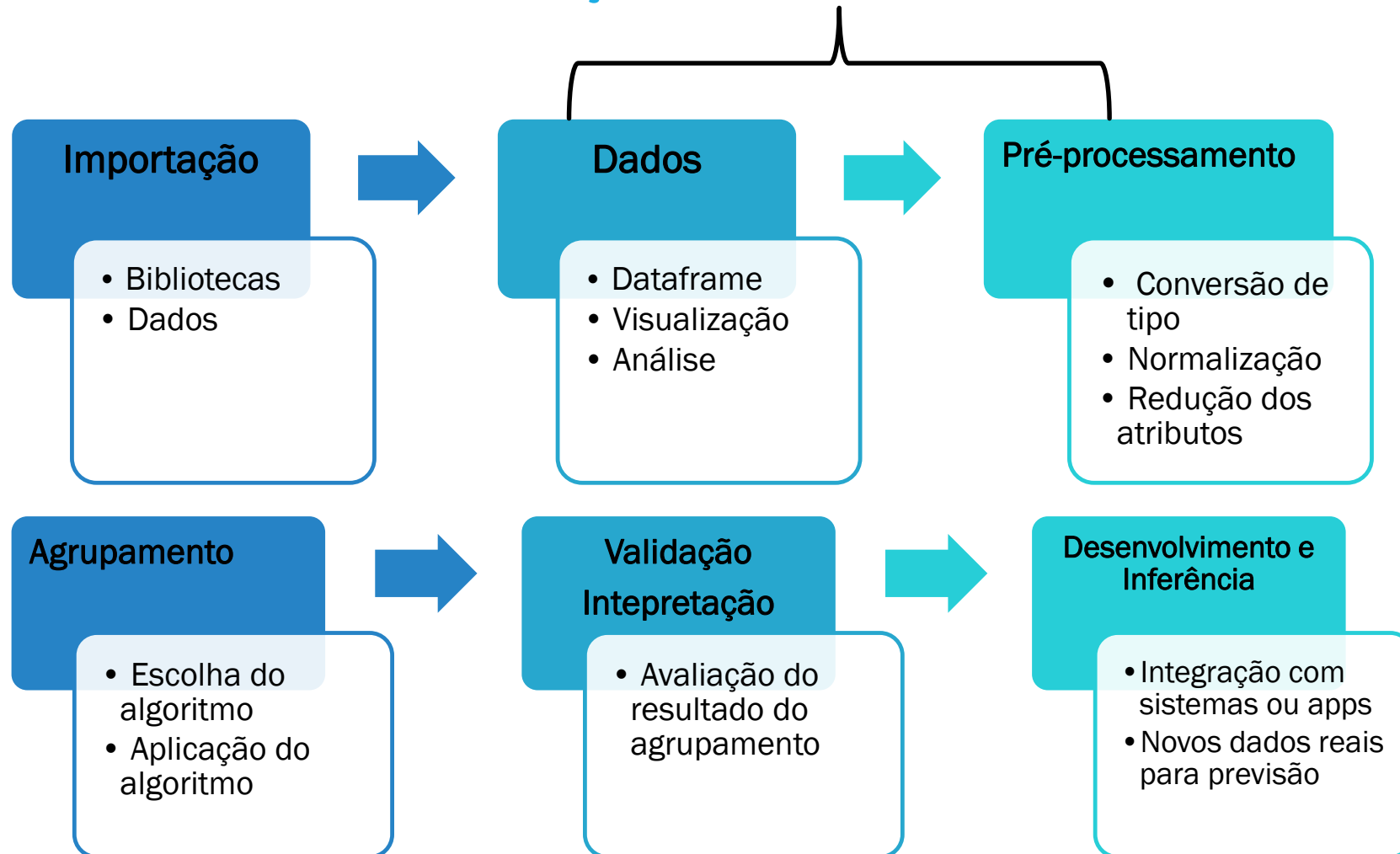
# MACHINE LEARNING (APRENDIZADO DE MÁQUINA)

## APRENDIZADO NÃO SUPERVISIONADO



# UTILIZANDO O APRENDIZADO NÃO SUPERVISIONADO

## DESCRIÇÃO DO PROCESSO



# UTILIZANDO O APRENDIZADO NÃO SUPERVISIONADO

## PRÉ-PROCESSAMENTO DOS DADOS

- ❖ O pré-processamento no aprendizado não supervisionado pode incluir:
  - ✓ Normalização;
  - ✓ Conversão de tipos;
  - ✓ Redução dos atributos.
- ❖ É importante ressaltar que no aprendizado não supervisionado técnicas de seleção, extração de características descritas para o tipo de aprendizado supervisionado não se aplicam, ou precisam ser adaptadas.

