



Redes Neurais Artificiais

AULA 02 – Introdução às RNA's

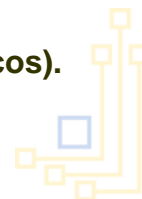
Prof. Ivan Nunes da Silva



1. Sistemas Inteligentes

O que são e qual sua constituição?

- **Conjunto de ferramentas computacionais que tentam simular alguns mecanismos inteligentes encontrados na natureza (seres humanos).**
- **É constituído por cinco grandes áreas:**
 - Redes Neurais Artificiais.
 - Sistemas de Inferência Nebulosos.
 - Computação Evolutiva (Alg. Genéticos).
 - Inteligência Coletiva (Swarm).
 - Agentes Autônomos.



2. Redes Neurais Artificiais (RNA)

Definição do ponto de vista de engenharia

- São modelos computacionais inspirados nos mecanismos de aprendizagem do cérebro humano.
- São modelos computacionais que tentam emular a forma com que o cérebro resolve problemas.

3

2. Redes Neurais Artificiais (RNA)

Inspiração no cérebro humano

- **O processamento de informações no cérebro humano é altamente complexo, não-linear e paralelo.**
 - O cérebro é constituído por aproximadamente 100 bilhões de neurônios (10^{11}).
 - Cada neurônio é interligado em média com outros 6.000 neurônios (60 trilhões de sinapses).
 - Número de neurônios equivale à quantidade de estrelas da via Láctea.
 - É um dos sistemas mais complexos que é atualmente conhecido pelo ser humano.

4

2. Redes Neurais Artificiais (RNA)

Comparação entre computador e cérebro

- **Tipo de Processamento**

- Computador → seqüencial (serial).
- Cérebro → paralelo.

- **Quantidade e Complexidade**

- Computador → um ou poucos processadores.
- Cérebro → $\approx 10^{10}$ neurônios; $\approx 10^3$ conexões/neurônio.

- **Velocidade de Processamento**

- Computador → $\approx 10^{-9}$ segundos.
- Cérebro → $\approx 10^{-6}$ segundos.

- **Eficiência Energética**

- Computador → $\approx 10^{-16}$ Joules (por operação/segundo).
- Cérebro → $\approx 10^{-6}$ Joules (por operação/segundo).

5

2. Redes Neurais Artificiais (RNA)

Principais características das RNA (I)

- **Adaptação por Experiência**

Parâmetros internos da rede são ajustados a partir da apresentação sucessiva de exemplos (amostras e medidas).

- **Capacidade de Aprendizado**

Aplicação de método de treinamento possibilita a rede extrair o relacionamento existente entre variáveis que compõem a aplicação.

- **Habilidade de Generalização**

Após o processo de treinamento, a rede é capaz de generalizar o conhecimento adquirido, possibilitando a estimação de soluções que eram até então desconhecidas

- **Organização de Dados**

Baseada em características marcantes de um conjunto de dados a respeito do processo, a rede é capaz de realizar sua organização interna para agrupamento de amostras que são similares/comuns.

6

2. Redes Neurais Artificiais (RNA)

Principais características das RNA (II)

- **Tolerância a Falhas**

Devido ao elevado nível de interconexões entre neurônios artificiais, a rede torna-se um sistema tolerante a falhas quando parte de sua estrutura interna for sensivelmente corrompida.

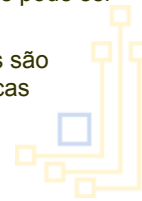
- **Armazenamento Distribuído**

O conhecimento dentro da rede é realizado de forma distribuída entre as sinapses dos neurônios artificiais, permitindo-se então robustez frente a eventuais neurônios que se tornaram inoperantes.

- **Facilidade de Prototipagem**

A implementação da maioria das arquiteturas neurais pode ser facilmente prototipadas em hardware ou em software

Após o processo de treinamento, os seus resultados são normalmente obtidos por algumas operações matemáticas elementares.



