

A decorative network diagram in the top-left corner, featuring a complex web of interconnected nodes and lines. Some nodes are highlighted with blue circles, and others with blue dots. The lines are thin and gray, creating a mesh-like structure.

Ciência de dados

Revisão

A decorative network diagram in the bottom-right corner, similar to the one in the top-left. It shows a network of nodes and lines, with some nodes highlighted by blue circles and others by blue dots.

Olá!

Me chamo Audrey

Graduanda em ciência da
computação e monitora da
disciplina ciência de dados.



@audreyemmely



@audrey-vasconcelos

Conteúdo

➤ **ÁRVORE DE DECISÃO**

é uma das abordagens de modelagem preditiva utilizada em estatística, mineração de dados e aprendizagem de máquinas

➤ **REDES NEURAIS**

são redes multicamadas de neurônios que usamos para classificar as coisas, fazer previsões, etc

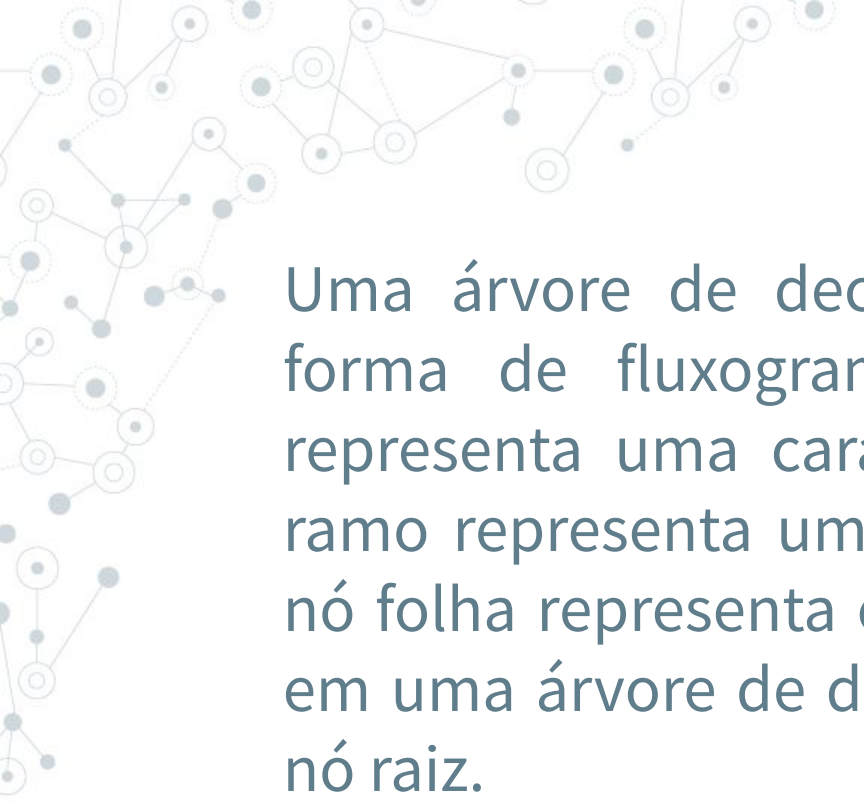
➤ **MÁQUINAS DE VETORES DE SUPORTE**

é um algoritmo de aprendizagem de máquina utilizado tanto para classificação quanto para regressão




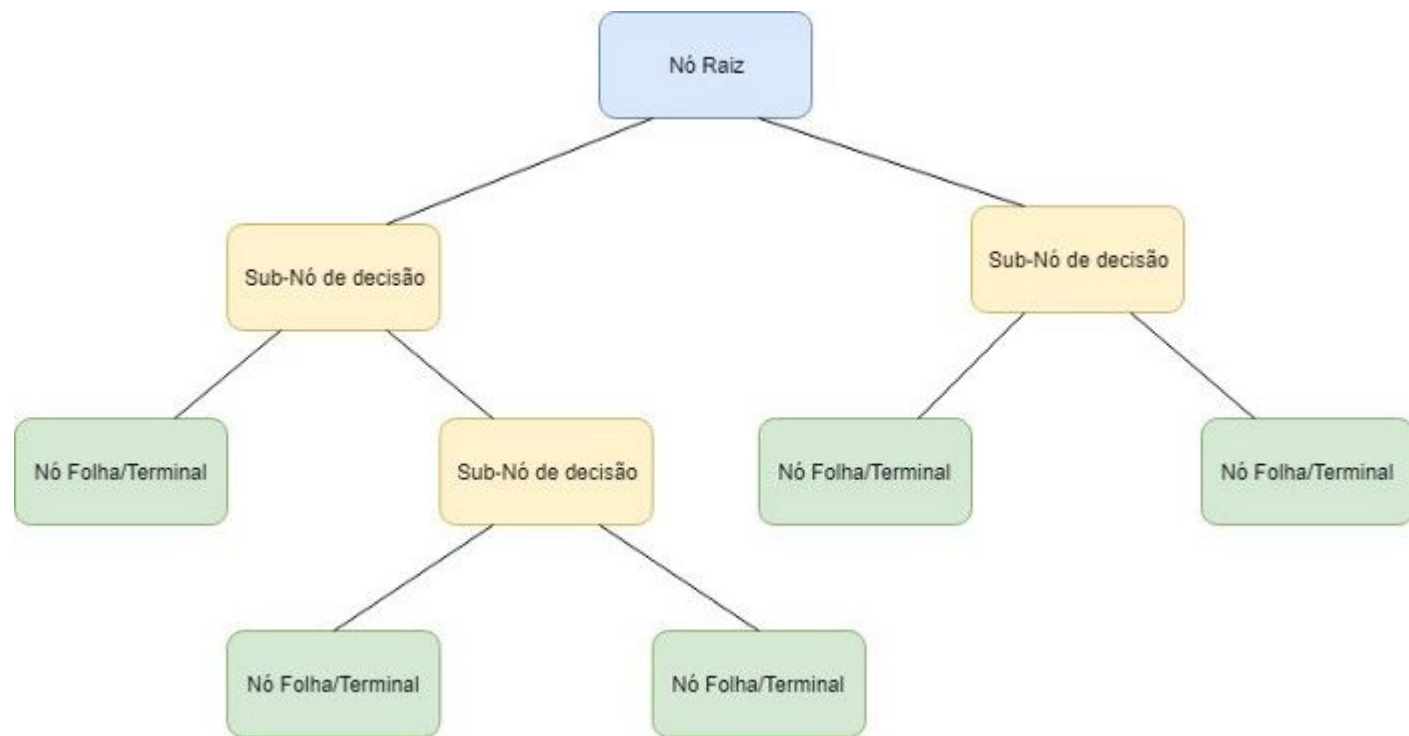
1.

