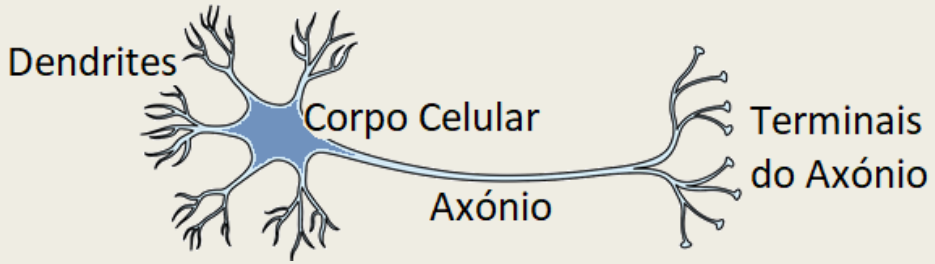
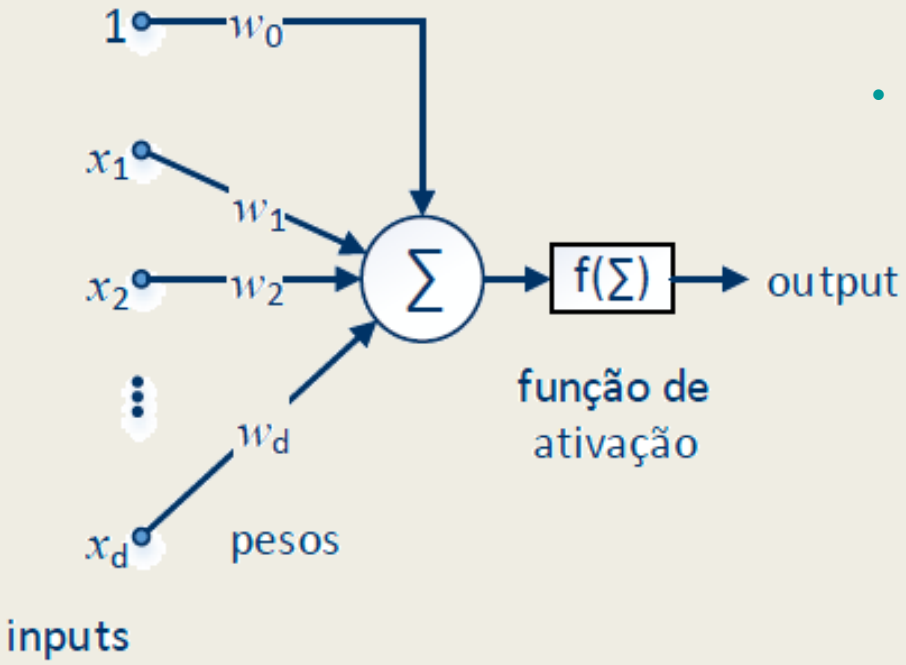
Redes Neuronais

- Uma Rede Neuronal (artificial) é uma técnica de aprendizagem inspirada na inteligência biológica,
  - *baseia-se na estrutura e no funcionamento do sistema nervoso, com o objetivo de simular a capacidade de aprendizagem do cérebro humano na aquisição de conhecimento*
  - *trata-se de uma estrutura extremamente interconectada de unidades computacionais, designadas neurónios artificiais, com capacidade de aprendizagem*
    - representando, assim, aproximações simplificadas das redes de neurónios que se encontram no cérebro humano
- Trata-se do método de aprendizagem automática mais poderoso e um dos mais utilizados na ML

O Neurónio

- A componente de processamento fundamental de uma Rede Neuronal (RN) é o neurónio
  - *Estas unidades, densamente interligadas através de um padrão de conexões, desempenham um papel muito simples que consiste em receber sinais das ligações de entrada e com eles calcular um novo valor para ser enviado para a saída*





- Especificamente,
  - *cada terminal de entrada do neurónio, simulando um dendrite, recebe um valor*
  - *Os valores recebidos são então ponderados e somados*
  - *sendo depois o valor do somatório, acrescido de um valor de offset (ou bias – o W_0), usado para calcular o valor de saída do neurónio, em analogia com o processamento realizado pelo corpo celular ou soma*
  - *Esse valor resultante é ainda transformado por uma função específica, denominada função de ativação, que assume normalmente uma das formas típicas ilustradas a seguir*

