



Inteligência
Artificial e Big
Data
Aula 03

Prof. Me Daniel Vieira

SENAI

Agenda

- 1- Crisp Dm
- 2- Crisp DM x KDD
- 3- Inteligência Artificial
- 4 - Machine Learning
- 5 -Aplicações de Machine Learning
- 6 - Algoritmos utilizados em aprendizado de máquina
- 7- Análise de conglomerados, PCA
- 8 - Regressão linear
- 9 - Regressão múltipla
- 10 Exercícios
- 11 - Código exemplo árvore de decisão

Crisp DM - Cross Industry Standard Process for Data Mining

Metodologia utilizada para mineração de dados

Entendimento do Negócio

- Objetivos do projeto
- Requisitos do cliente
- Escopo do projeto

Entendimento dos dados

- Coleta de dados
- Análise exploratória para entender sua estrutura e qualidade

Preparação dos dados

- Seleção de variáveis relevantes, tratamento de valores ausentes, transformação e limpeza dos dados

Modelagem

- Aplicação dos algoritmos para mineração de dados:
- Regressão, classificação, árvore de decisão, knn, k-means

Avaliação

- Após criar os modelos, é necessário avaliar sua eficiência Utilização de métricas para avaliação:
- Acurácia
- Precisão
- Recall

Crisp DM - Cross Industry Standard Process for Data Mining

Metodologia utilizada para mineração de dados



Implementação

- Implementação do modelo em produção.
- Integração em sistemas existentes, realização de treinamentos para os usuários e avaliação contínua do desempenho do modelo em ambiente real

Crisp DM

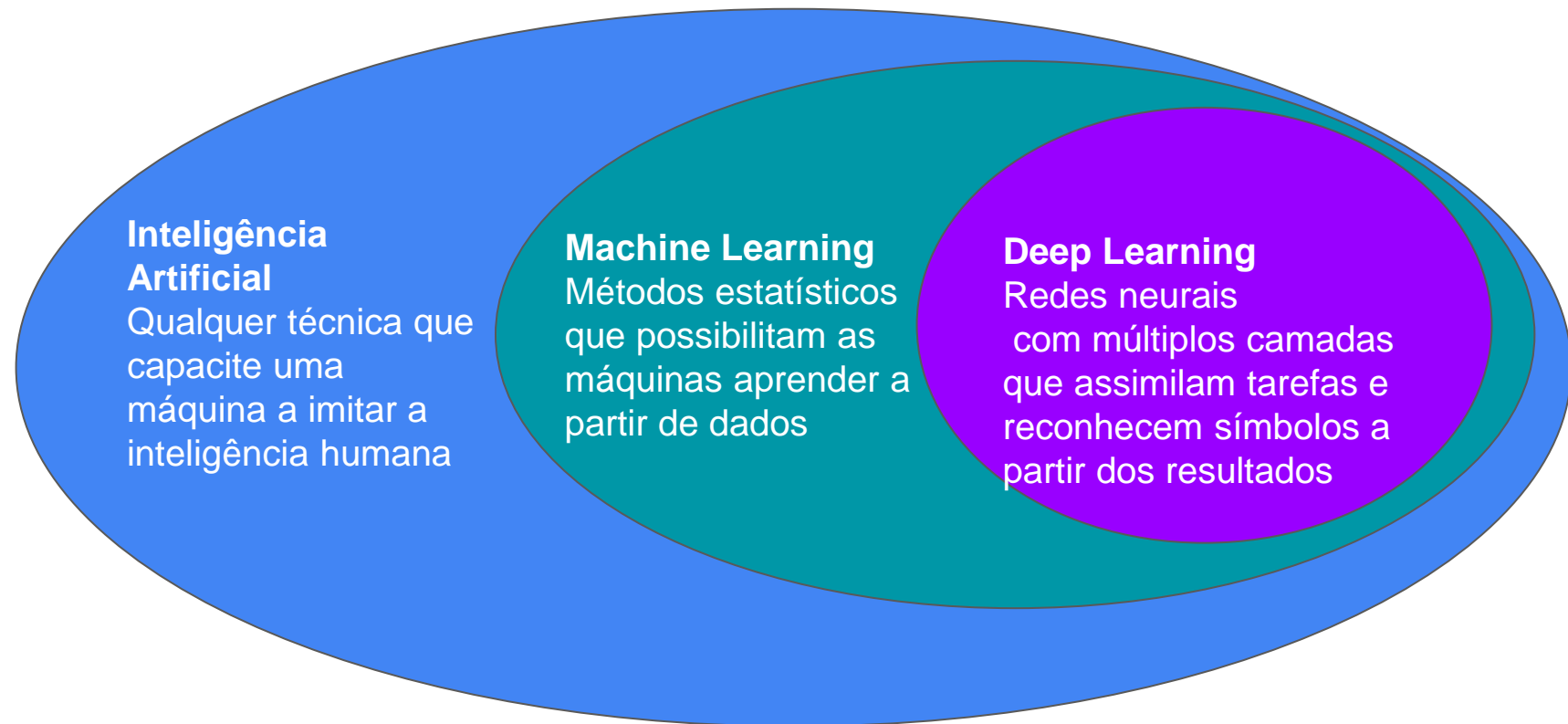
Metodologia utilizada para mineração de dados,
É cíclica, ou seja, após a conclusão da fase de implantação, é possível voltar a etapas anteriores para refinar ou aprimorar o projeto

Crisp DM ou KDD ?

KDD - Metodologia mais antiga, final de 1980 e início da década de 1990. É mais abrangente e envolve desde a etapa da seleção do banco de dados e de descoberta de conhecimento.