Árvore de decisão

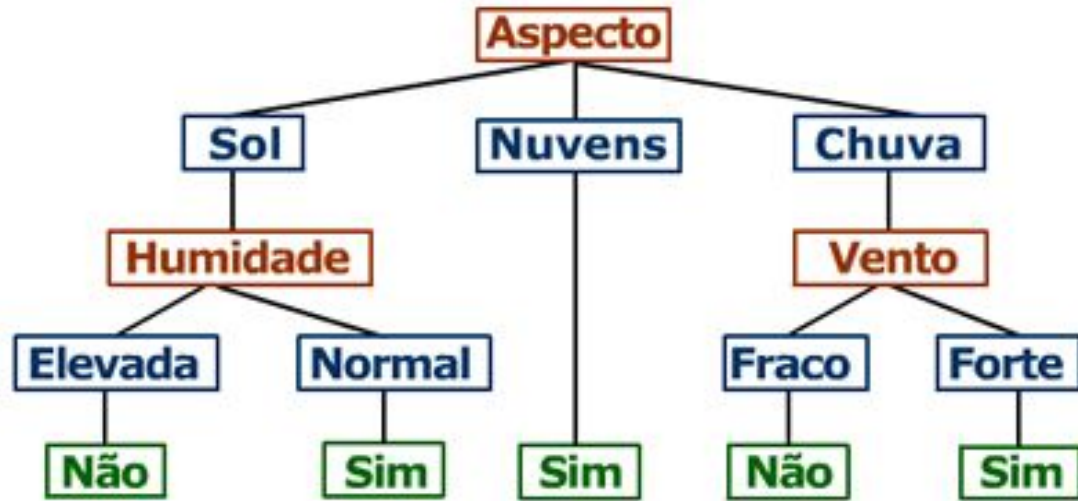
A decorative network diagram in the top-left corner, consisting of various sized circles (nodes) connected by thin lines, some solid and some dashed, in a light gray color.

Uma árvore de decisão é uma estrutura em forma de fluxograma onde um nó interno representa uma característica (ou atributo), o ramo representa uma regra de decisão, e cada nó folha representa o resultado. O nó mais alto em uma árvore de decisão é conhecido como o nó raiz.

A decorative network diagram in the bottom-right corner, similar to the one in the top-left, with nodes and connecting lines in a light gray color.



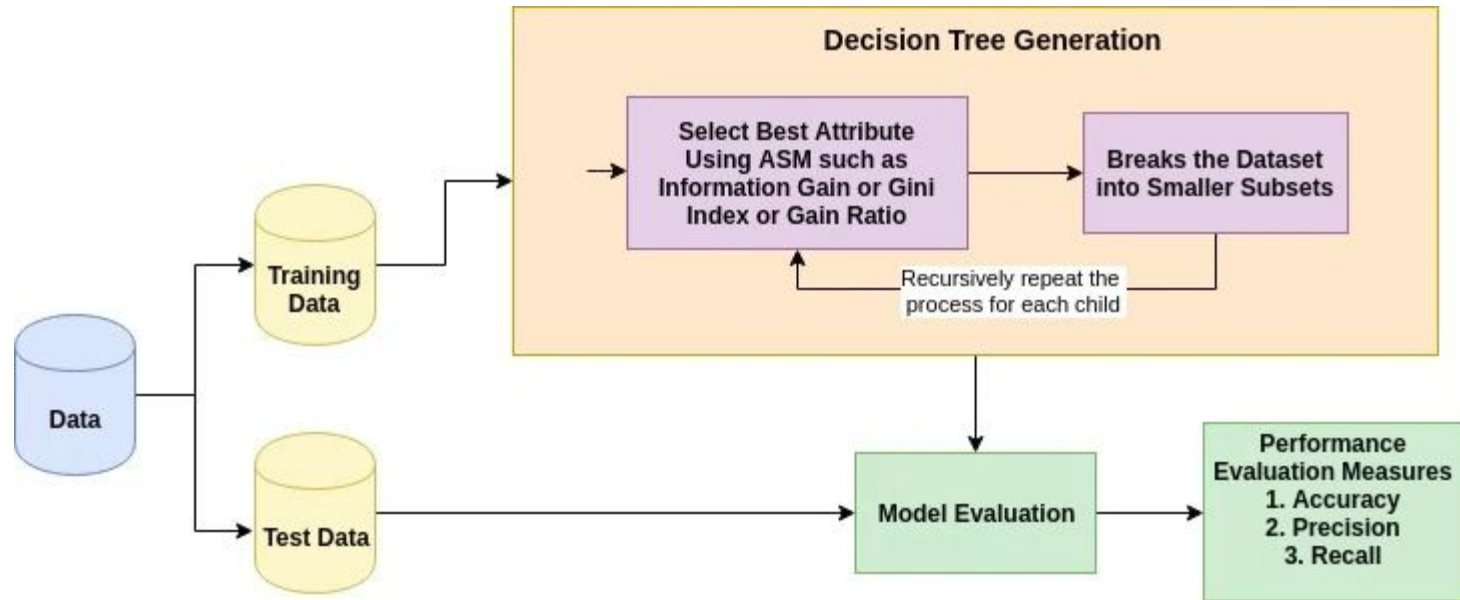
Árvore de decisão para saber se vai jogar tênis



Como funciona?