# UTILIZANDO O APRENDIZADO NÃO SUPERVISIONADO

## AGRUPAMENTO

- ❖ O agrupamento é a etapa central do processo de aprendizado de não supervisionado. Nela, um ou mais algoritmos de agrupamento são aplicados aos dados para a identificação dos clusters existentes nos dados.
- ❖ Os diferentes tipos de estruturas que podem ser encontradas por um algoritmo não supervisionado são, por exemplo, partições e hierarquias de participações.

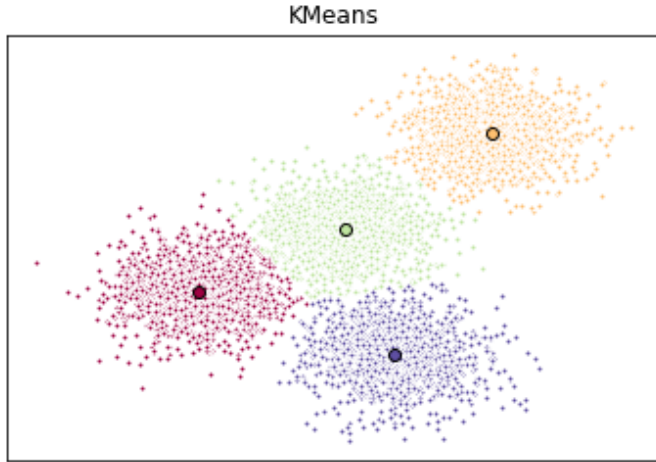


# UTILIZANDO O APRENDIZADO NÃO SUPERVISIONADO

## VALIDAÇÃO

- ❖ A análise e comparação de algoritmos não supervisionados são tarefas complexas e que dependem muito do domínio da aplicação e do conhecimento das técnicas de agrupamento empregadas.
- ❖ Uma característica importante, inerente ao aprendizado não supervisionado e que torna difícil a análise do desempenho e a comparação de algoritmos, é a ausência de uma estrutura ideal, que seja a resposta esperada para o agrupamento.
- ❖ Como o agrupamento é uma tarefa não supervisionada, não há uma classificação conhecida dos objetos.
- ❖ A análise de desempenho de algoritmos não supervisionados ainda é uma área aberta. Por isso, é essencial ter um bom entendimento da técnica que está sendo utilizada, conhecer detalhes sobre a obtenção dos dados e ter claramente definido o propósito do agrupamento que se deseja obter.

# AGRUPAMENTO – K-MEANS



Fonte: IBM

- ❖ k-Means funciona colocando aleatoriamente  $k$  centróides, um para cada cluster. Quanto mais afastados os clusters, melhor.
- ❖ O próximo passo é calcular a distância de cada ponto de dados (ou objeto) dos centróides.
- ❖ Em seguida, atribuir cada ponto de dados (ou objeto) ao centróide mais próximo, criando um grupo.
- ❖ Uma vez que cada ponto de dados tenha sido classificado em um grupo, recalcule a posição dos  $k$  centróides.
- ❖ A nova posição do centróide é determinada pela média de todos os pontos no grupo.
- ❖ Finalmente, isso continua até que os centróides não se movam mais.

# BIBLIOTECA DE MACHINE LEARNING

## scikit-learn

*Machine Learning in Python*

Getting Started

Release Highlights for 0.23

GitHub

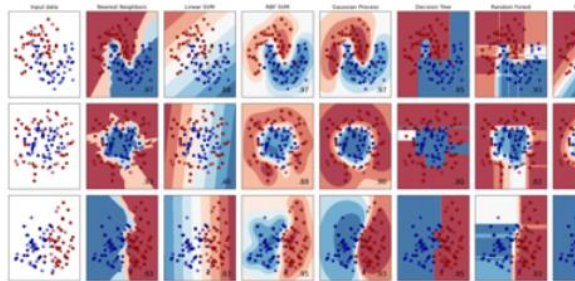
- Simple and efficient tools for predictive data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

### Classification

Identifying which category an object belongs to.

