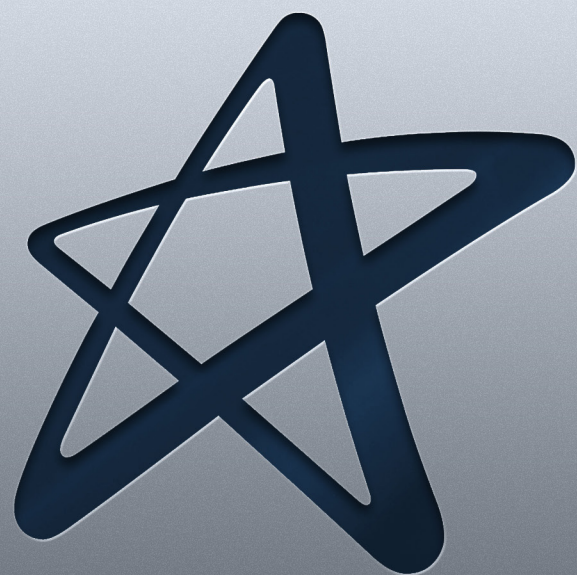


Redes Neurais



Cruzeiro do Sul Virtual
Educação a distância

Material Teórico



Introdução às Redes Neurais

Responsável pelo Conteúdo:

Prof. Dr. Alberto Messias da Costa Souza

Revisão Textual:

Prof. Me. Luciano Vieira Francisco

UNIDADE

Introdução às Redes Neurais



- Redes Neurais Artificiais;
- *Perceptrons*;
- Funções de Ativação e Neurônios Sigmoides.



OBJETIVOS DE APRENDIZADO

- Conhecer os conceitos de neurônios artificiais, as características iniciais de um *perceptron* e os neurônios artificiais sigmoides.
- Conhecer as funções de ativação e as mais comuns encontradas na literatura.

Orientações de estudo

Para que o conteúdo desta Disciplina seja bem aproveitado e haja maior aplicabilidade na sua formação acadêmica e atuação profissional, siga algumas recomendações básicas:



Assim:

- ✓ Organize seus estudos de maneira que passem a fazer parte da sua rotina. Por exemplo, você poderá determinar um dia e horário fixos como seu “momento do estudo”;
- ✓ Procure se alimentar e se hidratar quando for estudar; lembre-se de que uma alimentação saudável pode proporcionar melhor aproveitamento do estudo;
- ✓ No material de cada Unidade, há leituras indicadas e, entre elas, artigos científicos, livros, vídeos e sites para aprofundar os conhecimentos adquiridos ao longo da Unidade. Além disso, você também encontrará sugestões de conteúdo extra no item **Material Complementar**, que ampliarão sua interpretação e auxiliarão no pleno entendimento dos temas abordados;
- ✓ Após o contato com o conteúdo proposto, participe dos debates mediados em fóruns de discussão, pois irão auxiliar a verificar o quanto você absorveu de conhecimento, além de propiciar o contato com seus colegas e tutores, o que se apresenta como rico espaço de troca de ideias e de aprendizagem.

Redes Neurais Artificiais

Os primeiros estudos de redes neurais datam de 1943, em artigos de McCulloch e Pitts (REIS; ALBUQUERQUE; CASTRO, 2001), em que sugeriam a construção de uma máquina inspirada no cérebro humano.

Segundo Rojas (1996), Redes Neurais Artificiais (RNA) são modelos de processamento de informações inspirados em uma estrutura natural, o cérebro humano. Analogamente ao cérebro humano, uma rede neural é composta por unidades de processamento chamadas de **neurônios**, intimamente interconectadas e capazes de adquirir conhecimento ao longo do tempo.

Segundo Braga, Ludemir e Carvalho (2000), as RNA consistem em determinado número de neurônios, que calculam certas funções matemáticas, comumente não lineares, dispostas em uma ou mais camadas, interligadas por considerável número de conexões, geralmente unidirecionais. Na maioria dos modelos essas conexões estão associadas a pesos, os quais armazenam o conhecimento representado no modelo e servem para ponderar a entrada recebida por cada neurônio da rede.