A ideia básica por trás de qualquer algoritmo de árvore de decisão é a seguinte:

- ◎ Seleciona o melhor atributo usando as Medidas de Seleção de Atributos (ASM) para dividir os registros;
- ◎ Faz desse atributo um nó de decisão e divide o conjunto de dados em subconjuntos menores;
- ◎ Inicia a construção da árvore repetindo este processo recursivamente para cada filho até que uma das condições seja igualada:
 - Todas as tuplas pertencem ao mesmo valor de atributo.
 - Não há mais nenhum atributo restante.
 - Não há mais instâncias.



Medidas de Seleção de Atributos

Ganho de informação

Este método é o principal método usado para construir árvores de decisão. Ele reduz as informações e o número de testes que são necessários para classificar as tuplas. O atributo com o maior ganho de informação é selecionado.

Taxa de ganho

O ganho de informação pode às vezes resultar em porções inúteis para classificação. No entanto, a taxa de ganho divide o conjunto de dados de treinamento em partições e considera o número de tuplas do resultado em relação ao total de tuplas. O atributo com a maior taxa de ganho é usado como um atributo de divisão.

Índice de Gini

O índice de Gini considera uma divisão binária para cada atributo. Você pode calcular uma soma ponderada da impureza de cada partição.

Aplicações

Avaliar as oportunidades de crescimento em perspectiva

Dados históricos sobre vendas são usados em árvores de decisão que podem levar a mudanças radicais na estratégia de um negócio para ajudar na expansão e crescimento.

Usar dados demográficos para encontrar potenciais clientes

Na ausência de árvores de decisão, a empresa pode gastar seu marketing sem uma demografia específica em mente, o que afetará suas receitas gerais.

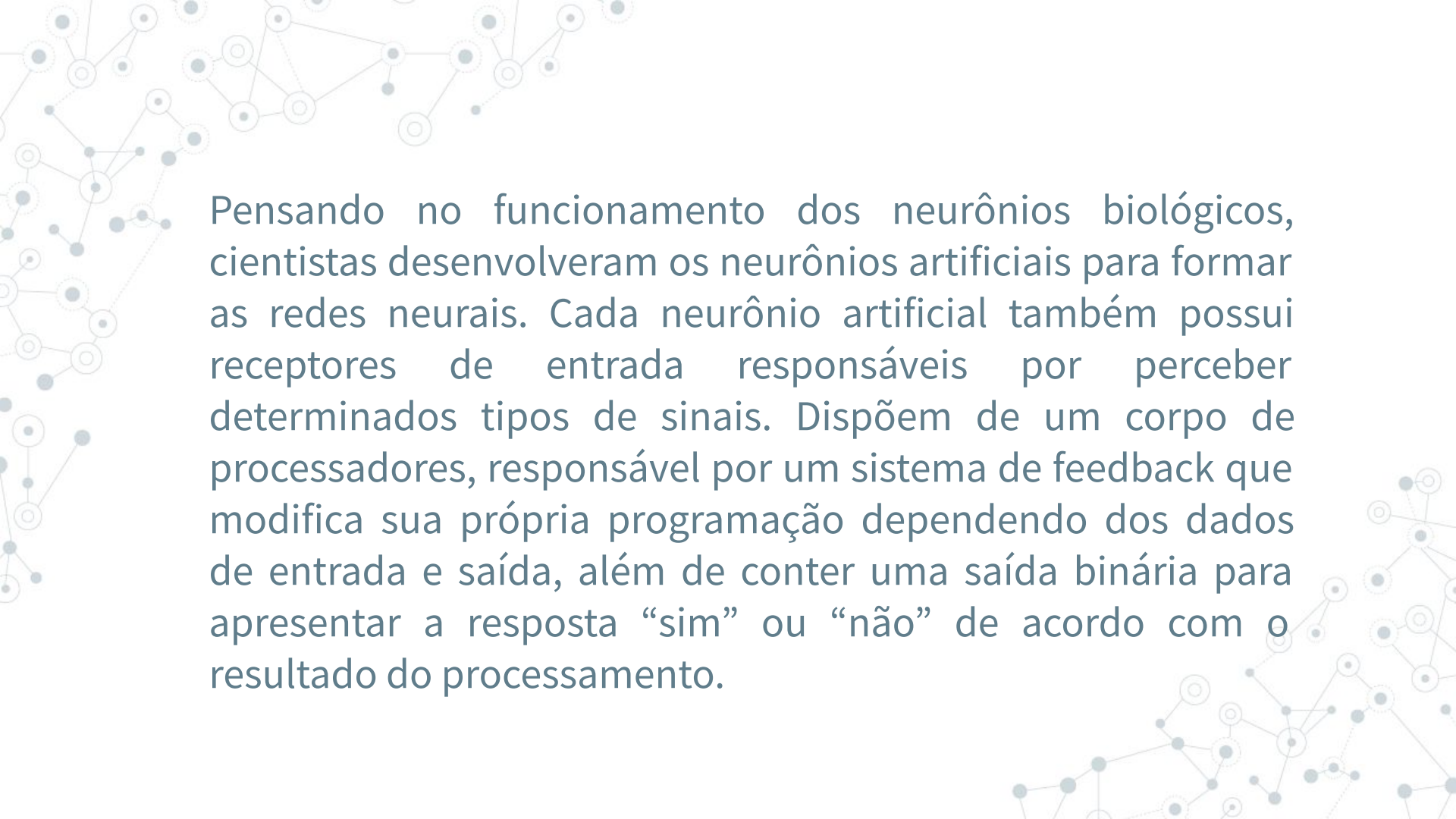
Serve como uma ferramenta de apoio em vários campos

Prever a probabilidade de um cliente inadimplente em um empréstimo, pode ajudar os credores a avaliar a qualidade de crédito de um cliente para evitar perdas, dentre outros.

