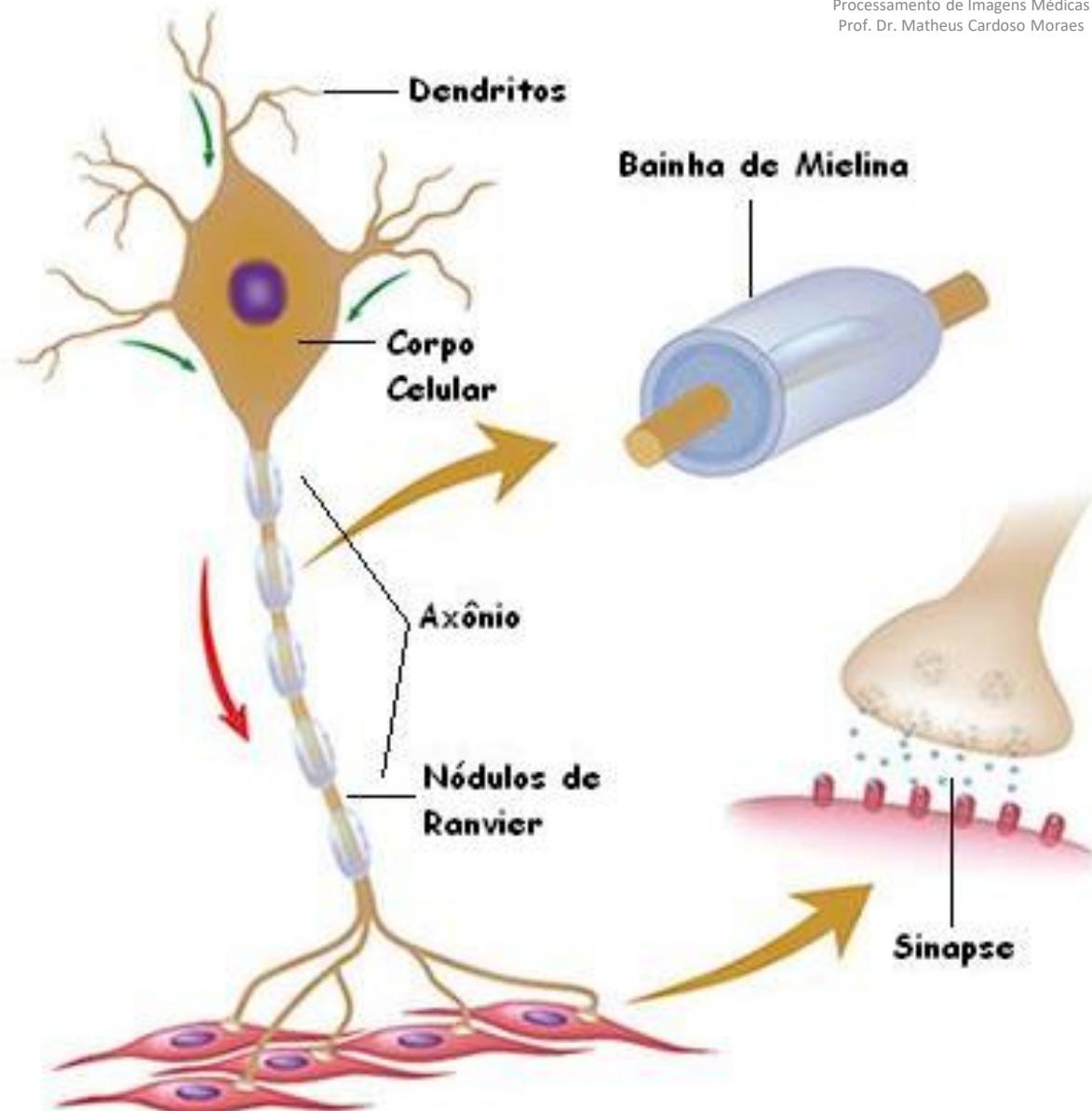


Introdução à Redes Neurais Artificiais

Prof. Dr. Matheus Cardoso Moraes

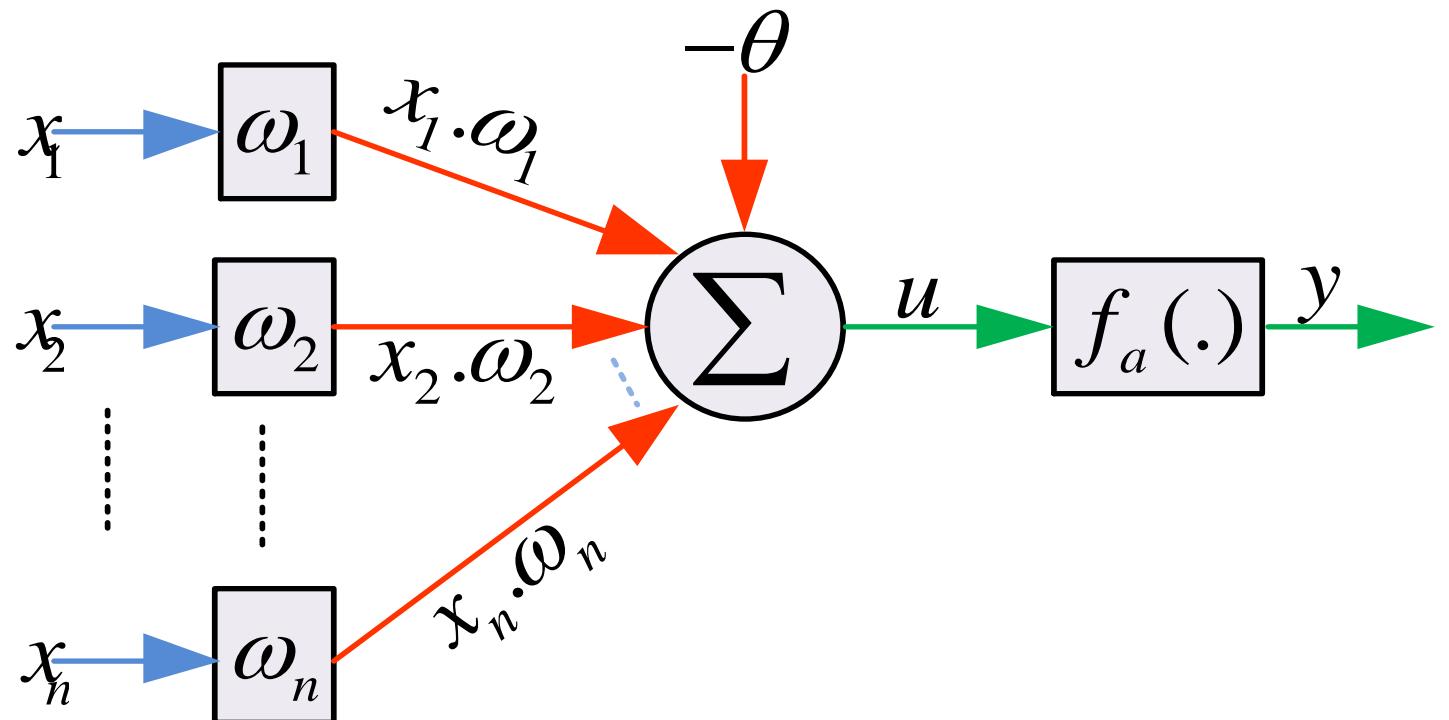
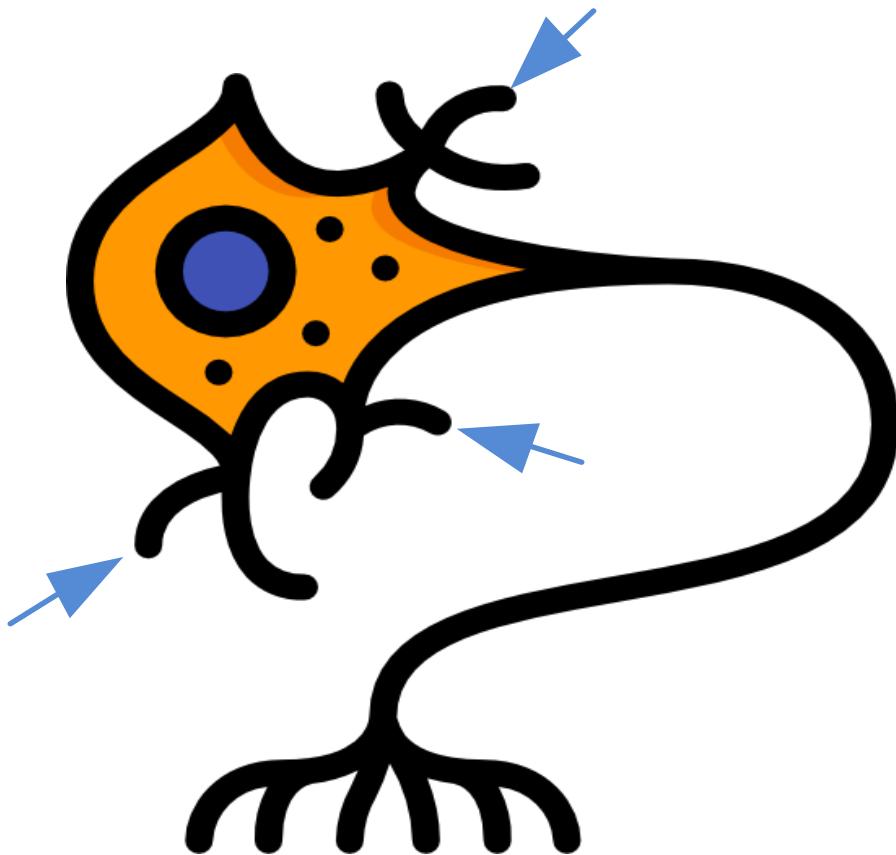
Estrutura Artificial Baseada no Neurônio Biológico

- Modelo artificial sistêmico e Matemático MCP(McCulloch e Pitts) baseado no neurônio biológico
 - Dendritos recebem estímulos
 - Corpo Celular (Soma) agraga estímulos
 - Corpo Celular e Axônio Ativam e Modulam impulso neuronal