Na Figura 1 se pode ver a representação de um neurônio artificial proposto em McCulloch em 1943, onde os sinais (X_1, X_2, \dots, X_p) são apresentados na entrada, de modo que cada sinal é multiplicado por um número, ou peso (W_1, W_2, \dots, W_p) que indica a sua influência na saída da unidade e posteriormente é feita a soma ponderada dos sinais, produzindo um nível de atividade; assim, se esse nível de atividade exceder determinado limite – ou o que podemos chamar de *threshold* –, a unidade produz uma resposta de saída específica.

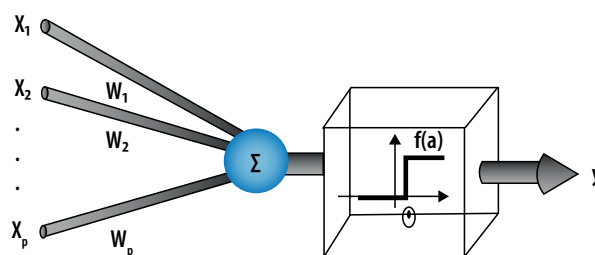


Figura 1 – Neurônio artificial

A definição dos pesos em cada neurônio é feita pelo processo de treinamento ou aprendizagem da rede neural. Inicialmente, a rede deve passar por uma fase de aprendizagem, onde um conjunto de exemplos é apresentado para a rede, a qual deverá extrair automaticamente as características necessárias para representar a informação fornecida.

Essa etapa de treinamento é um processo iterativo, no qual os pesos de cada neurônio são ajustados, existindo diversos algoritmos de aprendizado que se diferenciam basicamente pela maneira como os valores de pesos são modificados (BRAGA; LUDEMIR; CARVALHO, 2000).

Conforme visto em Russel e Norving (2004), existem duas categorias principais de redes neurais: redes acíclicas ou de alimentação direta, e redes cíclicas ou redes

recorrentes. Uma rede de alimentação direta representa a função de sua entrada atual; deste modo, não tem nenhum estado interno além dos pesos propriamente ditos. As redes recorrentes utilizam as suas saídas para alimentar de volta as suas próprias entradas.

No treinamento recorrente ou supervisionado, os vetores de entrada são inseridos, processados pelos neurônios e posteriormente as saídas são associadas às respostas corretas, a fim de modificar os pesos dos neurônios a cada iteração, ao se comparar as respostas aos resultados esperados.

No treinamento não supervisionado ou de alimentação direta, o conjunto de entrada é fornecido para que a rede consiga extrair propriedades de acordo com as representações internas sem que haja retroalimentação.

A implementação de redes neurais é uma das formas mais populares e efetivas de sistemas de aprendizagem, por possuir habilidade para executar em ambientes distribuídos, tolerar entradas ruidosas e aprender (RUSSEL; NORVING, 2004).

Segundo Reis, Albuquerque e Castro (2001), pode-se observar que a técnica de redes neurais artificiais possui considerável potencial e tem sido constantemente utilizada para o reconhecimento de padrões. Seguem algumas vantagens de se utilizar redes neurais para a tarefa de reconhecimento de padrões:

- Adaptabilidade: habilidade de se adaptar às novas informações;
- Velocidade: via o paralelismo maciço;
- Tolerância a falhas: capacidade de oferecer boas respostas mesmo com falta, confusão ou dados ruidosos;
- Otimalidade: taxa de erro em sistemas de classificação.

Os sistemas neurais oferecem uma abordagem adequada para o tratamento de aplicações da natureza do negócio da organização mencionada no estudo de caso. Uma RNA pode ser treinada utilizando grandes quantidades de exemplos significativos, o que contribui para uma avaliação bem ou mal sucedida dos históricos dos clientes e presente nas carteiras.

A rede neural é capaz de encontrar relações entre as informações e incorporar os aspectos subjetivos do processo de avaliação da organização. De acordo com M. Filho, Carvalho e Matias (1996), pode-se observar que as redes neurais são eficientes na resolução de problemas relacionados à análise de risco de crédito.

Na literatura, observa-se que os neurônios artificiais também são chamados de *perceptrons*, de modo que os analisaremos a seguir.

Perceptrons

Um *perceptron* funciona exatamente como o proposto na Figura 1, mas nos aprofundaremos em seu entendimento.