7

2. Redes Neurais Artificiais (RNA)

Histórico das redes neurais

- 1943 - Trabalho de McCulloch e Pitts. } Romântica
- 1956 - Reunião no Dartmouth College. } Barroca
- 1969 - Publicação do Livro Perceptron. } Trevas
- 1986 - Trabalhos de Rumelhart, Hopfield, Kohonen e outros. }
- 1987 - Primeiro Congresso Internacional. } Contemporânea



8

3. Potenciais Aplicações de RNA

Aproximador de Funções & Controle de Processos

- **Aproximador de Funções**

Mapeiam o relacionamento entre variáveis de um sistema a partir de um conjunto conhecido de seus valores representativos.

Envolvem normalmente o mapeamento de processos cuja modelagem por técnicas convencionais são de difícil obtenção.

Necessitam de domínio específico para variáveis de entrada.

Exemplo de aplicações: resolução de diversos tipos de problemas (em diferentes áreas do conhecimento).

- **Controle de Processos**

Consistem em identificar ações de controle que permitam o alcance dos requisitos de qualidade, de eficiência e de segurança do processo.

Exemplo de aplicações: controles empregados em robótica, aeronaves, elevadores, eletrodomésticos, satélites, etc.

9

3. Potenciais Aplicações de RNA

Classificação de Padrões & Agrupamento de Dados

- **Classificação de Padrões**

O objetivo desta aplicação consiste de associar um padrão de entrada (amostra) para uma das classes previamente definidas.

O problema a ser tratado possui um conjunto discreto e conhecido das possíveis saídas desejadas.

Exemplos de aplicações: reconhecimento de imagens, voz, escrita, etc.

- **Agrupamento de Dados (Clusterização)**

O alvo aqui consiste da identificação e detecção de similaridades e particularidades entre as diversas amostras do processo, objetivando-se agrupamento dos mesmos.

Exemplos de aplicações: identificação automática de classes (como em problemas de diagnóstico médico), compressão de dados e garimpagem de dados (*data mining*).

10

3. Potenciais Aplicações de RNA

Sistemas de Previsão & Otimização de Sistemas

● Sistemas de Previsão

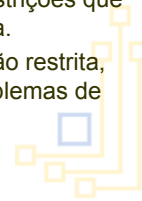
O objetivo consiste em estimar valores futuros de um processo levando-se em consideração diversas medidas prévias observadas em seu domínio.

Exemplos de aplicações: previsão de demanda de energia, previsões de mercados financeiros, previsões climáticas, séries temporais, etc.

● Otimização de Sistemas

O alvo consiste em minimizar ou maximizar uma função objetivo (custo), obedecendo-se também eventuais restrições que são impostas para o correto mapeamento do problema.

Exemplos de aplicações: problemas de otimização restrita, otimização combinatorial, programação dinâmica, problemas de seqüenciamento de produção, etc.



11

3. Potenciais Aplicações de RNA

Sistemas de Previsão & Otimização de Sistemas

● Memórias Associativas

A missão consiste em recuperar padrões corretos mesmo se os seus elementos constituintes forem apresentados de forma incompleta ou distorcida.

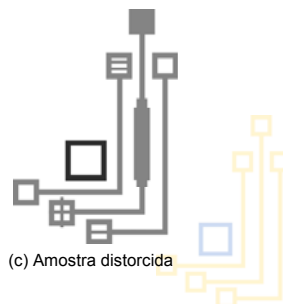
Exemplos de aplicações: processamento de imagens, transmissão de sinais, identificação de caracteres manuscritos, etc.



(a) Padrão original



(b) Amostra incompleta



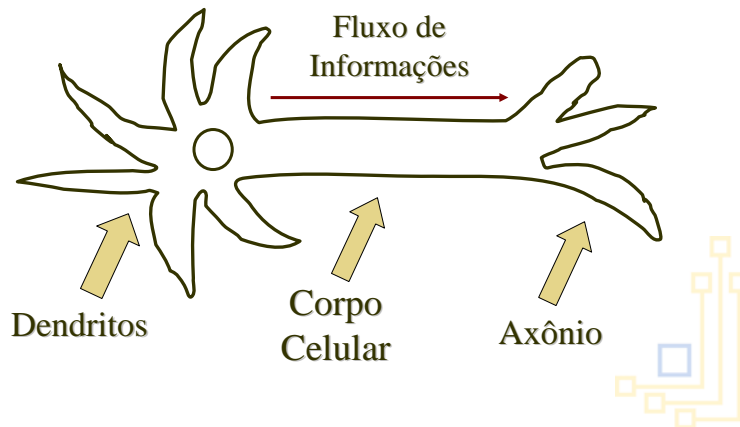
(c) Amostra distorcida

12

4. Neurônio Biológico

Características da célula nervosa

- Representação do neurônio biológico:

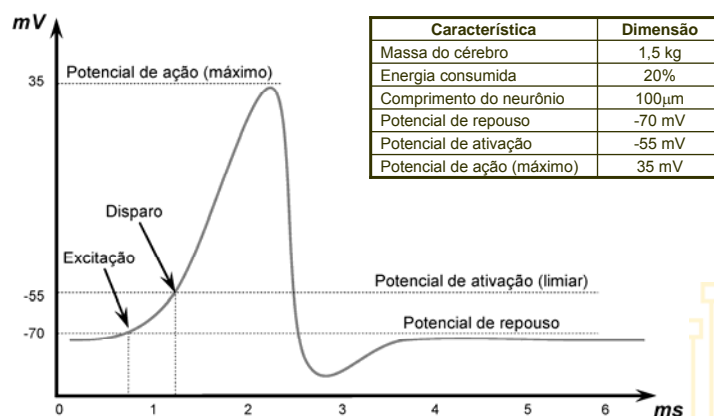


13

4. Neurônio Biológico

Variação do potencial de ação do neurônio

- Etapas de variação (potencial de ação):

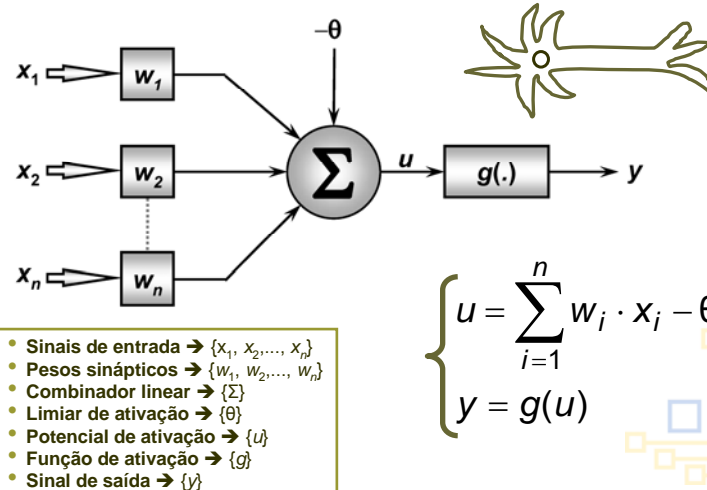


14

5. Neurônio Artificial

Representação do neurônio artificial

● Modelo matemático do neurônio artificial:

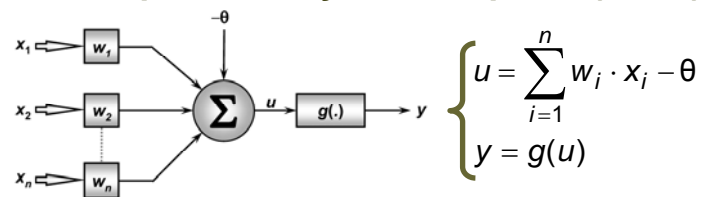


15

5. Neurônio Artificial

Resumo do funcionamento do neurônio artificial

● Passos para obtenção da resposta (saída):



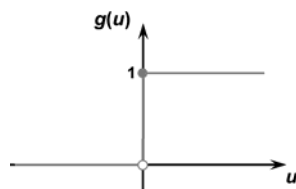
- Apresentação de um conjunto de valores que representam as variáveis de entrada do neurônio.
- Multiplicação de cada entrada do neurônio pelo seu respectivo peso sináptico.
- Obtenção do potencial de ativação produzido pela soma ponderada dos sinais de entrada, subtraindo-se o limiar de ativação.
- Aplicação de uma função de ativação apropriada, tendo-se como objetivo limitar a saída do neurônio.
- Compilação da saída a partir da aplicação da função de ativação neural em relação ao seu potencial de ativação.