Crisp DM - Mais atual e utilizada na indústria em projetos de Mineração de dados

Inteligência Artificial

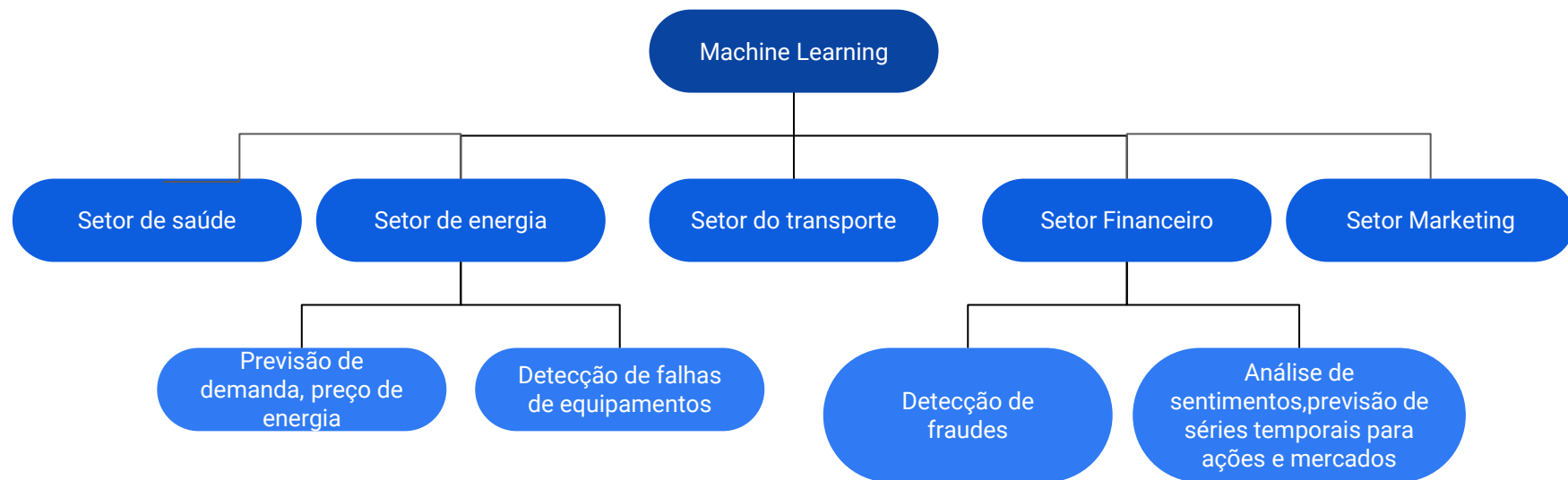


Machine Learning (Aprendizado de máquina)

Métodos estatísticos que possibilitam as máquinas aprender a partir de dados



Machine Learning (Aprendizado de máquina)



Algoritmos utilizados em aprendizado de máquina

Algoritmos Machine Learning

Regressão linear

Regressão logística

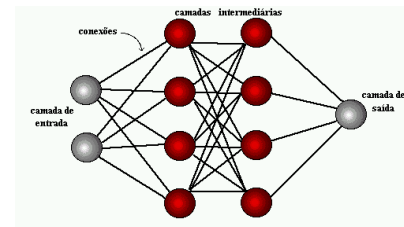
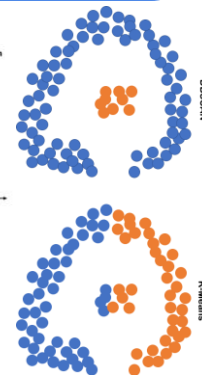
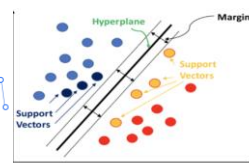
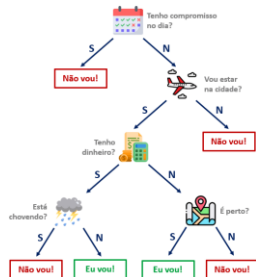
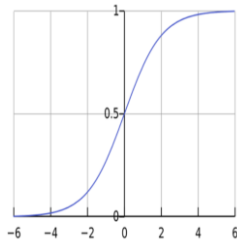
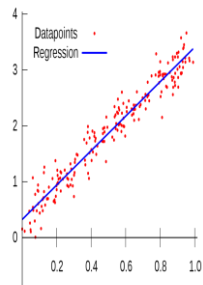
Árvore de decisão

Random Forest

Support Vector Machine

Clusterização DBSCAN

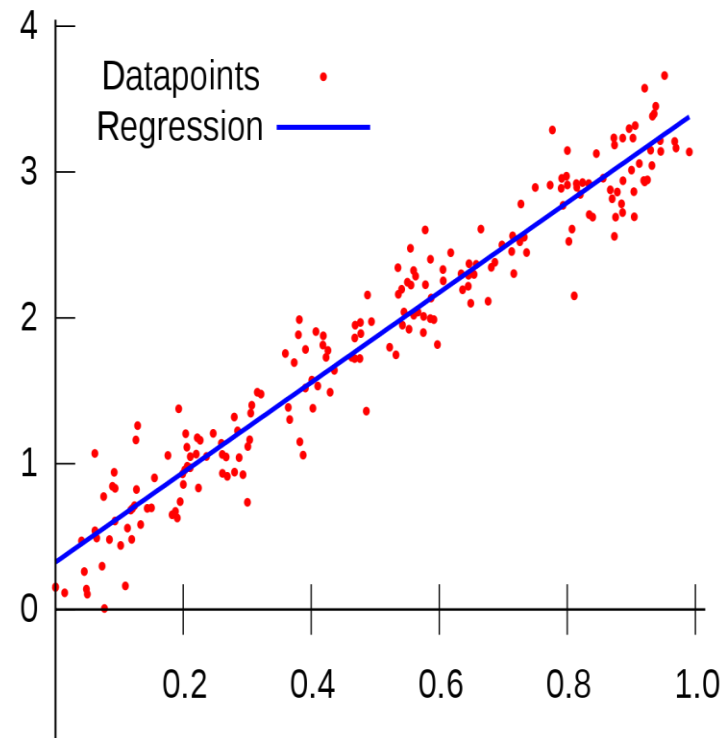
Redes neurais artificiais



Algoritmos utilizados em aprendizado de máquina

Regressão Linear

Utilizado em problemas onde o objetivo é prever um valor contínuo. Ele encontra a relação linear entre as variáveis de entrada e a variável de saída. Ex: Caso dos Imóveis