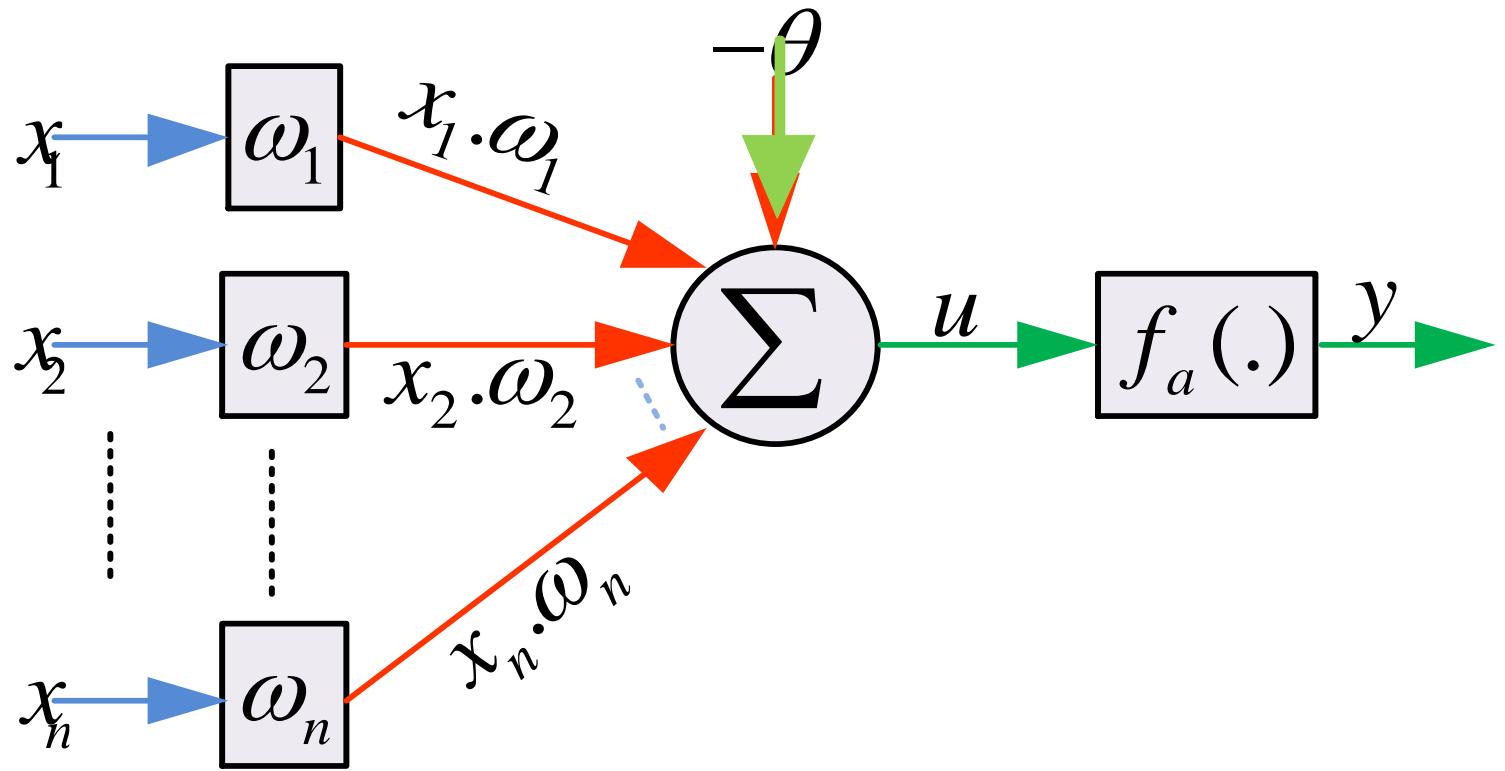
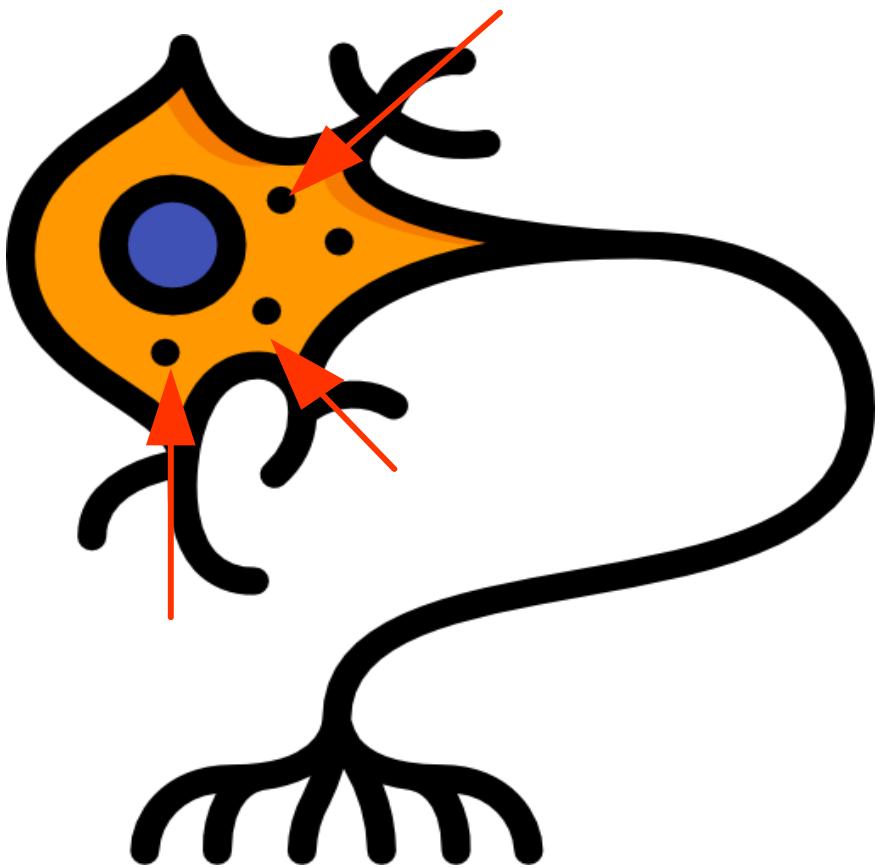
Estrutura

- **Estimulo** → Dendritos recebem estímulos dos neurotransmissores através das fendas sinápticas.
- **Sinais de Entrada** → $\{x_1, x_2, \dots, x_n\}$ → neurotransmissores
- **Pesos Sinápticos** → $\{\omega_1, \omega_2, \dots, \omega_n\}$ → fenda sináptica



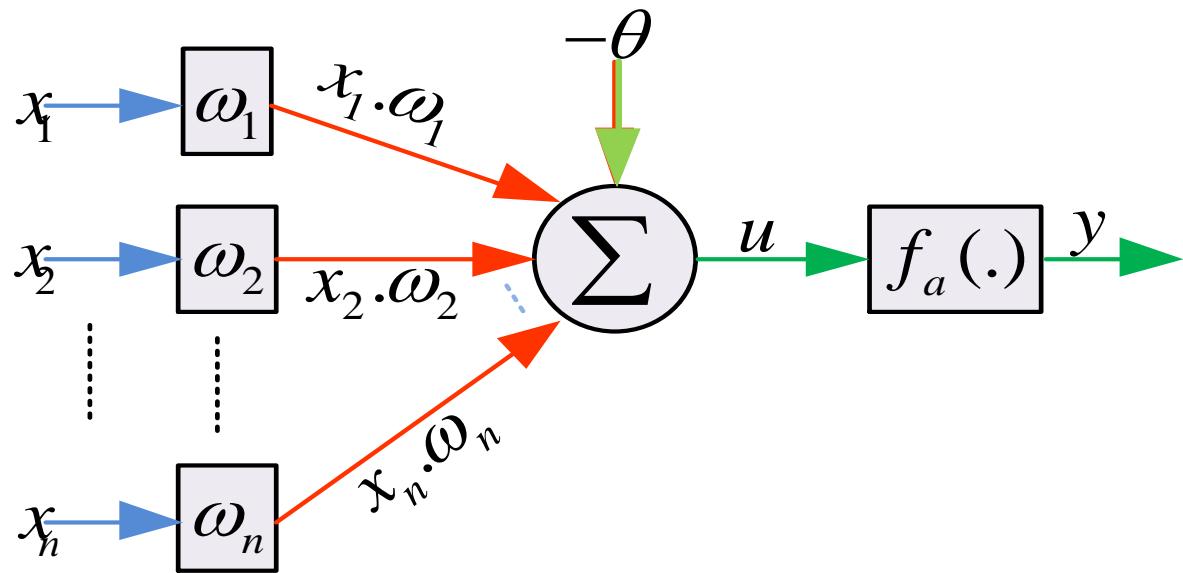
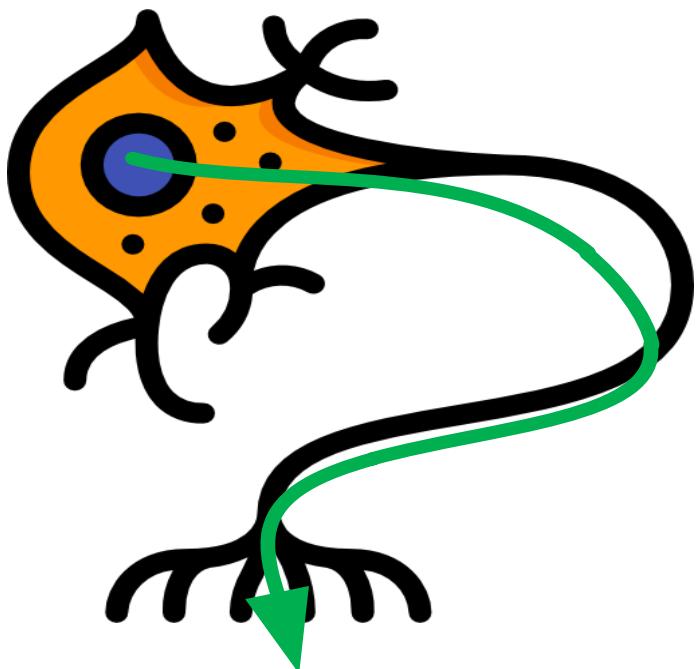
Estrutura

- Impulso Local → É gerado o impulso nervoso local em direção ao corpo neuronal.
- Energia do Impulso $\{x_i \cdot \omega_i\}$ → Combinação entre neurotransmissor e fenda sináptica



