Sistemas de Informação Inteligentes

Sumário

1.Lógica Fuzzy

~ Rompimento com a rigidez da lógica clássica

•. Utilização de valores intermediários entre os dois extremos

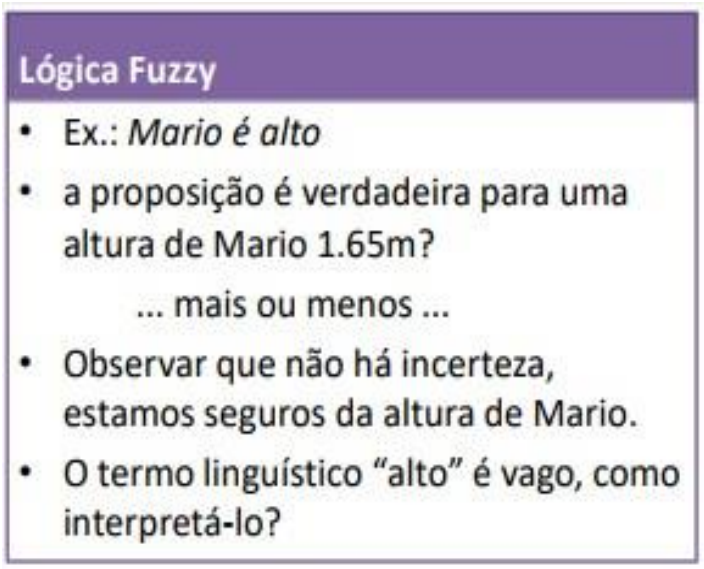
~ O "porquê" da lógica fuzzy?

•. Imprecisão do mundo real

•. Dificuldade de modelamento utilizando a lógica tradicional

// Lógica não clássica (aquelas que não são definidas apenas em termos de verdadeiro ou falso).

// Os valores das variáveis representam o grau de pertinência em relação ao conjunto, não devendo ser confundido com probabilidade ou crença.



•. Intenso uso de palavras ao invés de números (Termos: frio, quente, morno, alto, longe, devagar, etc....).

•. Manipulação de infinitos valores entre 0 e 1.

1.2. Conjuntos Fuzzy

Um conjunto Fuzzy corresponde a ideia de alargar a noção de conjunto, permitindo a representação de conceitos definidos por fronteiras difusas, tais como os que surgem a linguagem natural ou conceitos qualitativos.

A função de pertença a um conjunto fuzzy indica com que grau um conceito específico é membro de um conjunto.

- O grau de pertença 0 indica que o valor não pertence ao conjunto.

- O grau 1 significa que o valor é uma representação completa do conjunto.

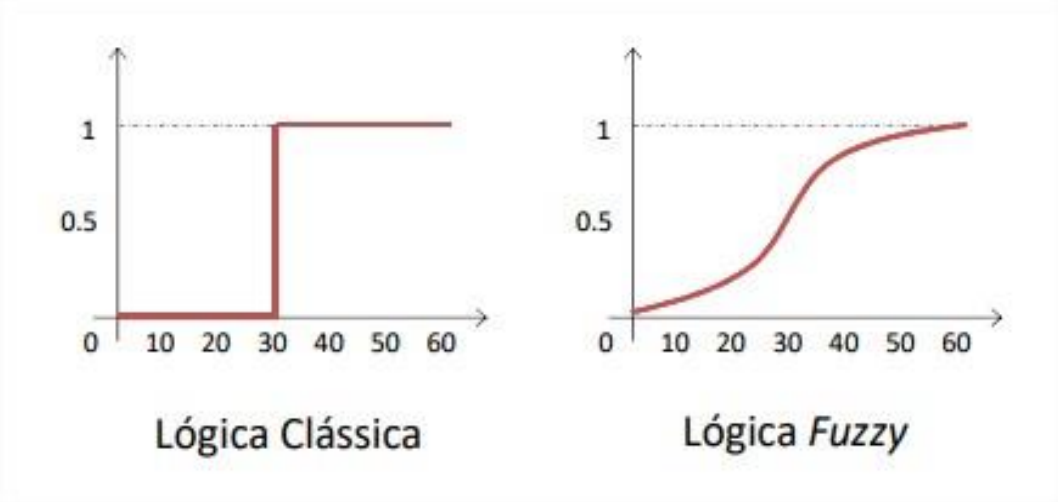
Lógica Clássica: elemento pertence ou não a um conjunto

- Conjunto: "Alto"

- Ex: João é alto/ João não é alto

Lógica fuzzy: elemento pertence, não pertence ou está parcialmente presente em um conjunto.

- Ex: João é um pouco alto.



Conjuntos Normais: função característica, equivale a medida de pertença associada ao conjunto A.

Conjunto Vago: quando os elementos têm um grau de pertença relativamente ao conjunto.

// Permitem modelar os aspectos através de valores vagos (i.e., termos imprecisos que capturam os valores possíveis do elemento real).

1.3. Variáveis Linguísticas

São elementos centrais da técnica de modelagem dos sistemas pois uma variável linguística é o nome do conjunto fuzzy.

- Transmitem o conceito de qualificadores.

•. Algumas variáveis linguísticas do conjunto "longo" com qualificadores:

- Muito longo

- Um tanto longo

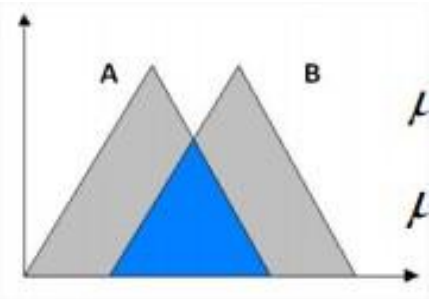
- Ligeiramente longo

Permitem que a linguagem da modelagem fuzzy expresse a semântica usada por especialistas.