Regressão Linear

A regressão linear simples é definida pela função linear:

$$Y = \beta_0 X + \beta_1 + \varepsilon$$

β_0 e β_1 são duas constantes desconhecidas que representam a inclinação da regressão, enquanto ε (épsilon) é o termo de erro.

Você pode usar a regressão linear simples para modelar a relação entre duas variáveis, como estas:

Precipitação e rendimento de colheitas

Idade e altura em crianças

Temperatura e expansão do mercúrio metálico em um termômetro

Regressão Múltipla

Regressão logística

Os cientistas de dados usam a regressão logística para medir a probabilidade de um evento ocorrer. A previsão é um valor entre 0 e 1, em que 0 indica um evento improvável de acontecer e 1 indica a probabilidade máxima de que esse evento aconteça. Equações logísticas usam funções logarítmicas para calcular a linha de regressão.

Veja alguns exemplos:

A probabilidade de uma vitória ou derrota em uma partida esportiva

A probabilidade de aprovação ou reprovação em um exame

A probabilidade de uma imagem ser uma fruta ou um animal

Regressão Múltipla

Regressão logística

Os cientistas de dados usam a regressão logística para medir a probabilidade de um evento ocorrer. A previsão é um valor entre 0 e 1, em que 0 indica um evento improvável de acontecer e 1 indica a probabilidade máxima de que esse evento aconteça. Equações logísticas usam funções logarítmicas para calcular a linha de regressão.

Veja alguns exemplos:

A probabilidade de uma vitória ou derrota em uma partida esportiva

A probabilidade de aprovação ou reprovação em um exame

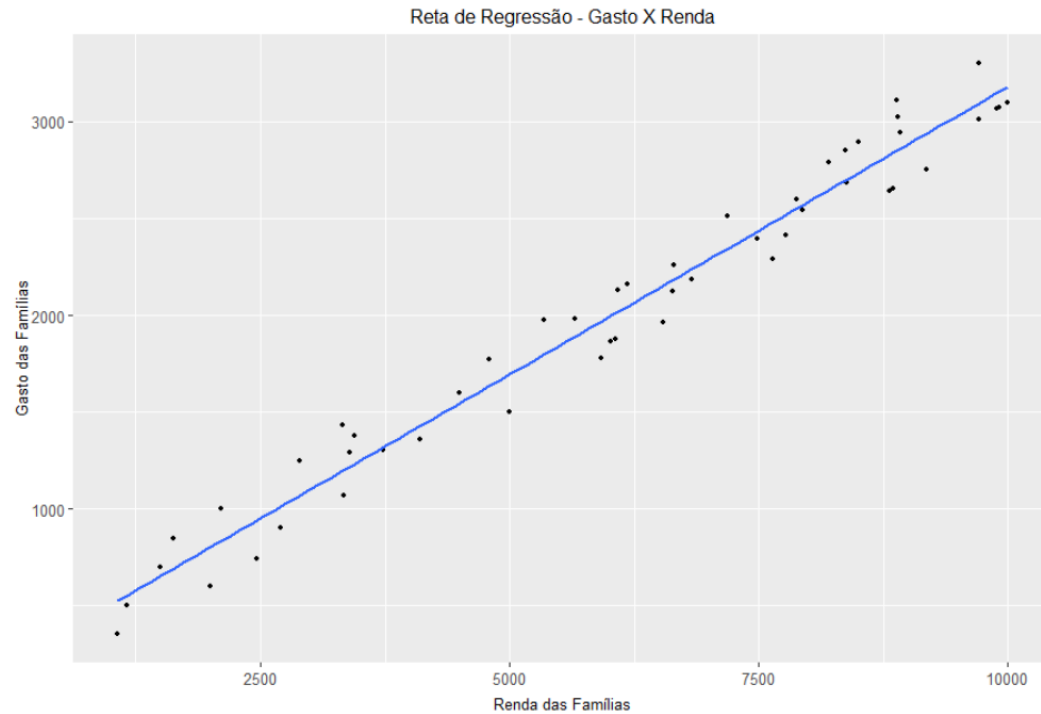
A probabilidade de uma imagem ser uma fruta ou um animal

Exemplo 1 - Gasto em função da renda

	▲	Gasto	▼	Renda	▼
1		3011		9714	
2		1305		3728	
3		1879		6062	
4		2654		8845	
5		2849		8378	

$$\text{Gasto} = \beta_0 + \beta_1 \text{Renda}$$

Exemplo 1 - Gasto em função da renda



Exemplo 1 - Gasto em função da renda

$$Y = \beta_0 + \beta_1 X + e$$

Variável
reposta

Intecepto

Coeficiente
angular

Variável
explicativa

Erro

B0 - é o valor de Y quando Xi é zero, já o B1 é o coeficiente angular e nós informa a taxa de variação e quão inclinada a reta está