16

6. Funções de Ativação Neural

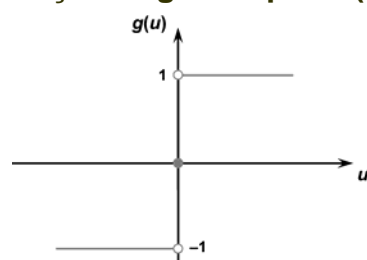
Funções parcialmente diferenciáveis (I)

- Função Degrau (Heavyside):



$$g(u) = \begin{cases} 1, & \text{se } u \geq 0 \\ 0, & \text{se } u < 0 \end{cases}$$

- Função Degrau Bipolar (Sinal):



$$g(u) = \begin{cases} 1, & \text{se } u > 0 \\ 0, & \text{se } u = 0 \\ -1, & \text{se } u < 0 \end{cases}$$

$$g(u) = \begin{cases} 1, & \text{se } u \geq 0 \\ -1, & \text{se } u < 0 \end{cases}$$

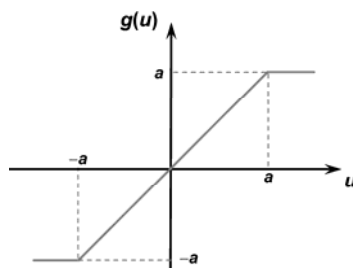
Para ser usada em classificação de padrões.

17

6. Funções de Ativação Neural

Funções parcialmente diferenciáveis (II)

- Função Rampa Simétrica:



$$g(u) = \begin{cases} a, & \text{se } u > a \\ u, & \text{se } -a \leq u \leq a \\ -a, & \text{se } u < -a \end{cases}$$

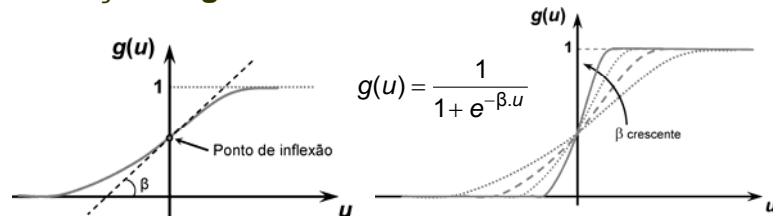
Os valores retornados são iguais aos próprios valores dos potenciais de ativação quando os mesmos estão definidos no intervalo $[-a, a]$, limitando-se aos valores limites em caso contrário.

18

6. Funções de Ativação Neural

Funções totalmente diferenciáveis (I)

● Função Logística:



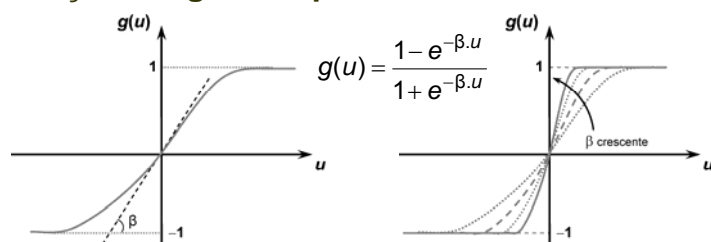
- Parâmetro β é uma constante real relacionada à inclinação da função logística frente ao seu ponto de inflexão.
- Resultado de saída assumirá sempre valores reais entre zero e um.
- Formato geométrico tende a ser similar àquele da função degrau quando β for muito elevado (tender ao infinito).

19

6. Funções de Ativação Neural

Funções totalmente diferenciáveis (II)

● Função Tangente Hiperbólica:



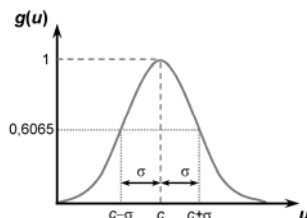
- Parâmetro β está também relacionado à inclinação da função logística frente ao ponto de inflexão.
- Resultado de saída assumirá sempre valores reais entre -1 e 1.
- Formato geométrico tende a ser similar àquele da função degrau bipolar quando β for muito elevado (tender ao infinito).