A decorative network diagram in the top-left corner, featuring a complex web of interconnected nodes and lines. The nodes are represented by small circles, some of which are larger and have concentric circles, suggesting different levels or types of nodes. The lines are thin and gray, connecting the nodes in a non-linear fashion.

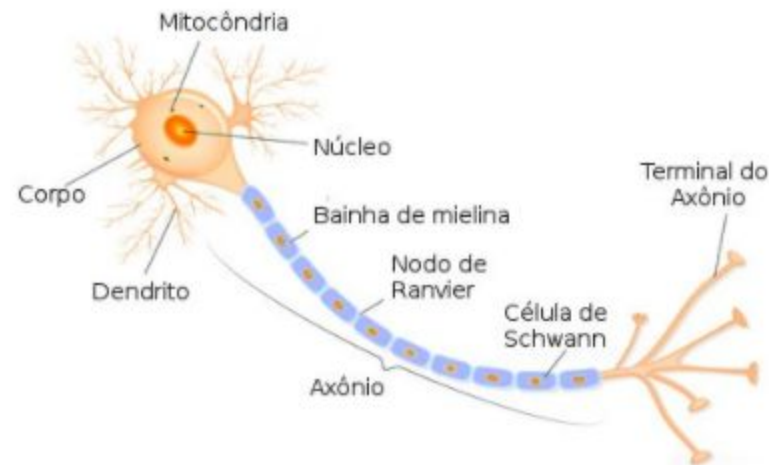
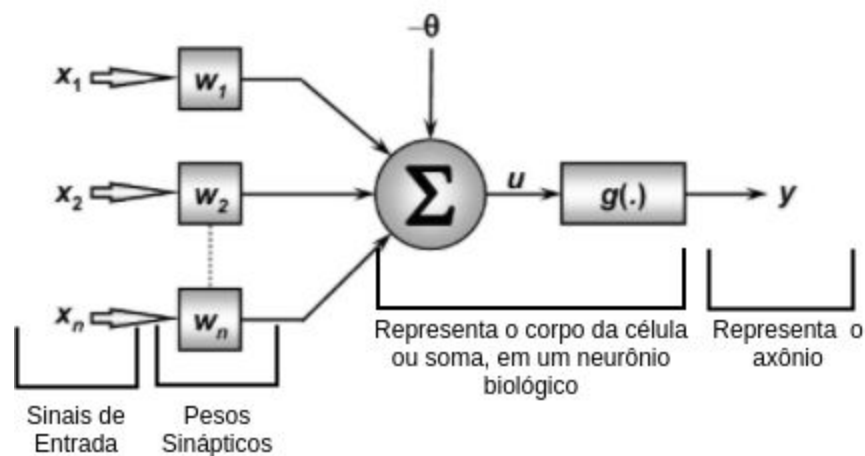
2.

Redes neurais



Pensando no funcionamento dos neurônios biológicos, cientistas desenvolveram os neurônios artificiais para formar as redes neurais. Cada neurônio artificial também possui receptores de entrada responsáveis por perceber determinados tipos de sinais. Dispõem de um corpo de processadores, responsável por um sistema de feedback que modifica sua própria programação dependendo dos dados de entrada e saída, além de conter uma saída binária para apresentar a resposta “sim” ou “não” de acordo com o resultado do processamento.

Neurônio Artificial x Neurônio Biológico



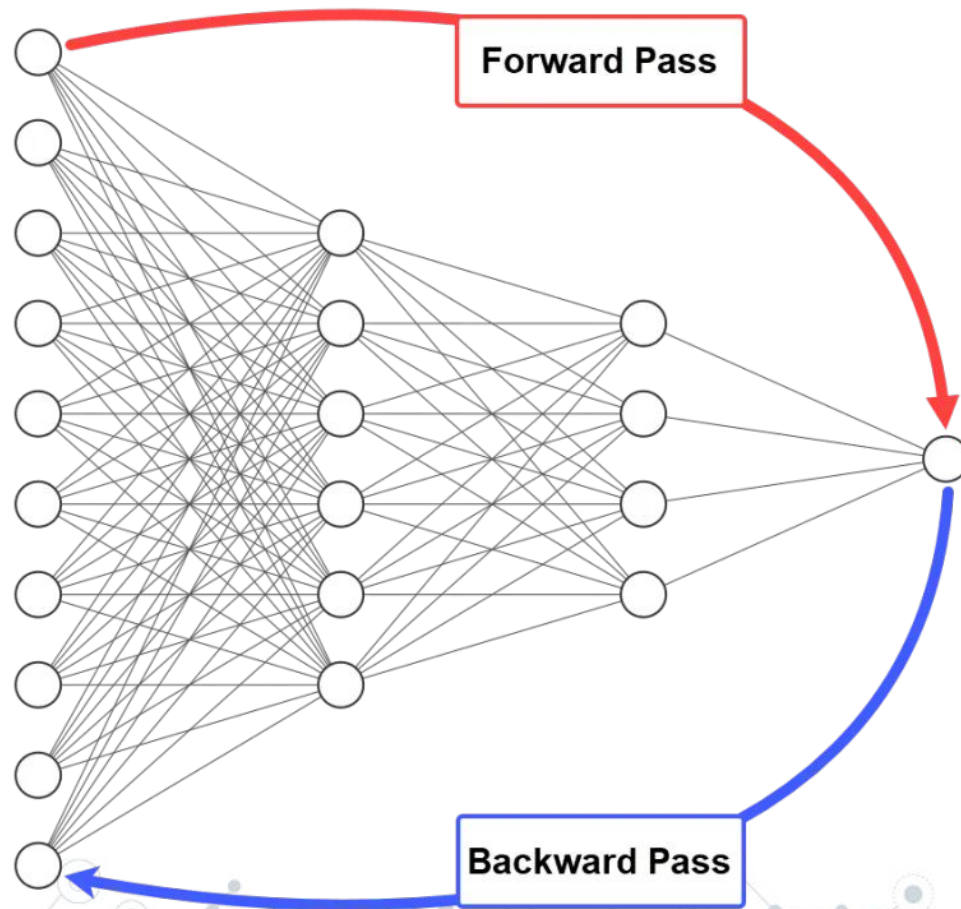
Tipos principais de redes neurais

Feedforward (Forward propagation)

A rede neural Feedforward é uma rede que não é recursiva. Os neurônios desta camada só foram conectados aos neurônios da camada seguinte, e não formam um ciclo. Em Feedforward, os sinais viajam em apenas uma direção, em direção à camada de saída.

