

# Unidade III

## 7 APRENDIZADO DE MÁQUINA

Aprendizado de máquina (machine learning) é uma subárea da inteligência artificial que se concentra em desenvolver técnicas e algoritmos que permitem que as máquinas aprendam com dados, sem serem explicitamente programadas. Ela é dividida em três categorias principais: aprendizado supervisionado, aprendizado não supervisionado e aprendizado por reforço.



### Lembrete

Aprendizado é o processo pelo qual um sistema de IA é capaz de melhorar seu desempenho em uma determinada tarefa ao longo do tempo, com base em exemplos ou experiência prévia. Isso envolve o uso de algoritmos e técnicas de machine learning (aprendizado de máquina), que permitem que o sistema aprenda a partir de dados e exemplos, e melhore sua capacidade de realizar uma tarefa específica.

O aprendizado de máquina manipula uma grande quantidade de dados para realizar o treinamento de um sistema computacional, possibilitando que o sistema reconheça determinados padrões em um grupo de dados. Os métodos utilizados geralmente são complexos; dessa forma, é muito importante o planejamento, análise e preparação dos dados (BISHOP, 2006).

Aprendizado supervisionado acontece quando um algoritmo é treinado com dados rotulados, ou seja, aqueles que já têm uma classificação ou saída desejada. Seu objetivo é fazer com que o algoritmo aprenda a generalizar essas saídas para novos dados não vistos anteriormente. Exemplos de técnicas de aprendizado supervisionado incluem regressão linear, árvores de decisão e redes neurais.

Aprendizado não supervisionado ocorre quando um algoritmo é treinado com dados não rotulados. Seu objetivo é explorar os dados e encontrar padrões ou estruturas escondidas neles. Exemplos de técnicas de aprendizado não supervisionado incluem clustering, detecção de outliers e redução de dimensionalidade.

Aprendizado por reforço se dá quando um algoritmo é treinado por meio de uma série de ações e recompensas. Seu objetivo é fazer com que o algoritmo aprenda a tomar ações que maximizem a recompensa. Exemplos de técnicas de aprendizado por reforço incluem Q-learning e algoritmos de política ótima.

O aprendizado de máquina está se tornando cada vez mais importante devido à proliferação de dados e à necessidade de automatizar tarefas e processos. Ele está sendo usado em áreas como reconhecimento de fala, processamento de linguagem natural, visão computacional, saúde, finanças e segurança cibernética. Com o aumento da capacidade de processamento e a disponibilidade de grandes quantidades de dados, o aprendizado de máquina está cada vez mais acessível e transformando a forma como as empresas e os indivíduos tomam decisões.

### 7.1 Aprendizado supervisionado

Aprendizado supervisionado é uma categoria de aprendizado de máquina em que o algoritmo é treinado com dados rotulados, ou seja, aqueles que já têm classificação ou saída desejada. Seu objetivo é fazer com que o algoritmo aprenda a generalizar essas saídas para novos dados não vistos anteriormente. Exemplos de técnicas de aprendizado supervisionado incluem regressão linear, árvores de decisão e redes neurais.

Existem dois tipos de tarefas de aprendizado supervisionado: classificação e regressão. Classificação é quando o algoritmo é treinado para atribuir dados a uma das várias categorias possíveis. Por exemplo, classificar as imagens de animais em cães ou gatos. Regressão é quando o algoritmo é treinado para prever um valor contínuo, como o preço de uma casa ou a temperatura futura.

O processo de aprendizado supervisionado é composto de três etapas: treinamento, validação e teste. Na etapa de treinamento, o algoritmo é exposto aos dados rotulados e ajusta seus parâmetros para minimizar o erro entre as saídas previstas e as saídas reais. Na etapa de validação, o algoritmo é avaliado em um conjunto de dados separado para garantir que generaliza corretamente e não está se ajustando excessivamente aos dados de treinamento. Na etapa de teste, o algoritmo é avaliado em um conjunto de dados ainda não visto a fim de avaliar sua capacidade de generalizar para novos dados.

#### Abordagem com base em árvores de decisão

A abordagem com base em árvores de decisão é uma técnica de aprendizado supervisionado de máquina que se baseia em dividir um conjunto de dados em subconjuntos cada vez menores através da aplicação de regras de decisão. Essas regras são aplicadas a cada nó da árvore, desde a raiz até a folha, que contém a classificação ou a decisão final. Seu objetivo é encontrar regras que melhor se ajustem aos dados e permitam uma classificação precisa.

A construção de uma árvore de decisão é feita em duas etapas: geração e poda. Na etapa de geração, a árvore é construída a partir do conjunto de treinamento. Isso é feito dividindo o conjunto de dados em subconjuntos cada vez menores através da aplicação de regras de decisão. Cada subconjunto é então associado a uma folha que contém a classificação ou a decisão final. Na etapa de poda, a árvore é simplificada para evitar overfitting. Isso é feito removendo nós que não contribuem significativamente para a precisão da classificação.

Uma das principais vantagens da abordagem com base em árvores de decisão é a facilidade de interpretação. A árvore resultante é fácil de entender e interpretar, o que é útil para tomar decisões.

Além disso, essas árvores são robustas a outliers e não requerem muitos pressupostos sobre os dados. No entanto, elas podem ser propensas a overfitting se não forem cuidadosamente podadas.

Ela serve para classificação de dados, mas também pode ser usada em tomada de decisão. Além disso, as árvores de decisão são empregadas como base para outros algoritmos de aprendizado, como Random Forest e Gradient Boosting Machine (GBM).

## Algoritmo ID3

O algoritmo ID3 (Iterative Dichotomiser 3) é um algoritmo de aprendizado supervisionado de árvores de decisão desenvolvido por Ross Quinlan em 1986. Ele é usado para construir uma árvore de decisão a partir de um conjunto de dados rotulados. Seu objetivo é encontrar as regras de decisão que melhor se ajustem aos dados e permitam uma classificação precisa.

O algoritmo ID3 funciona da seguinte maneira:

- 1) Escolha a melhor característica para dividir o conjunto de dados atual. Isso é feito calculando a medida de ganho de informação para cada característica e selecionando a característica com o maior ganho.
- 2) Divida o conjunto de dados atual de acordo com os valores da característica escolhida. Crie um nó para cada valor da característica e associe o subconjunto de dados correspondente a esse nó.
- 3) Repita os passos 1 e 2 para cada subconjunto de dados até que todos os nós sejam folhas ou até que a precisão desejada seja alcançada.

Uma das principais vantagens do algoritmo ID3 é o fato de ele ser simples e fácil de entender e interpretar. Além disso, é eficiente e escalável para grandes conjuntos de dados. No entanto, tem a propensão a overfitting se o conjunto de dados contiver ruído ou as regras de decisão forem muito complexas.

## 7.2 Aprendizado não supervisionado

Aprendizado não supervisionado é uma categoria de aprendizado de máquina em que um algoritmo é treinado com dados não rotulados. Seu objetivo é explorar os dados e encontrar padrões ou estruturas escondidas neles. Exemplos de técnicas de aprendizado não supervisionado incluem clustering, detecção de outliers e redução de dimensionalidade.

Clustering é uma técnica de agrupamento em que os algoritmos tentam encontrar padrões ou estruturas nos dados, agrupando-os em conjuntos de dados similares. Exemplos de algoritmos de clustering incluem k-means e agrupamento hierárquico.

A detecção de outliers é uma técnica de detecção de anomalia que busca identificar pontos de dados significativamente diferentes dos demais. Exemplos de algoritmos de detecção de outliers incluem Isolation Forest e Local Outlier Factor (LOF).

A redução de dimensionalidade é uma técnica usada para reduzir a complexidade dos dados, removendo redundâncias e mantendo a maior parte da informação importante. Exemplos de algoritmos de redução de dimensionalidade incluem Principal Component Analysis (PCA) e t-Distributed Stochastic Neighbor Embedding (t-SNE).