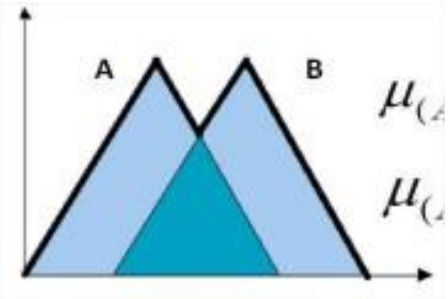
Encapsula as propriedades dos conceitos imprecisos numa forma usada computacionalmente.

1.4. Operadores dos Conjuntos Fuzzy

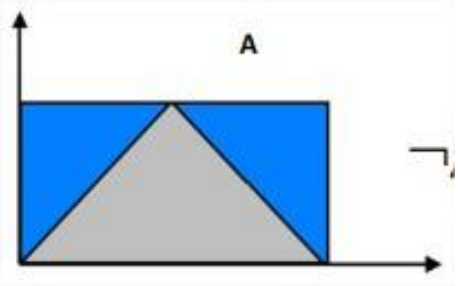
Intersecção



União



Complemento



1.5. Raciocínio Fuzzy

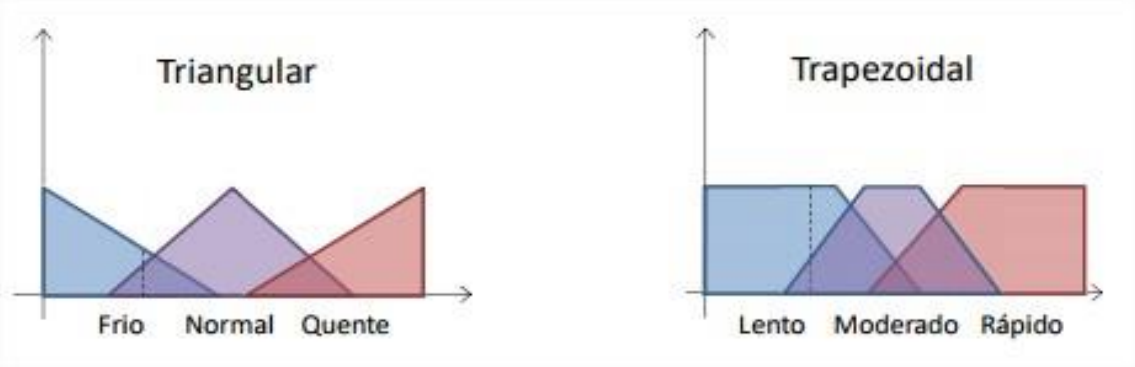
1) Fuzzificação

Durante a modelagem é a etapa na qual as variáveis linguísticas são definidas de forma subjetiva, bem como as funções membro (funções de pertença).

- Análise de Problema

- Definição das Variáveis

Na definição das funções de pertença para cada variável diversos tipos de espaço podem ser gerados.



- Ex: Temperatura, x=37°

- Conjuntos fuzzy = frio, morno, quente

- (37°) = 0,2/frio, 0.4/morno, 0.8/quente

2) Inferência

Etapa na qual as proposições são definidas e, depois, examinadas de modo paralelo. É o procedimento para chegar a conclusões a partir de regras **if – then** e corresponde ao "Raciocínio Fuzzy".

- Definição das proposições.

- Análise das regras:

-> O mecanismo chave do modelo Fuzzy são as regras. São elas que estabelecem o relacionamento entre as variáveis de modelo e regiões Fuzzy.

- Criação da região resultante.

Agregação: Calcula a importância de uma determinada regra para a situação corrente.

Composição: Calcula a influência de cada regra nas variáveis de saída.

3) Defuzzificação

Fase na qual as regiões resultantes são convertidas em valores para a variável de saída do sistema. Esta etapa corresponde a ligação funcional entre as regiões Fuzzy e o valor esperado.

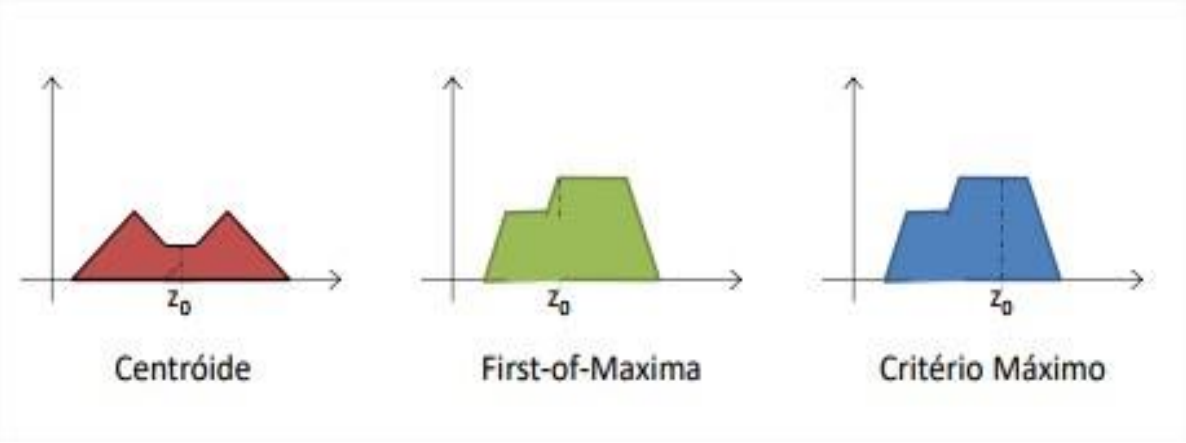
Técnicas de Defuzzificação:

- Centróide

- First-of-Maxima

- Middle-of-Maxima

- Critério Máximo



2.Árvore de Decisão

•. Indução a partir de um conjunto de dados rotulados (classificados)

- Aprendizado supervisionado.