20

6. Funções de Ativação Neural

Funções totalmente diferenciáveis (III)

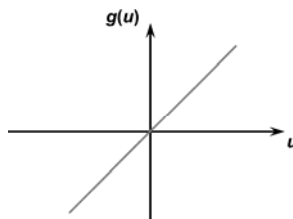
- **Função Gaussiana:**



$$g(u) = e^{-\frac{(u-c)^2}{2\sigma^2}}$$

- Parâmetro c define o centro (média) da função.
- Parâmetro σ^2 especifica a variância (espalhamento).

- **Função Linear (Identidade):**



$$g(u) = u$$

- Produz resultados idênticos aos valores do potencial de ativação $\{u\}$.

21

7. Arquiteturas de RNA's

Arquitetura, Topologia e Treinamento

- **Arquitetura de Redes Neurais:**

Define a forma como os seus diversos neurônios estão arranjados, ou dispostos, uns em relação aos outros.

Os arranjos são essencialmente estruturados através do direcionamento do fluxo sináptico.

- **Topologia de Redes Neurais:**

Define as diferentes formas de composições estruturais que uma rede poderá assumir, frente a uma determinada arquitetura.

Exemplo: dada uma determinada arquitetura, uma das redes pode ser composta de 10 neurônios e a outra de 20 neurônios.

- **Processo de Treinamento:**

Consiste da aplicação de um conjunto de passos ordenados com o intuito de ajustar os pesos e os limiares de seus neurônios.

Visa então sintonizar a rede para que as suas respostas estejam próximas dos valores desejados.

Exprime o próprio algoritmo de aprendizagem da rede.

22

7. Arquiteturas de RNA's

Estruturação de camadas

- **Camada de Entrada:**

Responsável pelo recebimento de informações (dados), sinais, características ou medições advindas do meio externo.

Valores recebidos são geralmente normalizados em relação aos valores limites produzidos pelas funções de ativação.

Esta normalização implica numa melhor precisão numérica frente às operações matemáticas realizadas pela rede.

- **Camadas Escondidas, Intermediárias, Ocultas ou Invisíveis:**

São aquelas camadas compostas de neurônios e que possuem a responsabilidade de extrair as características associadas ao processo ou sistema a ser inferido.

- **Camada de Saída:**

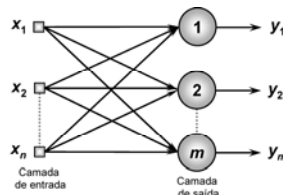
Esta camada é também constituída de neurônios, sendo responsável por produzir/apresentar os resultados finais da rede.

23

7. Arquiteturas de RNA's

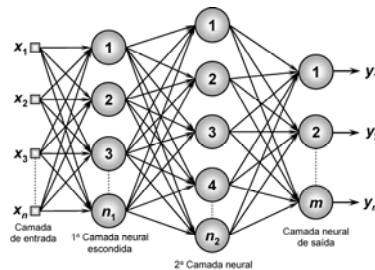
Arquiteturas FeedForward (Alimentação à frente)

- **Camada Simples:**



- ❖ Constituída de uma camada de entrada e única camada de neurônios, que é a própria camada de saída.
- ❖ Fluxo de informações segue sempre numa única direção (unidirecional).
- ❖ **Exemplos de arquiteturas:** Perceptron, ADALINE.
- ❖ **Aplicações:** classificação de padrões, filtragem linear.

- **Camadas Múltiplas:**



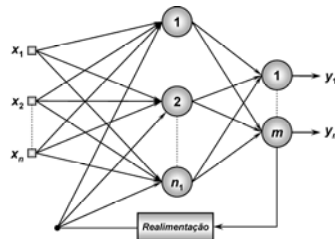
- ❖ Presença de uma ou mais camadas neurais escondidas.
- ❖ Quantidade de camadas escondidas e de neurônios dependem, sobretudo, do tipo e complexidade do problema.
- ❖ **Exemplos de arquiteturas:** Perceptron multicamadas, redes de base radial (*RBF*).
- ❖ **Aplicações:** aproximação de funções, classificação de padrões, controle de processos, otimização, etc.