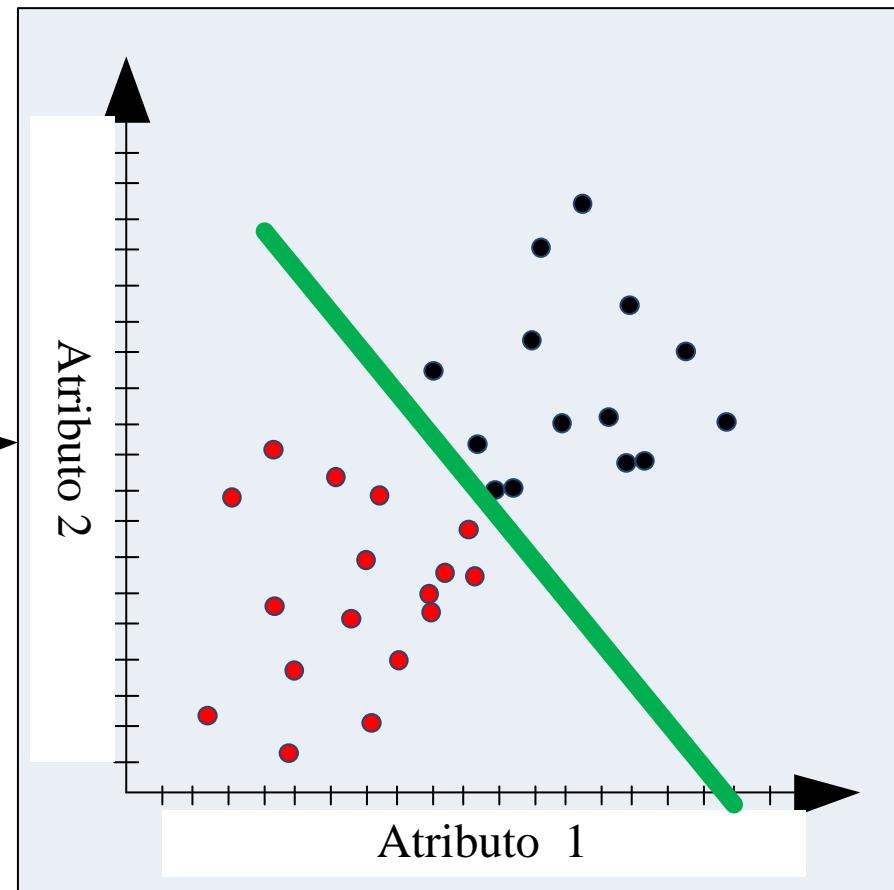
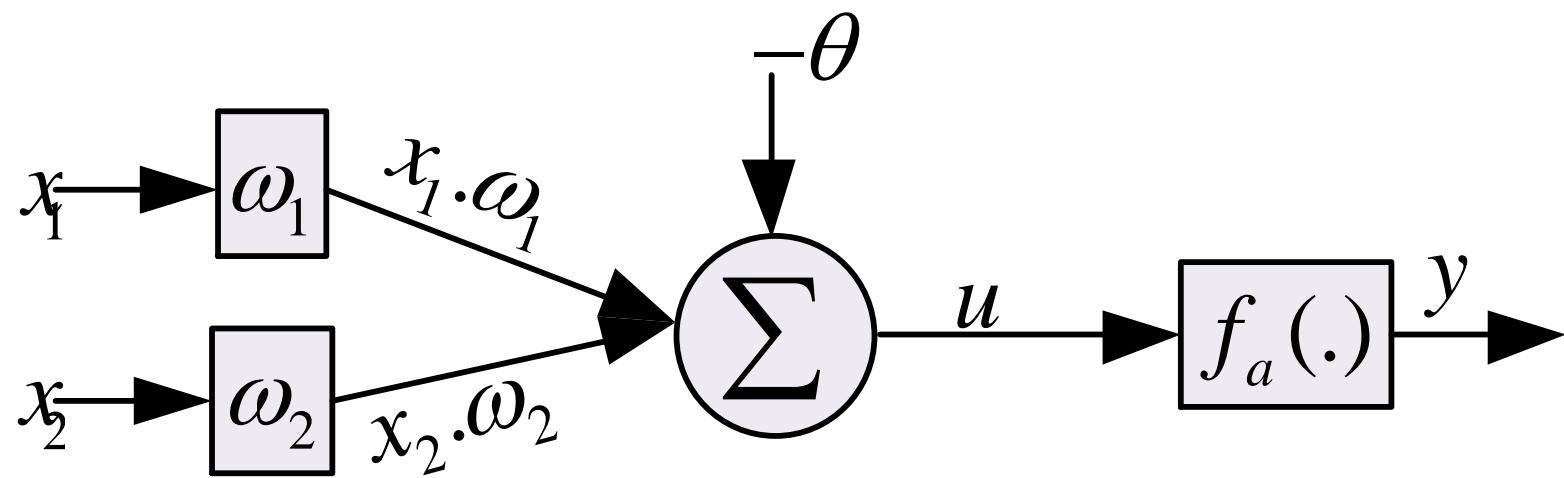
Estrutura

- Ativação Neuronal → Impulso resultante no corpo neuronal, uma combinação de fatores ativam o impulso com diferentes magnitudes.
- Limiar de Ativação $\{\theta\}$ → Limiar de ativação do corpo neuronal.
- Potencial de Ativação $\{u\}$ → Agrega impulsos dos diferentes dendritos
- Função de Ativação $\{f_a(\cdot)\}$ → Função que Modula o impulso no Axônio para a saída $\{y\}$.



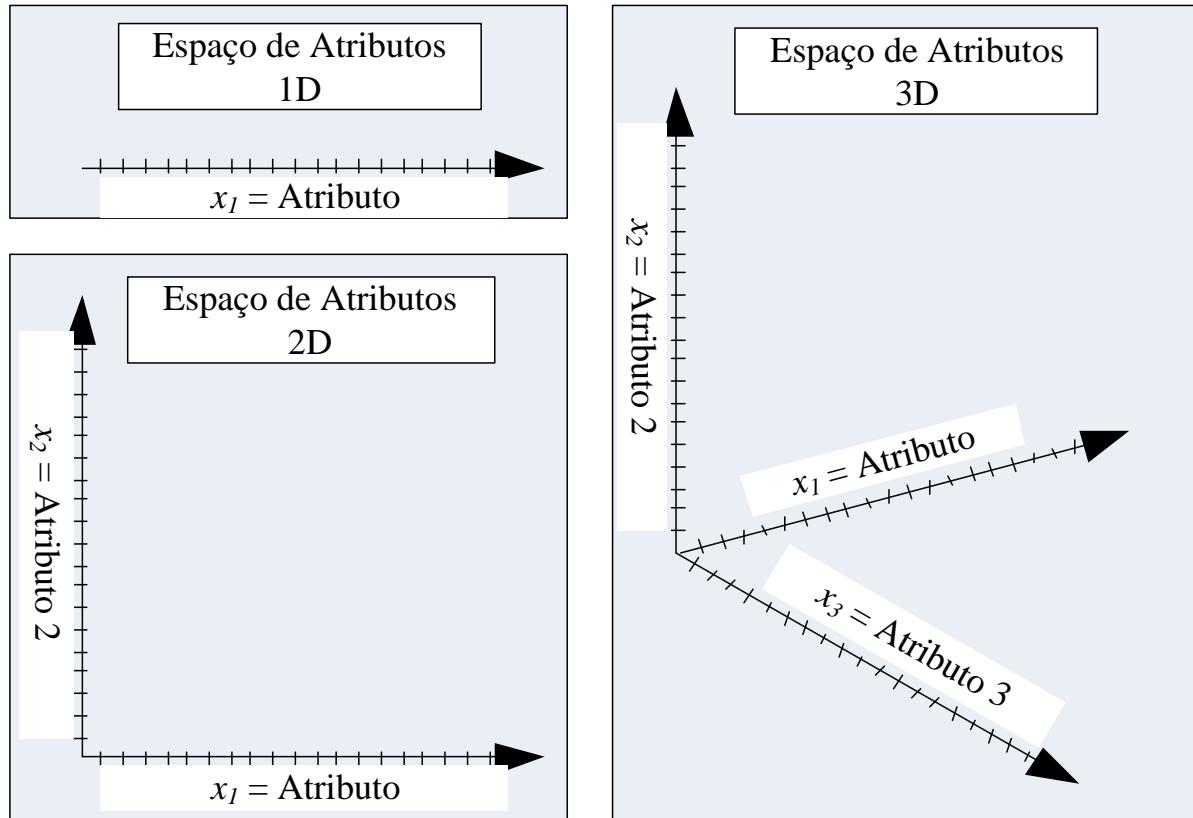
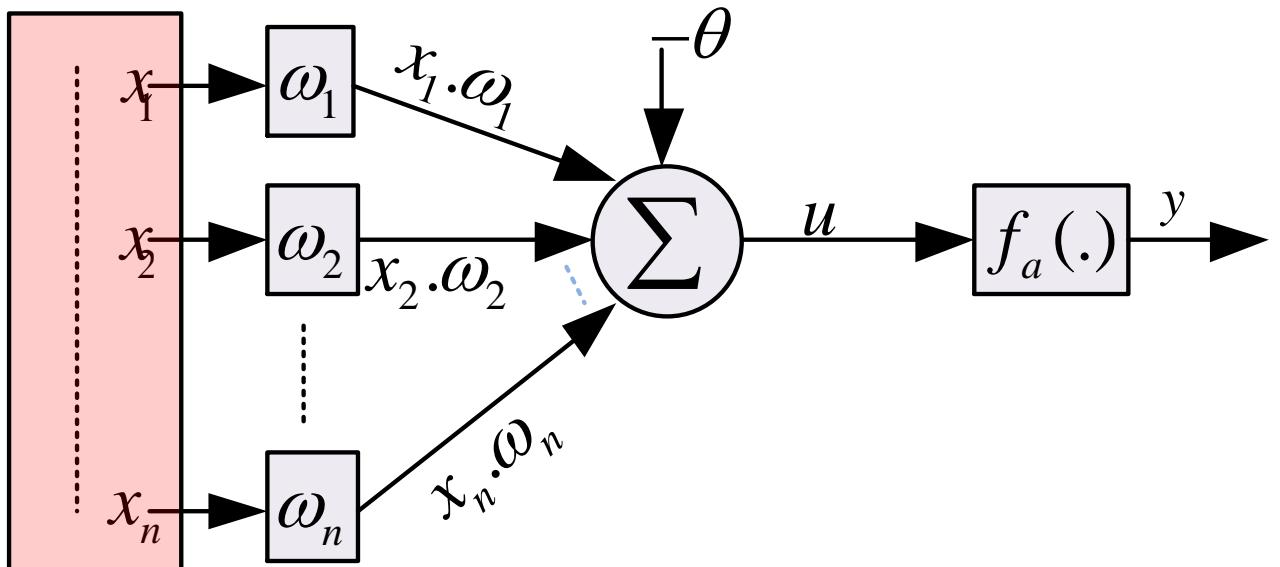
Estrutura Artificial no Contexto Computacional