O *perceptron* recebe várias entradas binárias, x_1 , x_2 , x_3 e produz uma única saída binária, conforme a Figura 2:

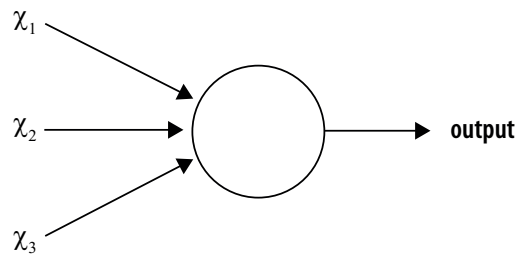


Figura 2 – Representação gráfica do perceptron com três entradas

Neste exemplo, o perceptron possui três entradas, x_1 , x_2 , x_3 . Em geral, poderia ter mais ou menos valores de entrada. A regra simples para calcular a saída é introduzir pesos W_1 , W_2 ... em números reais que expressem a importância das respectivas entradas para a saída. A saída do neurônio 0 ou 1 será determinada se a soma ponderada $\sum_j W_j x_j$ for menor ou maior que algum valor-limite. Assim como os pesos, o limiar é um número real, que é um parâmetro do neurônio. Segue a Figura 3 com o modelo matemático:

$$resultado = \begin{cases} 0 & \text{se } \sum_j W_j x_j \leq \text{limiar} \\ 1 & \text{se } \sum_j W_j x_j > \text{limiar} \end{cases}$$

Modelo matemático de um perceptron

Este é o modelo matemático básico e simples do funcionamento de um neurônio artificial, ou *perceptron*.

Vamos a um exemplo prático de uma situação hipotética que você já poderia ter passado: imaginemos que o final de semana está chegando e você foi convidado(a) para um evento em sua família. Você gosta destes eventos familiares e está tentando decidir se vai ou não. Você pode tomar a sua decisão ponderando três fatores:

1. O tempo está bom?
2. O(A) seu(sua) namorado(a) quer acompanhá-lo(a)?
3. É possível chegar de transporte público no evento? Dado que você não possui um automóvel.

Podemos representar esses três fatores por variáveis binárias correspondentes x_1 , x_2 e x_3 . Por exemplo, teríamos $x_1 = 1$ se o tempo estiver bom, ou $x_1 = 0$ se o tempo estiver ruim. A variável $x_2 = 1$ se o(a) seu(sua) namorado(a) quiser ir e $x_2 = 0$ caso não. Da mesma forma, $x_3 = 1$ para a possibilidade de ir de transporte público e $x_3 = 0$ caso não.

Pensemos, então, nos pesos de cada uma das variáveis: suponha que você adore esses eventos, mesmo que o(a) seu(sua) namorado(a) não esteja interessado(a) e o encontro seja difícil de ocorrer com frequência. Porém, você não gosta de mau tempo,

de modo que não há como ir se o tempo estiver ruim. Para essa situação hipotética, podemos usar *perceptrons* para modelar esse tipo de tomada de decisão. Uma maneira de fazer isso é escolher um peso $W_1 = 6$ para o clima e $W_2 = 2$ e $W_3 = 2$ para as outras condições. O maior valor de W_1 indica que o clima é consideravelmente importante para você, mais do que o(a) seu(sua) namorado(a) se juntar a você ou a proximidade do transporte público.

Por fim, suponha que você escolha um limite de 5 para o *perceptron*. Com essas opções, o *perceptron* implementará o modelo de tomada de decisão desejado, produzindo 1 sempre que o tempo estiver bom, e 0 sempre que o tempo estiver ruim. Não fará diferença para a saída se o(a) seu(sua) namorado(a) quiser ir, ou se o transporte público estiver próximo.

Variando os pesos e o limiar, podemos obter diferentes modelos de tomada de decisão. Por exemplo, suponha que escolhemos um limite de 3: o *perceptron* decide, então, que você deveria ir ao evento sempre que o tempo estivesse bom, ou quando o evento estivesse próximo ao transporte público e o(a) seu(sua) namorado(a) estivesse disposto(a) a lhe acompanhar. Em outras palavras, seria um modelo diferente de tomada de decisão. Portanto, diminuir o limiar significa que você está mais disposto(a) a ir ao evento.