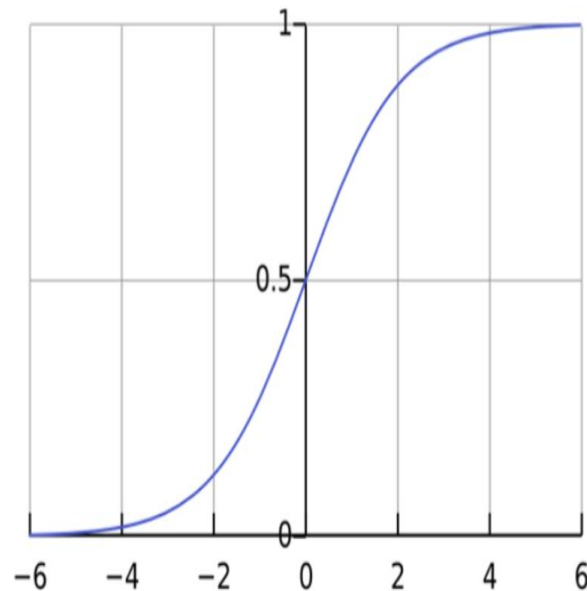
Algoritmos utilizados em aprendizado de máquina

Regressão Logística

Utilizado em problemas de classificação binária, onde o objetivo é prever uma classe ou categoria. Ela usa a função logística para modelar a relação entre as variáveis de entrada e a probabilidade de pertencer a uma classe

$$P(t) = \frac{K P_0 e^{rt}}{K + P_0 (e^{rt} - 1)}$$

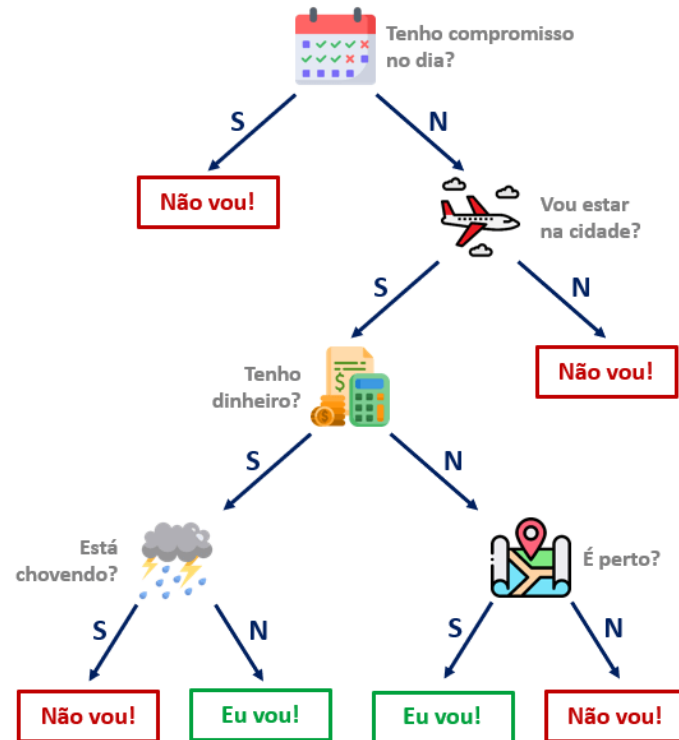
$$\lim_{t \rightarrow \infty} P(t) = K$$



Algoritmos utilizados em aprendizado de máquina

Árvore de decisão

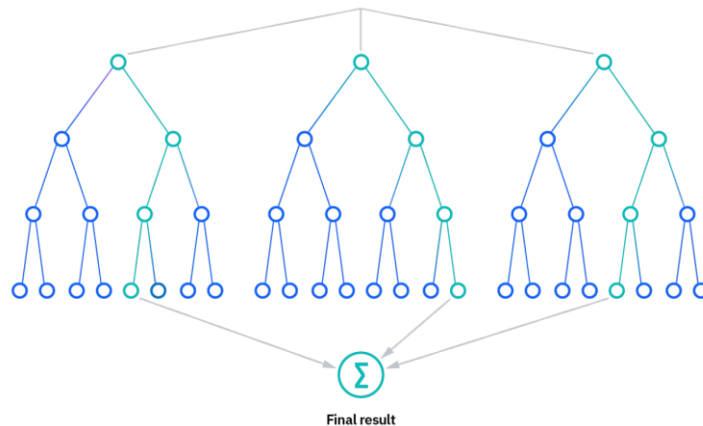
É um algoritmo de aprendizado supervisionado que cria uma estrutura na forma de árvore para tomar decisões. Cada nó interno representa um teste em uma característica, cada ramo representa um resultado possível desse teste e cada folha representa uma classe ou valor de saída



Algoritmos utilizados em aprendizado de máquina

Random Forest

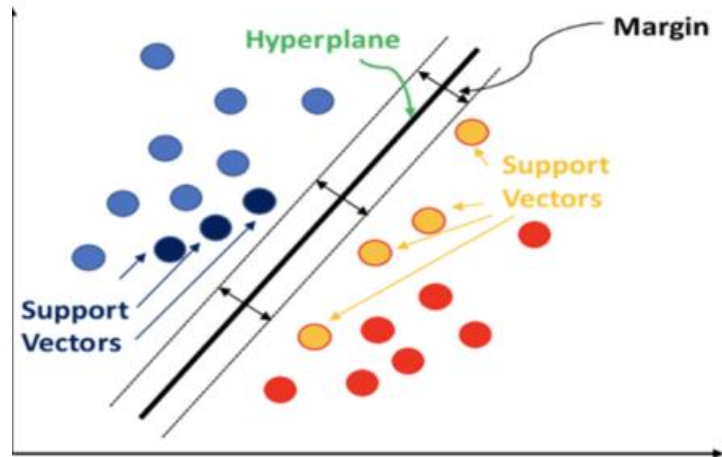
É uma técnica de aprendizado conjunto que utiliza várias árvores de decisão. Cada árvore é treinada em uma amostra aleatória dos dados e a classificação final é determinada por votação ou média das previsões das árvores