- Definição:
- Classificador de aprendizado supervisionado
 - Posiciona e Molda o hiperplano para a classificação



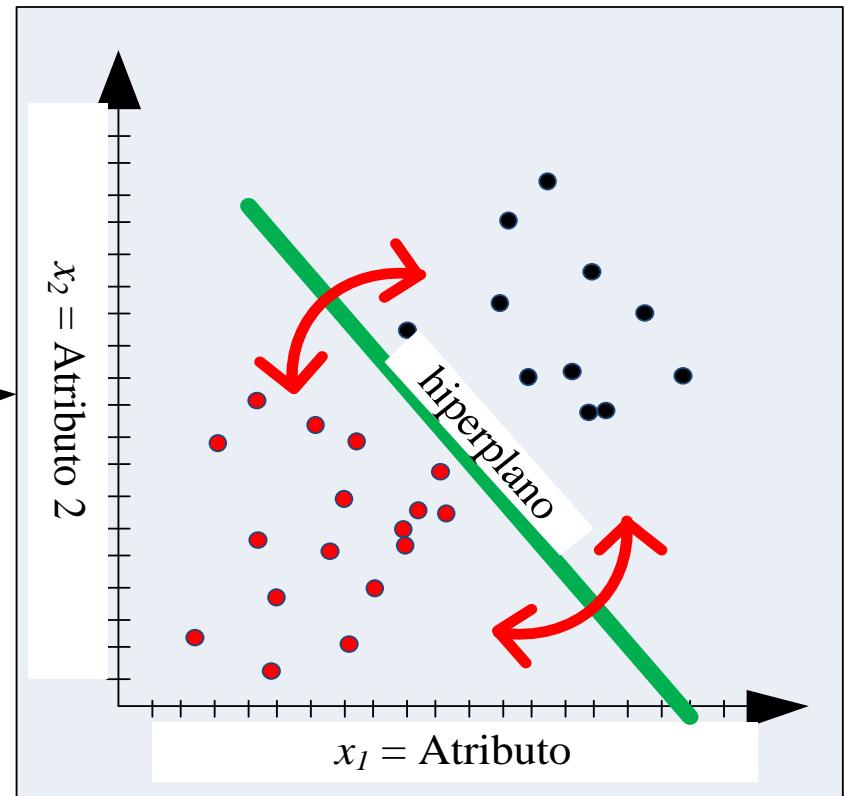
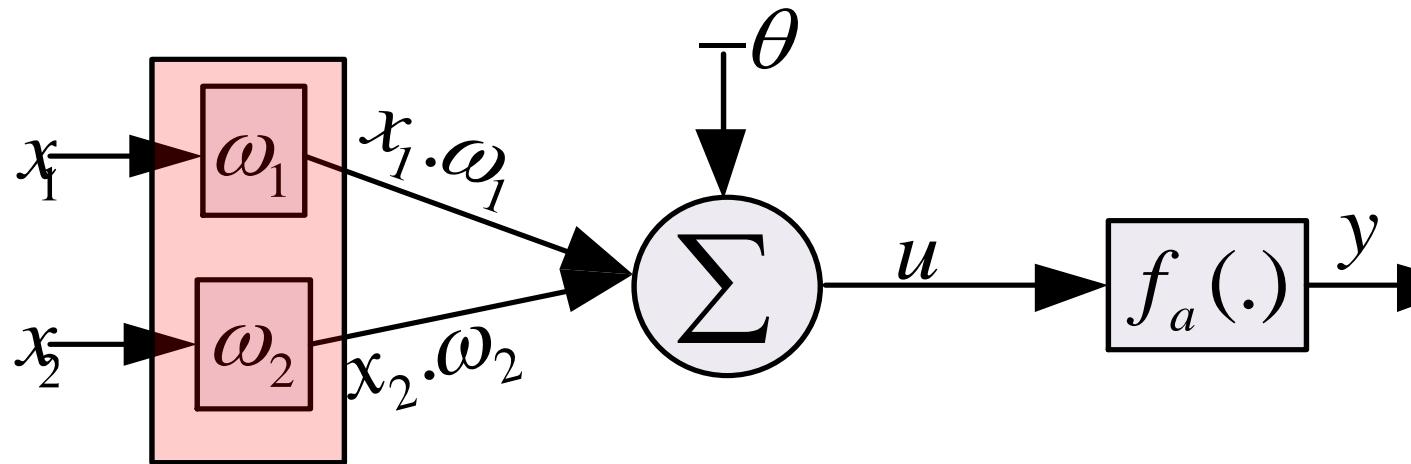
Entradas ou Atributos

- $\{x_1, x_2, \dots, x_n\} \rightarrow$ As entradas são atributos:
 - Definidos na Construção da Rede \rightarrow Engenharia de Atributos
 - Quantidade define dimensão do espaço
 - Componentes dos vetores
 - Normalizar entre 0 – 1, pois podem vir de grandezas diferentes.



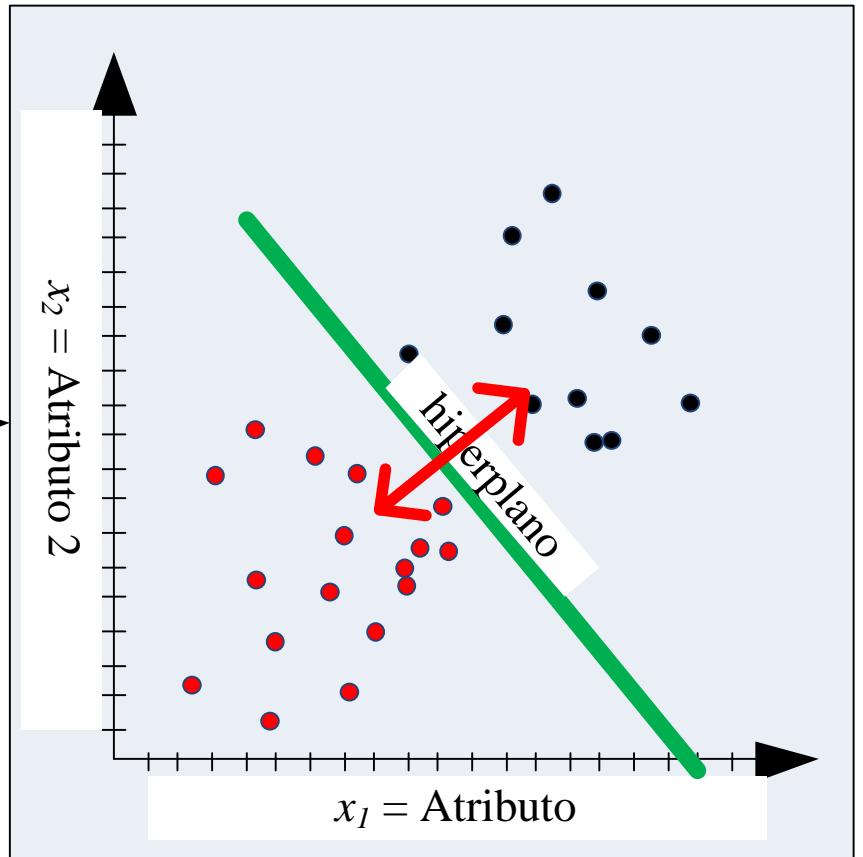
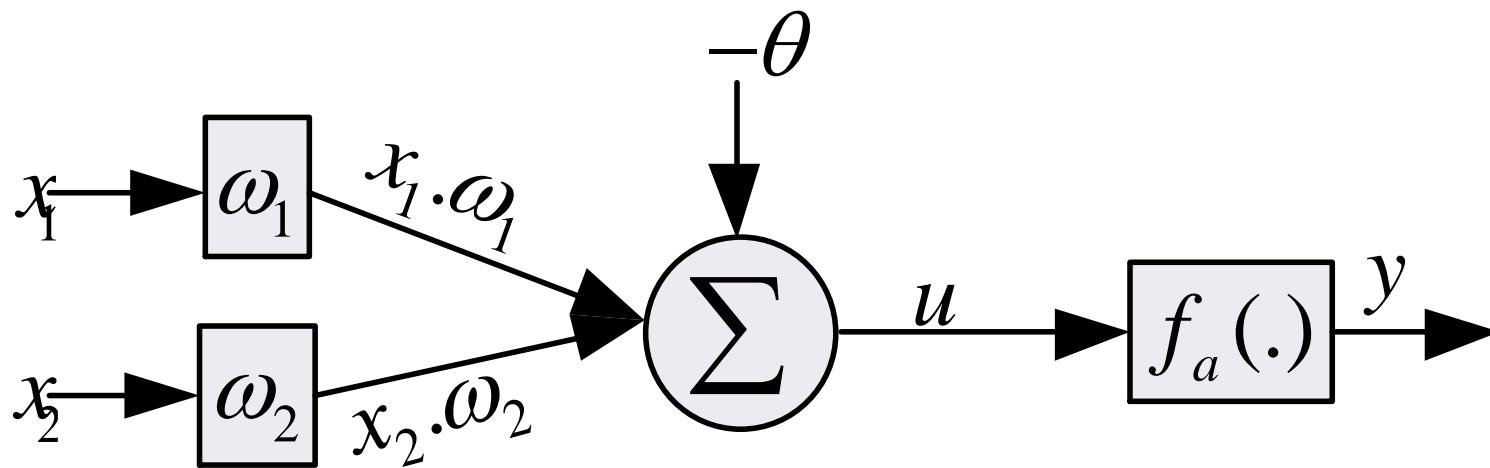
Pesos