O aprendizado não supervisionado é útil quando os dados não estão rotulados ou quando for necessário explorar os dados para encontrar padrões escondidos. No entanto, ele é menos preciso do que o aprendizado supervisionado e pode ser difícil de interpretar. Além disso, é importante ter uma boa compreensão dos dados e do problema antes de selecionar o algoritmo de aprendizado não supervisionado apropriado.

### Clusterização conceitual

A clusterização conceitual é uma técnica de aprendizado não supervisionado que se concentra em encontrar padrões ou estruturas escondidas em dados não rotulados através do agrupamento de dados baseado em sua similaridade semântica. Ela é diferente das técnicas de clusterização tradicionais, como k-means e agrupamento hierárquico, que se baseiam apenas na similaridade dos dados.

Baseia-se na ideia de que os dados podem ser agrupados de acordo com seu significado semântico, e não apenas com base na similaridade dos dados. Por exemplo, dois documentos podem ser semelhantes em termos de sua estrutura, mas diferentes em seu conteúdo.

A clusterização conceitual é realizada em três etapas principais: pré-processamento, agrupamento e pós-processamento.

- **Pré-processamento:** os dados são limpos e preparados para o agrupamento. Isso inclui a remoção de ruído, a normalização dos dados e a remoção de palavras irrelevantes. Além disso, é importante realizar a representação dos dados, como a utilização de técnicas de processamento de linguagem natural para criar vetores de recursos.
- **Agrupamento:** os dados são agrupados de acordo com sua similaridade semântica. Isso é feito utilizando algoritmos de clusterização conceitual, como Latent Dirichlet Allocation (LDA), Correlated Topic Model (CTM) e Probabilistic Latent Semantic Analysis (PLSA). Esses algoritmos buscam encontrar tópicos ocultos nos dados e agrupar os dados de acordo com eles.
- **Pós-processamento:** os resultados são analisados e interpretados. Isso inclui a avaliação da qualidade dos clusters gerados, a interpretação dos tópicos encontrados e a identificação de relações entre os clusters. Além disso, é possível realizar a etapa de rotulação dos clusters, atribuindo-lhes nomes que representem o conteúdo dos elementos agrupados.

Além dessas etapas, é importante ressaltar a necessidade de ter uma boa compreensão dos dados e do problema antes de selecionar o algoritmo de clusterização conceitual apropriado, bem como avaliar a qualidade do agrupamento através de métricas de avaliação de cluster.

## COBWEB

COBWEB é um algoritmo de clusterização conceitual desenvolvido por Andrew Moore em 1993. Ele é baseado no algoritmo de classificação hierárquica e projetado para agrupar dados de acordo com sua similaridade semântica.

O algoritmo COBWEB funciona da seguinte maneira:

- Inicialmente, cada instância é considerada como um cluster individual.
- Encontra-se a instância mais semelhante ao cluster atual.
- As instâncias são agrupadas em um novo cluster.
- A similaridade entre os novos clusters é calculada através de um critério de similaridade, como distância de Jaccard.
- Se a similaridade entre os clusters é maior que um limiar predefinido, eles são fundidos.
- Esse processo é repetido até que não haja nenhuma fusão possível ou até que o número de clusters alcance o valor desejado.

O COBWEB é capaz de lidar com dados categóricos e numéricos, sendo projetado para lidar com dados com uma estrutura hierárquica. Além disso, consegue lidar com dados incompletos e ruidosos, e é escalável para grandes conjuntos de dados. Ele é usado em uma variedade de aplicações, como recuperação de informações, processamento de linguagem natural e análise de mercado.

Um ponto importante a ser mencionado é que o COBWEB é uma técnica de agrupamento hierárquico, ou seja, gera uma estrutura de hierarquia de clusters e não apenas um conjunto de clusters finais. Ademais, é importante definir critérios para a parada do algoritmo, pois ele pode continuar agrupando clusters até que não haja mais dados para serem agrupados.

## 8 APRENDIZADO POR REFORÇO

O aprendizado por reforço é uma categoria de aprendizado de máquina em que um agente aprende a tomar ações apropriadas em um ambiente através da experimentação e do recebimento de recompensas ou punições. Seu objetivo é maximizar a recompensa acumulada ao longo do tempo.

Ele é baseado na ideia de que um agente pode aprender a tomar ações apropriadas através de tentativa e erro. O agente é colocado em um ambiente e recompensado ou punido de acordo com suas ações, então usa essas recompensas ou punições para ajustar sua estratégia e melhorar suas ações futuras.

Em inteligência artificial, os tipos principais de aprendizado por reforço são:

- **Aprendizagem por valor:** o agente aprende a atribuir valores a diferentes estados ou ações no ambiente. Ele usa esses valores para escolher a ação que maximiza a recompensa esperada. Algoritmos como Q-learning e SARSA são exemplos de suas técnicas.
- **Aprendizagem por política:** o agente aprende a escolher a ação apropriada em cada estado. Ele faz isso aprendendo uma política, que é uma função que mapeia estados para ações. Algoritmos como REINFORCE e Proximal Policy Optimization (PPO) são alguns de seus exemplos.
- **Aprendizagem por Actor-Critic:** abordagem híbrida que combina aspectos de aprendizagem por valor e aprendizagem por política. O Actor é responsável pela escolha das ações, já o Critic, pela avaliação das ações do Actor.
- **Aprendizagem por imitação:** abordagem em que o agente aprende a tomar ações apropriadas observando o comportamento de um agente treinado. Ele tenta imitar o comportamento observado, ao invés de aprender por meio de tentativa e erro.
- **Aprendizagem por inversão de controle:** abordagem em que o agente aprende a controlar um sistema dinâmico, ao invés de passar a tomar ações diretamente. Ele aprende a mapear estados desejáveis a ações para alcançá-los.

Cada abordagem tem suas próprias vantagens e desvantagens, sendo importante escolher a mais adequada para o problema em questão.

### 8.1 Elementos básicos da aprendizagem por reforço

A aprendizagem por reforço é uma categoria de aprendizado de máquina que se concentra em ensinar um agente a tomar ações apropriadas em um ambiente através da experimentação e do recebimento de recompensas ou punições. São alguns de seus elementos básicos:

- **Agente:** entidade que toma ações no ambiente. Ele é responsável por selecionar e executar ações com base em suas crenças sobre o estado do ambiente.
- **Ambiente:** contexto no qual o agente funciona. Ele fornece informações sobre o estado atual do sistema e responde às ações do agente com recompensas ou punições.
- **Ação:** escolha feita pelo agente para interagir com o ambiente. Ela pode ser uma ação discreta ou contínua.
- **Recompensa:** feedback positivo fornecido pelo ambiente em resposta às ações do agente. É utilizada pelo agente para avaliar o sucesso de suas ações e tomar decisões futuras.
- **Política:** estratégia utilizada pelo agente para selecionar ações. Ela pode ser uma função que mapeia estados para ações ou uma distribuição de probabilidade sobre ações.

- **Função de valor:** avaliação do agente sobre o valor de cada estado ou ação. Ela é utilizada pelo agente para avaliar o sucesso de suas ações e tomar decisões futuras.
- **Algoritmo de aprendizagem:** mecanismo utilizado pelo agente para aprender a partir das recompensas e punições. Ele pode ser baseado em valor, política ou híbrido.

## Jogo da velha

Jogo da velha é um jogo simples para duas pessoas, também conhecido como "jogo da tic-tac-toe" ou "noughts and crosses". Ele é jogado em uma grade 3x3, e cada jogador tem uma marcação, geralmente um "X" ou um "O". Vence o primeiro a conseguir colocar três de suas marcações em uma linha horizontal, vertical ou diagonal.

Em inteligência artificial, o jogo da velha pode ser usado como um problema de teste para algoritmos de busca, aprendizado de máquina e inteligência artificial. Ele pode ser jogado de forma completamente aleatória, em que o computador escolhe uma posição livre randomicamente, ou usando algoritmos mais sofisticados, como algoritmos min-max ou redes neurais.