Algoritmos utilizados em aprendizado de máquina

Support Vector Machine

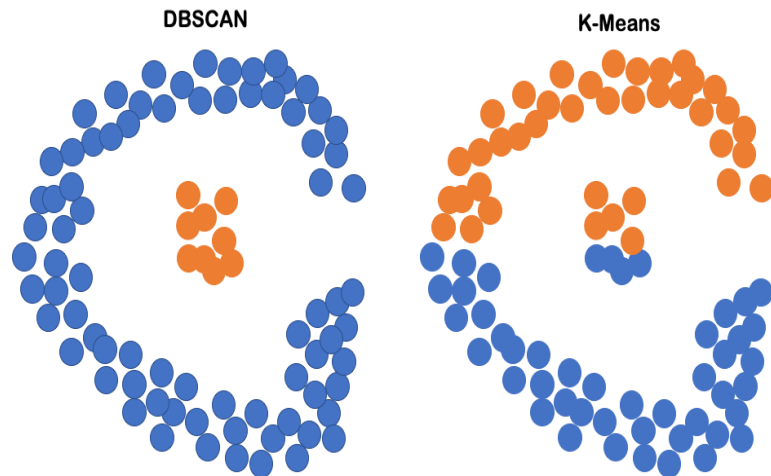
É uma técnica de aprendizado utilizada para problemas de classificação, onde o objetivo é encontrar um hiperplano de separação entre duas classes que maximize a margem entre elas. Ele pode mapear as amostras de entrada em um espaço de alta dimensão, usando funções de kernel.



Algoritmos utilizados em aprendizado de máquina

Clusterização

São utilizados para agrupar conjuntos de dados não rotulados com base na similaridade. Algoritmos populares K-means hierárquico e DBSCAN.



Análise de conglomerados - Cluster analysis

Cluster analysis - Metodologia e ferramentas utilizadas na exploração e na extração de padrões dos dados.

É normalmente associado a aprendizagem de máquina não supervisionado e utilizado para detecção de características da população e obtenção de conjuntos de agrupamento de dados, no qual cada agrupamento é denominado cluster.



Os clusters são agrupamentos compostos por um número não fixo de objetos, definidos de acordo com medidas de similaridade.

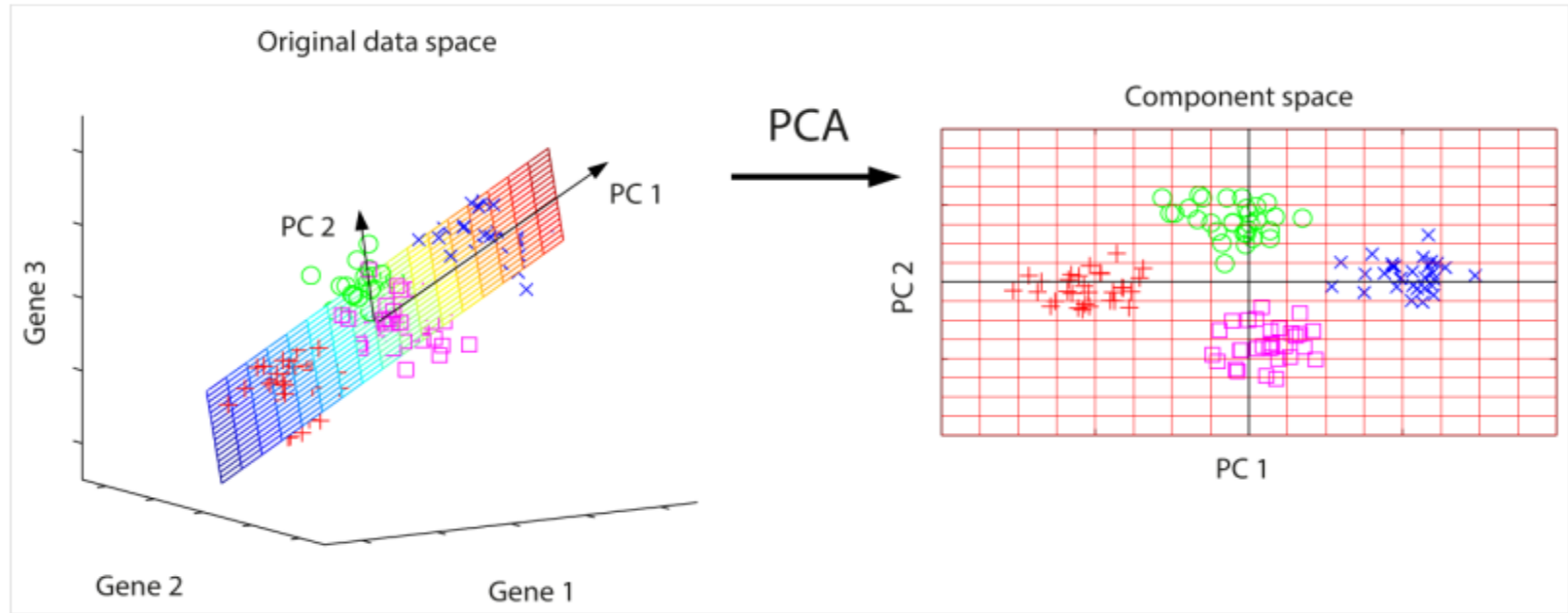
Adaptado de LI, 2019.

PCA - Análise dos componentes principais

É uma técnica de análise multivariada que utiliza princípios da álgebra linear para transformar variáveis possivelmente relacionadas em outras variáveis chamadas de componentes principais

Objetivo dessa análise é a redução de uma massa de dados minimizando perdas de informações do conjunto original.