- $\{\omega_1, \omega_2, \dots, \omega_n\} \rightarrow$ Os pesos ponderam a contribuição individual de cada entrada para a classificação.
 - Atuam na inclinação do hiperplano para separação de classes.
 - Valores optimizados no treinamento



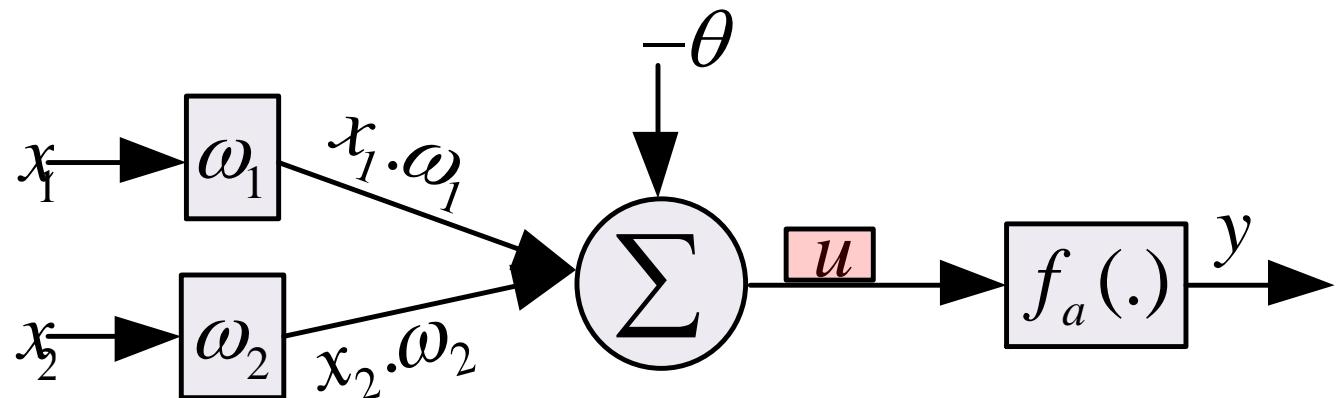
Bias

- $\{\theta\} \rightarrow$ Bias, limiar para classificação de classes.
 - Desloca o hiperplano para separação de classes.
 - É o gatilho da Ativação.
 - Valor optimizado no treinamento



Potencial de Ação

- $\{u\} \rightarrow$ Potencial de Ação, unifica as energias das entradas:
 - Parâmetro que computa a magnitude da combinação de estímulos, pesos e bias.
 - Parâmetro da Função de Ativação.
 - Correspondente a um parâmetro de uma função de pertinência.
 - Representação matemática intrínseca ao Nurônio

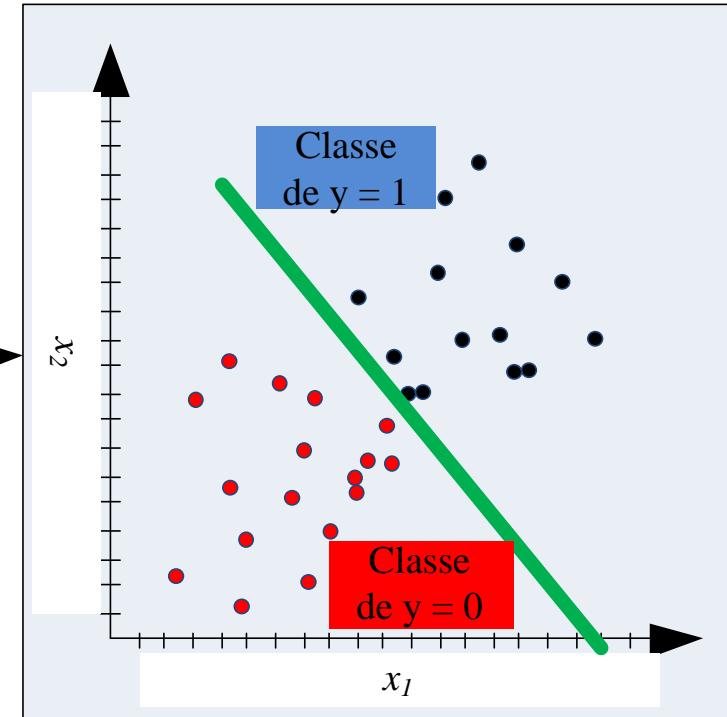
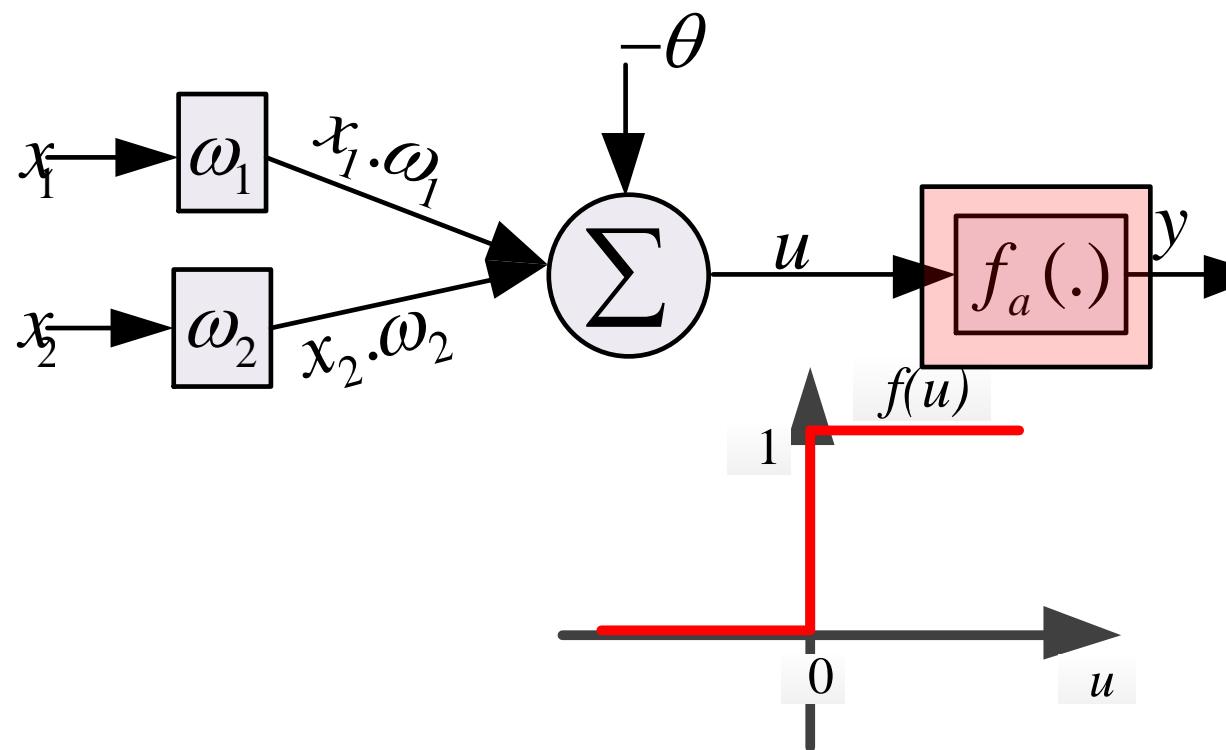


$$u = \sum_{i=1}^n x_i \cdot \omega_i - \theta$$

$$u = x_1 \cdot \omega_1 + x_2 \cdot \omega_2 - \theta$$

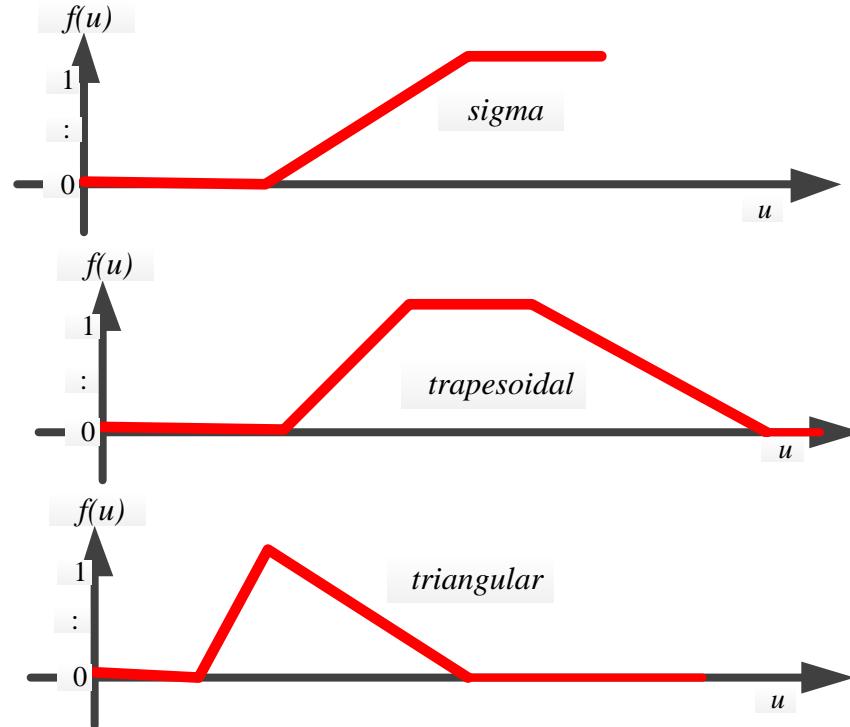
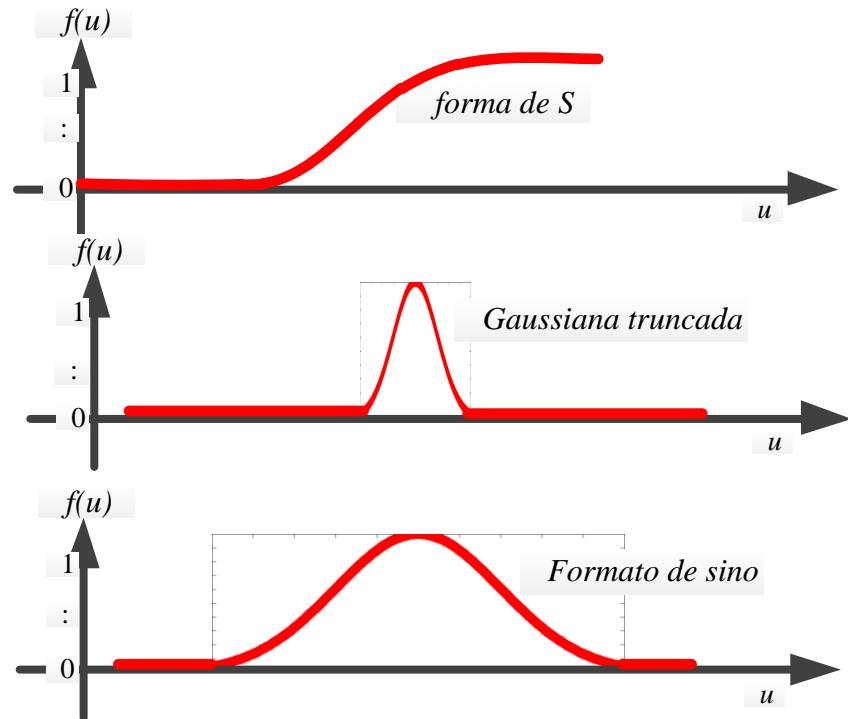
Função de Ativação

- $\{f(u)\} \rightarrow$ Função de Ativação, função que ativa e modula a saída.
- Decide a classe de um individuo em função de u .
 - Individuo \rightarrow vetor de atributos.