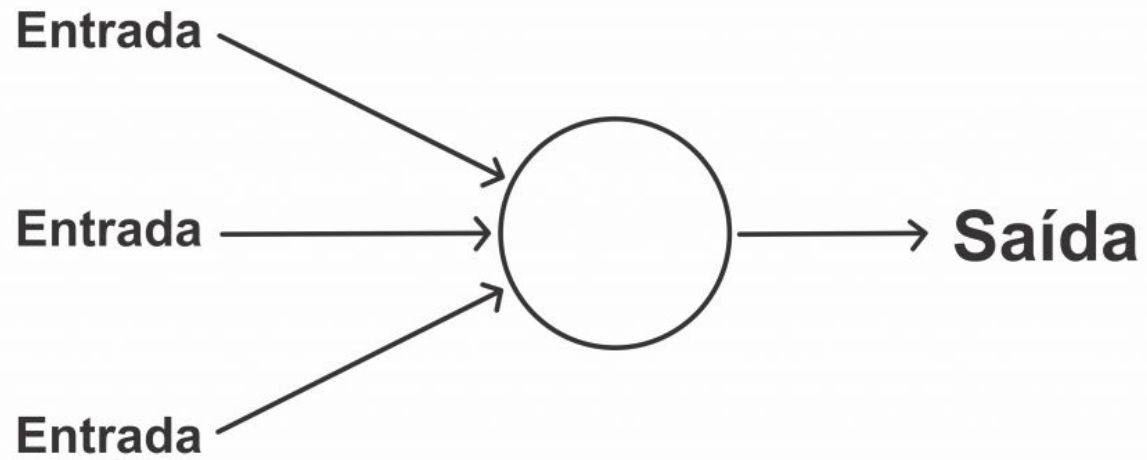
Feedback (Back propagation)

As redes neurais de Feedback contêm ciclos. Os sinais viajam em ambas as direções, introduzindo loops na rede. A rede neural de Feedback também é conhecida como redes neurais recorrentes.



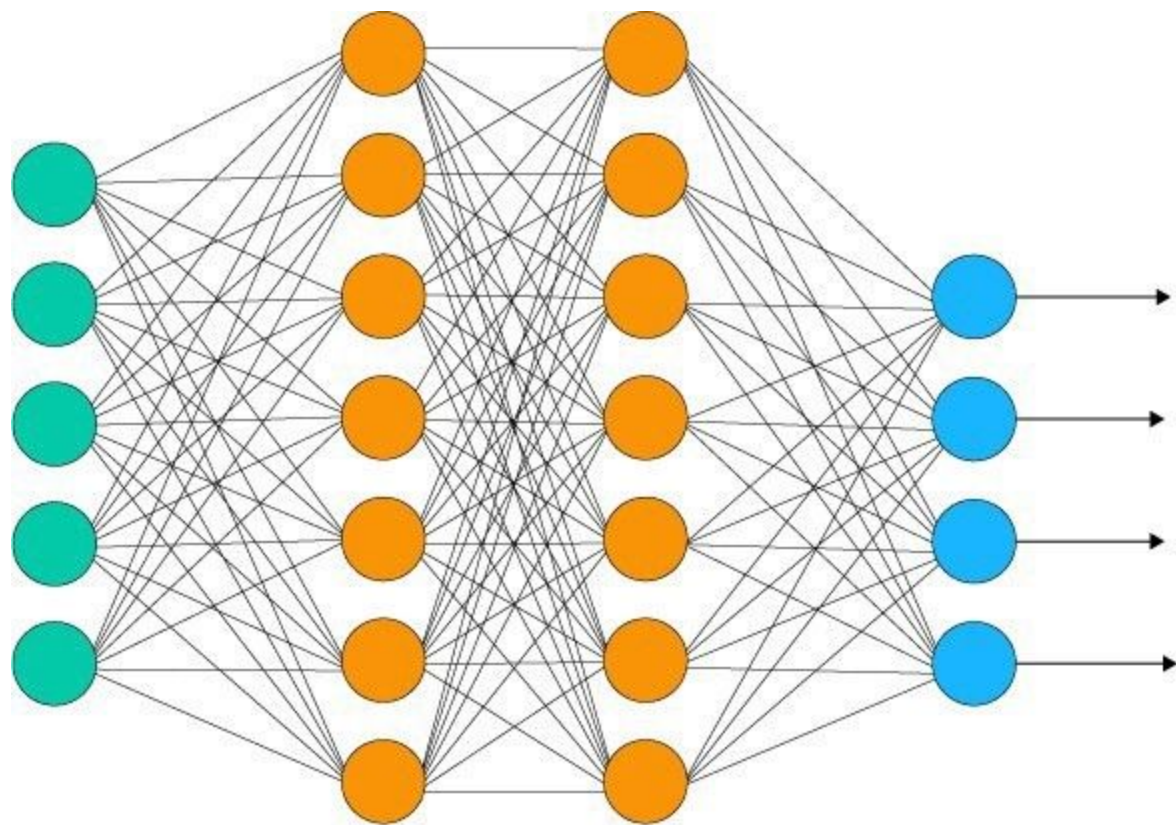
O neurônio básico: Perceptron

É uma das arquiteturas mais simples de redes neurais e lida com um único neurônio, utilizando padrões lineares para obtenção dos resultados. Cada entrada está conectada a um peso sináptico. O valor de cada neurônio de saída é linear, por conta disso, a arquitetura de um Perceptron é incapaz de aprender padrões complexos. Este modelo já não é muito utilizado, pois existem outras arquiteturas mais eficientes.