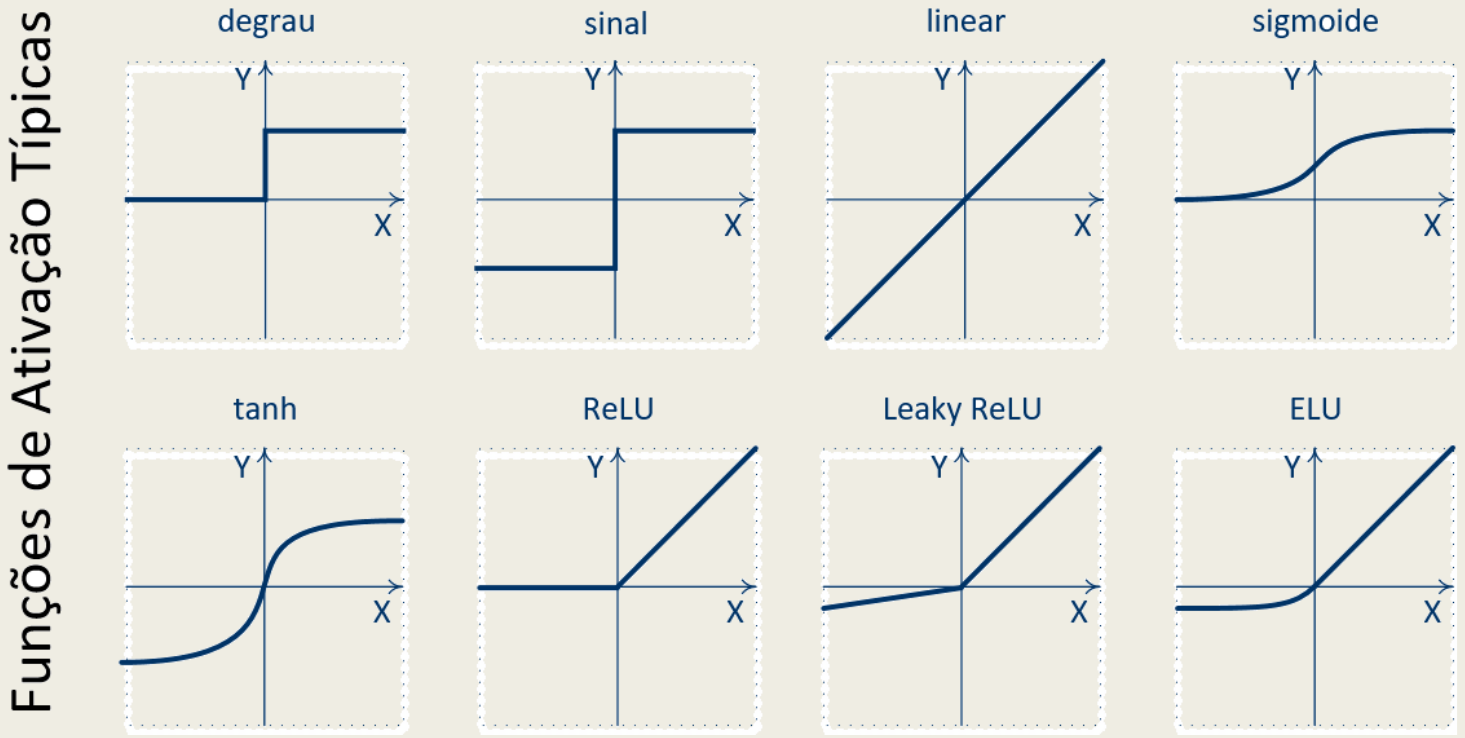
Funções de Ativação

- O *input* da função de ativação pode então ser expresso de forma simples pela seguinte expressão:

$$s = \mathbf{w}^t \cdot \mathbf{x} + w_0 = \sum_{i=1}^{d} w_i x_i + w_0$$

em que $x = [x_1, x_2, x_3, \ldots, x_d]^t$ *é o vetor com o conjunto de* d *entradas do neurónio,* $w = [w_1, w_2, w_3, \ldots, w_d]^t$ *o vetor de pesos associados e* w_0 *o valor de offset ou bias.*

  - *São estes pesos, associados às entradas dos neurónios, que constituem o principal meio de armazenamento de conhecimento numa rede.*



