Introdução às Redes Neurais Artificiais

Perceptron de Múltiplas Camadas

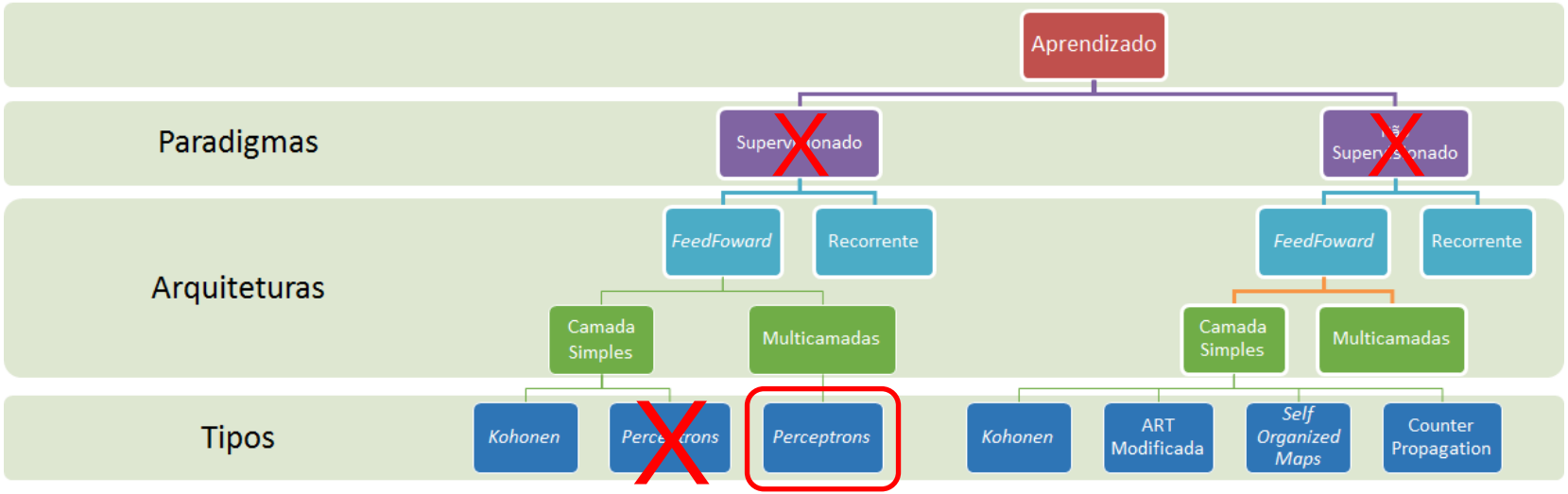
Prof. João Marcos Meirelles da Silva
jmarcos@id.uff.br

Universidade Federal Fluminense

www.latelco.uff.br
www.professores.uff.br/jmarcos

Mapa de Aprendizado

Redução de Dimensionalidade | Tipos de Dados | Detecção de Anomalias



Taxa de Aprendizado

1. Quanto menor η menor será a modificação nos pesos das sinapses, e mais suave será a trajetória no espaço de busca do problema → Treinamento mais demorado e tendência maior a uma maior acurácia da rede;

2. Quanto maior η maior será a modificação nos pesos das sinapses → Treinamento mais rápido e tendência à instabilidade.

Uma forma de aumentar a taxa de aprendizado e evitar o perigo de instabilidade é a chamada Regra Delta Generalizada.

Regra Delta Generalizada

$$\Delta w_{ji} = \alpha \Delta w_{ji} + \eta \delta_j y_i \qquad (1)$$

Onde $\alpha > 0$ é chamado de *constante de momento*. Se $\alpha = 0$, então a equação (1) se reduz à Regra Delta já conhecida.

Regra Delta Generalizada

Lembrando da aula passa que:

$$y_i = -\frac{\partial \varepsilon}{\partial w_{ji}} \qquad (2)$$

Temos que:

1. Se $\partial E / \partial w_{ji}$ possui o mesmo sinal algébrico em iterações consecutivas, Δw_{ji} cresce em magnitude;

2. Se $\partial E / \partial w_{ji}$ possui sinais algébricos opostos em iterações consecutivas, Δw_{ji} decresce em magnitude.

Critérios de Parada

Diversos critérios de parada para o treinamento da rede existem. Os mais comuns são:

1. Norma Euclidiana do vetor gradiente menor que um valor específico;

2. Taxa de alteração em E_{avg} menor que um valor específico (tipicamente: 0,1% a 1%);

3. Teste de generalização da rede.

Problemas no Treinamento

Problemas usuais durante o treinamento das redes *feedfoward* usando o algoritmo *backpropagation*:

1. *Overtraining* (também chamado de *overfitting*);

2. Convergência prematura (também chamado de paralisia).

Overtraining

1. O erro sobre o conjunto de treinamento torna-se bem pequeno, mas quando os dados do conjunto de teste são apresentados, o erro é grande $\Rightarrow$ Problemas de generalização;

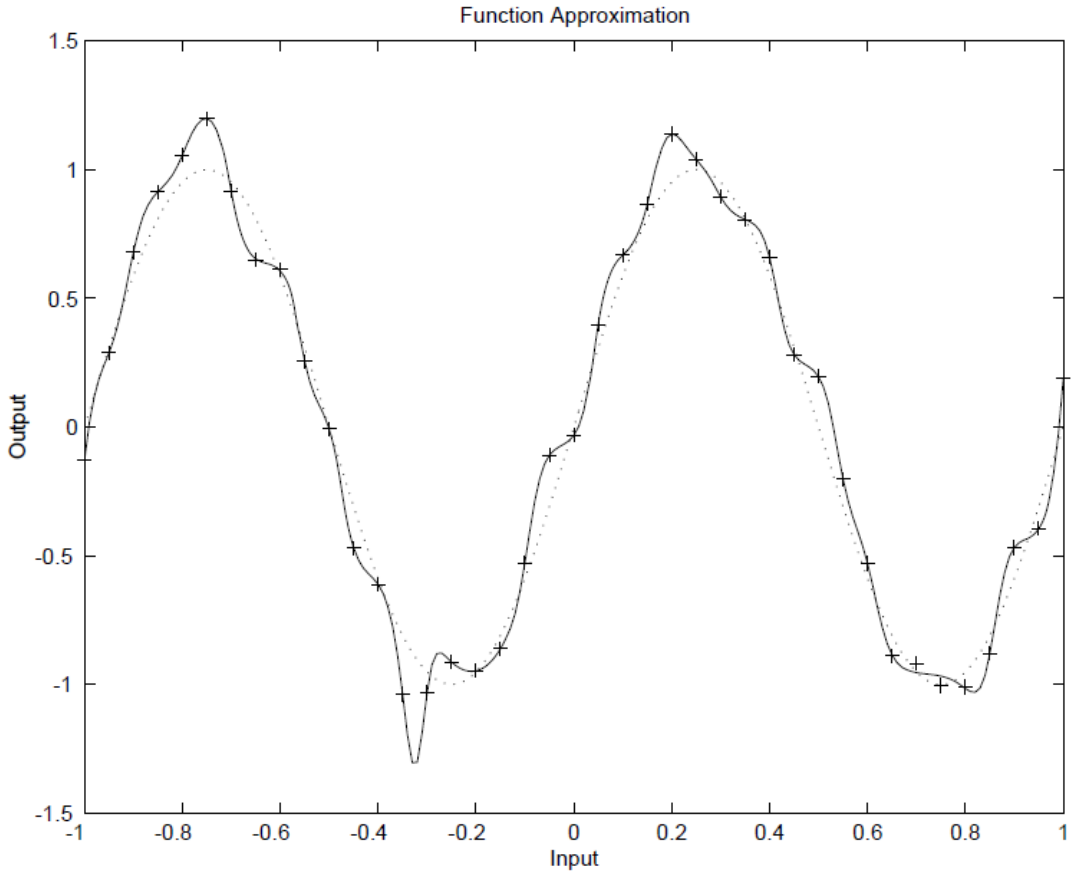
2. A rede "memoriza" os dados do conjunto de treinamento, mas não consegue generalizar em novas situações;