Algoritmos min-max são algoritmos de busca em árvore usados para encontrar a jogada ótima para o jogador atual, levando em consideração todas as possíveis jogadas futuras e seus resultados esperados. Ele maximiza a recompensa do jogador atual, ao mesmo tempo em que minimiza aquela do adversário.

Redes neurais são usadas para aprender a jogar o jogo da velha, treinando-as com exemplos de jogadas vitoriosas e perdedoras, o que permite que o computador faça previsões precisas sobre a melhor jogada a ser feita em uma dada situação.

O jogo da velha é considerado um problema simples, mas pode se tornar desafiador quando adicionado de uma camada de inteligência artificial. Além disso, pode ser usado como uma metáfora para outros problemas mais complexos, como jogos de tabuleiro e problemas de tomada de decisão em negócios etc.

## Aplicando realidade aumentada (AR) e inteligência artificial (IA)

A inteligência artificial (IA) é um campo da computação que se concentra na criação de sistemas que imitam o comportamento inteligente humano. Ela tem muitas aplicações, incluindo jogos. Um bom jogo para testar e desenvolver algoritmos de IA é o jogo da velha. Ela pode ser aplicada ao jogo da velha de várias maneiras, como usando algoritmos de busca e redes neurais, e a realidade aumentada pode ser usada para melhorar a experiência de jogo.

A realidade aumentada (AR) é uma tecnologia que combina informações virtuais e reais para criar uma experiência de jogo mais imersiva. Ao aplicar AR ao jogo da velha, é possível transformar o tabuleiro em uma tela virtual que pode ser jogada em qualquer lugar e permitir que os jogadores vejam a jogada de terceiros em tempo real.



### Observação

As tecnologias que oferecem experiências imersivas são realidade virtual (RV), realidade aumentada (RA) e realidade mista (RM), sendo que cada uma funciona de maneira diferente:

A primeira, realidade virtual (RV), é uma tecnologia que cria um ambiente virtual totalmente imersivo e separado do mundo real. O usuário pode interagir com esse local por meio de dispositivos como óculos de RV ou controladores de movimento. A RV é uma experiência que pode ser usada em treinamento, jogos e simulações. Exemplos incluem jogos de RV como *Beat Saber* e *Job Simulator*, e simulações de treinamento de voo.

Já a realidade aumentada (RA) é uma tecnologia que sobrepõe informações digitais ao mundo real. Ela é geralmente usada por meio de smartphones ou tablets com câmeras e aplicativos específicos que reconhecem objetos no mundo real e adicionam elementos virtuais a eles. Exemplos incluem aplicativos de RA que permitem aos usuários visualizar móveis em sua casa antes de comprá-los, ou aplicativos de RA que mostram informações sobre um monumento ou edifício histórico enquanto o usuário o explora.

Por fim temos a realidade mista (RM), que combina elementos da RV e da RA, permitindo que usuários interajam com objetos virtuais em um ambiente real. Ela usa dispositivos como o HoloLens da Microsoft, que apresenta objetos virtuais no mundo real e possibilita que o usuário interaja com eles por meio de gestos e movimentos físicos. Exemplos incluem aplicações de RM para treinamento de manufatura, uma vez que um trabalhador pode interagir com um modelo virtual de uma peça antes de trabalhar com o item real.

Algumas aplicações populares de realidade aumentada incluem:

- **Jogos:** jogos de RA combinam elementos virtuais com o ambiente real, permitindo que os jogadores interajam com personagens virtuais e objetos. Exemplos incluem *Pokémon Go*, que possibilita aos jogadores capturar Pokémon em locais do mundo real, e *Ingress*, que incentiva os jogadores a explorar locais importantes da cidade.
- **Publicidade:** as campanhas publicitárias de RA podem permitir aos consumidores visualizar produtos em um ambiente real, experimentar diferentes variações de produtos e obter informações adicionais sobre os produtos. Por exemplo, a IKEA lançou um aplicativo de RA que permite aos consumidores visualizar como um móvel ficará em sua casa antes de comprá-lo.



- **Educação:** a RA pode ser utilizada para criar experiências de aprendizado mais interativas e imersivas. Por exemplo, um aplicativo de RA pode permitir aos alunos explorar órgãos do corpo humano em 3D ou visualizar o sistema solar considerando a proporção dos planetas.
- **Turismo:** aplicativos de RA podem fornecer informações úteis aos turistas sobre locais históricos ou atrações turísticas. Por exemplo, um aplicativo de RA pode fornecer informações sobre um monumento ou um museu à medida que o usuário explora o local.
- **Saúde:** a RA pode ser utilizada em aplicações médicas para visualizar imagens médicas em 3D e orientar cirurgiões durante procedimentos. Por exemplo, um cirurgião pode usar um sistema de RA para visualizar um modelo 3D do corpo do paciente durante uma cirurgia.

## 8.2 Redes neurais

As redes neurais são uma técnica de inteligência artificial baseada na estrutura do cérebro humano. Elas consistem em camadas de nós ou "neurônios" conectados entre si, que trabalham juntos para realizar tarefas específicas, como reconhecimento de imagens, processamento de linguagem natural e tomada de decisão.

Existem vários tipos de redes neurais, incluindo redes neurais feedforward, que passam informações de entrada através de camadas sucessivas de neurônios sem voltar para trás, e redes neurais recorrentes, que permitem a informações de estado retornarem à rede.

Aprendizado de máquina é a forma como as redes neurais aprendem a realizar tarefas específicas. Isso é feito por meio de um processo de treinamento em que a rede é exposta a um grande conjunto de exemplos de entrada e saída, e os pesos dos conectores entre os neurônios são ajustados para minimizar a diferença entre a saída desejada e aquela produzida pela rede.

### O neurônio artificial

O neurônio artificial é uma unidade básica de uma rede neural artificial, que é projetada para simular a estrutura e função de um neurônio biológico. Cada neurônio artificial tem entradas, saídas e pesos que conectam suas entradas às suas saídas.

A operação básica de um neurônio artificial é realizar uma soma ponderada das entradas e aplicar uma função de ativação a esse resultado. A função de ativação é matemática e determina se o neurônio está "ativado" ou "inativo", ou seja, se ele irá transmitir uma saída ou não (HAYKIN, 1994).

Em um neurônio artificial, a função de ativação é uma função matemática que é aplicada à soma ponderada dos sinais de entrada para determinar a saída do neurônio. Ela é responsável por determinar se o neurônio deve ou não ser ativado e, se sim, qual deve ser a intensidade da ativação.

Há diversas funções de ativação que podem ser usadas em um neurônio artificial, e cada uma delas apresenta características específicas. As mais comuns são:

- **Função degrau (step function):** a saída do neurônio é 1 se a soma ponderada dos sinais de entrada for maior que um determinado limiar, e 0 caso contrário. Essa função é simples e eficiente, mas não é diferenciável e pode apresentar problemas durante o treinamento de redes neurais.
- **Função sigmoidal:** tem formato de S e é diferenciável em todos os pontos. Ela é usada principalmente em problemas de classificação binária, em que a saída do neurônio deve ser um valor entre 0 e 1. Essa função também é usada em redes neurais profundas, que ajuda a reduzir a variância e melhorar a estabilidade do treinamento.
- **Função tangente hiperbólica:** similar à função sigmoidal, mas com valores de saída entre -1 e 1. Ela é amplamente utilizada em redes neurais devido à sua capacidade de modelar relações não lineares e sua propriedade de simetria em torno do eixo y.
- **Função ReLU (Rectified Linear Unit):** é definida como  $f(x) = \max(0, x)$ . Ela é amplamente utilizada em redes neurais profundas devido à sua eficiência computacional e à sua capacidade de lidar com o problema do gradiente desvanecente.
- **Função softmax:** é usada em problemas de classificação multiclasse, em que a saída do neurônio deve ser uma distribuição de probabilidade sobre as possíveis classes. Essa função converte as saídas dos neurônios em valores de probabilidade normalizados, garantindo que a soma de todas as saídas seja igual a 1.