-> Propriedades

•. Instancias são representados por pares atributo-valor.

•. A hipótese/função objetivo (classe) tem, preferencialmente valores discretos.

- Atributos contínuos podem ser usados fazendo o nó dividir o domínio do atributo entre dois intervalos baseados em um limite.

- Árvores de classificação tem valores discretos nas folhas, árvores de regressão tem valores reais nas folhas.

•. Algoritmos para encontrar árvores consistentes são **eficientes** e podem processar grandes quantidades de dados de treinamento.

-> Estrutura

Uma árvore de decisão contém:

- Nós-folha que correspondem às classes.

- Nós de decisão que contem testes sobre atributos.

2.1. Algoritmo ID3

- Qual atributo deveria ser testado como raiz da árvore?

Para saber a resposta, usa a medida estatística ganho de informação, a qual mede quão bem um dado atributo separa o conjunto de treinamento de acordo com a classe.

```
(01) função DTree(exemplos, atributos): retorna uma árvore
(02) início
(03)   se todos exemplos pertencem a uma única classe então
(04)     retorna um nó folha com essa classe;
(05)   senão
(06)     se o conjunto de atributos estiver vazio então
(07)       retorna um nó folha com a classe mais comum entre os exemplos.
(08)     senão
(09)       escolha um atributo F e crie um nó R para ele
(10)       para cada possível valor v_i de F faça
(11)         seja exemplos_i o subconjunto de exemplos que tenha valor v_i para F
(12)         coloque uma aresta E a partir do nó R com o valor v_i.
(13)         se exemplos_i estiver vazio então
(14)           coloque uma folha ligado a aresta E com a classe mais comum entre
                                                                        os exemplos.
(15)         senão
(16)           chame DTree(exemplos_i, atributos - {F}) e ligue a árvore resultante
                                                          como uma sub-árvore sob E.
(17)     retorne a sub-árvore com raiz R.
(18) fim.
```

2.2. Entropia

$$Entropy(S) = -p_{(+)}\log_2(p_{(+)}) - p_{(-)}\log_2(p_{(-)})$$

Genericamente, para qualquer número c de classes de um conjunto de dados, a entropia de S é dada pela fórmula:

$$Entropy(S) = \sum_{i=1}^{c} -p_i \log_2(p_i)$$

onde *pi* é a proporção de instancias de S pertencendo a classe i e c é o número total de classes.

- Ex: Dada uma coleção S com 14 exemplos, sendo que o atributo de classe é constituído por 9 casos positivos e 5 casos negativos, [9+,5-], a entropia de S é:

(9+5=14[total]).

$$Entropy(S) = -p_+ \log_2(p_+) - p_- \log_2(p_-)$$

$$Entropy(S) = -\frac{9}{14}\log_2\left(\frac{9}{14}\right) - \frac{5}{14}\log_2\left(\frac{5}{14}\right)$$

$$Entropy(S) = 0.940$$

2.3. Ganho de Informação

A medida de ganho de informação pode ser definida como a redução esperada na entropia causada pelo particionamento de exemplos de acordo com um determinado atributo.

- O ganho de informação deve ser calculado para cada atributo do conjunto de atributos da coleção S.

- O atributo que resultar no maior ganho de informação é selecionado como atributo de teste.

$$Gain(S, A) = Entropy(S) - \sum_{v \in Valores(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

3. Naive Bayes

•. Fundamenta-se na teoria das probabilidades.

•. Opera-se calculando as probabilidades para as hipóteses induzidas.

Características:

- Cada exemplo de treinamento pode incrementar ou decrementar a probabilidade de uma hipótese.

- Conhecimento a priori pode ser combinado com os dados observados para determinar a probabilidade final de uma hipótese.

- Pode-se acomodar hipóteses que fazem predições probabilísticas.

3.1. Um pouco de probabilidade ...

Um evento é um conjunto de possibilidades que tem uma probabilidade associada.

- Por exemplo, quando jogamos uma moeda temos um de dois eventos: cara ou coroa.

Outro aspecto importante é a relação entre dois eventos.

- Por exemplo, a dependência entre chuva e a formação de nuvens.

As relações entre dois eventos podem ser:

- Disjuntos (ou exclusivos): um não pode acontecer ao mesmo tempo que o outro.

- Independentes: podem ocorrer ao mesmo tempo, mas a ocorrência de um não afeta a possibilidade de ocorrência do outro.

- Dependentes: se a ocorrência de um afeta o outro.

Se os eventos são independentes, a probabilidade de ocorrerem é dada por:

$$P(A \cap B) = P(A) * P(B)$$

4. Teorema de Bayes

Para dois eventos:

$$P(A \cap B) = P(A|B) * P(B) = P(B|A) * P(A)$$
$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$
$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)}$$

Se eventos A1... An são mutuamente exclusivos e suas probabilidades somam 1, então:

$$P(B) = \sum_{i=1}^{n} P(B|A) * P(A)$$

Exemplo:

Classificar os seguintes valores:
X = (Idade <= 30, Renda = Media, Estudante = sim, Crédito = bom)
Y = Compra_Computador?