Funções de Ativação

- O *output* do neurónio dependerá então da função de ativação que for escolhida:
  - *Com a função degrau, o sinal será 0 enquanto a entrada s for negativa (neurónio desativado), passando a 1 logo que se torne positiva (neurónio ativado)*
  - *A função sinal converte a entrada nas saídas −1 e +1, em concordância com o sinal, negativo ou positivo, da entrada*
  - *O uso da função linear limitar-se-á a passar para a saída um sinal igual (ou, no mínimo, proporcional) ao de entrada*
  - *Já a função sigmoide representa uma aproximação contínua e diferençável da função degrau*
  - *A função tanh (tangente hiperbólica) é uma versão escalonada da sigmoide, com o intuito de a tornar simétrica em torno da origem (varia entre -1 e 1)*
  - *A função ReLU (unidade linear retificada) permite que só alguns neurónios sejam ativados ao mesmo tempo, tornando a rede esparsa e, por isso, computacionalmente mais eficiente*
  - *A função Leaky ReLU é uma versão melhorada da ReLU, que permite lidar melhor com os gradientes que se deslocam em direção ao zero.*
  - *A função ELU (unidade exponencial e linear) é outra variante da função ReLU, que substitui a 1ª parte da função por uma curva exponencial, com o intuito de aumentar a convergência e a capacidade de generalização da rede*

Escolha da Função de Ativação

Não há uma regra de ouro e a escolha dependerá sempre do problema a ser tratado. No entanto há algumas heurísticas que podem se seguidas, nessa escolha.

- As funções Degrau e Sinal são pouco usadas por não serem diferenciáveis, o que impossibilita o cálculo dos gradientes, necessários à atualização dos coeficientes por *backpropagation* (no processo de aprendizagem)
- Também é pouco usada a função Linear, pois a saída da rede não será mais do que uma combinação linear das entradas, independentemente das camadas que forem usadas, servindo assim unicamente para resolver problemas essencialmente lineares, de pouca complexidade
- Funções do tipo Sigmoide normalmente funcionam melhor em problemas de classificação
- A função ReLU, sendo de aplicação geral, é das mais usadas atualmente
- Uma regra que pode ser seguida, passa por começar-se por usar a função ReLU e só depois passar-se para outras funções de ativação, para ver se os resultados melhoram.

Perceptron

- Um único neurónio com a sua função de ativação em forma de degrau também constitui aquilo que é conhecido por modelo Perceptron,
  - *trata-se da arquitetura mais simples de uma rede neuronal artificial*
  - *O Perceptron foi o modelo pioneiro desenvolvido nas décadas de 1950 e 1960 por Frank Rosenblatt, inspirado em trabalhos anteriores de Warren McCulloch e Walter Pitts.*
    - Dada a sua simplicidade, é ainda hoje útil para a compreensão inicial das redes neurais
  - *Como facilmente se percebe, o Perceptron é um classificador binário,*
    - pode receber como entradas as variáveis explicativas dum problema, e produzir como saída a respetiva classificação: 0 ou 1 (sim ou não).
- A função de ativação simples em forma de degrau, tendo sido historicamente usada nos primeiros modelos de *perceptrons*, passou gradualmente a ser substituída por outras funções mais adequadas.