### Saiba mais

A escolha da função de ativação depende do problema em questão e das características da rede neural. Cada função tem suas vantagens e desvantagens, e o uso de uma função inadequada pode levar a problemas de desempenho e convergência durante o treinamento. A fim de entender melhor, recomendamos o artigo a seguir:

NWANKPA, C. E. *et al.* *Activation functions: comparison of trends in practice and research for deep learning*. Cornell University, 2018. Disponível em: <https://bit.ly/3F9U2ae>. Acesso em: 6 mar. 2023.

Os pesos dos neurônios são valores numéricos que determinam a importância relativa das entradas para a saída. Durante o processo de treinamento, os pesos dos neurônios são ajustados para minimizar a diferença entre a saída desejada e aquela produzida pela rede.

Os neurônios artificiais são organizados em camadas, sendo que cada camada é composta de vários neurônios. A primeira delas é chamada de camada de entrada e é responsável por receber os dados de entrada. A última é a camada de saída, responsável por produzir a saída da rede. As camadas intermediárias são camadas ocultas e têm a função de realizar operações complexas e extrair características importantes dos dados de entrada.

O neurônio artificial é a unidade básica da rede neural artificial, que é responsável por realizar as operações matemáticas e tomar decisões. Trata-se de uma abstração da estrutura e função do neurônio biológico, que é usada para simular a capacidade de aprendizado e adaptação do cérebro humano.

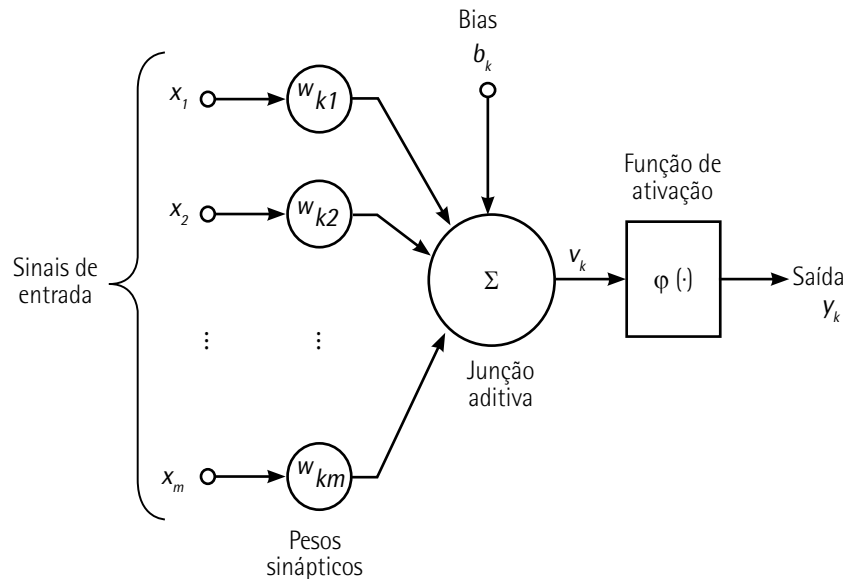


Figura 37 – Modelo de neurônio artificial

Fonte: Haykin (2007, p. 36).

## Treinamento de um perceptron

O perceptron é uma arquitetura simples de rede neural artificial, composta de uma única camada de neurônios, usada para resolver problemas de classificação binária. Trata-se de uma das primeiras arquiteturas de redes neurais desenvolvidas e tem sido amplamente utilizada como uma ferramenta básica para entender como as redes neurais aprendem.

O processo de treinamento de um perceptron é baseado no algoritmo de aprendizado de perceptron, que é um algoritmo de otimização supervisionado. Ele é usado para ajustar os pesos dos neurônios a fim de minimizar a diferença entre a saída desejada e a saída produzida pela rede. O algoritmo de aprendizado de perceptron é iterativo e executado até que a diferença entre a saída desejada e aquela produzida pela rede seja menor que um determinado limite.

O processo de treinamento de um perceptron consiste em três etapas:

- 1) **Inicialização:** os pesos dos neurônios são inicializados com valores aleatórios.
- 2) **Propagação:** os dados de entrada são passados através da rede, e a saída é produzida.
- 3) **Atualização:** os pesos dos neurônios são atualizados com base na diferença entre a saída desejada e a saída produzida pela rede.

A atualização dos pesos é realizada usando o algoritmo de aprendizado de perceptron, que é baseado no gradiente descendente. Ele usa a derivada dos erros em relação aos pesos para calcular a direção de atualização dos pesos. A taxa de aprendizado é um parâmetro que controla a velocidade de atualização dos pesos.

É importante notar que o perceptron somente é capaz de resolver problemas linearmente separáveis, isto é, aqueles nos quais os dados de treinamento podem ser separados por uma linha reta. Se os dados não são linearmente separáveis, o perceptron não será capaz de aprender uma solução correta.

### Backpropagation

Backpropagation é um algoritmo de aprendizado de máquina utilizado para ajustar os pesos de uma rede neural artificial. Ele é usado para treinar redes neurais feedforward, que têm camadas sucessivas de neurônios conectados entre si. O algoritmo é chamado de backpropagation porque ele "propaga o erro de volta" através da rede, ajustando os pesos de cada camada para minimizar o erro.

O processo de treinamento com backpropagation consiste em três etapas principais:

- 1) **Propagação para frente:** os dados de entrada são passados através da rede, e a saída é produzida. Isso é realizado através da aplicação das funções de ativação e dos pesos de cada neurônio na camada.
- 2) **Cálculo do erro:** a diferença entre a saída desejada e a saída produzida pela rede é calculada e usada como medida do erro.
- 3) **Propagação para trás:** o erro é propagado de volta através da rede, ajustando os pesos de cada neurônio para minimizar o erro. Isso é feito através do cálculo dos gradientes dos erros em relação aos pesos usando o método do gradiente descendente.

O backpropagation é um algoritmo iterativo, ou seja, o processo de treinamento é repetido várias vezes com diferentes conjuntos de dados até que o erro seja minimizado. A taxa de aprendizado é um parâmetro que controla a velocidade de atualização dos pesos.

Trata-se de um algoritmo eficiente e robusto para treinar redes neurais feedforward, é amplamente utilizado em uma variedade de aplicações, como reconhecimento de imagens, processamento de linguagem natural, reconhecimento de fala e tomada de decisão. Alguns avanços recentes, como o uso de técnicas de otimização e regularização, por exemplo, do Momentum, Adam, Adagrad e Adadelta, têm melhorado ainda mais o desempenho do backpropagation.

### Treinamento do MLP

MLP (Multi-Layer Perceptron) é uma arquitetura de rede neural artificial composta de várias camadas de neurônios, sendo que cada camada possui vários neurônios (como um cérebro humano). Trata-se de uma extensão do perceptron simples, que é constituída por uma única camada de neurônios. As camadas adicionais permitem que a rede realize operações mais complexas, o que possibilita resolver problemas não linearmente separáveis.

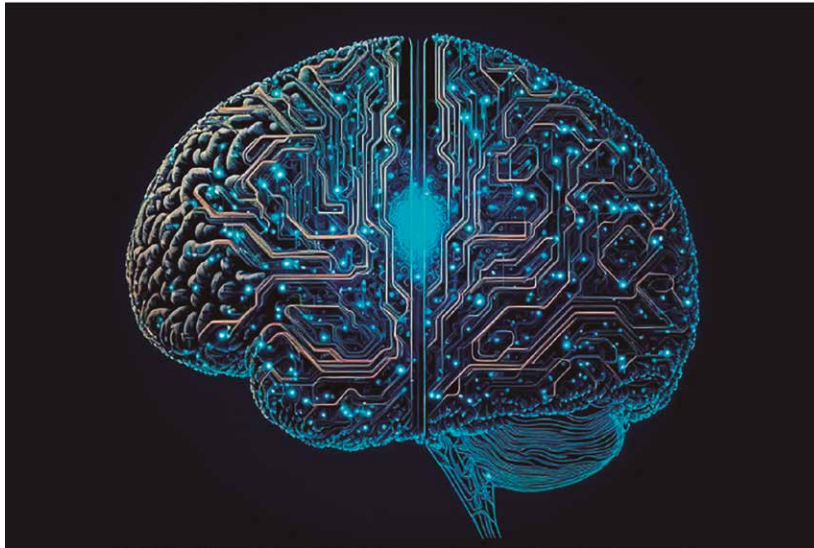


Figura 38

Disponível em: <https://bit.ly/3YwnbmH>. Acesso em: 6 mar. 2023.

