

# REDE MULTICAMADAS

**Prof. Dr. Ajalmar Rocha**

**Disciplina: Inteligência Computacional Aplicada (ICA)**

Programa de Pós-Graduação em Eng. de Telecomunicações (PPGET)

Instituto Federal do Ceará (IFCE)

Agosto/2013

# Problemas Linearmente Separáveis

- Problemas **linearmente separáveis** podem ser resolvidos por uma rede neural treinada com uma única camada.
- Classificar dados para as portas OR e AND são exemplos deste tipo de problema.

$$\text{Porta OR} = \left[ \begin{array}{cc|c} \mathbf{p} & \mathbf{q} & \mathbf{p \vee q} \\ \hline 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{array} \right]$$

$$\text{Porta AND} = \left[ \begin{array}{cc|c} \mathbf{p} & \mathbf{q} & \mathbf{p \wedge q} \\ \hline 0 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{array} \right]$$

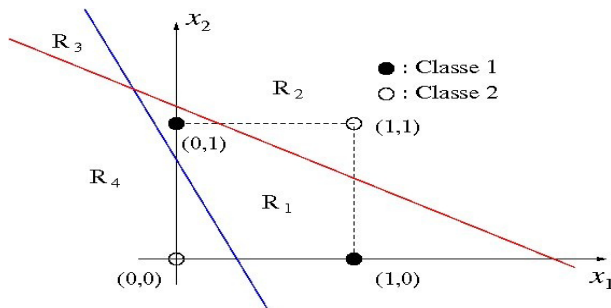
# Problemas Não Linearmente Separáveis

- Problemas não-linearmente separáveis não podem ser resolvidos por uma rede neural treinada com uma única camada.
- Este tipo de problema exige uma rede neural do tipo multicamadas com saídas não lineares.
- Classificar dados para a porta XOR é um exemplo deste tipo de problema.

|                    |  |          |          |                |
|--------------------|--|----------|----------|----------------|
| <b>Porta XOR =</b> |  | <b>p</b> | <b>q</b> | <b>p xor q</b> |
|                    |  | 0        | 0        | 0              |
|                    |  | 0        | 1        | 1              |
|                    |  | 1        | 0        | 1              |
|                    |  | 1        | 1        | 0              |

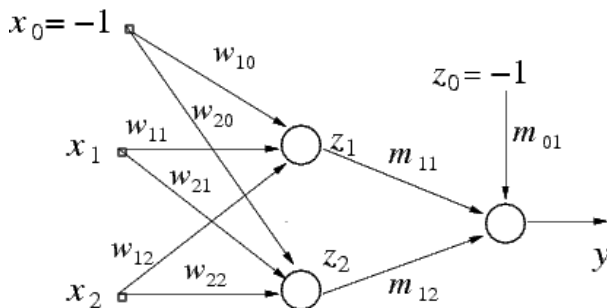
# Rede para Classificar dados para o XOR

- Com dois neurônios conseguimos dividir o espaço de entrada em quatro regiões,  $\{R_i\}_{i=1}^4$ .
- Considere uma reta resolvendo o problema para a **porta AND** e a outra para a **porta OR**.



## Rede para Classificar dados para o XOR

- Considere ainda uma rede descrita segundo a arquitetura abaixo.
- A saída para cada neurônio é calculada com base na função degrau.

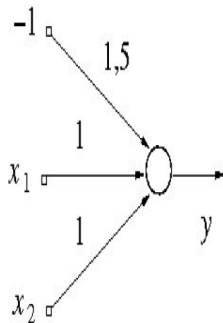


## Rede para Classificar dados para o XOR

- A saída para os neurônios da camada oculta são calculadas com base na função degrau.

$$\text{Porta AND} = \left[ \begin{array}{cc|c} \mathbf{x_1} & \mathbf{x_2} & \mathbf{z_1} \\ \hline 0 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{array} \right] \quad \text{Porta OR} = \left[ \begin{array}{cc|c} \mathbf{x_1} & \mathbf{x_2} & \mathbf{z_2} \\ \hline 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{array} \right]$$

## Pesos do Neurônio de M-P para porta AND (Lembram???)

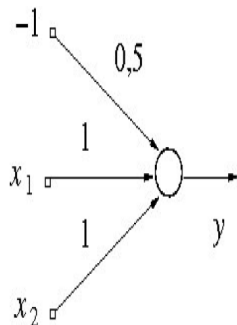


$$w_1 = w_2 = 1 \text{ e } \theta = 1,5$$

$$y = 1, \text{ se } u \geq 0.$$

$$y = 0, \text{ se } u < 0.$$

## Pesos do Neurônio de M-P para porta OR (Lembram???)



$$w_1 = w_2 = 1 \text{ e } \theta = 0,5$$

$$y = 1, \text{ se } u \geq 0.$$

$$y = 0, \text{ se } u < 0.$$



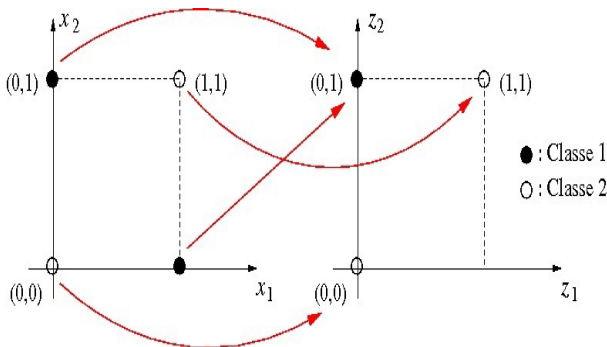
## Rede para Classificar dados para o XOR

- Perceba que para o neurônio de saída as entradas são dadas pelos valores de  $[z_1 \ z_2]^T$ , obtidos a partir de  $[x_1 \ x_2]^T$ .

$$\text{Porta XOR} = \left[ \begin{array}{cc|cc|c} \mathbf{x_1} & \mathbf{x_2} & \mathbf{z_1} & \mathbf{z_2} & \mathbf{y} \\ \hline 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{array} \right]$$

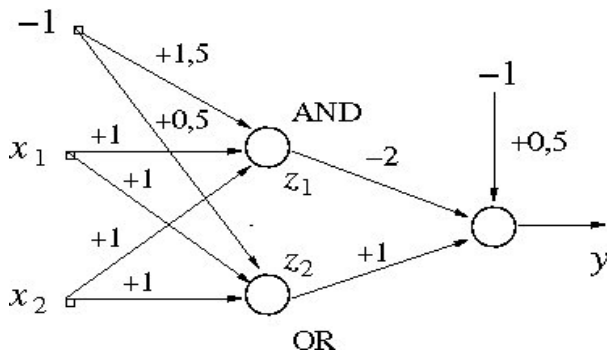
## Rede para Classificar dados para o XOR

- A representação dos dados em relação a  $z_1$  e  $z_2$  é mostrado a seguir.
- Como pode se perceber uma reta agora é capaz de classificar o problema.



## Rede para Classificar dados para o XOR

- Uma rede com duas camadas capaz de resolver o problema da **porta XOR** é apresentada a seguir.



OBRIGADO!