Multi-Layer Perceptron

O Multi-Layer Perceptron (MLP) parte do mesmo princípio do Perceptron, porém mais desenvolvido e com maior número de neurônios. Eles surgem com o intuito de lidar com os problemas não linearmente separáveis, adicionando camadas de neurônio ocultas ao modelo.



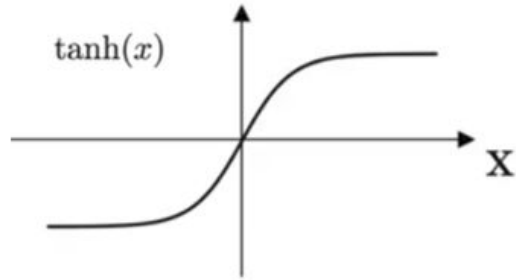
● Input Layer

● Hidden Layer

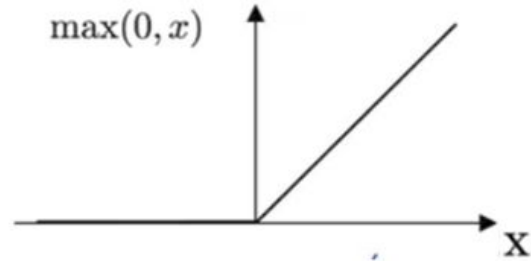
● Output Layer

Funções de ativação

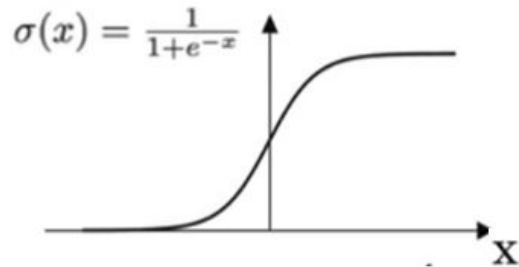
Hyper Tangent Function



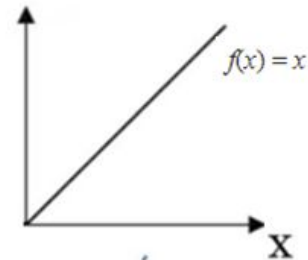
ReLU Function



Sigmoid Function



Identity Function



Como funciona?

Uma rede neural é treinada ajustando os pesos dos neurônios de acordo com o desempenho da rede com base nos exemplos de entrada. Por exemplo, se a rede classificar corretamente uma imagem, os pesos que contribuem para a resposta correta são aumentados, enquanto outros pesos são diminuídos.

Aplicações

Reconhecimento de padrões

As redes neurais são muito adequadas para problemas de reconhecimento de padrões, tais como reconhecimento facial, detecção de objetos, reconhecimento de impressões digitais, etc.

Processamento de linguagem natural

Oferecem uma ampla gama de aplicações em tarefas de Processamento de Linguagem Natural, tais como classificação de texto, reconhecimento de fala e verificação ortográfica.