**Applications:** Spam detection, image recognition.

**Algorithms:** SVM, nearest neighbors, random forest, and more...

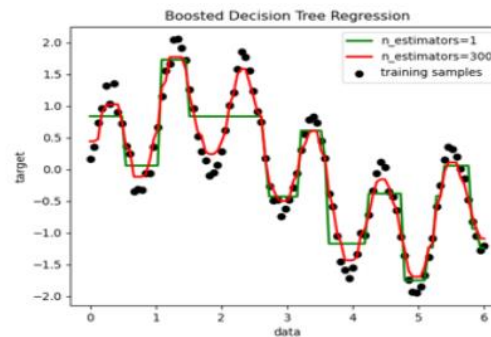


### Regression

Predicting a continuous-valued attribute associated with an object.

**Applications:** Drug response, Stock prices.

**Algorithms:** SVR, nearest neighbors, random forest, and more...



### Clustering

Automatic grouping of similar objects into sets.

**Applications:** Customer segmentation, Grouping experiment outcomes

**Algorithms:** k-Means, spectral clustering, mean-shift, and more...



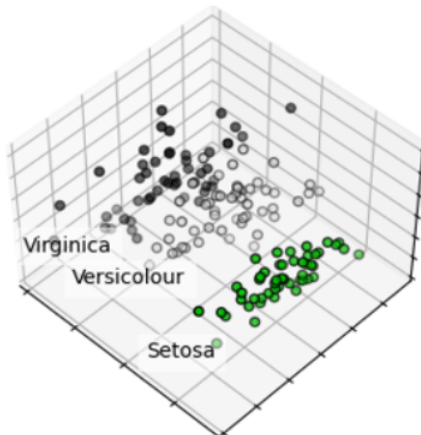
# BIBLIOTECA DE MACHINE LEARNING

## Dimensionality reduction

Reducing the number of random variables to consider.

**Applications:** Visualization, Increased efficiency

**Algorithms:** k-Means, feature selection, non-negative matrix factorization, and more...

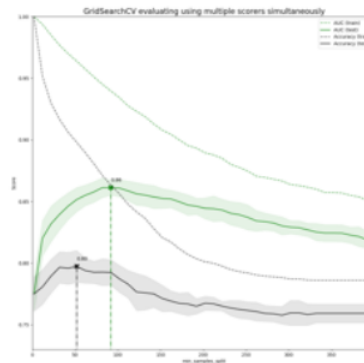


## Model selection

Comparing, validating and choosing parameters and models.

**Applications:** Improved accuracy via parameter tuning

**Algorithms:** grid search, cross validation, metrics, and more...

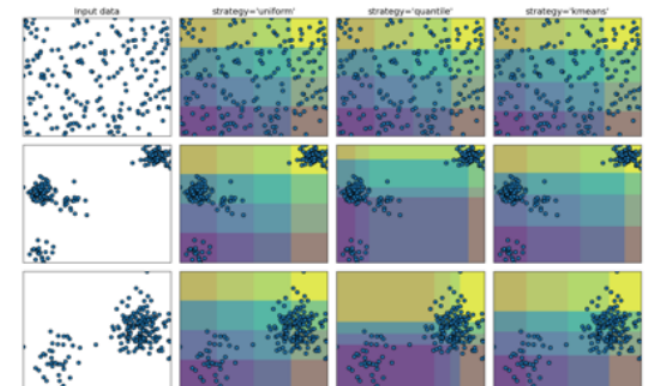


## Preprocessing

Feature extraction and normalization.