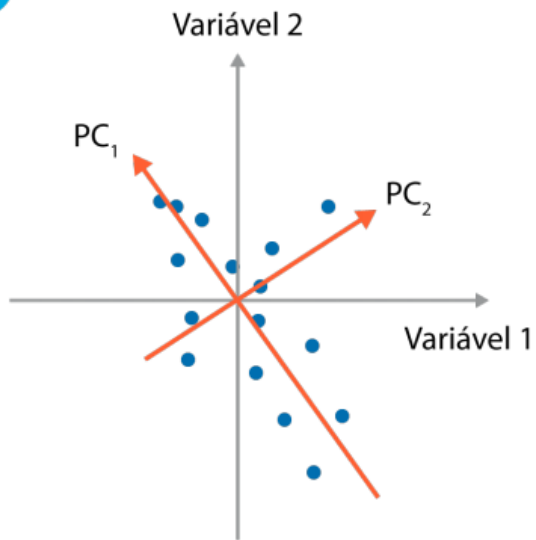
PCA - Análise dos componentes principais



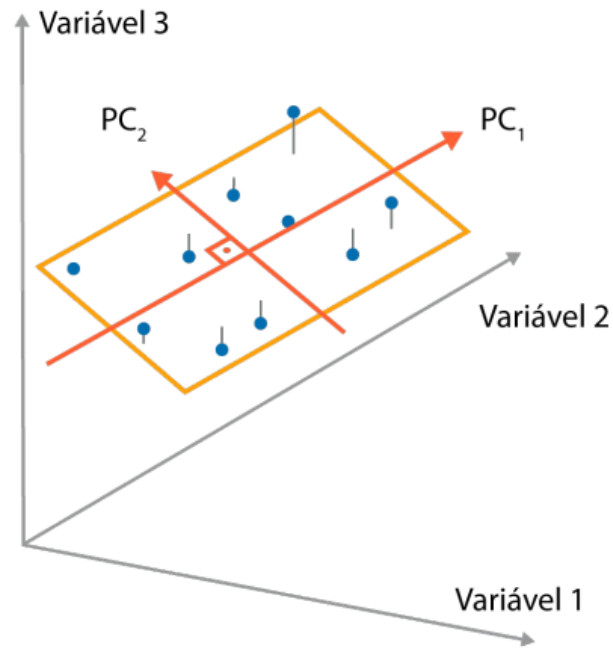
Adaptado de SCHOLZ, 2006.

PCA - Análise dos componentes principais

a



b



Exercícios

1) Criar um script para prever o nível de um tanque com vazão de 10,20,30,5,35,40 m³/ min

2) Criar um script para prever a temperatura de um forno a partir da quantidade de calor fornecida.

Q = [100,200,300,400,500,600,700]

T = [30,35,40,45,60,100,150]

3) Prever o valor de um imóvel com base na área

A=[50,100,120,300]

Valor = [180.000,300.000,375.000,600.000]

4) Prever os gastos dos clientes com base no salário.

Renda = [1200,5000,8000,10000,30000]

Gastos = [750,1500,4000,6000,18000]

Exercícios

5) Criar um script para prever a demanda de energia com base no consumo dos meses anteriores pelos clientes de 50 KW, 140KW, 30 KW, 75, KW ,123KW

Exercício 1

```
import numpy as np
from sklearn.linear_model import LinearRegression

# Dados de treinamento
vazao = np.array([10, 20, 30, 5, 35, 40]).reshape(-1, 1) # vazão de água em m3/min
nivel = np.array([50, 60, 70, 45, 80, 85]) # nível do tanque em metros

# Criar e treinar o modelo de regressão linear
modelo = LinearRegression()
modelo.fit(vazao, nivel)

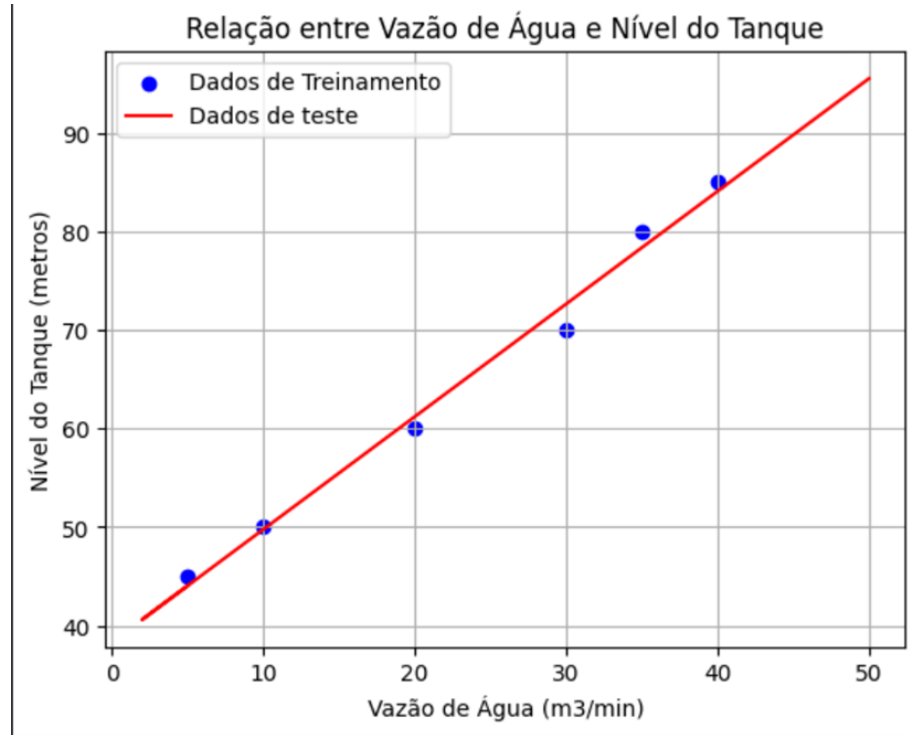
# Dados de teste
#vazao_teste = np.array([15, 25, 33]).reshape(-1, 1) # vazão de água em m3/min
vazao_teste = np.array([5, 2, 7, 30, 50]).reshape(-1, 1) # vazão de água em m3/min

# Fazer previsões com base na vazão de água
nivel_previsto = modelo.predict(vazao_teste)
```