3. Ocorre devido à capacidade da rede "ajustar" uma função sobre todos os dados, ou sobre dados contaminados com ruído (o que é natural em aplicações reais).

Overtraining

A rede neural deveria ajustar uma curva sobre a senóide (linha pontilhada). No entanto, ela ajusta uma curva passando exatamente sobre todos os pontos experimentais!

Isso ocorre muitas vezes quando a rede possui excesso de neurônios/camadas.



Overtraining

Como evitar o *overtraining* e melhorar a generalização?

1. Utilizar redes "suficientemente" pequenas;
2. Promover uma parada prematura;
3. Utilizar regularização.

Overtraining

Tamanho da rede neural

1. Em geral, uma única camada intermediária é suficiente para aproximar qualquer função. No entanto, em algumas situações, uma segunda camada intermediária pode facilitar o treinamento;

2. Mas quantos neurônios devemos dispor na camada intermediária?

Tentativa e Erro !

Overtraining

Tamanho da rede neural

1. Em geral, u nte para
   aproximar algumas
   situações, u facilitar
   o treinamer

2. Mas quant camada
   intermediár

Network Growing
x
Network Pruning !

Tentativa e Erro !

Overtraining

Parada Prematura ⇒ Método da Validação Cruzada!

Nesta técnicas, divide-se o conjunto de dados em 3 subconjuntos:

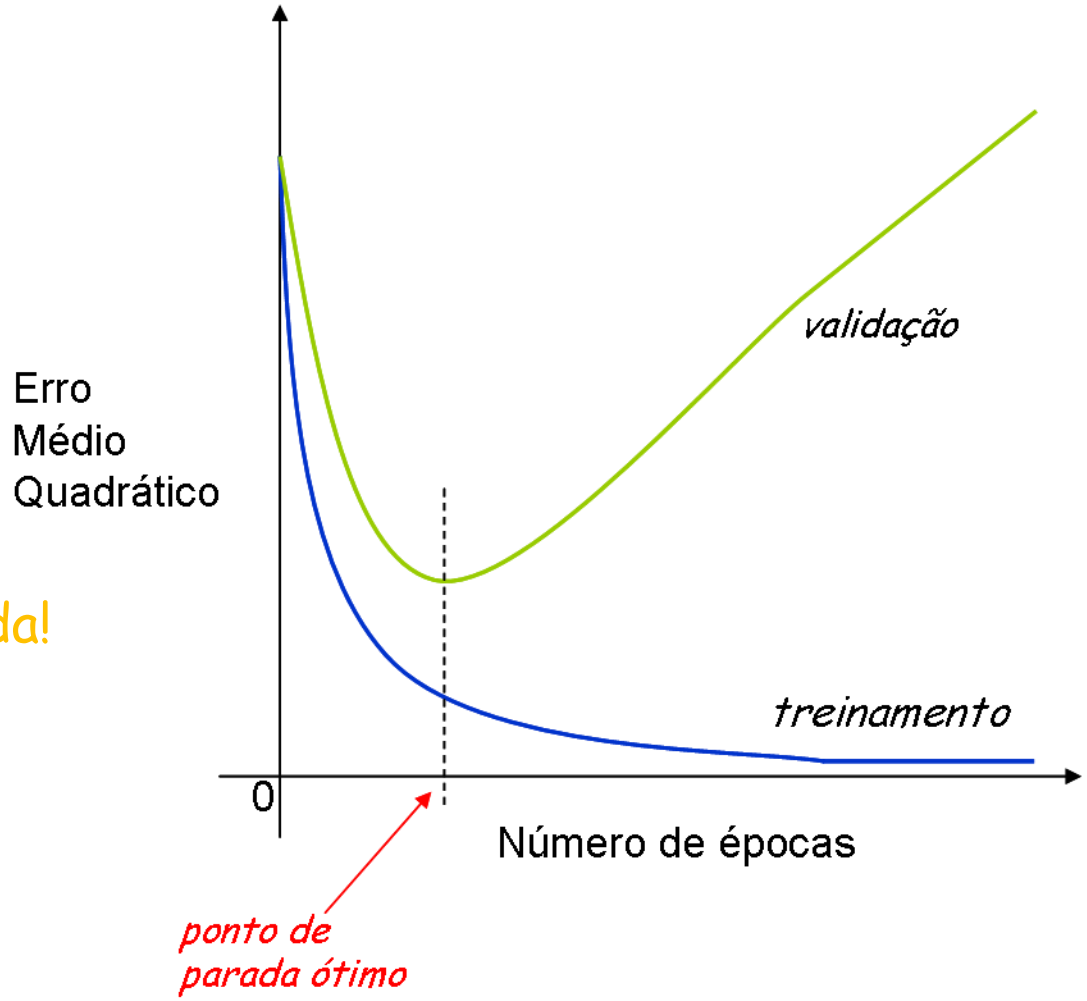
1. Treinamento (70%): Cálculo dos gradientes, pesos e bias;

2. Validação (15%): O erro sobre esse conjunto é monitorado;

3. Teste (15%): Usado para comparar diferentes modelos de redes (conjunto de controle).

Overtraining

Método da Validação Cruzada!