**Applications:** Transforming input data such as text for use with machine learning algorithms.

**Algorithms:** preprocessing, feature extraction, and more...



# BIBLIOTECA DE MACHINE LEARNING

## SCIKIT-LEARN

**scikit-learn**  
*Machine Learning in Python*

Getting Started Release Highlights for 0.23 GitHub

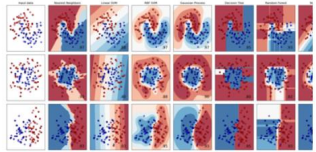
- Simple and efficient tools for predictive data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

### Classification

Identifying which category an object belongs to.

**Applications:** Spam detection, image recognition.

**Algorithms:** SVM, nearest neighbors, random forest, and more...

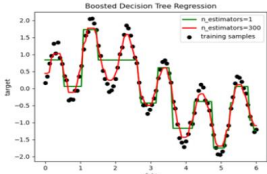


### Regression

Predicting a continuous-valued attribute associated with an object.

**Applications:** Drug response, Stock prices.

**Algorithms:** SVR, nearest neighbors, random forest, and more...

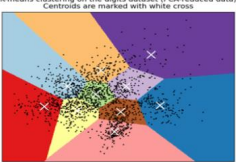


### Clustering

Automatic grouping of similar objects into sets.

**Applications:** Customer segmentation, Grouping experiment outcomes

**Algorithms:** k-Means, spectral clustering, mean-shift, and more...



- ❖ O **scikit-learn** ou apenas **sklearn** é uma biblioteca do Python específica para aplicações de Machine Learning.
- ❖ Possui pacotes e módulos para classificação, regressão e clustering.
- ❖ É código aberto e foi construída sobre os pacotes NumPy, SciPy e Matplotlib.

# BIBLIOTECA DE MACHINE LEARNING

## SCIKIT-LEARN – PRINCIPAIS APLICAÇÕES

- ❖ **Pré-processamento:** como já mencionado em várias aulas, é a etapa mais trabalhosa no processo de desenvolvimento de um modelo de **Machine Learning**. Nesta etapa, o **NumPy** e o **Pandas** são amplamente utilizados, porém o **sklearn** também possui funções para esta etapa, com foco no tratamento dos dados.
- ❖ **Regressão:** consiste em modelos que visam atribuir um valor quantitativo a um elemento. Com este tipo de modelo é possível prever o preço de um automóvel, altura de uma pessoa, quantidade de vendas de um produto, entre outros.
- ❖ **Classificação:** consiste em modelos capazes de detectar qual categoria pré-determinada um elemento pertence. Com este tipo de modelo é possível efetuar previsões que classificam se uma pessoa possui ou não determinada doença, se um cliente deixará ou não de consumir determinado produto, entre outros.



# BIBLIOTECA DE MACHINE LEARNING

## SCIKIT-LEARN – PRINCIPAIS APLICAÇÕES

- ❖ **Clusterização:** consiste em modelos para detecção de grupos cujos integrantes possuem características similares. Com este tipo de modelo é possível identificar clientes com comportamentos parecidos, grupos de risco de determinada doença, verificar padrões entre moradores de uma cidade, e muitos outros agrupamentos.
- ❖ **Redução de dimensionalidade:** consiste em modelos que visam reduzir o número de variáveis em um problema. Com esta redução é possível diminuir consideravelmente a quantidade de cálculos necessários para geração de um modelo, aumentando a eficiência, com uma perda mínima de assertividade.
- ❖ **Ajuste de parâmetros:** visa comparar, validar e escolher parâmetros e modelos, de maneira automatizada. Permite facilmente comparar diferentes parâmetros no ajuste de um modelo, encontrando dessa forma a melhor configuração para uma determinada aplicação.

# REFERÊNCIAS

**FACELI**, Katti et al. Inteligência artificial: uma abordagem de aprendizado de máquina. Rio de Janeiro, RJ: LTC, 2011.

**Didática Tech Inteligência Artificial & Data Science**. 2020. Disponível em: < <https://didatica.tech/underfitting-e-overfitting/>>. Acesso em: 21 nov 2020.



---

# DÚVIDAS





# OBRIGADA!!!!

Profa. Carla Oliveira

E-mails: [carla.olivei@gmail.com](mailto:carla.olivei@gmail.com) e [carlaos@unicid.edu.br](mailto:carlaos@unicid.edu.br)