Exercício 1

```
# Imprimir as previsões
for i in range(len(vazao_teste)):
    print(f"Vazão: {vazao_teste[i][0]} m3/min - Nível previsto: {nivel_previsto[i]} metros")
plt.scatter(vazao, nivel, color='blue', label='Dados de Treinamento')
plt.plot(vazao_teste,nivel_previsto, color = 'red',label='Dados de teste')
plt.xlabel('Vazão de Água (m3/min)')
plt.ylabel('Nível do Tanque (metros)')
plt.title('Relação entre Vazão de Água e Nível do Tanque')
plt.legend()
plt.grid(True)
plt.show()
```

Exercício 1



Código exemplo árvore de decisão

```
import matplotlib.pyplot as plt  
from sklearn import datasets  
from sklearn import tree
```

```
iris = datasets.load_iris()
```


Código exemplo árvore de decisão

```
data = iris.data
target = iris.target
data = iris.data
target
```

```
data = iris.data
data
# Imprima as primeiras linhas dos dados
print("Dados:")
print(data[:5]) # Imprime as cinco
primeiras linhas dos dados

# Imprima as primeiras linhas do alvo
print("\nAlvo:")
print(target[:5]) # Imprime as cinco
primeiras linhas do alvo
```

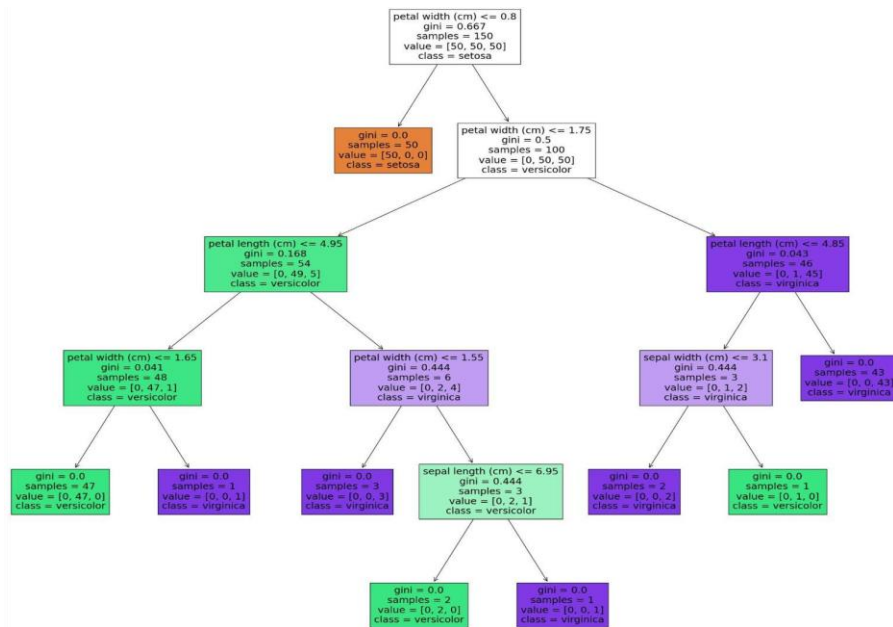
Código exemplo árvore de decisão

```
X,y = iris.data,iris.target  
fig = plt.figure(figsize=(30,25))  
tree.plot_tree(classificador,feature_names=iris.feature_names,  
class_names=iris.target_names,filled=True)
```

Código exemplo árvore de decisão

```
X,y = iris.data,iris.target
classificador = tree.DecisionTreeClassifier()
classificador.fit(X,y)
fig = plt.figure(figsize=(30,25))
tree.plot_tree(classificador,feature_names=
iris.feature_names,class_names=iris.target_names,filled=True)
plt.savefig('arvore.jpg')
```