O *perceptron* não é um modelo completo de tomada de decisão humana, mas o que este exemplo ilustra é como um *perceptron* pode pesar diferentes tipos de evidência para a tomada de decisão.

Deve parecer plausível que uma rede complexa de *perceptrons* possa tomar decisões sutis ou com maiores níveis de complexidade, de modo que a Figura 4 ilustra uma rede de *perceptrons*:

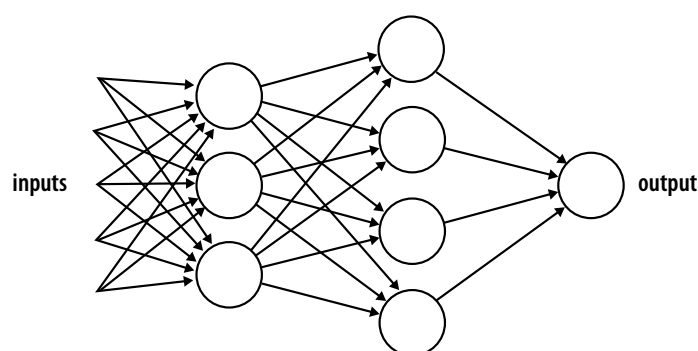


Figura 3 – Ilustração de uma rede de *perceptrons*

Nesta rede, a primeira coluna de perceptrons, ou primeira camada de *perceptrons*, toma três decisões simples, ponderando unicamente as evidências de entrada. Já cada um dos *perceptrons* na segunda camada toma decisão ponderando os resultados da primeira camada de tomada de decisão.

Assim, um perceptron na segunda camada pode tomar uma decisão em nível mais complexo e abstrato do que os *perceptrons* na primeira camada. Por sua vez, decisões ainda mais complexas podem ser tomadas pelo *perceptron* na terceira camada.

Portanto, uma rede de *perceptrons* de várias camadas pode se envolver em sofisticadas tomadas de decisão – analogamente ao que ocorre em um cérebro, que é uma complexa rede neural com bilhões de interligações e camadas de neurônios.

Observe ainda que, na Figura 4, os *perceptrons* das duas primeiras colunas possuem várias saídas, diferentemente do que foi definido anteriormente. Na verdade, ainda são de saída única. Ocorre que as múltiplas setas de saída são apenas uma maneira útil de indicar que a saída de um *perceptron* é usada como entrada para vários outros *perceptrons* – é menos pesado do que desenhar uma única linha de saída que depois se divide.

A condição de limiar mostrada anteriormente pode ser simplificada utilizando um produto escalar, $w \cdot x = \sum_j w_j x_j$, onde W e X são vetores em que, na verdade, são W os pesos e X as entradas. Outra alteração é incluir o conceito de bias do *perceptron* na condição do limiar, que aqui chamaremos apenas de b – segue a Figura 5 com a nova notação do *perceptron*:

$$output = \begin{cases} 0 & \text{se } w \cdot x + b \leq 0 \\ 1 & \text{se } w \cdot x + b > 0 \end{cases}$$

Reescrita da notação do *perceptron*

Podemos definir bias como medida que permite a saída do *perceptron* como necessariamente 1, ou seja, ajusta-se a saída do *perceptron* com o valor do bias.

Outra característica interessante e que pode ser incorporada aos *perceptrons* corresponde às funções computacionais *and*, *or* e *nand*, ou, respectivamente, *e*, *ou* e *não e*. Segue um exemplo: imagine que temos um *perceptron* com duas entradas, cada uma com o seu peso -2 e um bias geral 3 – segue a Figura 6, com a representação gráfica deste *perceptron*:

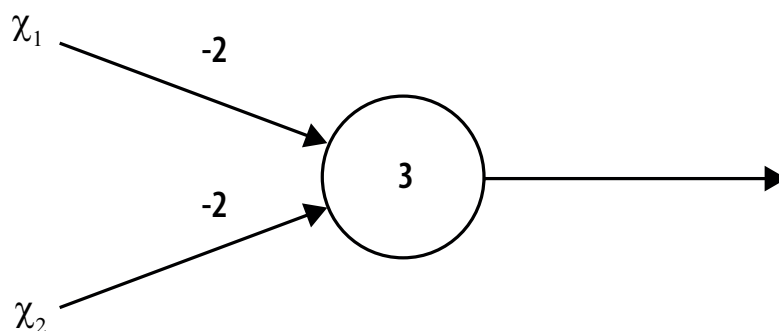


Figura 4 – *Perceptron* de exemplo

Observe, então, que a entrada $0\ 0$ produzirá a saída 1, sendo: $(-2) \times 0 + (-2) \times 0 + 3 = 3$, onde 3 é positivo. Aplicando semelhantemente as entradas $0\ 1$ e $1\ 0$ possuirão a saída 1; porém, se a entrada for $1\ 1$, produzirá a saída 0, veja: $(-2) \times 1 + (-2) \times 1 + 3 = -1$, ou seja, este *perceptron* implementa uma porta *nand*.

Nesse *perceptron* de exemplo observamos que podem ser utilizados para calcular funções lógicas simples, mas com uma rede de *perceptrons* pode-se calcular funções lógicas complexas. Ademais, vale destacar que um *perceptron* não se restringe a uma função de porta *nand*.

Podemos utilizar os *perceptrons* para conceber algoritmos de aprendizado que possam ajustar automaticamente os pesos e bias em uma rede de neurônios artificiais. Tal afinação ou aprendizagem ocorre a partir de estímulos externos, sem a intervenção direta do programador. Esses algoritmos de aprendizagem nos permitem usar os neurônios artificiais de maneira diferente de uma porta *nand* convencional, ou seja, não simplesmente usar circuitos com portas lógicas, as redes neurais podem aprender a resolver problemas, o que não seria possível com circuitos lógicos convencionais.

Funções de Ativação e Neurônios Sigmoides

Outra característica importante a ser destacada no processo de aprendizagem de um neurônio artificial é a função de ativação ou função sigmoide (não linear) que, na verdade, faz pequenas alterações nos valores dos pesos ou dos bias, de modo a se aproximar do resultado correto ou esperado – neste caso, pensando em apenas um único *perceptron*.

Ao considerar uma rede de neurônios, o restante da rede pode ser significativamente alterado com uma pequena mudança de peso ou bias em um único neurônio, o que dificulta o processo de aprendizagem ou de alterações graduais dos pesos e bias.

Uma função – ou um neurônio – sigmoide é dada por uma função matemática diferente da notação matemática inicialmente proposta para um *perceptron* – segue a Figura 7 com a função matemática:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Função matemática sigmoide

Analogamente a um *perceptron*, onde as entradas são X_1 e X_2 – com os respectivos pesos W_1 e W_2 e um bias b –, temos o padrão ilustrado na Figura 8:

$$\frac{1}{1 + \exp(-\sum_j w_j x_j - b)}$$

Analogia de neurônio sigmoide com a função de neurônio *perceptron*

Note que há diferença de notação, mas que são definições relativamente próximas.

Segue a Figura 9 com uma representação gráfica de uma função sigmoide, diferentemente de uma função linear, que mais se assemelha às saídas de um *perceptron*:

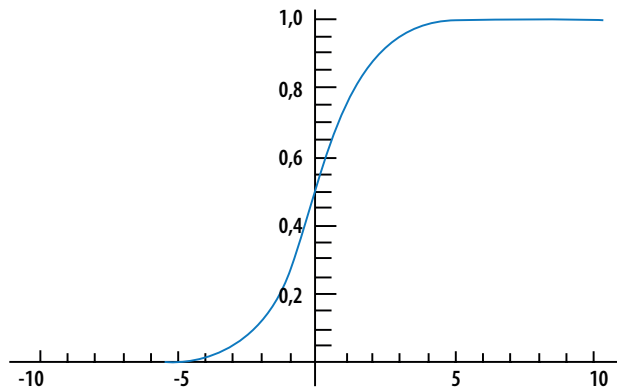


Figura 5 – Representação gráfica da saída de uma função sigmoide

Observa-se, então, que através de uma função sigmoide a saída é mais suavizada. Um neurônio sigmoide possui comportamento aproximado ao de um neurônio *perceptron* e o mais importante a observar no gráfico é a forma da plotagem suavizada, o que permitirá ao neurônio sigmoide a suavidade nas alterações dos pesos e bias que, por sua vez, teriam mudanças mais sutis ou controladas em suas saídas mais sutis, facilitando o processo de aprendizagem em uma rede de neurônios.