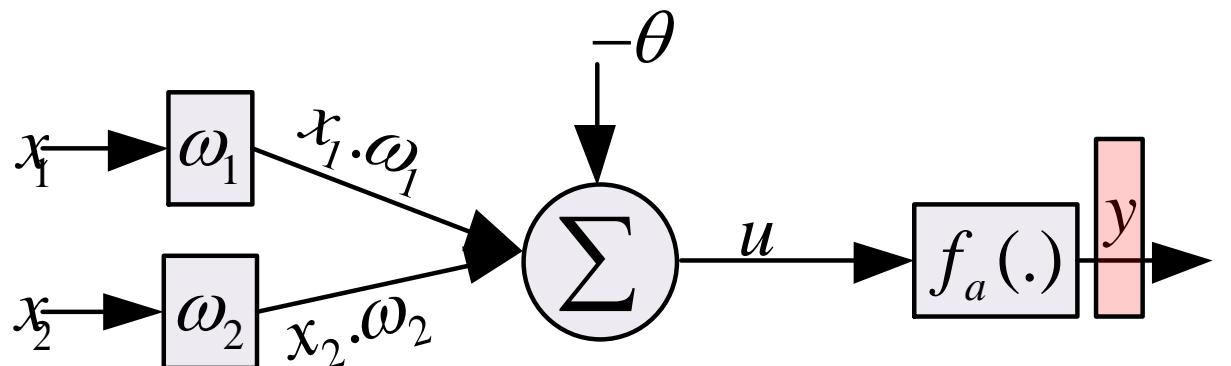
Tipos de Função de Ativação

- $\{f(u)\} \rightarrow$ Funções de Ativação
 - Molda o hiperplano para separação de classes.
 - Definidos na Construção da Rede \rightarrow Deve ser escolhida para melhorar separação entre as classes
 - Correspondente a uma função de Pertinência

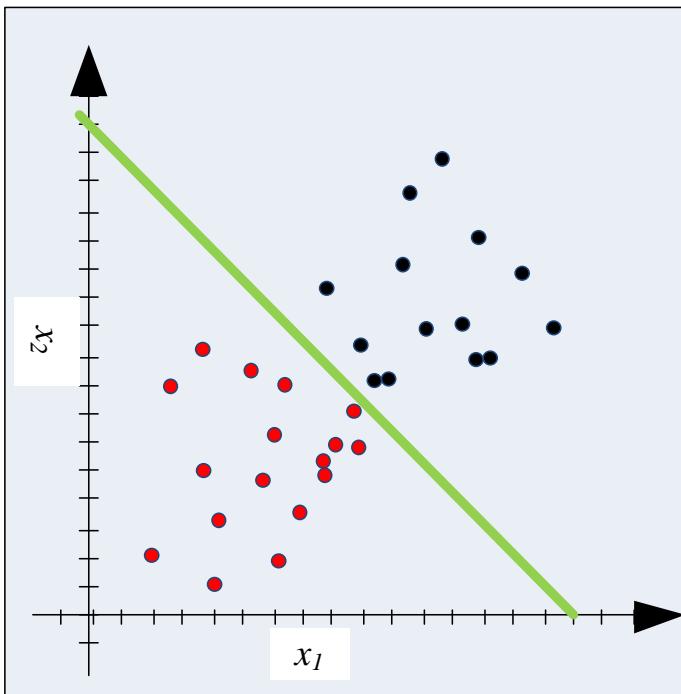


Saída

- $y \rightarrow$ Saída.
- Classifica indivíduos de acordo com:
 - Entrada
 - Função de Ativação
 - Pesos e bias após treinamento
- O Espaço de Atributos com o hiperplano é uma representação das possibilidades de saída da rede construída e treinada.

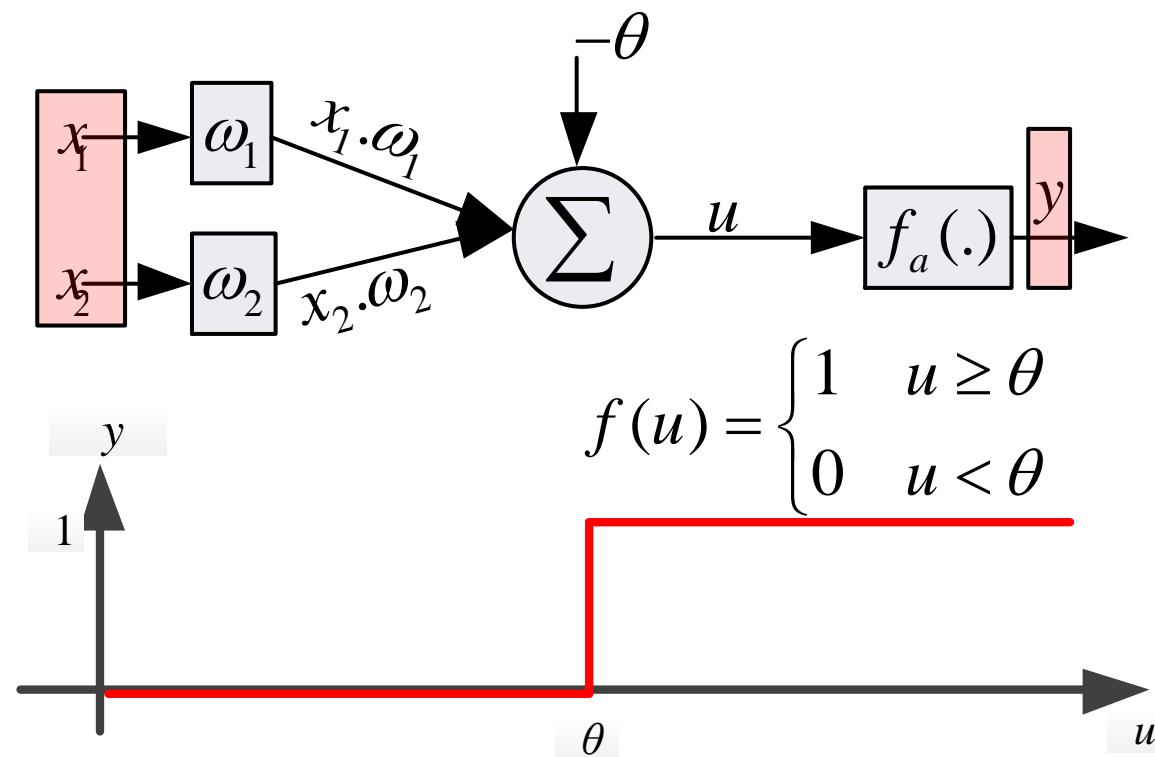


$$y = f_a \left(\sum_{i=1}^n x_i \cdot \omega_i - \theta \right)$$



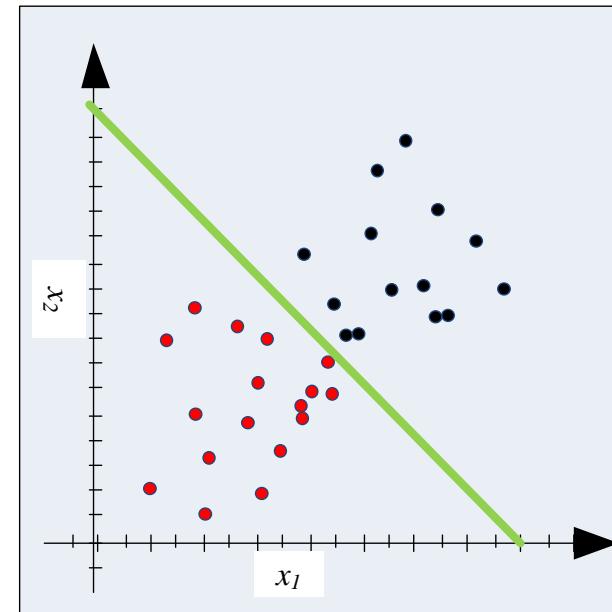
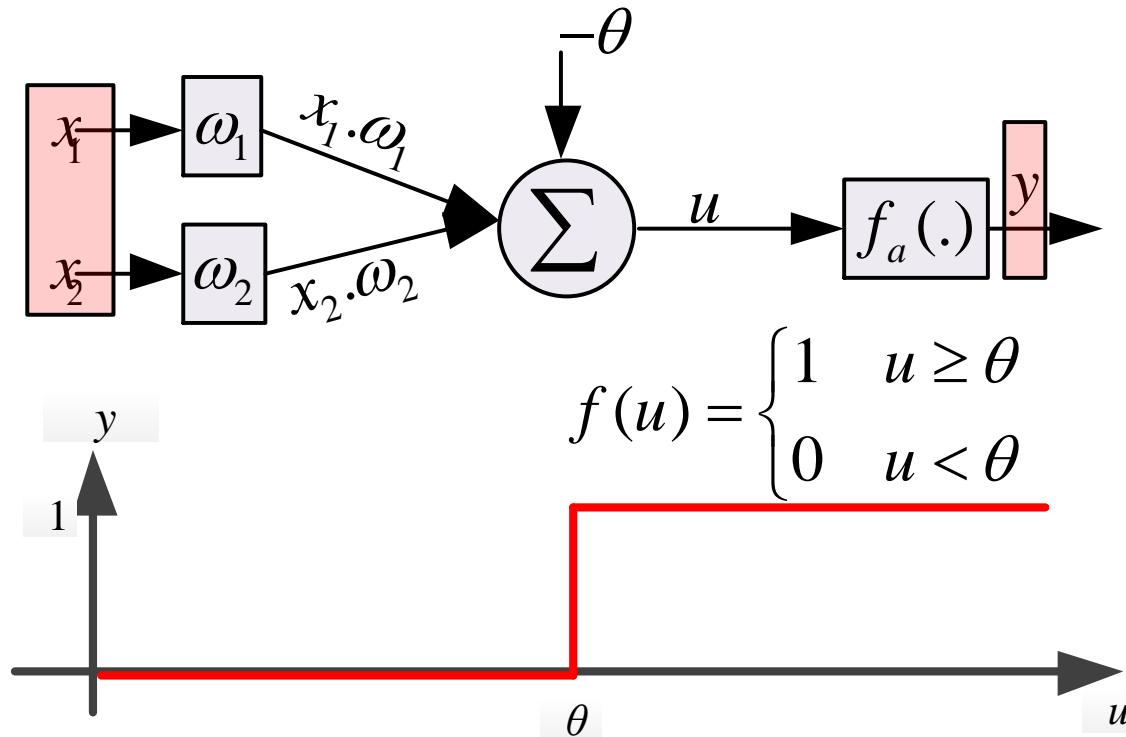
Modelo Perceptron Formal