Overtraining

Regularização

1. É uma outra forma de minimizar as chances de *overtraining* e melhorar a generalização da rede;

2. Envolve a modificação da função de erro total que o treinamento busca minimizar.

$$\varepsilon_{avg} = mse = \frac{1}{N}\sum_{j=1}^{N} e_j^2 \qquad (3)$$

Overtraining

Regularização

É possível melhorar a capacidade de generalização da rede modificando a função de erro utilizada, adicionando-se um termo que consiste na média da soma dos quadrados dos pesos e dos bias:

$$\varepsilon_{reg} = \gamma \frac{1}{N} \sum_{i=1}^{N} e_i^2 + (1-\gamma) \frac{1}{P} \sum_{j=1}^{P} w_j^2 \qquad (4)$$

onde N é o número de neurônios da camada de saída, e P é o número total de pesos da rede (incluindo os bias).

Overtraining

Regularização

Utilizando esta nova função de erro, a rede passa a treinar com valores de pesos menores e mais suáveis $\Rightarrow$ Respostas mais suáveis e menor propensão ao *overtraining*.

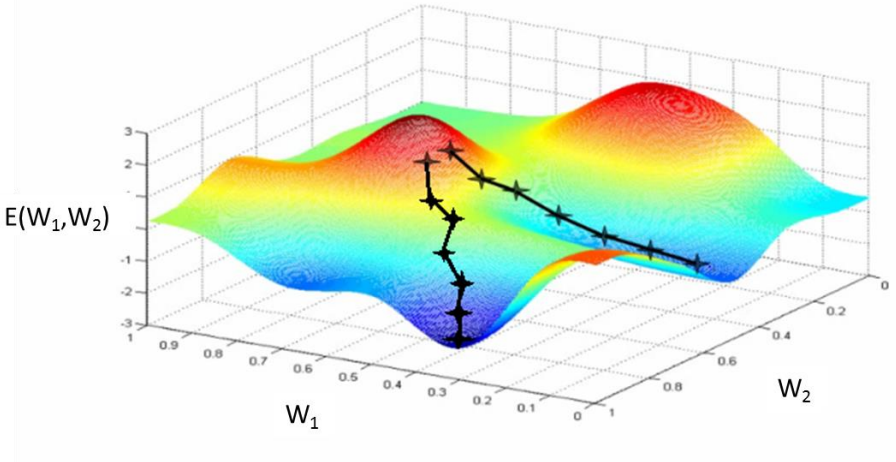
OBS: No entanto, há uma dificuldade a mais em determinarmos o valor ótimo de γ:

- Se $\gamma \cong 0$ → A rede neural não aprende;
- Se $\gamma \cong 1$ → *Overtraining*

Convergência Prematura

- Problemas de mínimos locais;
- A rede opera com desempenho inadequado;
- Como comparar as diversas soluções possíveis?

Resposta: Acompanhando o valor do erro no final de cada treinamento !

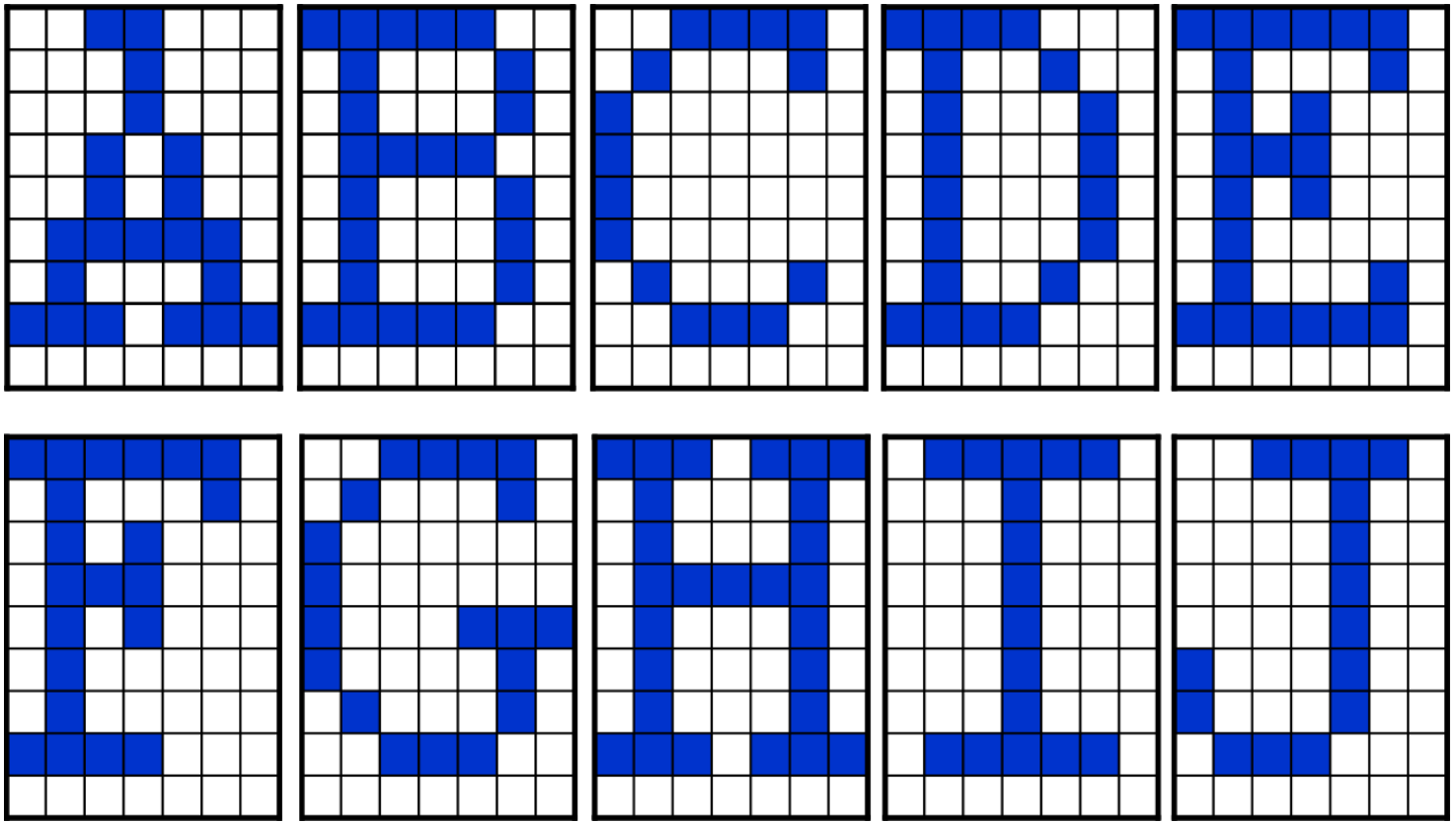


Convergência Prematura

Como evitar a paralisia?

- Utilizar uma "grande" quantidade de dados diferentes durante o treinamento;
- Acompanhar a evolução do erro durante o treinamento (método da validação cruzada);
- Executar o treinamento diversas vezes e optar pelos pesos que resultaram no menor erro encontrado.