Vale destacar ainda que as saídas dos neurônios sigmóides não se limitam a 0 e 1, podendo retornar valores reais entre 0 e 1, ou seja, valores como 0,15234 e 0,98816, o que pode ser mais útil em diversos problemas computacionais.

Outra dica de utilização seria a seguinte: qualquer valor menor que 0,5 pode ser classificado como 0 ou falso e valor maior que 0,5 pode ser classificado como 1 ou verdadeiro.

Na literatura não há apenas esses dois tipos de função de ativação para um neurônio, podendo-se utilizar outros tipos de funções de ativação, pois uma função sigmoide pode ter saídas indesejadas, dependendo do tipo de entrada como, por exemplo, estacionar as saídas em zero, de acordo com a entrada, o que pararia o processo de aprendizagem. Podemos encontrar outras funções como:

- **Tanh:** que é próxima de uma função sigmoide, porém, as suas saídas não têm apenas a variação de número entre 0 e 1, possuindo saídas reais entre -1 e 1, dada pela função: $\tanh(x) = 2 \sigma(2x) - 1$;
- **ReLU:** função de unidade linear retificada, que não necessariamente é uma função linear; ou seja, uma variação nos pesos ou bias pode ser propagada para neurônios em outras camadas e ativá-los ou não, a depender da proximidade da saída: se estiver em zero, ativará ou não determinado neurônio, sendo dada pela função: $f(x) = \max(0, x)$;
- **Leaky ReLU, ou “com vazamento”:** ReLUs com vazamento são tentativas de corrigir o problema de “ReLU moribundo”. Em vez de a função ser zero quando $x < 0$, uma ReLU com vazamento terá uma pequena inclinação negativa (de 0,01 ou mais), de modo a mitigar os problemas encontrados na função ReLU inicialmente proposta. É dada pela função: $f(x) = 1 (x < 0) (\alpha x) + 1 (x \geq 0) (x)$;

- **Maxout:** o neurônio *maxout* generaliza a função ReLU e a sua versão *leaky ReLU*. O neurônio *maxout* calcula a função: $\max(w_1^T x + b_1, w_2^T x + b_2)$. Ademais, desfruta de todos os benefícios de uma unidade ReLU (regime linear de operação, sem saturação) e não possui as suas desvantagens (ReLU moribundo).

Material Complementar

Indicações para saber mais sobre os assuntos abordados nesta Unidade:



Livros

Inteligência artificial aplicada: uma abordagem introdutória

Leia o quinto capítulo do livro Inteligência artificial aplicada: uma abordagem introdutória.

Inteligência artificial

Leia o décimo primeiro capítulo do livro Inteligência artificial.



Leitura

Lista abrangente de funções de ativação em redes neurais com prós e contras

<https://bit.ly/2YK8vEv>

Redes neurais: um breve histórico

<https://bit.ly/2YFbHkQ>

Referências

BRAGA, A. P.; LUDEMIR, T. B.; CARVALHO, A. P. L. F. **Redes neurais artificiais – teoria e aplicações**. Rio de Janeiro: LTC, 2000.

M. FILHO, E. F.; CARVALHO, A. C. P. L. F.; MATIAS, A. B. **Utilização de redes neurais artificiais na análise de risco de crédito a pessoas físicas**. São Carlos, SP; Ribeirão Preto, SP: SCE/ICMSC/USP; FEA/RP/USP, 1996.

REIS, C. F.; ALBUQUERQUE, M. P.; CASTRO, S. B. **Introdução ao reconhecimento de padrões utilizando redes neurais**. Rio de Janeiro: Centro Brasileiro de Pesquisas Físicas, 2001.

ROJAS, R. *Neural networks – a systematic introduction*. Berlin: Springer Verlag, 1996.

RUSSEL, S. J.; NORVING, P. **Inteligência artificial**. 2. ed. [S.l.]: Campus, 2004.



Cruzeiro do Sul
Educatonal