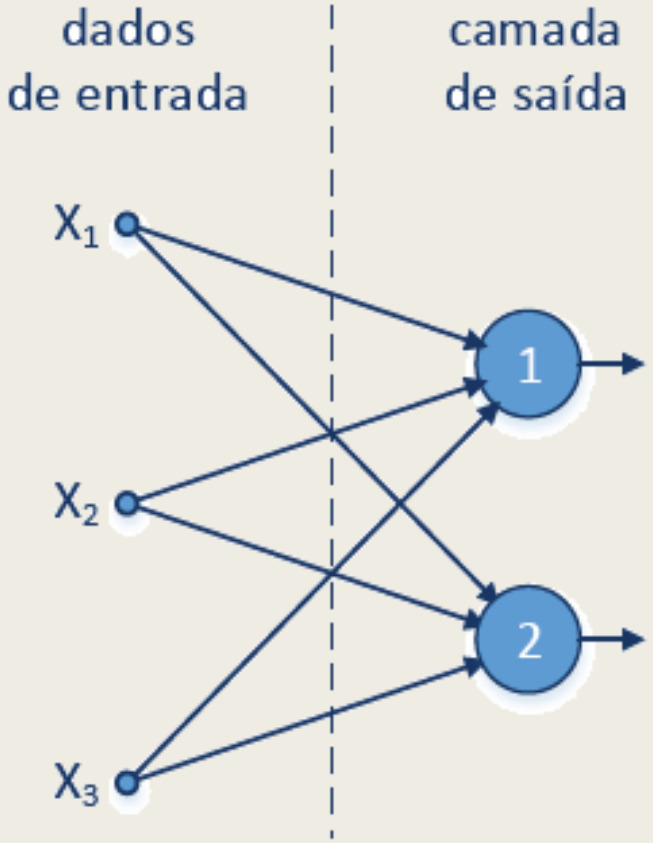
Arquitetura das Redes Neuronais

- Estando os neurónios interligados em rede, as entradas x de um neurónico interior da rede não serão mais do que as saídas dos d neurónios que o precedam e que a ele estejam ligados.
  - *Já se for um neurónio da camada inicial, as entradas x corresponderão às entradas do próprio problema*
  - *e tratando-se de um neurónio da última camada, o seu valor de saída representará um dos valores estimados pela rede.*
- As várias unidades de processamento computacional, ou neurónios artificiais, estão dispostas numa ou mais camadas e interligadas por um grande número de conexões
  - *O número de camadas, o número de neurónios em cada camada, o grau de conectividade e a presença ou não de conexões de retro-propagação definem a* ***arquitetura****, ou* ***topologia****, de uma RN*
    - Existem várias arquiteturas de RN, ou topologias, com características diferentes. Uma das características mais diferenciadoras é se as redes contêm apenas ligações para a frente (redes progressivas ou *feedforward*), ou se existem ciclos de realimentação, formando nesse caso as redes com retropropagação ou redes recorrentes.

Redes neuronais de uma só camada

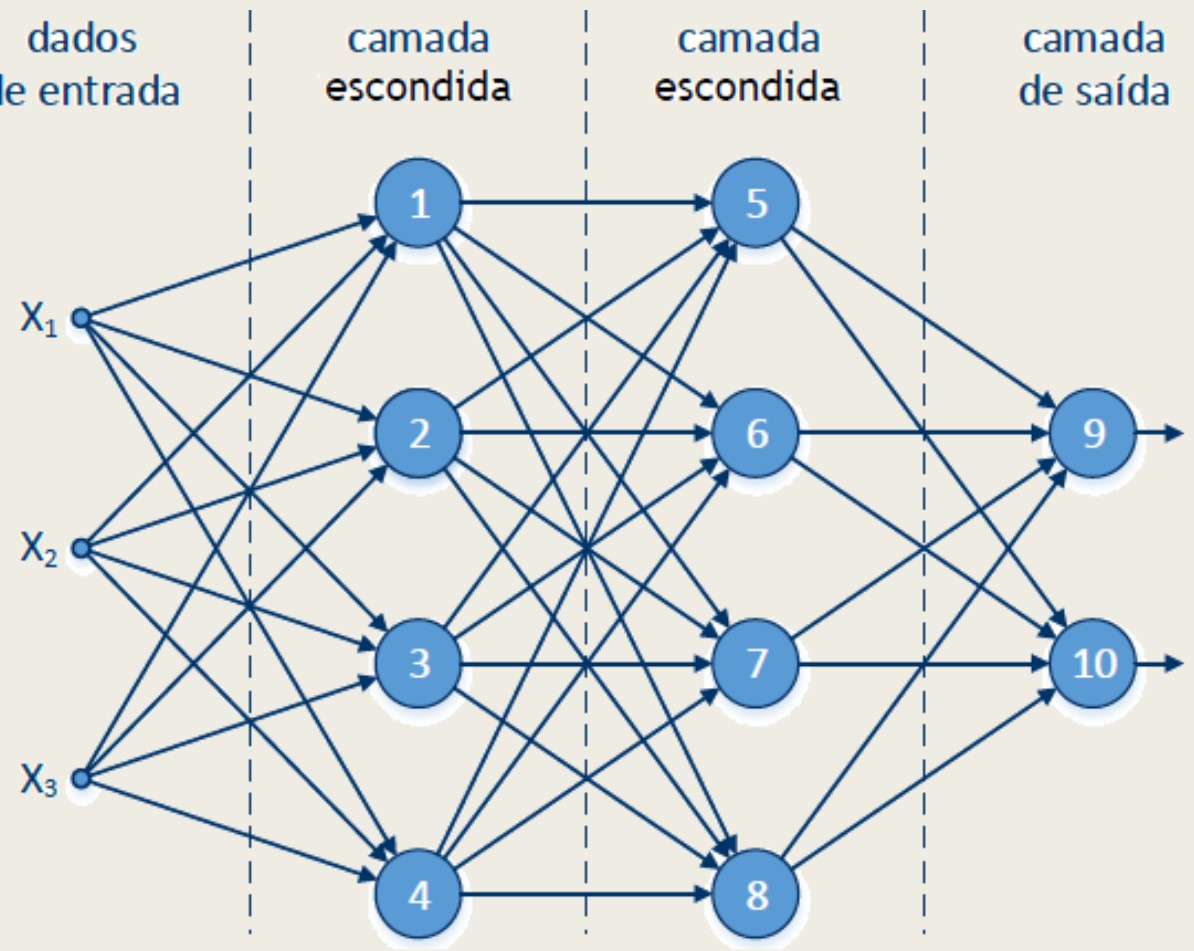
- Na sua forma mais simples uma rede progressiva de uma só camada é composta por
  - *uma "camada" de entrada, cujos valores de saída são fixados externamente,*
  - *e por uma camada de saída, contendo todos os neurónios da rede.*