Exemplo: Compressão de Imagens



Exemplo: Compressão de Imagens

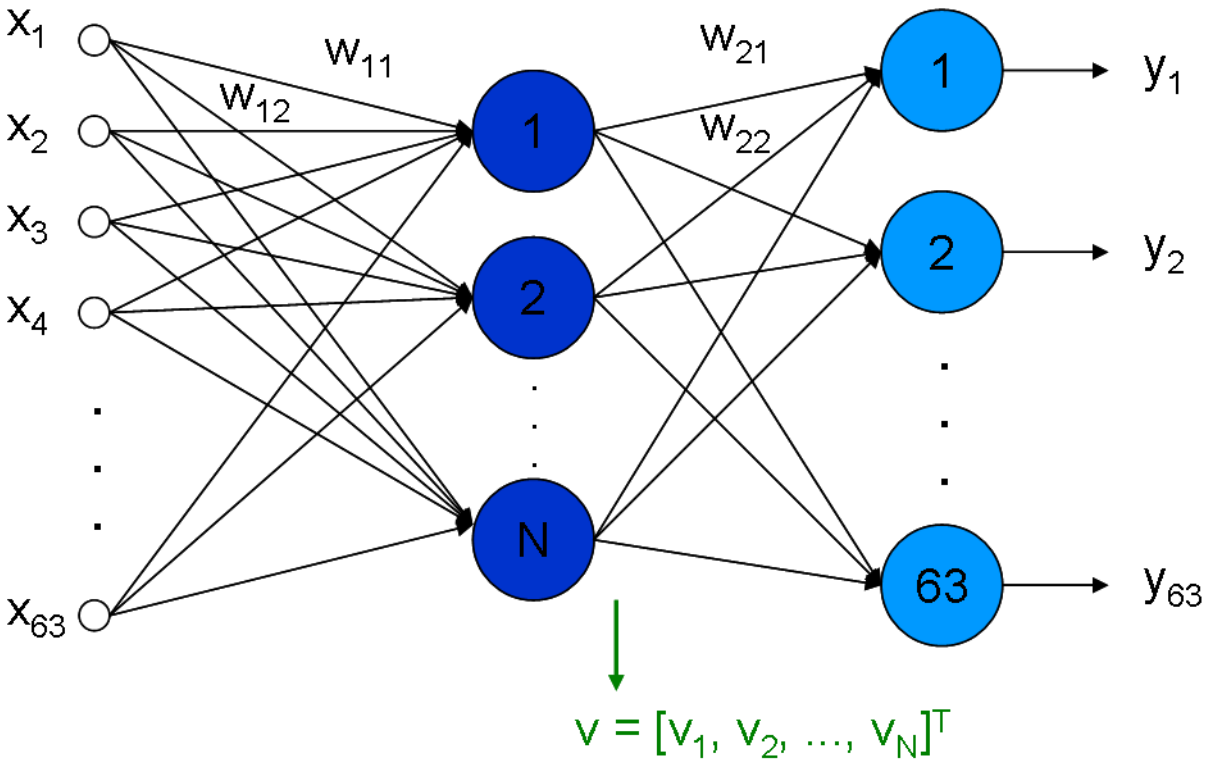
Observações

- Padrões formados por 63 pixels (matrizes de 9 x 7);
- Sabe-se que um conjunto de N padrões ortogonais podem ser mapeados em Log2 N neurônios na camada escondida;
- Como os padrões aqui não são ortogonais, esta relação deve ser tomada como um valor mínimo:

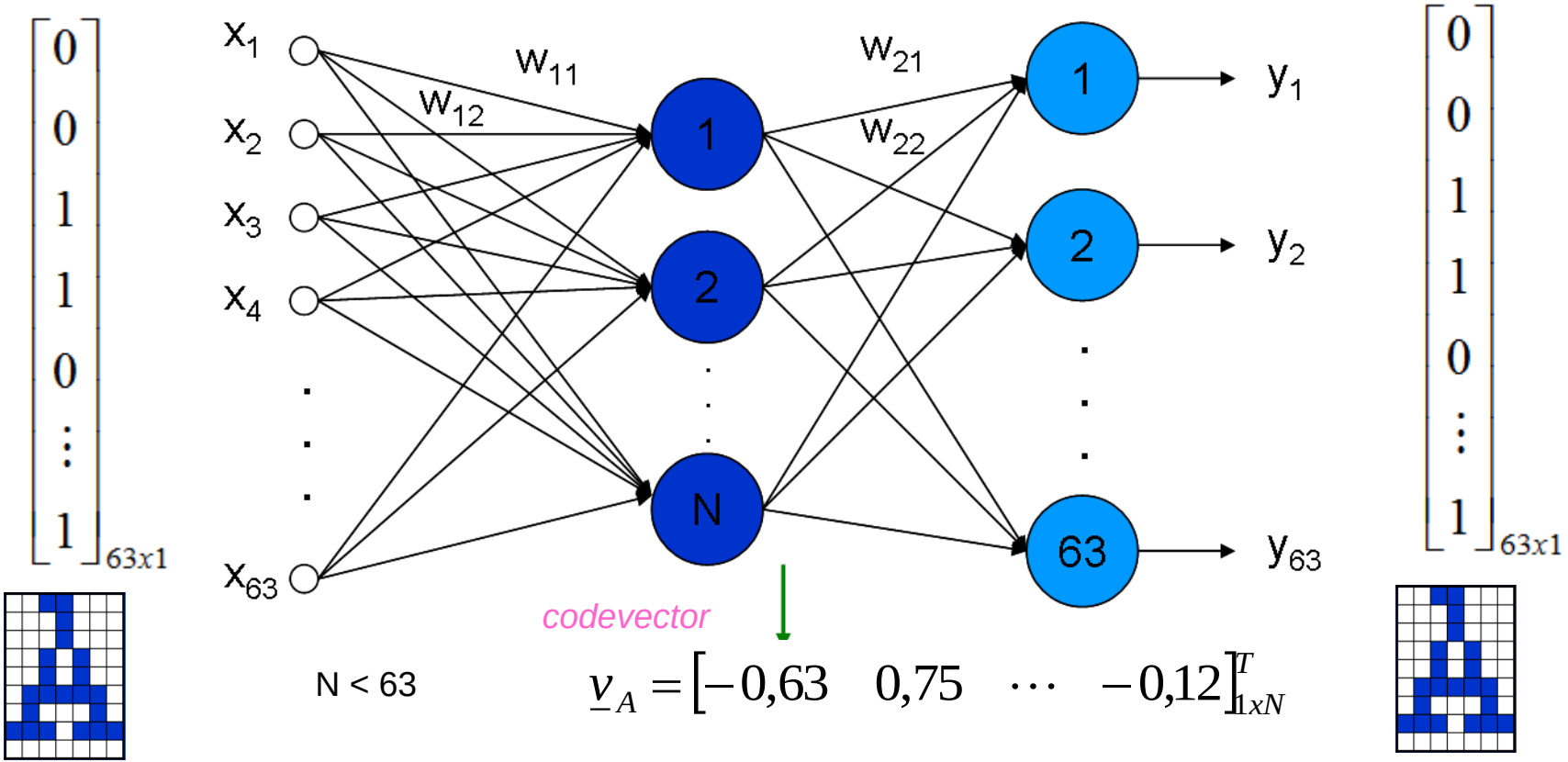
$$\log_2 63 = 5{,}9733 \approx 6$$

A rede deverá ser, no mínimo, de 63 x 6 x 63 !!!