24

7. Arquiteturas de RNA's

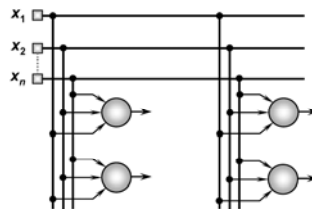
Arquitetura Recorrente & Arquitetura Reticulada

● Arquitetura Recorrente:



- ❖ Saídas dos neurônios são realimentadas como sinais de entrada para outros neurônios.
- ❖ Realimentação as qualifica para processamento dinâmico de informações.
- ❖ **Exemplos de arquiteturas:** Hopfield, Perceptron multicamadas com realimentação.
- ❖ **Aplicações:** sistemas de previsão, otimização, controle de processos, etc.

● Arquitetura Reticulada:



- ❖ Levam em consideração a forma em que a disposição espacial dos neurônios está organizada.
- ❖ Visa propósitos de extração de características.
- ❖ **Exemplos de arquiteturas:** Kohonen.
- ❖ **Aplicações:** problemas de agrupamento, classificação de padrões, otimização de sistemas, etc.

25

8. Processos de Treinamento

Aspectos de conjunto de treinamento/teste

● Finalidade do Treinamento:

Consiste da aplicação de passos ordenados a fim de ajustar (sintonizar) os parâmetros livres (pesos sinápticos e limiares) dos neurônios.

Após treinamento, a rede está apta para generalizar soluções (as quais não eram conhecidas).

● Conjuntos de Treinamento e Teste:

- ❖ **Conjunto Total de Amostras** → Representa todos os dados disponíveis sobre o comportamento do processo a ser mapeado. Será dividido em subconjuntos de treinamento e de teste.
- ❖ **Subconjunto de Treinamento** → Usado essencialmente para o processo de aprendizado da rede. Composto aleatoriamente com cerca de 60 a 90% das amostras do conjunto total.
- ❖ **Subconjunto de Teste** → Usado p/ verificar se a generalização de soluções por parte da rede já estão em patamares aceitáveis. Composto de 10 a 40% das amostras do conjunto total.
 - Estes dados não participam do treinamento da rede.

26

8. Processos de Treinamento

Aspectos do treinamento supervisionado

- **Treinamento Supervisionado:**

Consiste em se ter disponível, considerando cada amostra dos sinais de entrada, as respectivas saídas desejadas.

Cada amostra de treinamento é então composta pelos sinais de entradas e suas correspondentes saídas.

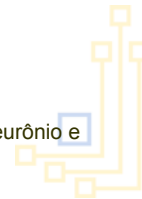
Comporta como se houvesse um “professor” ensinando para a rede qual seria a resposta correta para cada amostra apresentada.

- **Passos do Treinamento Supervisionado:**

1. Apresente uma amostra de treinamento.
2. Calcule a saída produzida pela rede.
3. Compare com a saída desejada.
4. Se estiver dentro de valores aceitáveis:

Então → Termine o processo de aprendizado.

Senão → Ajuste os pesos sinápticos e limiares dos neurônio e volte ao passo 1.



27

8. Processos de Treinamento

Aspectos do treinamento não-supervisionado

- **Treinamento Não-Supervisionado:**

Diferentemente do supervisionado, as respectivas saídas desejadas são inexistentes.

A rede deve se auto-organizar em relação às particularidades existentes entre os elementos do conjunto total de amostras, identificando subconjuntos (*clusters*) que contenham similaridades.

- **Passos do Treinamento Não-Supervisionado:**

1. Apresente todas as amostras de treinamento.
2. Obtenha as características que marcam as amostras de treinamento.
3. Agrupe todas as amostras com características em comum.
4. Coloque as amostras comuns em classes.



28

8. Processos de Treinamento

Aprendizagem off-line & on-line

- **Off-line (Usando lote de padrões):**

Os ajustes efetuados nos pesos e limiares dos neurônios só são efetivados após a apresentação de todo o conjunto de treinamento.

Cada passo de ajuste leva em consideração o total de desvios observados nas amostras de treinamento frente aos respectivos valores desejados para as suas saídas.

- **On-line (Usando padrão-por-padrão):**

Ao contrário da *off-line*, os ajustes nos pesos e limiares dos neurônios são efetuados após a apresentação de cada amostra de treinamento.

É normalmente utilizada quando o comportamento do sistema a ser mapeado varia de forma bastante rápida.



29

Fim da Apresentação



30