- Têm a limitação de apenas conseguirem classificar objetos linearmente separáveis (ou seja, só se houver um hiperplano que separe os dados das duas classes)



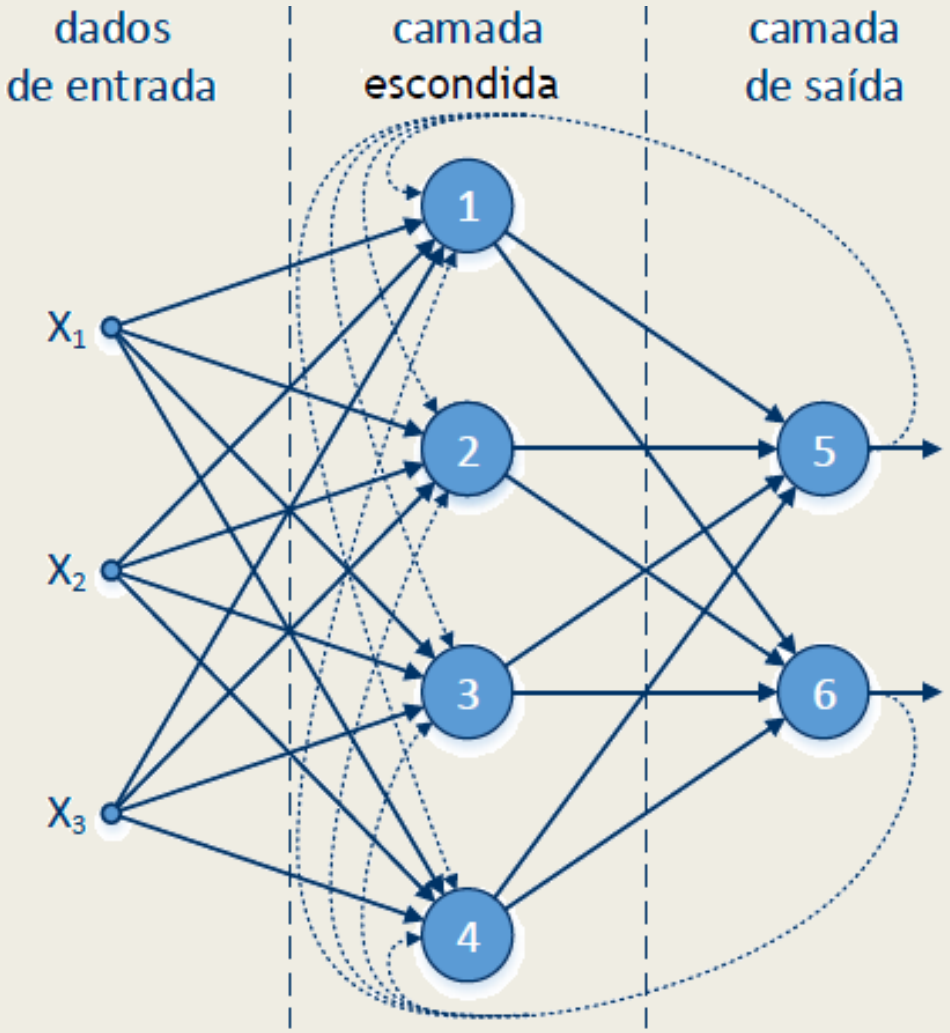
Redes neuronais multicamada

- Distinguem-se pelo facto de possuírem uma ou mais camadas intermédias (designadas escondidas), cujos neurónios são também designados neurónios intermédios
  - *a função destes é de intervir de forma útil entre a entrada e a saída da rede*
  - *Representam uma subárea muito importante da IA, a Deep Learning (aprendizagem profunda), que não desenvolveremos*
    - uma ferramenta disruptiva que atualmente está a revolucionar imenso as nossas vidas.
  - *Ao se acrescentarem camadas intermédias está-se a aumentar a capacidade da rede para modelar funções de maior complexidade.*
- As redes mais frequentemente usadas, desta categoria, são as Multilayer Perceptron (MLP)



Redes neuronais recorrentes

- São realimentadas das saídas para as entradas
  - *Devido a esta característica, respondem a estímulos de forma dinâmica,*
    - ou seja, após se aplicar uma nova entrada, a saída é recalculada e usada para modificar novamente entrada



Configuração de uma Rede Neuronal

- O primeiro passo para que as RN possam induzir um modelo a partir de um conjunto de dados é definir a sua **arquitetura**
  - *A escolha correta da **função de ativação** e da **topologia** da rede é decisiva para uma aprendizagem ou treino bem sucedido*
    - Por exemplo, se a rede for muito pequena poderá ser incapaz de resolver o problema proposto.
    - Por outro lado, se a rede for demasiado grande, poderá revelar-se incapaz de generalizar (*overfitting*) e aumentar drasticamente o tempo de processamento.
  - *A arquitetura da rede é, por isso, determinante na capacidade de processamento e aprendizagem de uma RN.*
    - Geralmente, a arquitetura mais promissora é encontrada através de um processo exaustivo de tentativa e erro, onde diferentes configurações são investigadas, comparadas e avaliadas para selecionar aquela que apresente melhor capacidade preditiva.