- Entrada ~**Booleana**: $x \in \{0,1\}^n$
- Saída Modulada por $f(u)$ **Degrau**:
 - Saída Booleana $y: \{0,1\}^n \rightarrow \{0,1\}$ para $\omega \in R^n$
 - Hiperplano linear (reta)
 - Portas E, OU, NÃO E, NÃO OU



Modelo Perceptron Variação

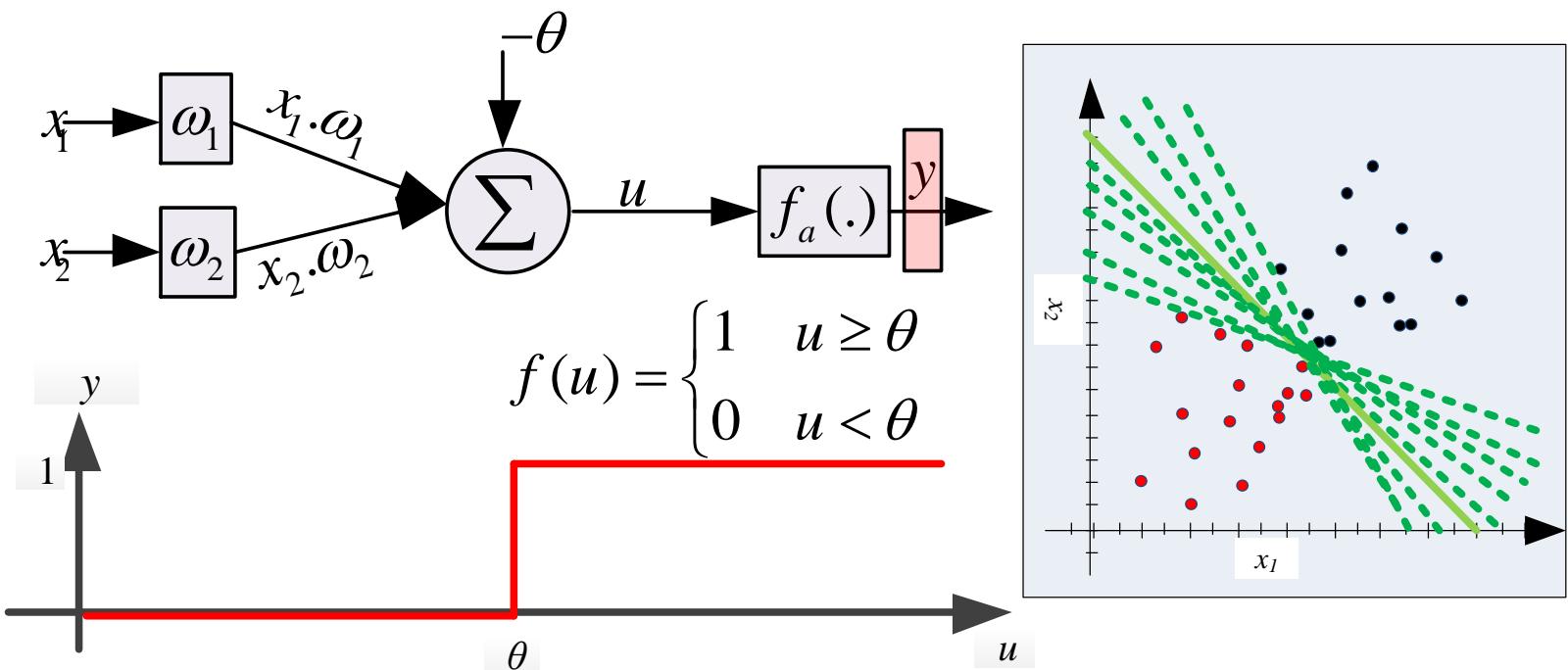
- Entrada pode ser **não Booleana**: $\{x \in R^n \mid 0 \leq x \leq 1\}$
- Saída Modulada por $f(u)$ **Degrau**:
 - Saída Booleana $y: \{0,1\}^n \rightarrow \{0,1\}$ para $\omega \in R^n$
 - Hiperplano linear (reta)
 - Classifica combinações de entradas



Modelo Perceptron Aprendizado

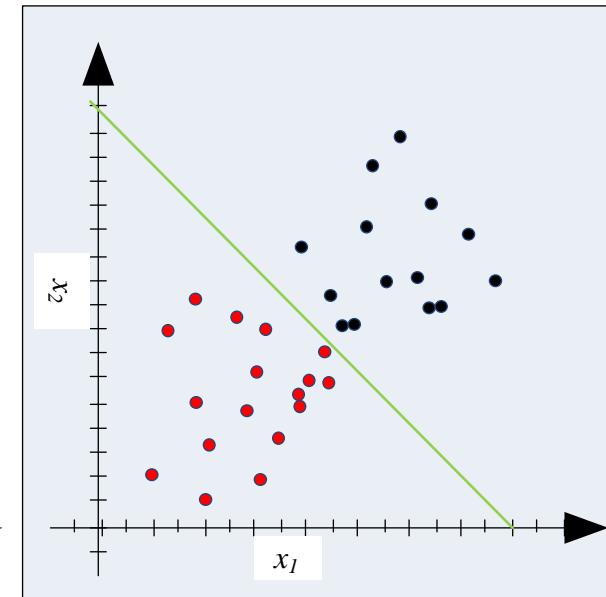
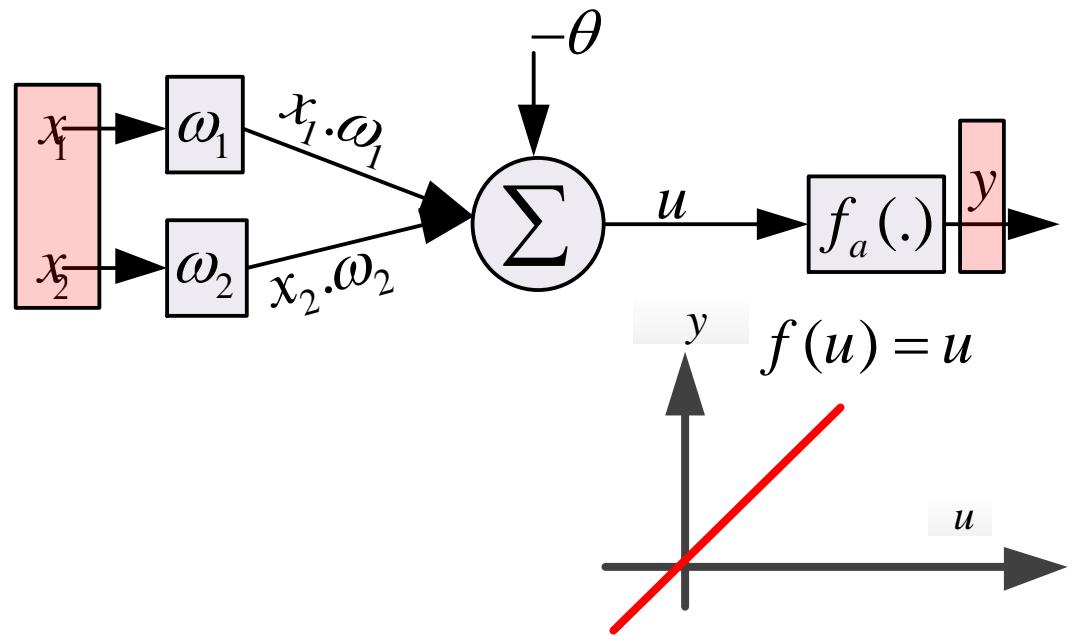
- Aprendizado:

- Por erro Individual $\rightarrow e^p$
- Com mesmo conjunto, Γ , pode gerar infinitos hiperplanos que satisfaçam a saída