O processo de treinamento de uma MLP é semelhante ao do perceptron, usando o algoritmo de backpropagation. Trata-se de um algoritmo de otimização supervisionado que ajusta os pesos dos neurônios para minimizar a diferença entre a saída desejada e a saída produzida pela rede. O algoritmo de backpropagation é iterativo e executado até que a diferença entre a saída desejada e aquela produzida pela rede seja menor que um determinado limite.

O processo de treinamento de uma MLP consiste em três etapas:

- 1) **Inicialização:** os pesos dos neurônios são inicializados com valores aleatórios.
- 2) **Propagação para frente:** os dados de entrada são passados através da rede, e a saída é produzida. Isso é realizado através da aplicação das funções de ativação e dos pesos de cada neurônio em qualquer camada.
- 3) **Propagação para trás:** o erro é propagado de volta através da rede, ajustando os pesos de cada neurônio para minimizar o erro. Isso é feito através do cálculo dos gradientes dos erros em relação aos pesos usando o método do gradiente descendente.

A diferença do perceptron para o MLP é que esta possui mais camadas, e assim tem mais capacidade de representação e pode realizar tarefas mais complexas. A taxa de aprendizado é um parâmetro que controla a velocidade de atualização dos pesos, sendo importante para evitar o overfitting.



### Observação

Overfitting é um problema comum em redes neurais e ocorre quando a rede aprende padrões específicos dos dados de treinamento que não generalizam bem para novos dados de teste. Em outras palavras, a rede neural se ajusta aos dados de treinamento, mas não consegue generalizar para novos exemplos.

Esse problema geralmente acontece quando a rede é muito complexa em relação ao tamanho do conjunto de dados de treinamento ou o conjunto de treinamento não é representativo ou suficiente no que diz respeito aos dados que a rede neural encontrará na vida real. Como resultado, a rede aprende padrões que são específicos para o conjunto de treinamento e não são relevantes para outros dados.

Ele pode ser identificado observando o desempenho da rede neural nos dados de treinamento e de teste. Se ele estiver muito bom nos dados de treinamento, mas ruim nos dados de teste, é provável que a rede esteja sofrendo de overfitting.

É importante notar que o treinamento de uma MLP pode ser computacionalmente caro, dependendo do tamanho da rede e do conjunto de dados de treinamento. Algumas técnicas, como de otimização e regularização, podem ser usadas para melhorar o desempenho e a eficiência do treinamento.

A camada de entrada não realiza processamento, apenas distribui os valores de entrada para os neurônios de primeira camada de processamento. Podemos conectar os neurônios artificiais de duas maneiras: com ou sem realimentação, podendo assim dividir as MLP em duas classes:

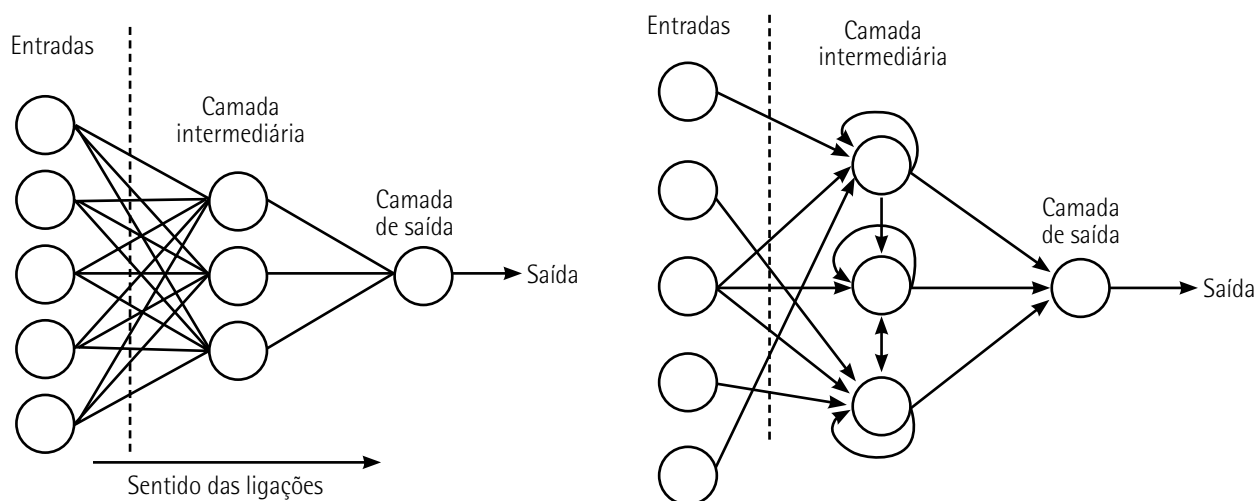


Figura 39 – À direita uma RNA não recorrente e à esquerda uma RNA recorrente

Fonte: Braga, Ludermir e Carvalho (2000, p. 12).

- **MLP não recorrentes ("Feedforward"):** as saídas não realimentam as entradas, e nem os neurônios da mesma camada, nem os neurônios das camadas anteriores são conectados. As entradas são aplicadas apenas aos neurônios da primeira camada.
- **MLP recorrente:** devido à realimentação das saídas para as entradas, as saídas dependem das entradas atuais e das saídas anteriores, e pode haver conexões de neurônios de uma mesma camada.

## Por que utilizar redes neurais?

As redes neurais são uma técnica de inteligência artificial baseada na estrutura do cérebro humano. Elas são amplamente utilizadas devido à sua capacidade de aprender e se adaptar a novos dados, tornando-as uma ferramenta valiosa para tarefas que exigem aprendizado automático. Algumas das principais razões para utilizar redes neurais incluem:

- **Capacidade de aprendizado:** capacidade de aprender e se adaptar a novos dados, o que as torna úteis para tarefas que exigem aprendizado automático. Por exemplo, elas podem ser usadas para reconhecimento de imagens, processamento de linguagem natural e tomada de decisão.
- **Robustez:** resistência a ruído e variações na entrada, o que as torna úteis para tarefas que enfrentam incerteza e ambiguidade.
- **Capacidade de generalização:** possibilidades de generalizar suas habilidades para novos dados, o que as torna úteis para tarefas que exigem generalização para novos exemplos.
- **Capacidade de representação:** potencial de representar relações complexas entre entradas e saídas, o que as torna úteis para tarefas que exigem representação de conhecimento.
- **Flexibilidade:** qualidade de serem usadas para resolver uma ampla variedade de tarefas, desde problemas simples até aqueles mais complexos, e podem ser facilmente adaptadas para novas tarefas.
- **Escalabilidade:** aptidão de serem escaladas para lidar com grandes conjuntos de dados, o que as torna úteis às tarefas que exigem grande capacidade de processamento.

As redes neurais são uma técnica poderosa e versátil de inteligência artificial, que podem ser usadas para resolver uma ampla variedade de problemas, e por isso são bastante utilizadas em várias áreas.

## Aplicações de redes neurais artificiais

As redes neurais artificiais são técnicas de inteligência artificial utilizadas em uma variedade de aplicações, devido à sua capacidade de aprender e se adaptar a novos dados. Algumas das suas principais aplicações incluem:

- **Visão computacional:** atuam como reconhecimento de imagens e detecção de objetos. Elas são capazes de reconhecer rostos, identificar objetos em imagens, rastrear objetos em vídeos e muito mais.