- Primeiramente fazemos o cálculo da probabilidade de sim e não da classe:

ID	Idade	Renda	Estudante	Crédito	Compra computador
1	<= 30	Alta	Não	Bom	Não
2	<= 30	Alta	Não	Bom	Não
3	31..40	Alta	Não	Bom	Sim
4	> 40	Média	Não	Bom	Sim
5	> 40	Baixa	Sim	Bom	Sim
6	> 40	Baixa	Sim	Excelente	Não
7	31..40	Baixa	Sim	Excelente	Sim
8	<= 30	Média	Não	Bom	Não
9	<= 30	Baixa	Sim	Bom	Sim
10	> 40	Média	Sim	Bom	Sim
11	<= 30	Média	Sim	Excelente	Sim
12	31..40	Média	Não	Excelente	Sim
13	31..40	Alta	Sim	Bom	Sim
14	> 40	Média	Não	Excelente	Não

P(Y=sim) e P(Y=não)
Probabilidades: P(Y=sim) = 9/14 = 0,643

9/14:

9 = quantidade de “sim”.

14 = total de valores apresentados.

ID	Idade	Renda	Estudante	Crédito	Compra computador
1	<= 30	Alta	Não	Bom	Não
2	<= 30	Alta	Não	Bom	Não
3	31..40	Alta	Não	Bom	Sim
4	> 40	Média	Não	Bom	Sim
5	> 40	Baixa	Sim	Bom	Sim
6	> 40	Baixa	Sim	Excelente	Não
7	31..40	Baixa	Sim	Excelente	Sim
8	<= 30	Média	Não	Bom	Não
9	<= 30	Baixa	Sim	Bom	Sim
10	> 40	Média	Sim	Bom	Sim
11	<= 30	Média	Sim	Excelente	Sim
12	31..40	Média	Não	Excelente	Sim
13	31..40	Alta	Sim	Bom	Sim
14	> 40	Média	Não	Excelente	Não

P(Y=sim) e P(Y=não)
Probabilidades: P(Y=sim) = 9/14 = 0,643
P(Y=não) = 5/14 = 0,357 = 1-P(Y=sim)

5/14:

5 = quantidade de “não”.

14 = total de valores apresentados.

- Depois calculamos a probabilidade dentro do problema que nos foi passado. Por exemplo, ele quer saber a probabilidade de uma pessoa ter a idade **<=30**.

ID	Idade	Renda	Estudante	Crédito	Compra computador
1	<= 30	Alta	Não	Bom	Não
2	<= 30	Alta	Não	Bom	Não
3	31..40	Alta	Não	Bom	Sim
4	> 40	Média	Não	Bom	Sim
5	> 40	Baixa	Sim	Bom	Sim
6	> 40	Baixa	Sim	Excelente	Não
7	31..40	Baixa	Sim	Excelente	Sim
8	<= 30	Média	Não	Bom	Não
9	<= 30	Baixa	Sim	Bom	Sim
10	> 40	Média	Sim	Bom	Sim
11	<= 30	Média	Sim	Excelente	Sim
12	31..40	Média	Não	Excelente	Sim
13	31..40	Alta	Sim	Bom	Sim
14	> 40	Média	Não	Excelente	Não

X = (**Idade <= 30**, Renda = Media, Estudante = sim, Crédito = bom)
Probabilidades: P[Idade <= 30 | Y = sim] = 2/9 = 0,222

2/9:

2 = número de "sim" que estão 'dentro' da condição (<=30).

9 = total de "sim".

ID	Idade	Renda	Estudante	Crédito	Compra computador
1	<= 30	Alta	Não	Bom	Não
2	<= 30	Alta	Não	Bom	Não
3	31..40	Alta	Não	Bom	Sim
4	> 40	Média	Não	Bom	Sim
5	> 40	Baixa	Sim	Bom	Sim
6	> 40	Baixa	Sim	Excelente	Não
7	31..40	Baixa	Sim	Excelente	Sim
8	<= 30	Média	Não	Bom	Não
9	<= 30	Baixa	Sim	Bom	Sim
10	> 40	Média	Sim	Bom	Sim
11	<= 30	Média	Sim	Excelente	Sim
12	31..40	Média	Não	Excelente	Sim
13	31..40	Alta	Sim	Bom	Sim
14	> 40	Média	Não	Excelente	Não

X = (**Idade <= 30**, Renda = Media, Estudante = sim, Crédito = bom)
Probabilidades: P[Idade <= 30 | Y = sim] = 2/9 = 0,222
P[Idade <= 30 | Y = não] = 3/5 = 0,6

3/5:

3 = número de “não” que estão dentro da condição (<=30).

5 = total de “não”.

- E assim sucessivamente para as outras condições que é pedido no problema.