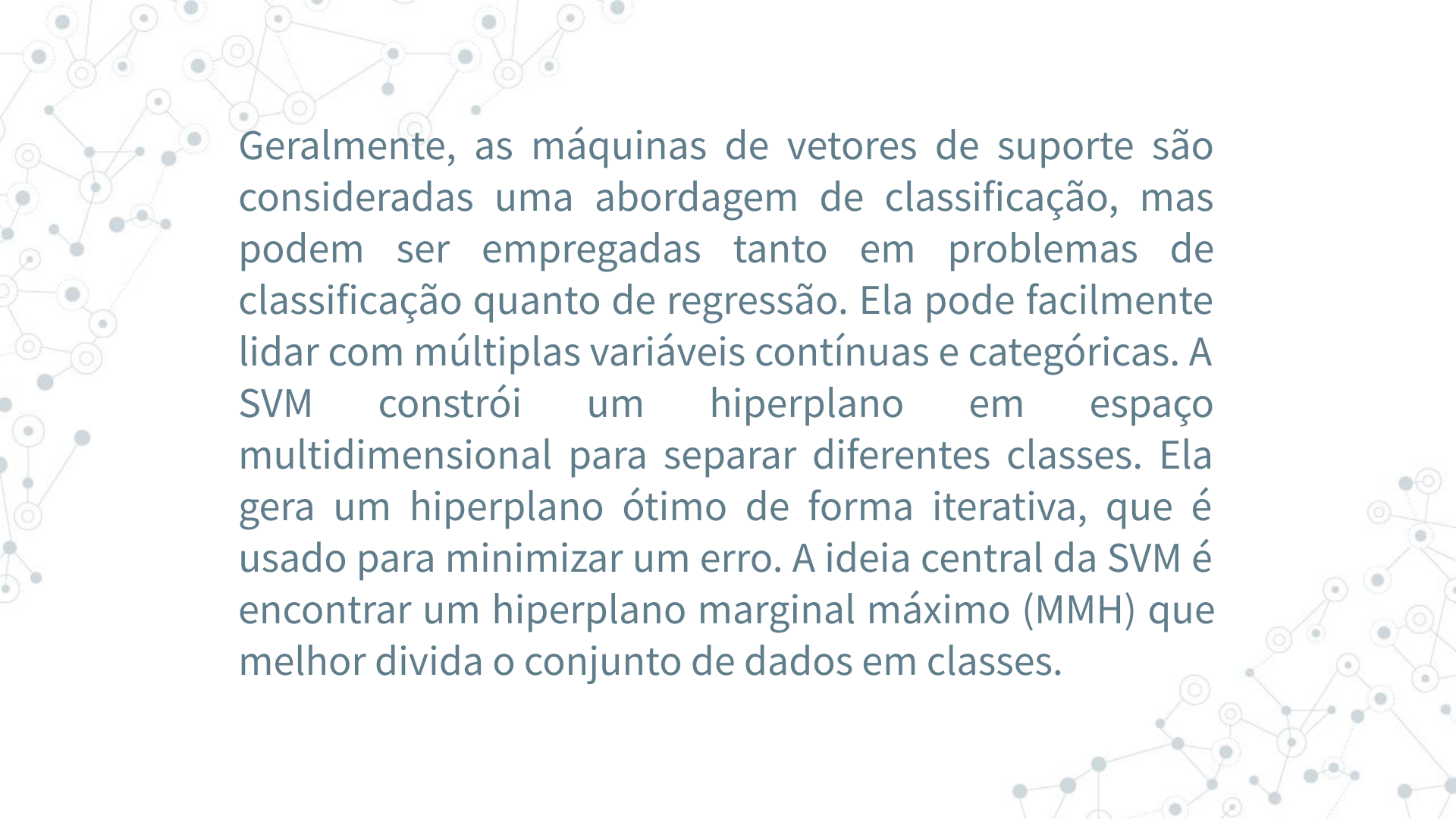
Predição de série temporal

Podem ser usadas para prever problemas de séries temporais, tais como preço de ações, previsão do tempo.

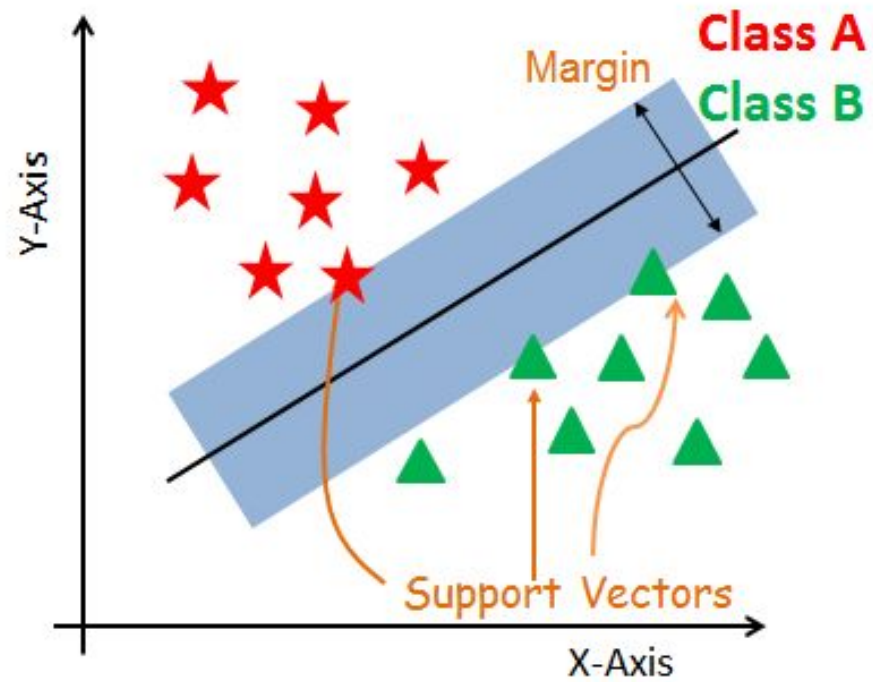
A decorative network diagram in the top-left corner, featuring a complex web of interconnected nodes and lines. The nodes are represented by small circles, some of which are larger and have concentric circles, suggesting different levels or types of nodes. The lines are thin and gray, connecting the nodes in a non-linear fashion.

3.

Máquinas de vetores de suporte



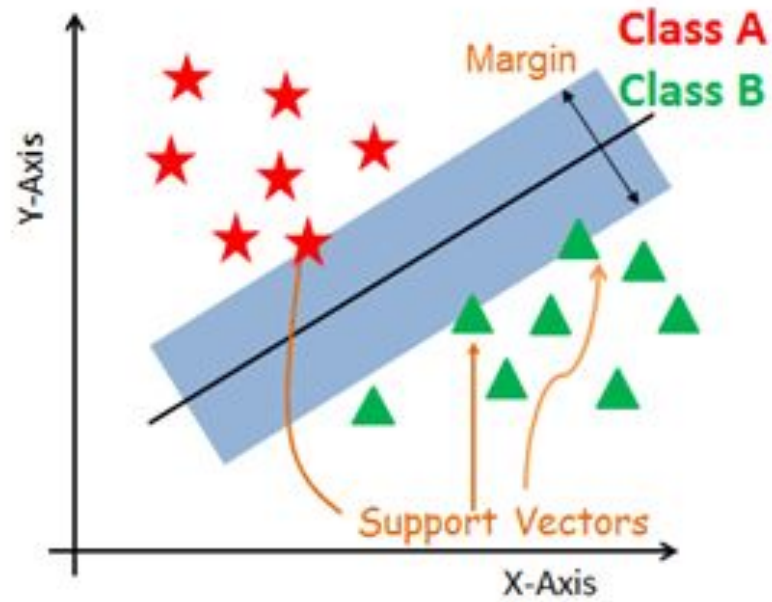
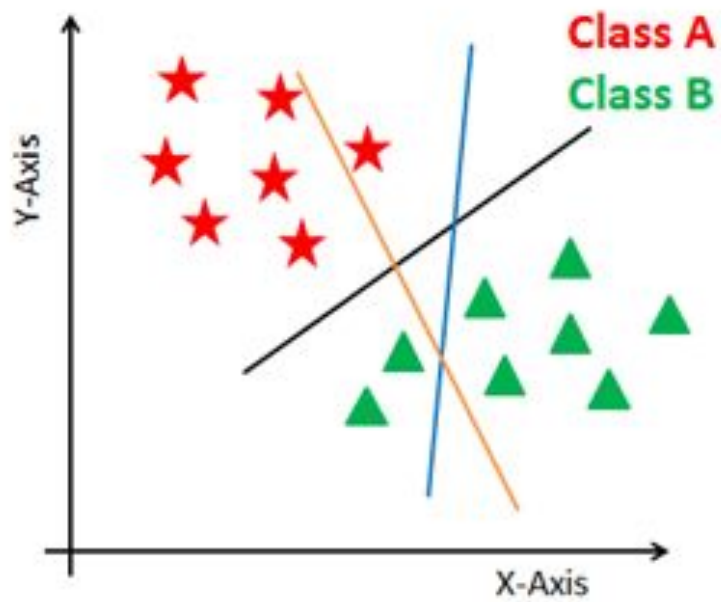
Geralmente, as máquinas de vetores de suporte são consideradas uma abordagem de classificação, mas podem ser empregadas tanto em problemas de classificação quanto de regressão. Ela pode facilmente lidar com múltiplas variáveis contínuas e categóricas. A SVM constrói um hiperplano em espaço multidimensional para separar diferentes classes. Ela gera um hiperplano ótimo de forma iterativa, que é usado para minimizar um erro. A ideia central da SVM é encontrar um hiperplano marginal máximo (MMH) que melhor divida o conjunto de dados em classes.