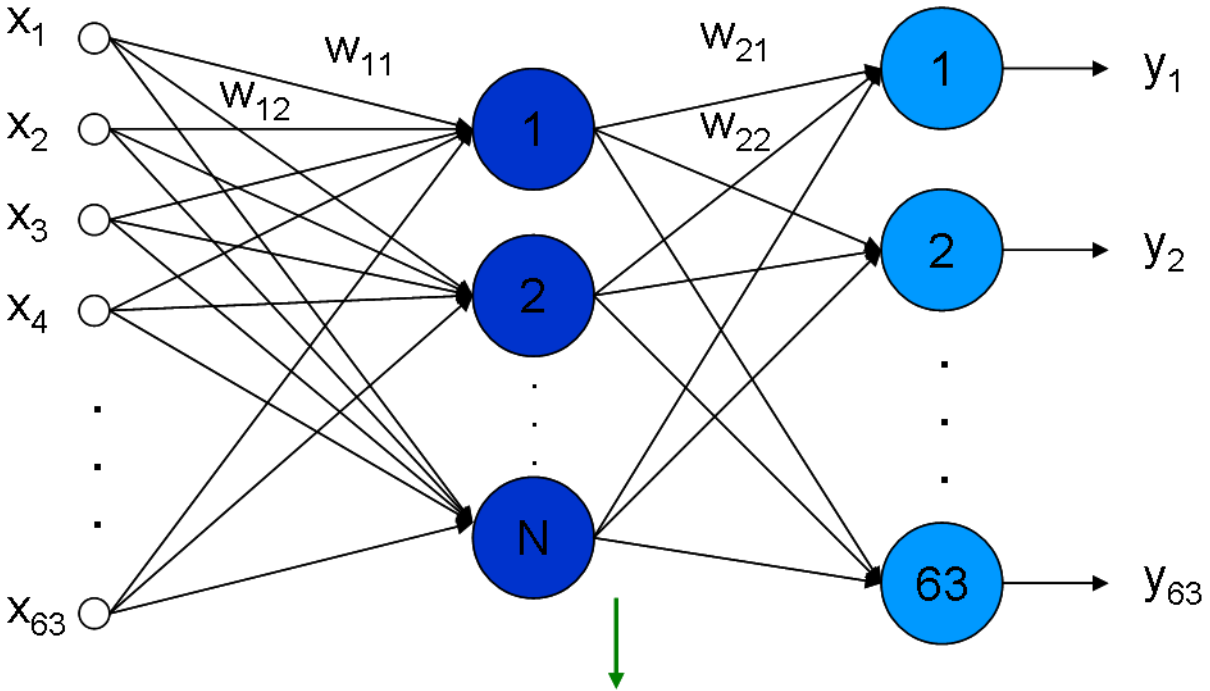
Exemplo: Compressão de Imagens



Exemplo: Compressão de Imagens



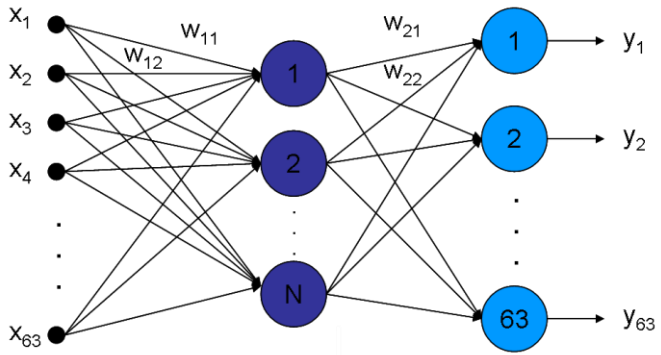
Exemplo: Compressão de Imagens



$$\underline{v} = \left[\underline{v}_A \quad \underline{v}_B \quad \cdots \quad \underline{v}_J \right]^T_{Nx10}$$

codebook

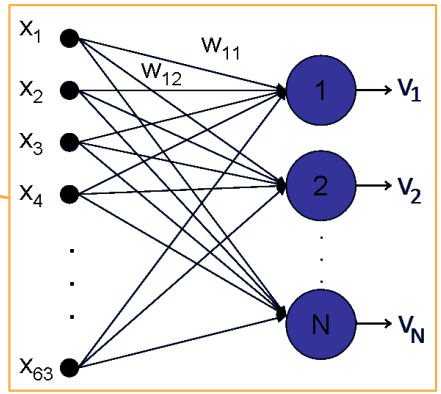
Exemplo: Compressão de Imagens



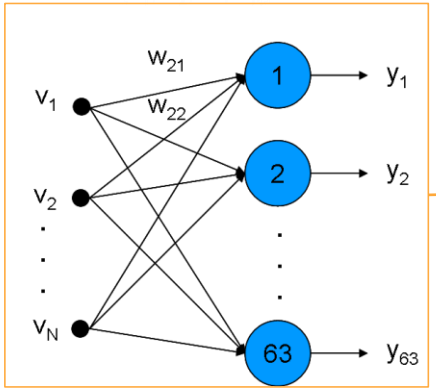
Após o treinamento, apenas metade da rede é necessária para o trabalho de compressão e descompressão !

A rede é então desmembrada em duas…

Compressão !!!