3.1. Teorema de Bayes na Classificação

•. Calculamos isoladamente o valor da probabilidade condicional de cada atributo, mas para que eles sejam calculados de forma interseccionada, temos:

$$P(x_1, x_2, \dots, x_d|C) = P(x_1|C) * P(x_2|C) * \cdots * P(x_d|C)$$

Com isso é possível chegar a uma forma mais geral do Teorema de Bayes:

$$P(Y|X) = \frac{P(Y)\prod_{i=1}^{d} P(X_i|Y)}{P(X)}$$

•. Assim temos que:

P(Idade ≤ 30, Renda = Media, Estudante = sim, Crédito = bom | Y=sim) =
P(Idade ≤ 30 | Y = sim) * P(Renda = Media | Y = sim) *
P(Estudante = sim | Y = sim) * P(Crédito = bom | Y = sim) =
0,222 * 0,444 * 0,667 * 0,667 = 0,0438 ∴ P(X|Y=*sim*) = **0,0438**

P(Idade ≤ 30, Renda = Media, Estudante = sim, Crédito = bom | Y=não) =
P(Idade ≤ 30 | Y = não) * P(Renda = Media | Y = não) *
P(Estudante = sim | Y = não) * P(Crédito = bom | Y=não) =
0,6 * 0,4 * 0,2 * 0,6 = 0,0288 ∴ P(X|Y=*não*) = **0,0288**

Multiplica-se todos os valores obtidos com a classe “sim” e todos os valores obtidos com a classe “não”. (Menos as da classe principal, que foi a primeira conta feita)

•. Como calcular a probabilidade de alguém com o perfil X?

- Pela lei da probabilidade total

Ou seja:

$$P(X) = P(X|Y=sim) * P(Y=sim) + P(X|Y=não) * P(Y=não)$$
$$P(X) = 0,0438 * 0,643 + 0,0288 * 0,357 = \mathbf{0,0384}$$

• *0,0438 * 0,643:*

0,0438: valor obtido da multiplicação de todos os valores “sim”.

0,643: valor obtido na primeira conta, referente a probabilidade de “sim” na tabela toda.

• *0,0288 * 0,357:*

0,0288: valor obtido da multiplicação de todos os valores “não”.

0,357: valor obtido na primeira conta, referente a probabilidade “não” na tabela toda.

Obtendo o resultado da soma =0,0384

Por fim calculamos então:

$$P(Y=sim|X) = P(X|Y=sim) * P(Y=sim) / P(X)$$
$$= 0,0438 * 0,643 / 0,0384 = 0,7334$$
$$P(Y=não|X) = P(X|Y=não) * P(Y=não) / P(X)$$
$$= 0,0288 * 0,357 / 0,0384 = 0,2676$$

*0,0438 * 0,643 / 0,0384:* A multiplicação feita anteriormente com os valores de "sim", dividido pelo valor encontrado na soma anterior, obtendo assim a probabilidade de "sim", ou seja, de que aquele caso mostrado aconteça.

*0,0288 * 0,357 / 0,0384:* A multiplicação feita anteriormente com os valores de "não", dividido pelo valor encontrando na soma anterior, obtendo assim a probabilidade de "não", ou seja, de que aquele caso mostrando não aconteça.

Como o valor de "sim" foi maior que o valor de "não" logo a probabilidade dessa pessoa comprar o computador é maior.

4. Classificador Naive Bayes

Um classificador Naive Bayes estima a probabilidade de classe condicional P (X|Y) a partir de uma determinada amostra de dados predizendo a classe mais provável.

- Pré-Considerações:

- Assume-se que os atributos são condicionalmente independentes.

- As probabilidades condicionais são estimadas para os atributos de acordo com a sua classificação.

- Discretos ou contínuos.

Atributos Discretos: É aquele atributo para o qual é possível estabelecer um conjunto de valores finito. Ex: sexo, cor da pele.

Atributos Contínuos: São considerados contínuos os atributos que possuem muitos ou infinitos valores possíveis.

5. Redes Neurais

Modelos inspirados no cérebro humano, criadas em analogia a sistemas neurais biológicos, que são capazes de aprendizagem.

Criadas com o objetivo de entender sistemas neurais biológicos através de modelagem computacional.

5.1. Redes Neurais Artificiais

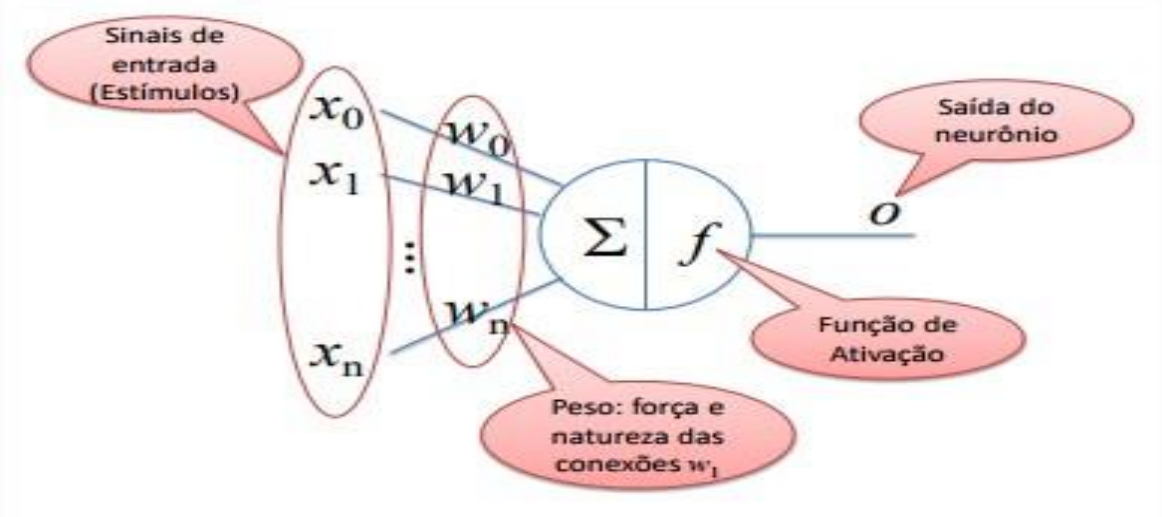
•. São tentativas de produzir sistemas de aprendizado biologicamente realistas.