      - *Embora seja a abordagem mais utilizada, esta procura cega tem a desvantagem de apresentar um elevado tempo de análise/processamento.*
      - *Alternativamente, pode recorrer-se a algoritmos ou ferramentas que realizem uma procura mais eficiente*

Aprendizagem numa Rede Neuronal

- As conexões entre os neurónios de uma RN possuem pesos associados, os quais desempenham, como já se referiu, um papel essencial na aquisição de conhecimento numa rede.
- São esses pesos que vão assumindo os valores adequados durante o processo de aprendizagem cujo objetivo principal é obter um conjunto de saídas desejadas e consistentes, a partir de um conjunto de entradas. Mais especificamente,
  - *após se atribuir um valor inicial aos pesos das ligações, normalmente por um processo aleatório, todos os vetores de observações presentes num conjunto de treino são apresentados à rede, por regra, mais do que uma vez (várias épocas)*
  - *sempre que um desses vetores é mostrado à rede, os neurónios da primeira camada escondida recebem esses valores e produzem as respetivas saídas, passando-as para os neurónios da camada seguinte,*
    - e assim sucessivamente até se chegar à última camada de neurónios
  - *o valor produzido à saída de cada um dos últimos neurónios é então comparado com o resultado que já é conhecido para o item em causa*
    - a diferença entre os valores verificados à saída e os valores reais conhecidos indica o erro cometido pela rede para o item apresentado, e é esse erro que, de alguma forma, será levado em conta no ajuste dos pesos numa aprendizagem supervisionada
    - durante todo este processo, os pesos da rede convergem de forma gradual para determinados valores, sendo, por regra, necessário apresentar à rede sucessivas épocas dos dados de treino, até que os vetores de entrada produzam as saídas desejadas.