Como funciona?

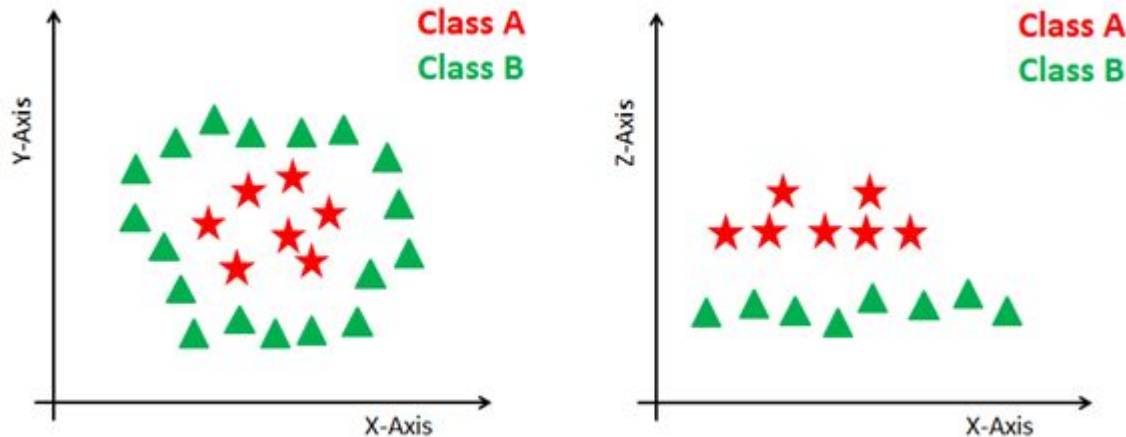
O objetivo principal é segregar o conjunto de dados fornecido da melhor maneira possível. A distância entre os pontos mais próximos é conhecida como margem. O objetivo é selecionar um hiperplano com a maior margem possível entre os vetores de suporte no conjunto de dados fornecido. O SVM procura o hiperplano marginal máximo nas seguintes etapas:

1. Gera hiperplanos que segregam as classes da melhor maneira;
2. Seleciona o hiperplano com a maior segregação dos pontos de dados mais próximos.



Lidando com planos não-lineares e inseparáveis

Alguns problemas não podem ser resolvidos usando hiperplano linear, conforme mostrado na figura abaixo (lado esquerdo).



Em tal situação, o SVM usa o truque do kernel para transformar o espaço de entrada em um espaço dimensional superior, conforme mostrado à direita. Os pontos de dados são plotados nos eixos x e z (z é a soma quadrada de x e y : $z = x^2 + y^2$). Agora você pode segregar facilmente esses pontos usando a separação linear.

Aplicações

Bioinformática

Inclui a classificação de proteínas e classificação de câncer. Usa SVM para identificar a classificação de genes, pacientes com base em genes e outros problemas biológicos.

Classificação de imagens

O uso de SVMs proporciona melhor precisão na busca de classificação de imagens, em comparação com as técnicas de busca tradicionais baseadas em consultas.

Reconhecimento de escrita à mão

Utiliza-se SVM para reconhecer caracteres escritos à mão.

Referências

Árvores de Decisão. Disponível em:

<<http://web.tecnico.ulisboa.pt/ana.freitas/bioinformatics.ath.cx/bioinformatics.ath.cx/indexf23d.html?id>>. Acesso em: 10 abr. 2021.

Deep Learning: Compreenda o que são e como funcionam as Redes Neurais. Arbit. Disponível em:

<<https://blog.arbit.com.br/deep-learning-redes-neurais/>>. Acesso em: 11 abr. 2021.

NAVLANI, Avinash. Decision Tree Classification in Python. Disponível em:

<<https://www.datacamp.com/community/tutorials/decision-tree-classification-python>>. Acesso em: 10 abr. 2021.

NAVLANI, Avinash. Neural Network Models in R. Datacamp. Disponível em:

<<https://www.datacamp.com/community/tutorials/neural-network-models-r>>. Acesso em: 10 abr. 2021.

NAVLANI, Avinash. Support Vector Machines with Scikit-learn. Datacamp. Disponível em:

<<https://www.datacamp.com/community/tutorials/svm-classification-scikit-learn-python#svm>>. Acesso em: 10 abr. 2021.

Real-Life Applications of SVM (Support Vector Machines). Data Flair. Disponível em:

<<https://data-flair.training/blogs/applications-of-svm/#:~:text=We%20use%20SVM%20for%20identifying,recognize%20handw,ritten%20characters%20used%20widely.>> . Acesso em: 10 abr. 2021.

STANKEVIX, Gabriel. Árvore de Decisão em R. Medium. Disponível em:

<<https://medium.com/@gabriel.stankevix/arvore-de-decis%C3%A3o-em-r-85a449b296b2>>. Acesso em: 10 abr. 2021.