- São baseados em modelos abstratos de como pensamos que o cérebro funciona.

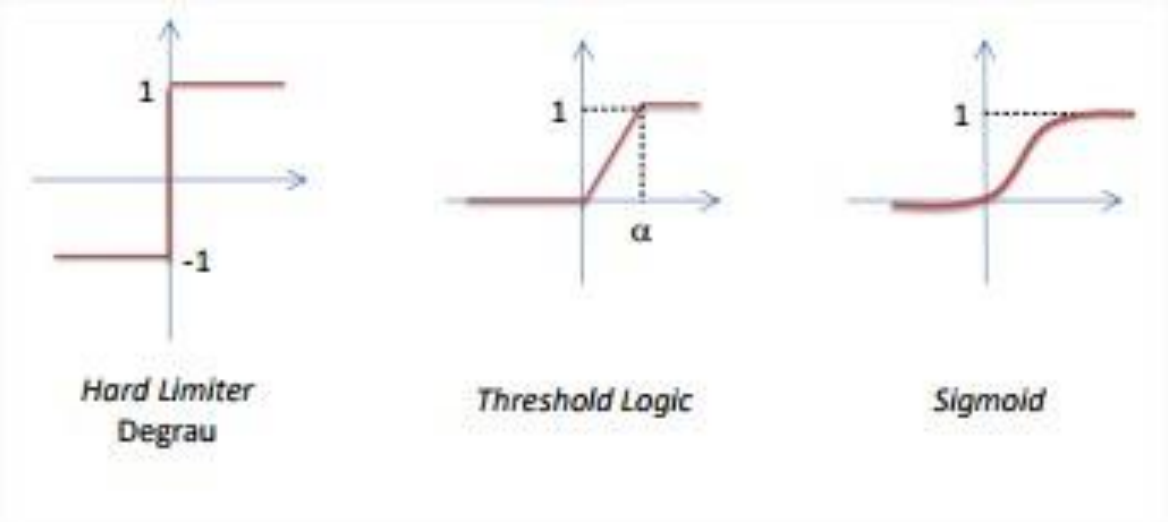
•. Abordagem baseada em uma adaptação do funcionamento de sistemas neurais biológicos.

- Perceptron: Algoritmo inicial para aprendizagem de redes neurais simples (uma camada) desenvolvido nos anos 50.

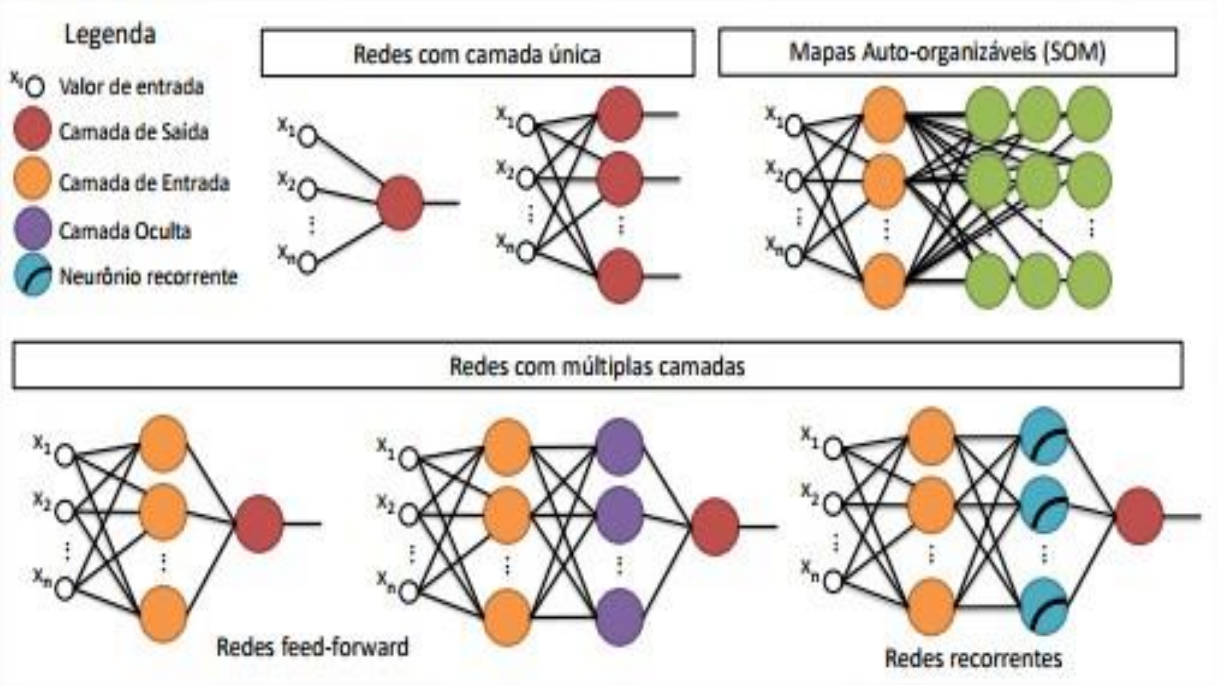
- Retroprogramação: Algoritmo mais complexo para aprendizagem de redes neurais de múltiplas camadas desenvolvido nos anos 80.