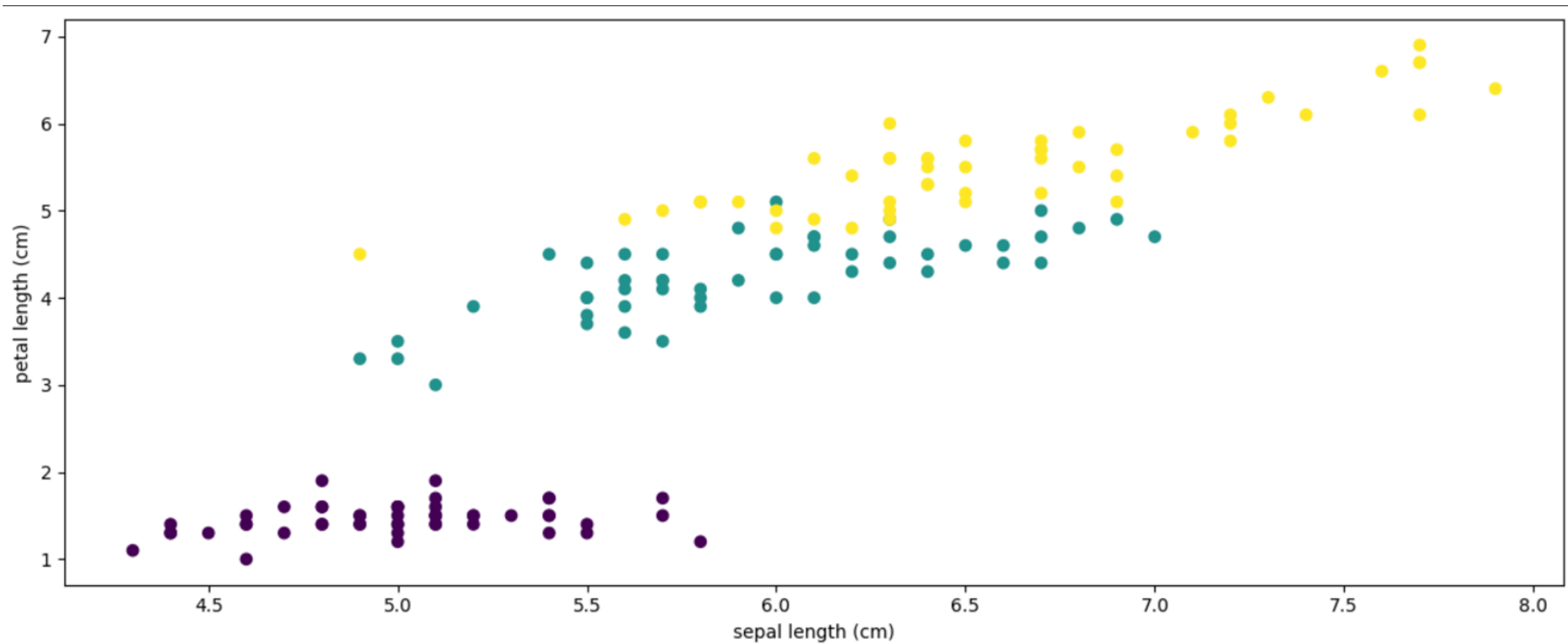
Código exemplo árvore de decisão



Código exemplo árvore de decisão

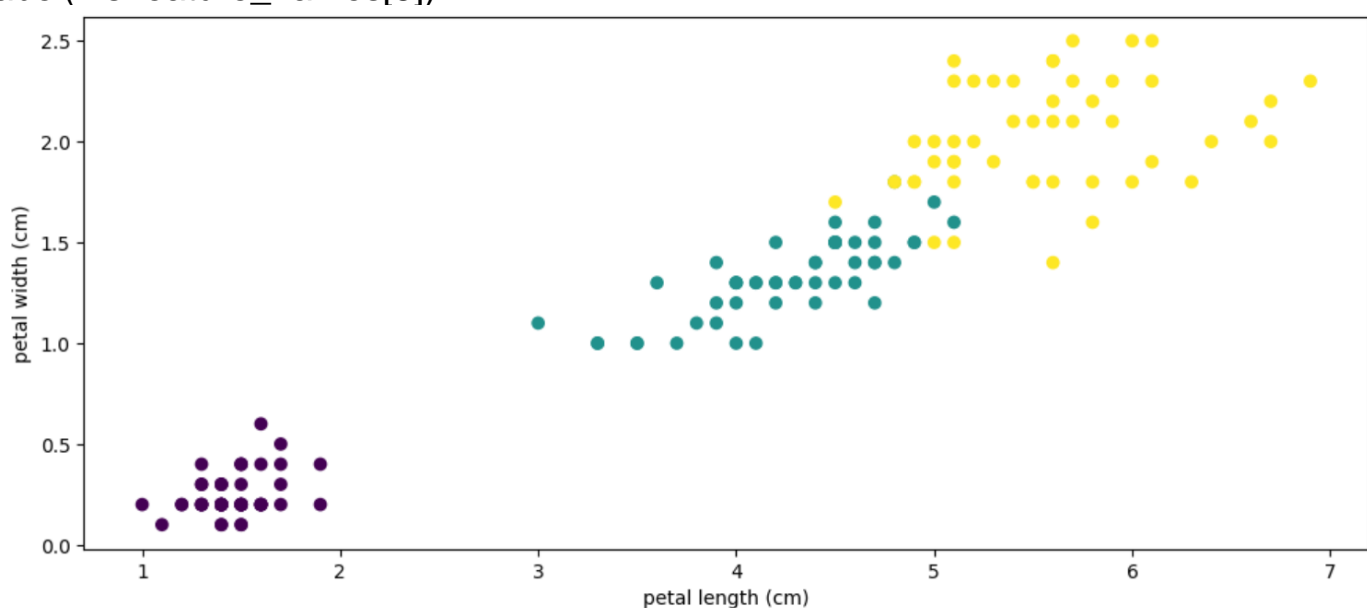
```
plt.figure(figsize=(15,10))  
plt.scatter(iris.data[:,0],iris.data[:,2],c = iris.target)  
plt.xlabel(iris.feature_names[0])  
plt.ylabel(iris.feature_names[2])  
plt.tight_layout()  
plt.show()
```

Código exemplo árvore de decisão



Código exemplo árvore de decisão

```
plt.figure(figsize=(15,10))  
plt.scatter(iris.data[:,2],iris.data[:,3],c=iris.target)  
plt.xlabel(iris.feature_names[2])  
plt.ylabel(iris.feature_names[3])
```



Exercícios

- 1) Criar um script para classificar uma bebida como boa, ruim, péssima a partir das notas dos clientes
- 2) Criar um script para classificar um filme como bom, ruim, péssimo a partir das notas dos clientes

Obrigado!

Prof. Me Daniel Vieira

Email: danielvieira2006@gmail.com

Linkedin: Daniel Vieira

Instagram: Prof daniel.vieira95