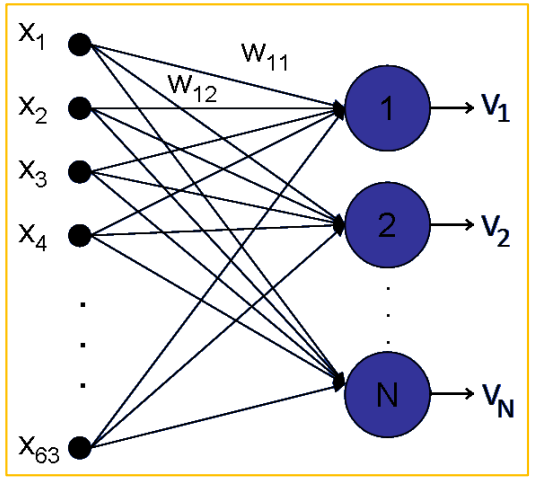


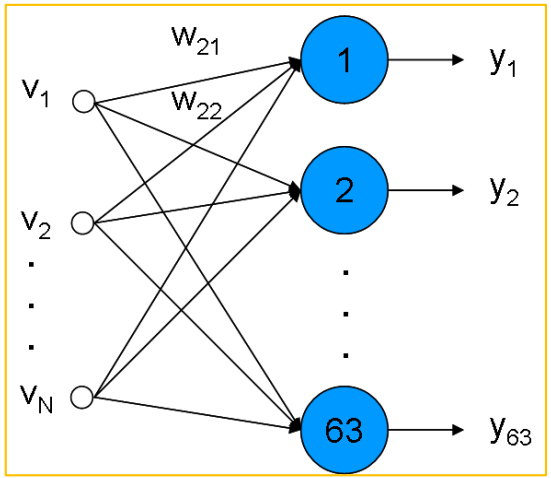
Descompressão !!!