- **Processamento de linguagem natural:** processam a linguagem natural, como tradução automática, gerador de texto, reconhecimento de voz e compreensão de linguagem.
- **Reconhecimento de fala:** transformam fala em texto e reconhecem comandos de voz, como reconhecimento de voz para dispositivos de assistente de voz.
- **Robótica:** controlam robôs e permitem que os robôs aprendam e se adaptem a novas tarefas.
- **Análise de dados:** analisam grandes conjuntos de dados, como dados de mercado financeiro, dados médicos e dados climáticos, para prever tendências e tomar decisões informadas.
- **Jogos:** desenvolvem jogos inteligentes, como jogos de tabuleiro e jogos de videogame, em que os jogadores podem praticar contra agentes de inteligência artificial.

### 8.3 Algoritmos genéticos

Os algoritmos genéticos (AGs) são uma técnica de inteligência artificial baseada no processo de evolução natural. Eles são utilizados para resolver problemas de otimização e busca de soluções em áreas como engenharia, ciência da computação e negócios.

Os AGs usam conceitos da biologia evolutiva, como seleção natural, reprodução e mutação, para encontrar soluções de problemas complexos. Eles criam uma população de soluções candidatas (chamadas de indivíduos) e, através de uma série de iterações, selecionam os indivíduos mais aptos para reprodução e combinação, enquanto outros são descartados. A cada iteração, a população evolui para se aproximar de uma solução ótima para o problema.



#### Lembrete

A iteração é um processo de repetição ou ciclo usado em diversas áreas do conhecimento, em que uma operação ou conjunto de operações é executado várias vezes para se aproximar de uma solução ou aprimorar um sistema.

Eles são empregados para resolver problemas de otimização, como encontrar os melhores parâmetros para um modelo, a melhor rota em um grafo ou a melhor estratégia em um jogo. Também são utilizados para resolver problemas de classificação e previsão, como classificar documentos ou prever preços de ações.

Além disso, conseguem lidar com problemas complexos e não lineares, podendo encontrar soluções ótimas mesmo quando não há uma solução analítica disponível. Eles também são flexíveis e facilmente adaptados para novos problemas. No entanto, os AGs podem ser computacionalmente caros e não ser a melhor escolha para problemas simples ou aqueles com restrições de tempo.

Existem vários tipos de algoritmos genéticos (AGs), cada um com suas características e aplicações específicas. Alguns dos principais tipos de AGs incluem:



- **Algoritmo genético clássico (AGC):** um dos tipos mais comuns de AGs, baseado nas ideias básicas da evolução natural, como seleção, reprodução e mutação. Ele é amplamente utilizado para resolver problemas de otimização e busca de soluções.
- **Algoritmo genético evolutivo (AGE):** variação do AGC que usa técnicas evolutivas adicionais, como o uso de diferentes taxas de mutação e cruzamento, a fim de aumentar a eficiência do algoritmo.
- **Algoritmo genético estocástico (AGE):** variação do AGC que utiliza métodos estocásticos para seleção, reprodução e mutação, a fim de evitar o problema de convergência precoce.
- **Algoritmo genético de mapa de otimização (AGMO):** AG baseado no mapa de otimização e amplamente utilizado para resolver problemas de otimização não linear.
- **Algoritmo genético de programação (AGP):** AG baseado na programação genética e amplamente utilizado para resolver problemas de programação.
- **Algoritmo genético multiobjetivo (AGMO):** AG projetado para resolver problemas de otimização multiobjetivo, em que mais de uma função deve ser otimizada simultaneamente.
- **Algoritmo genético de partícula (AGP):** AG baseado no algoritmo de otimização de enxame de partículas (PSO) e amplamente utilizado para resolver problemas de otimização não linear.

## Conceitos básicos

Os algoritmos genéticos (AGs) são uma técnica de inteligência artificial baseada no processo de evolução natural, que é amplamente utilizada para resolver problemas de otimização e busca de soluções. Os conceitos básicos de AG incluem:

- **População:** composta de um conjunto de soluções candidatas (chamadas de indivíduos) para o problema. Cada indivíduo é representado por um conjunto de características ou parâmetros, chamados de cromossomos.
- **Fitness:** medida da aptidão de um indivíduo para resolver o problema. Ela é geralmente calculada com base em uma função de fitness, que avalia o quão bem um indivíduo se adapta às restrições e objetivos do problema.
- **Seleção:** processo de escolha dos indivíduos mais aptos para reprodução. Ela geralmente é baseada em sua aptidão, mas pode considerar outros critérios, como diversidade genética.
- **Cruzamento (Crossover):** modo de combinar os cromossomos de dois indivíduos para produzir novas pessoas. Ele é geralmente realizado escolhendo aleatoriamente pontos de corte nos cromossomos dos pais e combinando as partes correspondentes dos cromossomos para produzir os cromossomos dos filhos.

- **Mutação:** método de alterar aleatoriamente um ou mais cromossomos de um indivíduo. Isso é feito para introduzir nova diversidade genética na população e evitar a convergência precoce.
- **Iteração:** processo de seleção, cruzamento e mutação é repetido em cada iteração, produzindo uma nova geração de indivíduos.

### Funcionamento do algoritmo

Os algoritmos genéticos (AGs) são uma técnica de inteligência artificial baseada no processo de evolução natural, amplamente utilizada para resolver problemas de otimização e busca de soluções. Seu funcionamento básico é o seguinte:

- **Inicialização:** começo do algoritmo criando uma população inicial de indivíduos aleatórios, cada um deles representado por um conjunto de características ou parâmetros, chamados de cromossomos.
- **Avaliação:** cada indivíduo é avaliado usando uma função de fitness, que mede sua aptidão para resolver o problema. Essa função é geralmente definida com base nas restrições e objetivos do problema.
- **Seleção:** processo de escolher os indivíduos mais aptos para reprodução. Ela geralmente é baseada na aptidão dos indivíduos, mas pode considerar outros critérios, como diversidade genética.
- **Cruzamento (Crossover):** modo de combinar os cromossomos de dois indivíduos para produzir novos indivíduos. Ele é geralmente realizado escolhendo aleatoriamente pontos de corte nos cromossomos dos pais e combinando as partes correspondentes dos cromossomos para produzir os cromossomos dos filhos.
- **Mutação:** método de alterar aleatoriamente um ou mais cromossomos de um indivíduo. Isso é feito para introduzir nova diversidade genética na população e evitar a convergência precoce.
- **Iteração:** processo de seleção, cruzamento e mutação que é repetido em cada iteração, produzindo uma nova geração de indivíduos. Cada geração é avaliada usando a função de fitness e o processo é repetido até que uma solução satisfatória seja encontrada ou o número máximo de iterações seja atingido.

Os AGs são capazes de lidar com problemas complexos e não lineares, e podem encontrar soluções ótimas mesmo quando não há uma.

### Seleção dos mais aptos

A seleção é um dos passos fundamentais do processo de algoritmo genético (AG). Ela é responsável por escolher os indivíduos mais aptos para reprodução e, consequentemente, pela formação de uma nova geração. Seu objetivo é garantir que os indivíduos mais aptos sejam reproduzidos com maior frequência, a fim de aumentar as chances de obter soluções ótimas para o problema.

Existem vários métodos de seleção utilizados em AGs, cada um com suas vantagens e desvantagens. Alguns dos métodos mais comuns incluem:

- **Seleção por roleta:** baseia-se na ideia de que indivíduos com maior aptidão têm maiores chances de serem selecionados para reprodução. Ele funciona atribuindo uma porcentagem de escolha para cada indivíduo, proporcional à sua aptidão.
- **Seleção por torneio:** envolve escolher aleatoriamente um grupo de indivíduos e selecionar aquele com a maior aptidão para reprodução. Isso é repetido várias vezes para formar uma nova geração.
- **Seleção estocástica universal:** emprega o princípio de seleção proporcional à aptidão, mas usa uma técnica estocástica para escolher indivíduos, evitando que aqueles com maior aptidão sejam selecionados com muita frequência.
- **Seleção por elitismo:** consiste em preservar os indivíduos mais aptos de uma geração para a próxima, garantindo que as características desejadas não sejam perdidas.