Funções de Ativação:



Aprendizagem Hebbiana: Quando dois neurônios conectados disparam ao mesmo tempo, a conexão sináptica entre eles aumenta.



6. Perceptron

McCollough e Pitts mostraram como neurônios simples desse tipo poderiam calcular funções lógicas e serem usados como máquinas de estado.

- Portas Lógicas: AND, OR, NOT.

- Podemos construir qualquer circuito lógico, maquina sequencial e computadores com essas portas.

•. Uma rede neural deve produzir, para cada conjunto de entradas apresentado, o conjunto de saídas desejado.

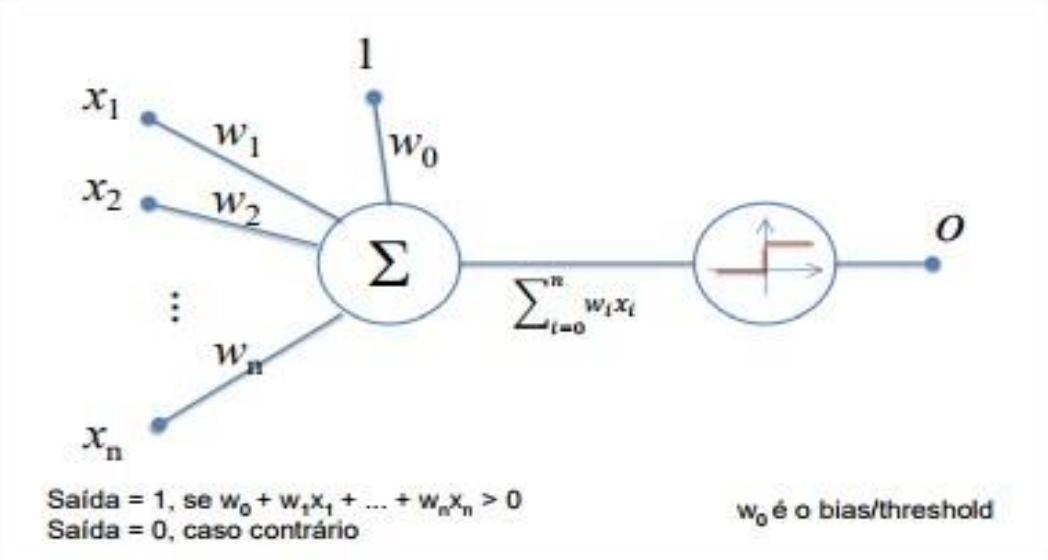
•. O objetivo é aprender pesos sinápticos de tal forma que a unidade de saída produza a saída correta para cada exemplo.

- Quando a saída é produzida é diferente da desejada, os pesos da rede são modificados.

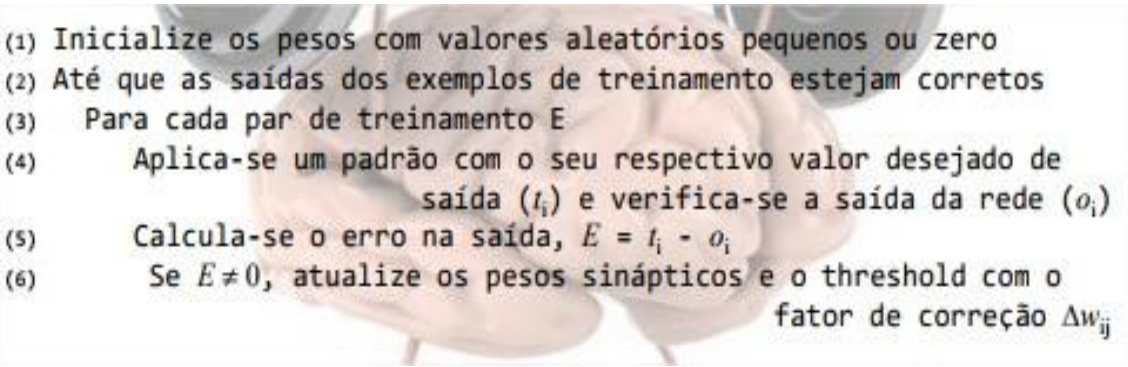
$$w_{t+1} = w_t + fator\_de\_correção$$

- O algoritmo faz atualizações iterativamente até chegar aos pesos corretos.

Modelo do Neurônio



Algoritmo de Aprendizado



Regras de aprendizagem de Perceptrons

•. O processo equivale a:

- Se a saída estiver correta, não fazer nada.

- Se a saída estiver alta, baixar os pesos das saídas ativas.

- Se a saída estiver baixa, aumentar os pesos das saídas ativas.

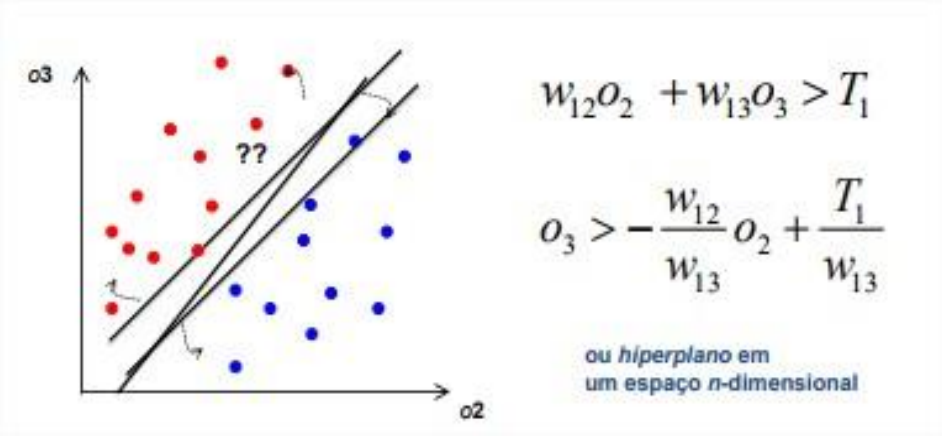
•. Atualizar pesos usando:

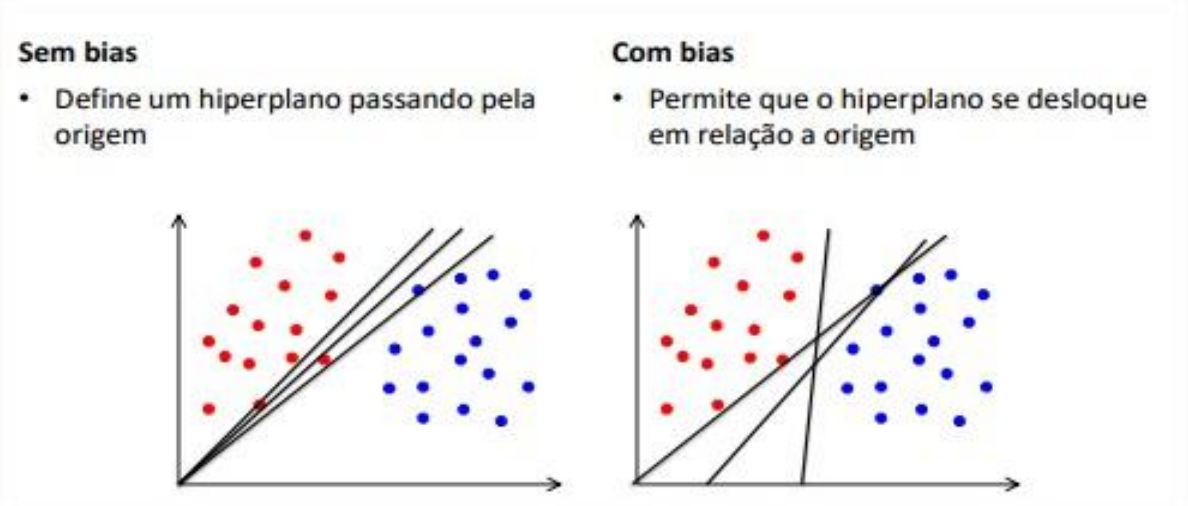
$$w_{ij} = w_{ij} + \Delta w_{ij}$$

$$\Delta w_{ij} = \eta x_i (t_j - o_j)$$

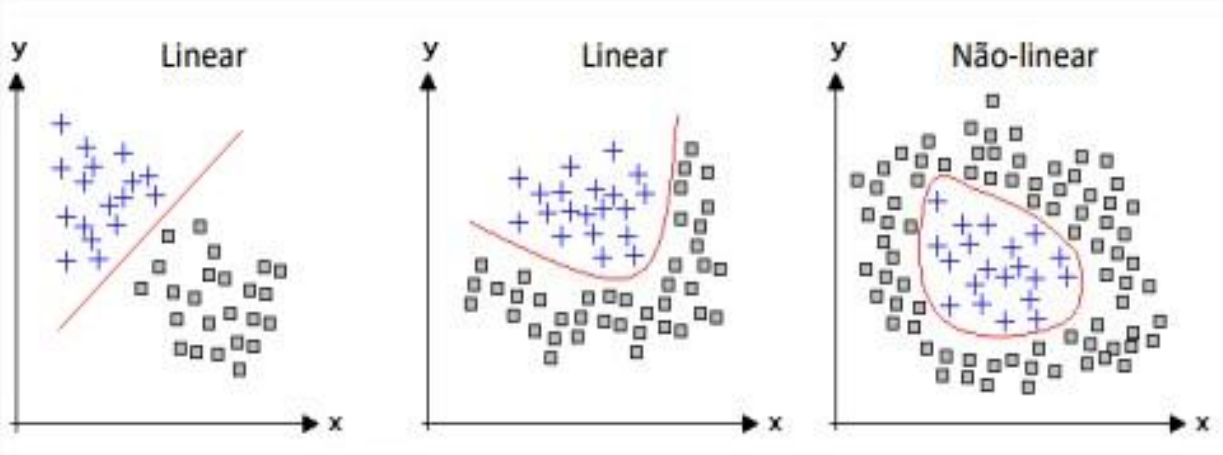
onde η é a "taxa de aprendizagem"
t_j é a saída especificada para a unidade j

Separador Linear: Como o Perceptron usa uma função de limite linear, ele procura por um separador linear que discrimine as classes.





Lineares VS Não-Lineares



Limitações do Perceptron

Não pode aprender conceitos que não é capaz de representar.

Minsksy e Papert escreveram um livro analisando o Perceptron e descrevendo funções que ele não podia aprender.

Esses resultados desencorajaram o estudo de redes neurais e as regras simbólicas se tornaram o principal paradigma de IA.

Teoremas

•. **Convergência do Perceptron:** Se os dados forem linearmente separáveis, então o algoritmo do Perceptron irá corrigir para um conjunto consistente de pesos.

•. **Ciclo do Perceptron:** Se os dados não forem linearmente separáveis, o algoritmo irá repetir um conjunto de pesos e limites no final de uma época e, como consequência entra em um loop infinito.