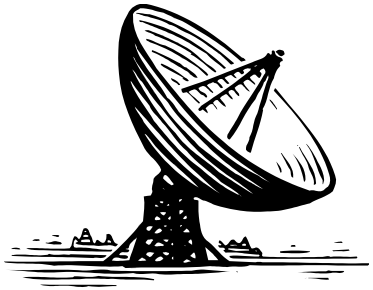


Transmissor

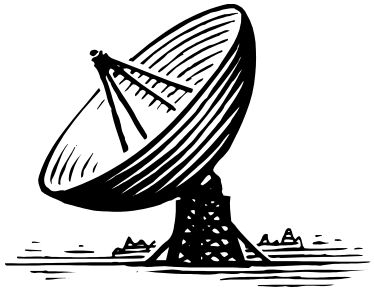
Receptor





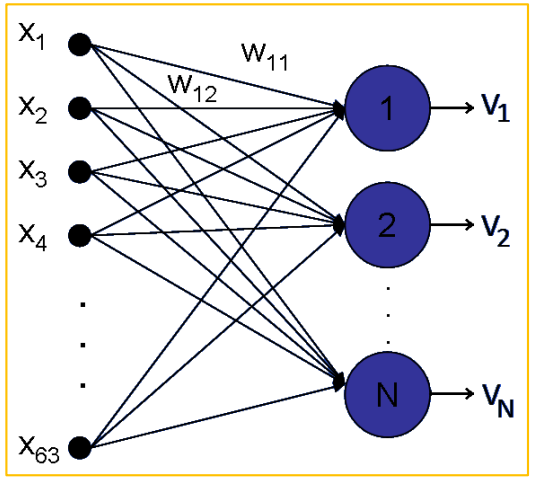


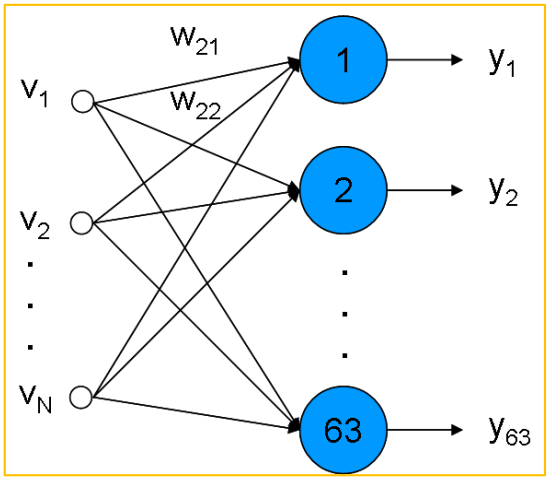
Pesos da RNA

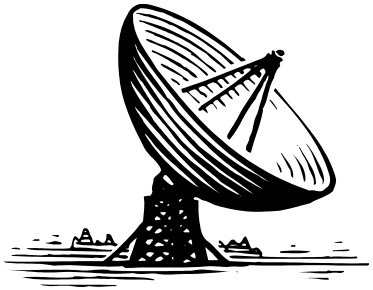


Transmissor

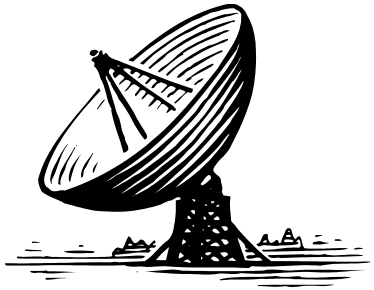
Receptor





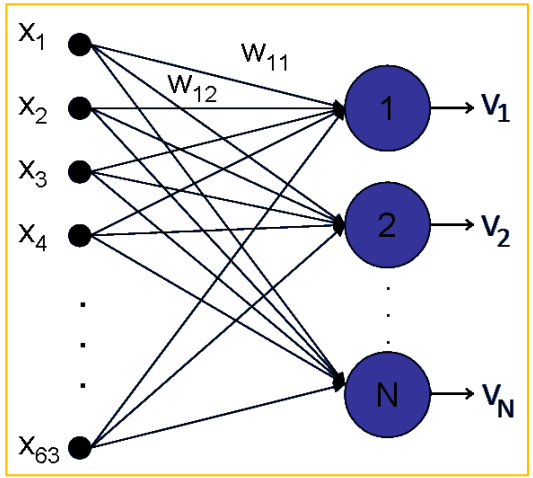


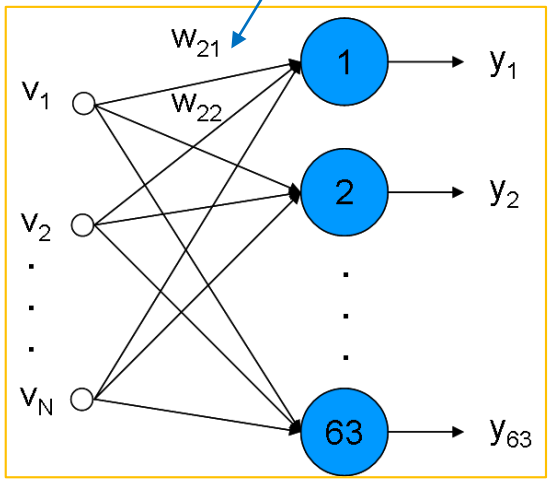
Transmissor

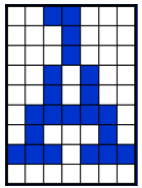


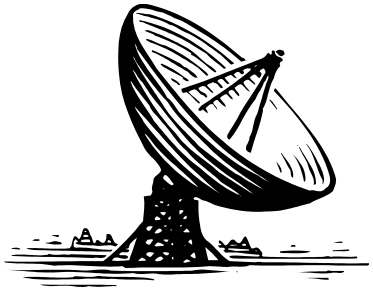
Receptor

Pesos da RNA

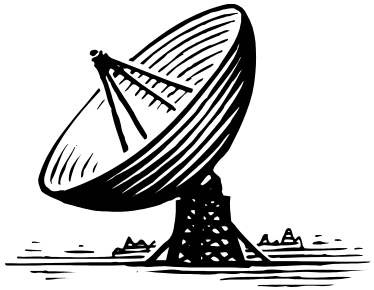




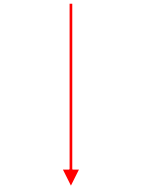


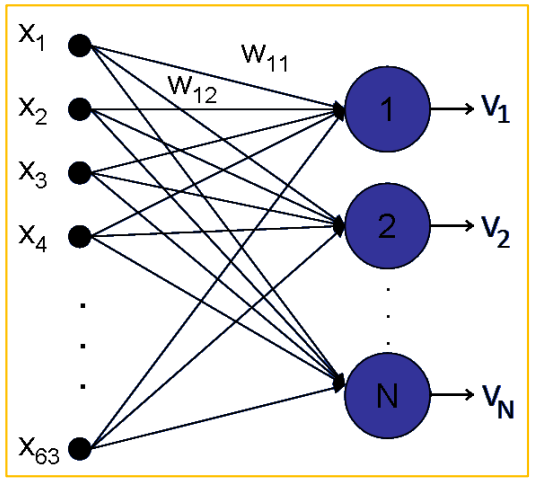


Transmissor

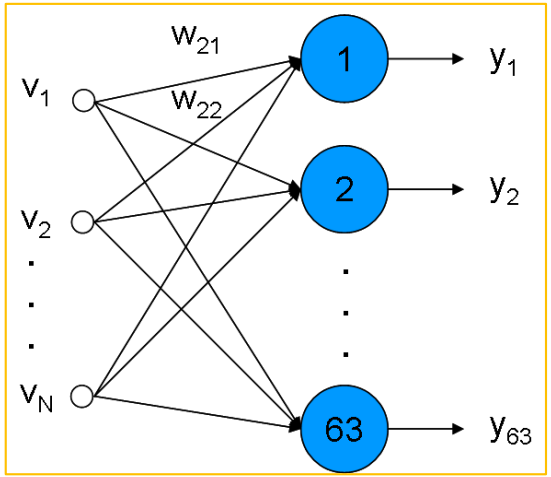


Receptor

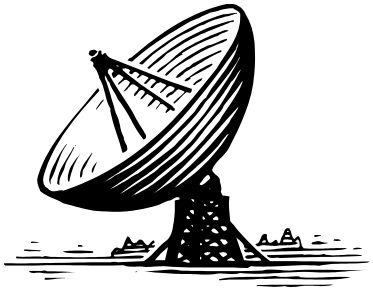




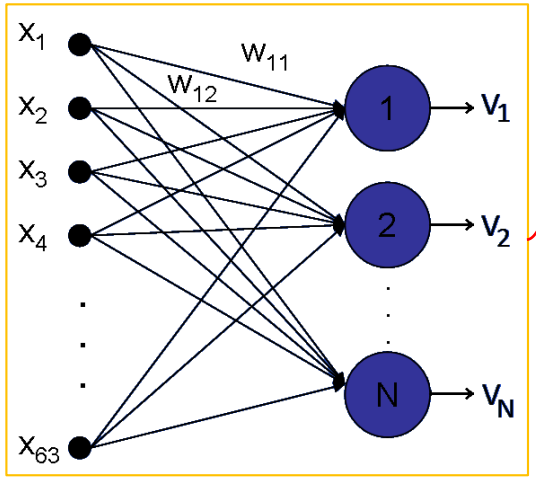
$\underline{V}_A$



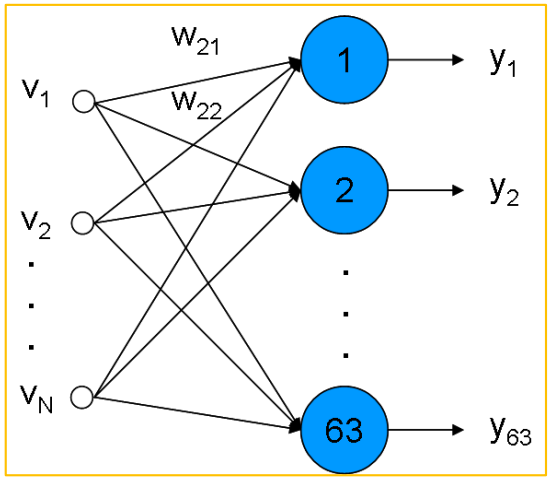
Transmissor

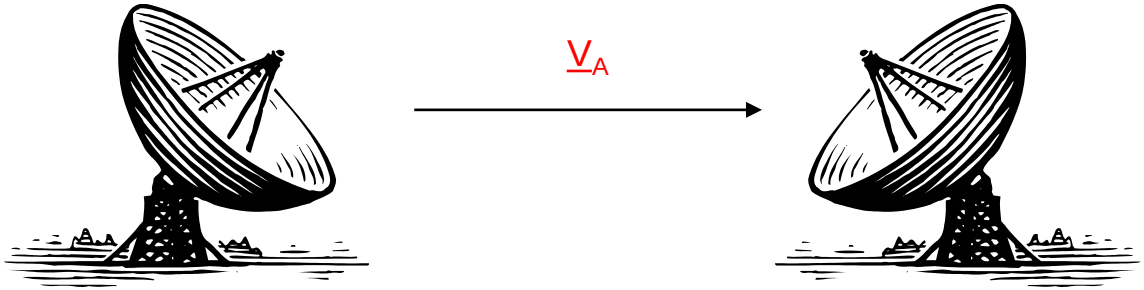


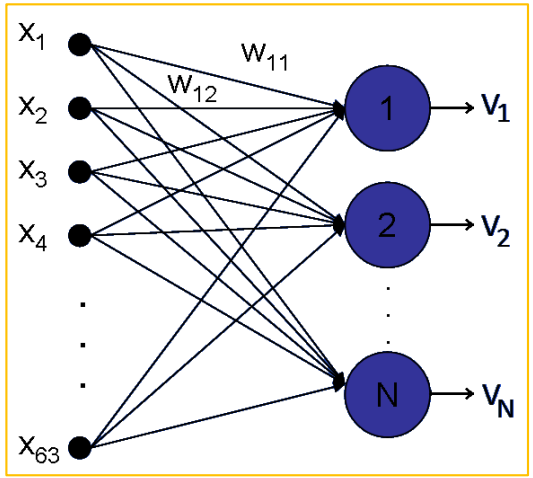