A escolha do método de seleção a ser utilizado depende do problema específico e do contexto em que o AG está sendo aplicado. Alguns são mais adequados para problemas com restrições fortes.

## Parâmetros genéticos

Os parâmetros genéticos são uma parte importante do algoritmo genético (AG), usados para controlar o comportamento do algoritmo. Eles incluem a taxa de crossover, taxa de mutação, tamanho da população, número de gerações, entre outros. A seguir, serão descritos alguns dos principais parâmetros genéticos:

- **Taxa de crossover:** probabilidade de que um par de indivíduos sofra crossover (ou seja, troca de informação genética) durante a reprodução. É geralmente definida como um valor entre 0 e 1. Valores mais altos normalmente resultam em maior diversidade genética, mas aumentam a chance de introduzir características indesejadas na população.
- **Taxa de mutação:** hipótese de que um cromossomo sofra mutação (ou seja, alteração aleatória) em cada geração. É geralmente definida como um valor entre 0 e 1. Valores mais altos normalmente resultam em maior diversidade genética, mas aumentam a chance de introduzir características indesejadas na população.
- **Tamanho da população:** parâmetro que controla o número de indivíduos presentes na população. Um tamanho de população maior pode aumentar a diversidade genética, mas pode elevar o tempo de processamento.
- **Taxa de cruzamento:** característica que controla a frequência com que ocorre o cruzamento entre indivíduos. Uma taxa de cruzamento alta pode aumentar a diversidade genética, bem como elevar o risco de perda de informação útil.

- **Número de gerações:** fator que refere-se ao número de iterações que o algoritmo executa durante o processo de otimização. Cada geração é composta de um conjunto de indivíduos (soluções candidatas) avaliados de acordo com uma função de aptidão, que determina quão bem cada pessoa resolve o problema em questão.

### Aplicações

Os algoritmos genéticos (AGs) são uma técnica de inteligência artificial baseada no processo de evolução natural, amplamente usada para resolver problemas de otimização e busca de soluções. Eles são aplicados em uma variedade de campos, incluindo:

- **Engenharia:** utilizados para otimizar o desempenho de sistemas mecânicos, elétricos e de controle. Por exemplo, a fim de projetar antenas, otimizar a configuração de sistemas de controle de processos industriais e encontrar soluções para problemas de planejamento de projetos.
- **Finanças:** utilizados para otimizar a alocação de ativos, encontrar estratégias de negociação e prever o desempenho de ações.
- **Ciência da computação:** utilizados para resolver problemas de otimização em inteligência artificial, como redes neurais, algoritmos de aprendizado de máquina e algoritmos de busca.
- **Bioinformática:** utilizados para analisar dados de sequenciamento de DNA e proteínas, identificar genes relacionados a doenças e desenvolver novos medicamentos.
- **Robótica:** utilizados para otimizar a performance de robôs, desenvolver estratégias de navegação e planejar trajetórias.
- **Agricultura:** utilizados para otimizar a produção agrícola, planejar o uso de recursos e desenvolver estratégias de irrigação.
- **Produção:** utilizados para planejar e otimizar a produção, ordenar a logística e desenvolver estratégias.



#### Saiba mais

A fim de entender mais sobre algoritmos genéticos como, por exemplo, os esquemas, os operadores genéticos, o cruzamento e a mutação, recomenda-se a leitura do capítulo 14 do seguinte livro:

COPPIN, B. *Inteligência artificial*. Rio de Janeiro: LTC, 2010.



## Resumo

Nesta unidade, vimos os tipos de aprendizado de máquina, que é um subcampo da inteligência artificial concentrado em desenvolver algoritmos capazes de aprender a partir de dados sem serem explicitamente programados. Eles possuem três tipos principais: supervisionado, não supervisionado e por reforço.

O aprendizado de máquina supervisionado envolve o treinamento de um modelo usando um conjunto de dados rotulados. Cada exemplo de treinamento é uma entrada com uma saída esperada correspondente. O modelo aprende a associar as entradas às saídas corretas durante o treinamento, ajustando seus parâmetros para minimizar algum critério de erro. Em seguida, ele pode ser usado a fim de prever a saída correspondente para novos exemplos de entrada. Por exemplo, um modelo de classificação supervisionado poderia ser treinado com um conjunto de imagens de animais rotuladas como "gato" ou "cachorro", aprenderia a associar as características das imagens com os rótulos correspondentes e, em seguida, poderia ser usado para classificar novas imagens de animais.

Já o aprendizado de máquina não supervisionado envolve a identificação de padrões ou estruturas em um conjunto de dados sem rótulos. O modelo é treinado apenas com as entradas e precisa encontrar estruturas ou agrupamentos significativos no conjunto de dados. Ele pode ser usado para descobrir insights em grandes conjuntos de dados ou reduzir a dimensionalidade dos dados. Por exemplo, um modelo de clusterização não supervisionado poderia ser usado para agrupar clientes com base em seus hábitos de compra, analisando os padrões de compra dos clientes para encontrar grupos de clientes semelhantes entre si.

Observamos ainda que aprendizado por reforço é uma técnica de aprendizado de máquina em que um agente interage com um ambiente dinâmico e aprende a escolher ações que maximizam a recompensa numérica em longo prazo. O agente toma decisões com base em uma política de ação, que determina a probabilidade de escolher cada ação possível em qualquer estado do ambiente. Seus elementos básicos incluem o agente, o ambiente, as ações que o agente pode tomar, as recompensas que ele recebe do ambiente, o estado atual do ambiente e a política de ação que o agente usa para escolher suas ações.

Também foram abordados as redes neurais e os algoritmos genéticos. As redes neurais são modelos matemáticos inspirados no cérebro humano

que são usados em muitos tipos de aprendizado de máquina, incluindo aprendizado profundo. Eles são compostos de camadas de neurônios artificiais, que processam informações e transmitem sinais através das conexões entre eles. As redes neurais são treinadas usando um conjunto de dados rotulado, ajustando seus pesos para minimizar algum critério de erro. Já os algoritmos genéticos são uma técnica de otimização baseada na seleção natural e na evolução biológica. Eles trabalham com uma população de soluções candidatas, representadas por cromossomos, que evoluem ao longo do tempo por meio de processos de mutação e cruzamento. Cada solução é avaliada com base em uma função de aptidão, que determina quão bem ela resolve o problema em questão. As soluções mais aptas são selecionadas para se reproduzir e gerar novas soluções para a próxima geração.

Por fim, entendemos que tanto as redes neurais quanto os algoritmos genéticos podem ser usados com o aprendizado por reforço para melhorar o desempenho do agente. As redes neurais podem ser empregadas para aproximar a função de valor do agente ou a política de ação, enquanto os algoritmos genéticos podem ser usados a fim de ajustar os hiperparâmetros do modelo de aprendizado por reforço.



## Exercícios

**Questão 1.** (Enade 2021, adaptada) Leia o texto a seguir, a respeito de plataformas virtuais de aprendizagem.

O Massive Open Online Course (MOOC) é composto de plataformas virtuais de aprendizagem que têm como objetivo disponibilizar, para grande número de alunos, a oportunidade de ampliar os seus conhecimentos de forma gratuita. Uma definição que pode ser adotada para MOOC é de que representa experiências de aprendizagem inovadoras com base nas TICs, em plataformas web 2.0 e em redes sociais. A participação em um MOOC é aberta para qualquer interessado e envolve grande quantidade de material didático.

Adaptado de: CORDEIRO, R. F.; AGUIAR, Y. P. C.; SARAIVA, J. A. G. Perspectivas da avaliação de usabilidade em MOOCs. *Nuevas Ideas en Informática Educativa*, v. 12, p. 465-470.