Algoritmos de treino duma Rede Neuronal

- Atualmente existe um conjunto diversificado de algoritmos de treino, que usam regras de aprendizagem diferentes
  - *um dos mais populares para as redes MLP é o algoritmo de de retro propagação (backpropagation), que minimiza o erro quadrático entre os valores reais esperados e os valores produzidos pela rede*
    - A ideia central subjacente a este algoritmo é a de que o erro produzido pelos neurónios das camadas escondidas é estimado retro propagando, sucessivamente, os erros cometidos na camada de saída.
    - Trata-se de um algoritmo iterativo que opera em duas fases, primeiro para a frente (***forward***) e de seguida para trás (***backward***),
      - *Na fase forward, um vetor do conjunto de treino é apresentado à rede, a qual calcula, com os pesos atuais, a saída correspondente e o erro associado;*
      - *depois, na fase backward, o erro é sucessivamente retro propagado, calculando o gradiente do erro cometido em cada neurónio em relação aos pesos das suas entradas,*
        - é a partir da estimação desses erros intermédios que se encontra o ajuste adequado para os pesos da rede.

Resumindo…

Uma RN é definida pela sua arquitetura e pelo algoritmo de aprendizagem

- *Por conseguinte, para se construir uma RN é necessário começar por especificar o tipo e o número de neurónios e particularizar a forma como se interligam, ficando assim definida a sua topologia.*
- *Seguidamente atribuem-se valores iniciais aos pesos das ligações (por norma aleatórios) e inicia-se com a aprendizagem da rede, fase em que cada um dos itens do conjunto de treino é iterativamente apresentado à rede*
  - Para cada um desses itens de treino, os pesos são atualizados de forma a minimizar o erro entre a previsão da rede e o resultado conhecido.
  - O processo é repetido, voltando-se a mostrar sucessivamente à rede os mesmos itens de treino (várias épocas), até os pesos convergirem para valores adequados ou até se ter atingido um número limite de iterações ou épocas (o algoritmo pode não convergir).

Taxa de aprendizagem

- Existe ainda um parâmetro importante normalmente associado ao algoritmo de aprendizagem, denominado taxa de aprendizagem, que permite acelerar ou desacelerar o treinamento da rede
  - *Trata-se, normalmente, de uma constante entre 0 e 1, que o analista pode escolher.*
  - *Quanto maior for essa constante, maior será a alteração dos pesos em cada iteração,*
    - conseguindo-se dessa forma aumentar a velocidade de aprendizagem,
    - mas também aumentar a possibilidade de ocorrerem oscilações que tornem o algoritmo instável ou não convergente
  - *Por outro lado, se a taxa de aprendizagem for demasiado baixa,*
    - a possibilidade do algoritmo convergir é de facto maior,
    - mas, em contrapartida, a rede pode vir a demorar demasiado tempo a aprender.
  - *Uma outra opção interessante, que tenta mitigar as dificuldades enunciadas, passa pela utilização de uma taxa variável que vá diminuindo o seu valor à medida que o algoritmo se vai aproximando do valor de convergência.*

Hiperparâmetros

- Entre os parâmetros mais importantes a afinar numa rede neuronal, na procura do melhor modelo de ML, encontram-se:
  - *o número de camadas escondidas e números de neurónios em cada uma dessas camadas;*
  - *a taxa de aprendizagem;*
  - *o tipo de função de ativação a usar nos neurónios*
  - *e o número máximo de iterações ou épocas usadas no treino da rede.*

Vantagens e desvantagens das Redes Neuronais

Vantagens

- *as RN possuem várias características que justificam o seu elevado desempenho e a sua popularidade*
  - A capacidade de aprendizagem e generalização, o processamento maciçamente paralelo, a flexibilidade, a adaptabilidade e a tolerância a falhas e ruídos, são algumas dessas caraterísticas.

Desvantagens

- *o elevado tempo de processamento*
- *as dificuldades associadas à configuração da arquitetura da rede*
- *a dificuldade de interpretação dos modelos obtidos*
  - tal com as SVM, são usualmente designadas como técnicas do tipo caixa-preta.
- *não consegue lidar com atributos categóricos (requer a aplicação da técnica "one hot encoding")*
- *preditores com diferentes escalas compromete o seu desempenho (requer normalização)*