Receptor

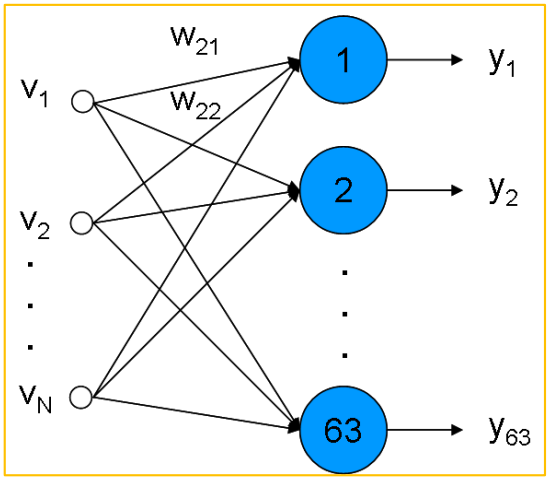


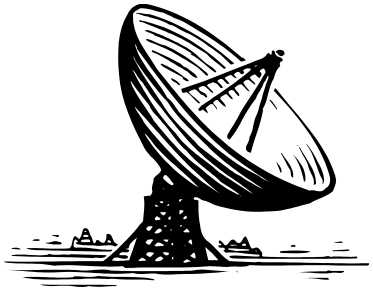
$\underline{V}_A$



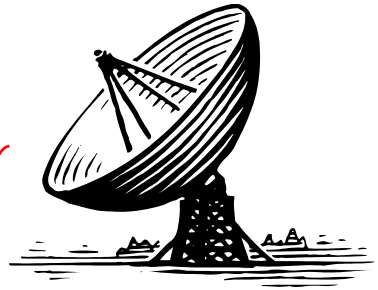

$\underline{V}_A$
Transmissor
Receptor


x_1
x_2
x_3
x_4
x_{63}
w_{11}
w_{12}
1
2
N
V_1
V_2
V_N

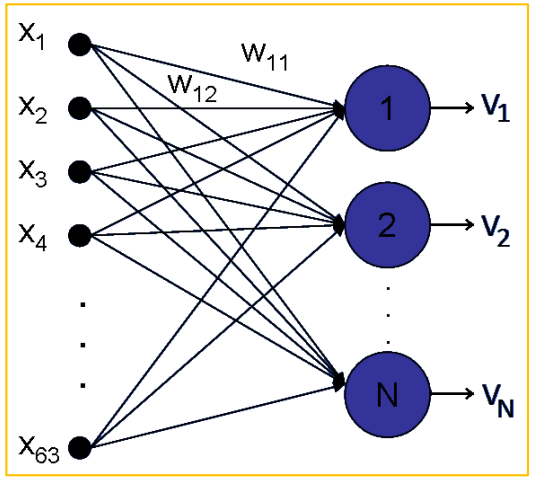

v_1
v_2
v_N
w_{21}
w_{22}
1
2
63
y_1
y_2
y_{63}



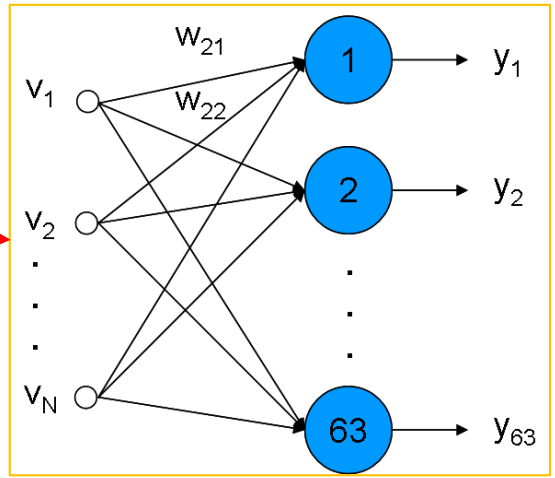
Transmissor

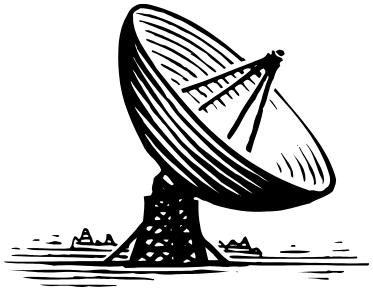


Receptor

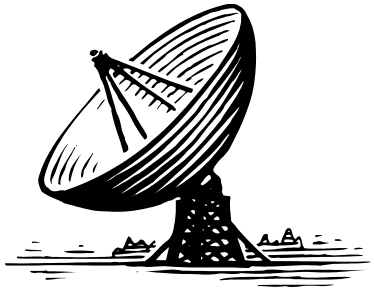


$\underline{V}_A$

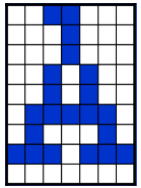


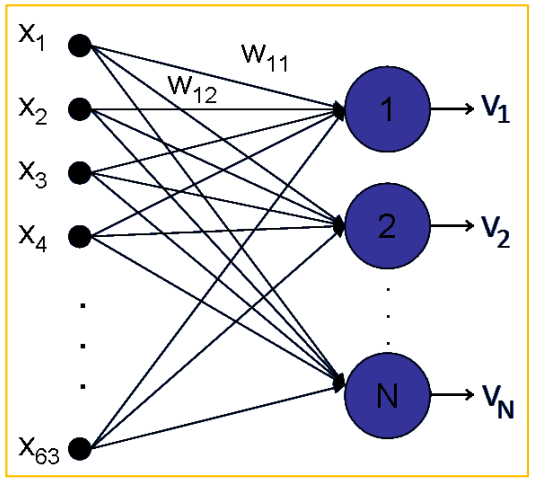


Transmissor

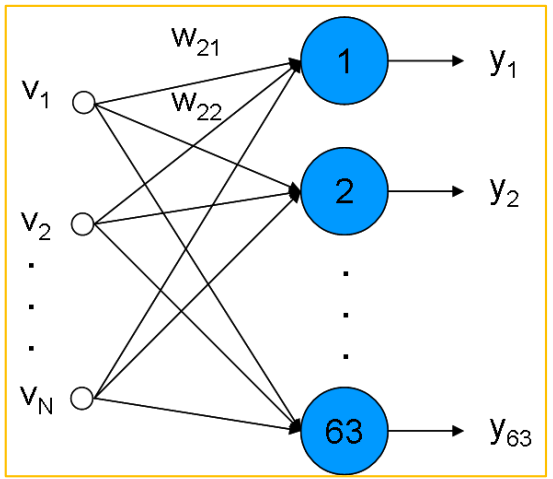


Receptor





$\underline{V}_A$



Referências

1. *Neural Networks and Learning Machines*, 3rd. Edition, Simon Haykin

2. *Fundamental of Neural Networks - Architectures, Algorithms and Applications*, Laurene Fausett

3. *Pattern Classification*, Richard O. Duda, Peter E. Hart, David G. Stork