Acerca da avaliação em plataformas MOOCs, avalie as asserções e a relação proposta entre elas.

I – O poder de massificação das plataformas MOOCs gera um grande potencial acerca de automatização de avaliação nestes ambientes.

porque

II – O grande número de estudantes que podem ser submetidos à avaliação nos MOOCs permite que, com o decorrer do tempo e o uso de inteligência artificial, sejam coletados dados suficientes para criar um sistema capaz de corrigir avaliações automaticamente, mesmo que essas contenham questões discursivas.

A respeito dessas asserções, assinale a opção correta.

- A) As asserções I e II são proposições verdadeiras, e a asserção II é uma justificativa correta da I.
- B) As asserções I e II são proposições verdadeiras, e a asserção II não é uma justificativa correta da I.
- C) A asserção I é uma proposição verdadeira, e a II é uma proposição falsa.
- D) A asserção I é uma proposição falsa, e a II é uma proposição verdadeira.
- E) As asserções I e II são proposições falsas.

Resposta correta: alternativa A.

### Análise das asserções

I – Asserção verdadeira.

Justificativa: o poder de massificação das plataformas MOOCs gera uma possibilidade de coleta massiva de dados gerados pelos usuários, que podem ser usados com o intuito da automatização de avaliação nesses ambientes.

II – Asserção verdadeira.

Justificativa: aprendizado de máquina (machine learning) é uma subárea da inteligência artificial que se concentra em desenvolver técnicas e algoritmos que permitem que as máquinas aprendam com dados. Nas plataformas MOOCs, ao longo do tempo, os dados gerados pelos estudantes podem ser coletados, gerando quantidade massiva de dados disponíveis a respeito das avaliações da plataforma. O uso adequado do machine learning será, então, capaz de extrair informações a respeito desses dados, possibilitando a criação de um sistema de correção automática de avaliações, mesmo para os casos de questões dissertativas.

Relação entre as asserções.

Se grande número de estudantes que podem ser submetidos à avaliação nos MOOCs permite que sejam coletados dados suficientes para criar um sistema capaz de corrigir avaliações automaticamente, então o poder de massificação das MOOCs gera grande potencial acerca de automatização de avaliação nestes ambientes. Temos, portanto, que a asserção II justifica a asserção I.



**Questão 2.** (IBFC 2020, adaptada) No contexto da inteligência artificial, analise o diagrama simplificado e esquematizado a seguir.

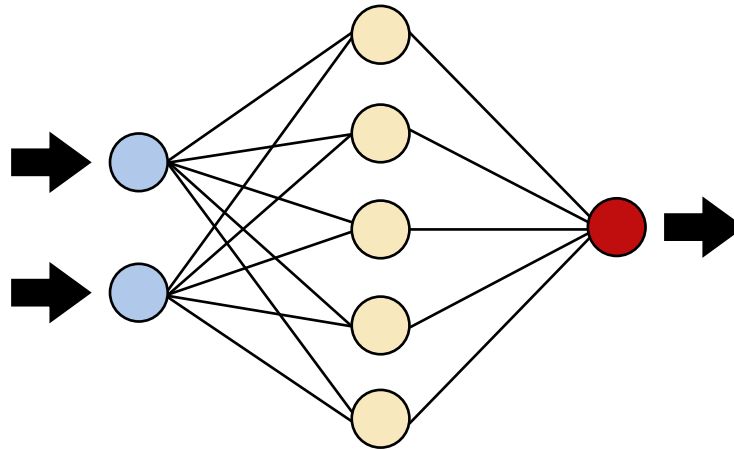


Figura 40

Assinale a alternativa correta sobre a qual rede se refere o diagrama.

- A) Anel.
- B) Estrela.
- C) Barramento.
- D) Neural.
- E) Heurística.

Resposta correta: alternativa D.

### Análise da questão

As redes neurais são um tipo de machine learning baseado na estrutura do cérebro humano. Elas consistem em camadas de nós, ou "neurônios", conectados entre si, que trabalham em conjunto para realizar tarefas específicas, como o reconhecimento de imagens, o processamento de linguagem natural e a tomada de decisão.

Cada círculo da imagem da questão representa um neurônio artificial, que é a unidade básica de uma rede neural artificial, projetado para simular a estrutura e a função de um neurônio biológico.

Por meio de algoritmos, as redes neurais podem reconhecer padrões escondidos e correlações em dados brutos, agrupá-los e classificá-los. Com o tempo, conseguem aprender e melhorar suas conclusões.

Os neurônios artificiais são organizados em camadas, sendo que cada camada é composta de vários neurônios. A primeira é chamada de camada de entrada, mostrada à esquerda na figura, e é responsável por receber os dados de entrada. A última camada é a camada de saída, mostrada à direita na figura, e é responsável por produzir a saída da rede. Já a camada intermediária, composta de neurônios representados em azul, é oculta, tem a função de realizar operações ou extrair características dos dados de entrada.

## REFERÊNCIAS

### TEXTUAIS

- ARISTÓTELES. *Ética a Nicômaco*. São Paulo: Nova Cultural, 1987. (Coleção: Os pensadores).
- BISHOP, C. *Pattern Recognition and Machine Learning* (Information Science and Statistics). Secaucus: Springer-Verlag New York, Inc., 2006.
- BRAGA, A.; LUDERMIR, T.; CARVALHO, A. *Redes neurais artificiais: teoria e aplicações*. Rio de Janeiro: LTC, 2000.
- CHEESEMAN, P.; KANEFSKY, B.; TAYLOR, W. Where the really hard problems are. In: INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE, 12., 1991, Sydney. *Anais [...]*. Sydney: Morgan Kaufmann, 1991.
- COPPIN, B. *Inteligência artificial*. Rio de Janeiro: LTC, 2010.
- DUDA, R.; HART, P.; STORK, D. *Pattern classification: second edition*. New York: John Wiley Sons, Inc., 2000.
- FREITAS, F. L. *Ontologias e a Web Semântica*. Santos: Universidade Católica de Santos, 2007. Disponível em: <https://bit.ly/3JoXiku>. Acesso em: 6 mar. 2023.
- HAYKIN, S. *Neural networks: a comprehensive foundation*. New York: MacMillan, 1994.
- HAYKIN, S. *Redes neurais: princípios e prática*. Porto Alegre: Bookman, 2007.
- JENNINGS, N. R. Coordination techniques for distributed artificial intelligence. *Foundations of distributed artificial intelligence*, p. 187-210, May. 1996.
- JÓNSSON, A. et al. Planning in interplanetary space: theory and practice. In: INTERNATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE PLANNING SYSTEMS, 5., 2000, [s.l.]. *Anais [...]*. [s.l.]: AIPS, 2000.
- KAPLAN, A.; HAENLEIN, M. Siri, siri, in my hand: Who's the fairest in the land? On the interpretations, illustrations, and implications of artificial intelligence. *Business Horizons*, v. 62, n. 1, p. 15-25, 2019.
- MCCARTHY, J. et al. *Lisp 1.5 programmer's manual*. Massachusetts: MIT, 1962.
- NWANKPA, C. E. et al. *Activation functions: comparison of trends in practice and research for deep learning*. Cornell University, 2018. Disponível em: <https://bit.ly/3F9U2ae>. Acesso em: 6 mar. 2023.
- O'HARE, G. M. P.; JENNINGS, N. R. *Foundations of distributed artificial intelligence*. Nova Jersey: Wiley, 1996.
- RUSSELL, S.; NORVIG, P. *Inteligência artificial*. 3. ed. Barueri: Grupo GEN, 2013.
- SAMUEL, A. Some studies in machine learning using the game of checkers. *IBM Journal*, v. 3, n. 3. jul. 1959.



Handwriting practice lines consisting of 30 horizontal blue lines. Each line is preceded by a small blue dot, serving as a guide for letter height and placement.





Informações:  
[www.sepi.unip.br](http://www.sepi.unip.br) ou 0800 010 9000