Modelo Adaline Formal

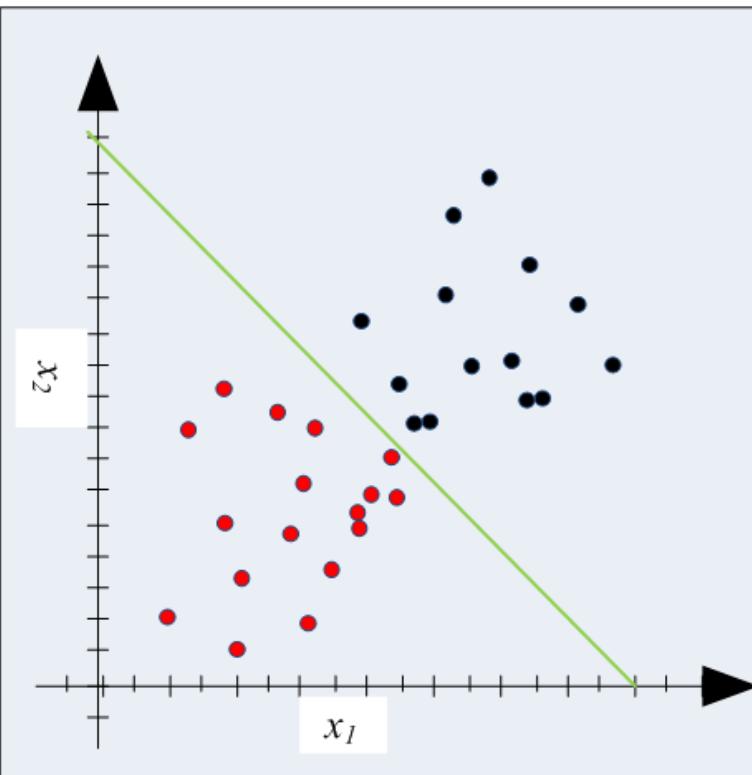
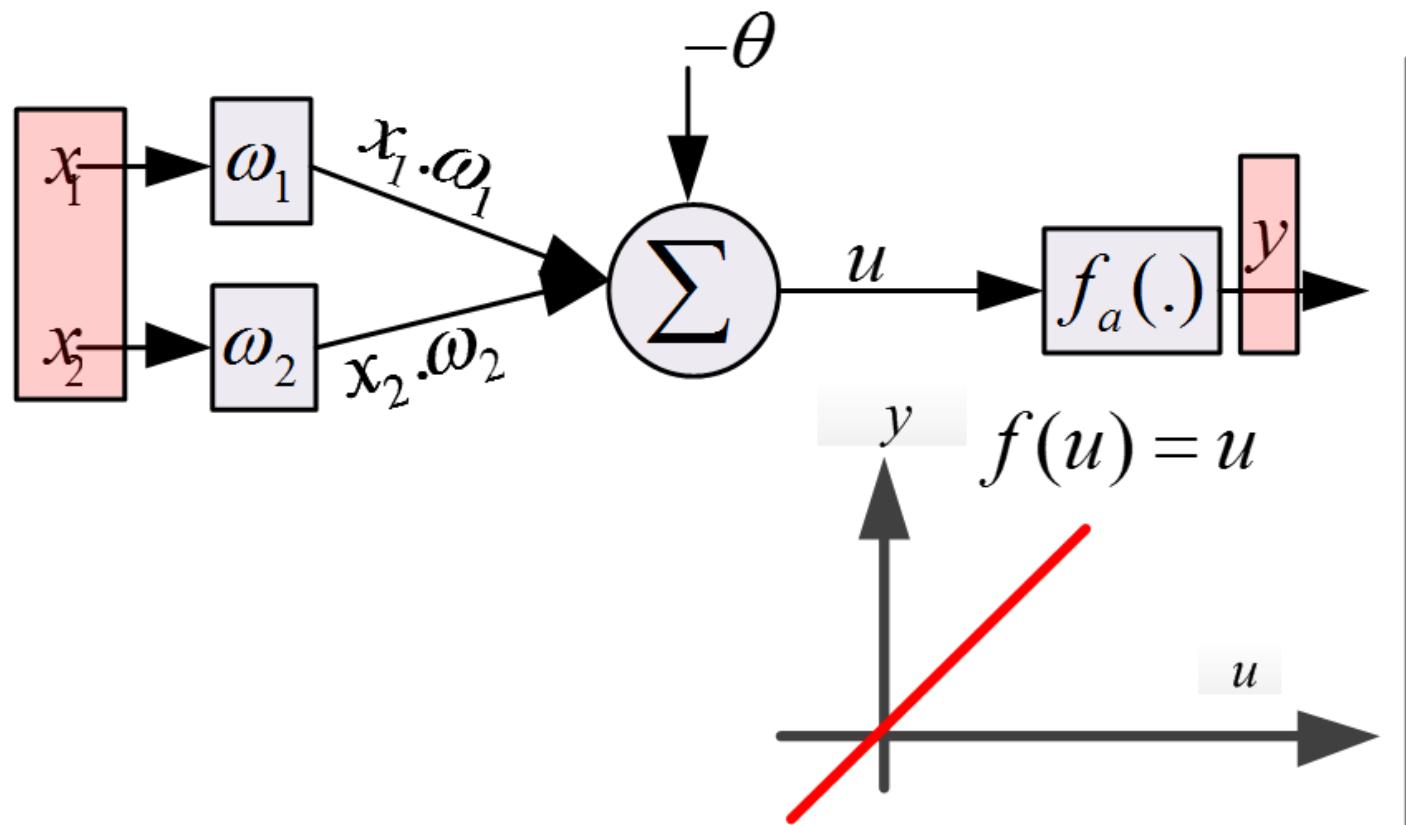
- Entrada pode ser **Linear**: $\{x \in R^n \mid -1 \leq x \leq 1\}$
- Saída Modulada por $f(u)$ **Linear**:
 - Saída Linear $\{y \in R^n \mid -1 \leq y \leq 1\}$
 - Hiperplano linear (reta)
 - A reta representa o ponto médio de uma rampa que gradua as saídas entre -1 e 1.
 - **Aplicação: Probabilidade de um individuo pertencer a uma classe de duas classes.**
(reta)



Modelo Adaline Aprendizado

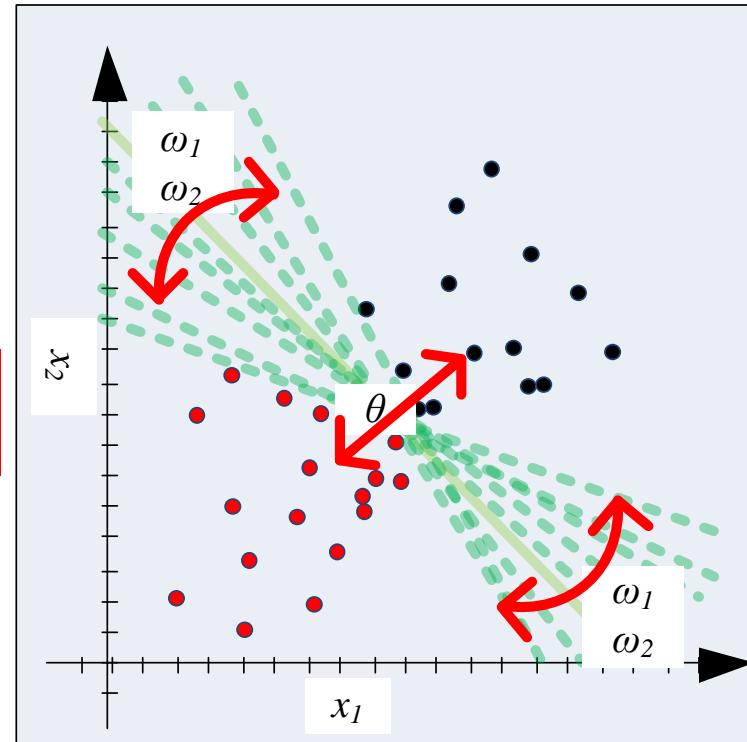
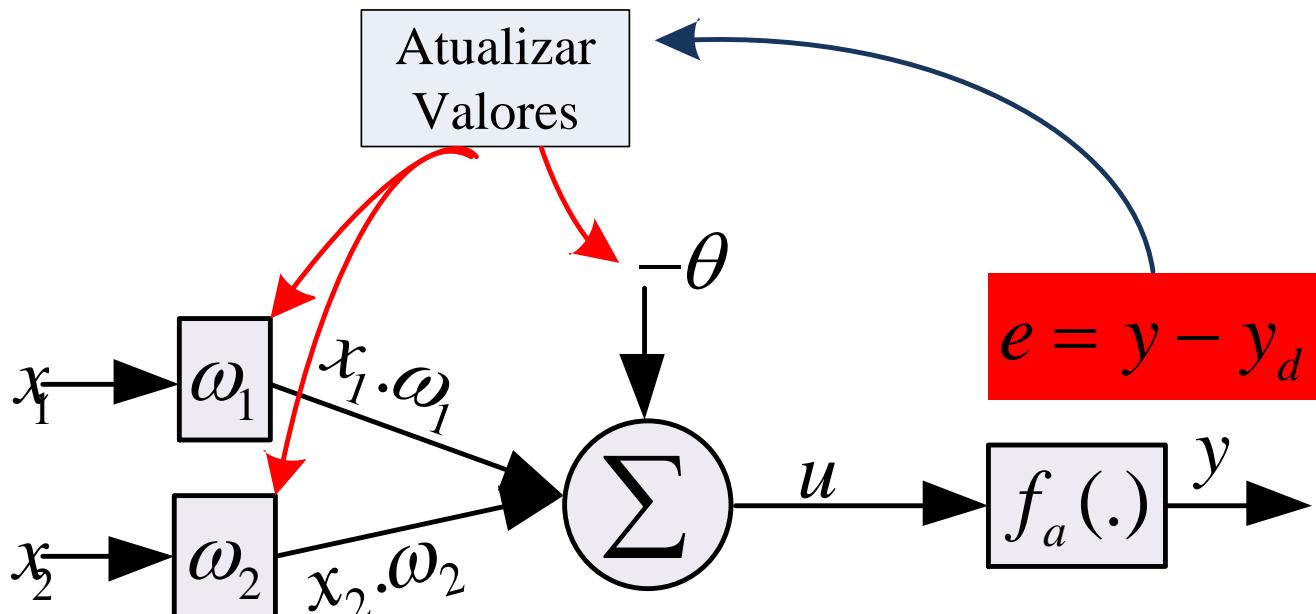
- Aprendizado:

- Por erro médio quadrático $\rightarrow e_{qm}$
- Com mesmo conjunto, Γ , gera sempre um hiperplano que satisfaça a saída



Treinamento ou Aprendizado: Visão Geral

- Computar/Alterar valores dos pesos e bias visando uma saída desejada para um conjunto de entrada.
 - Computar erro entre saída obtida y e saída desejada y_d .
 - Alterar valores de pesos e bias, visando a minimização do erro da saída.

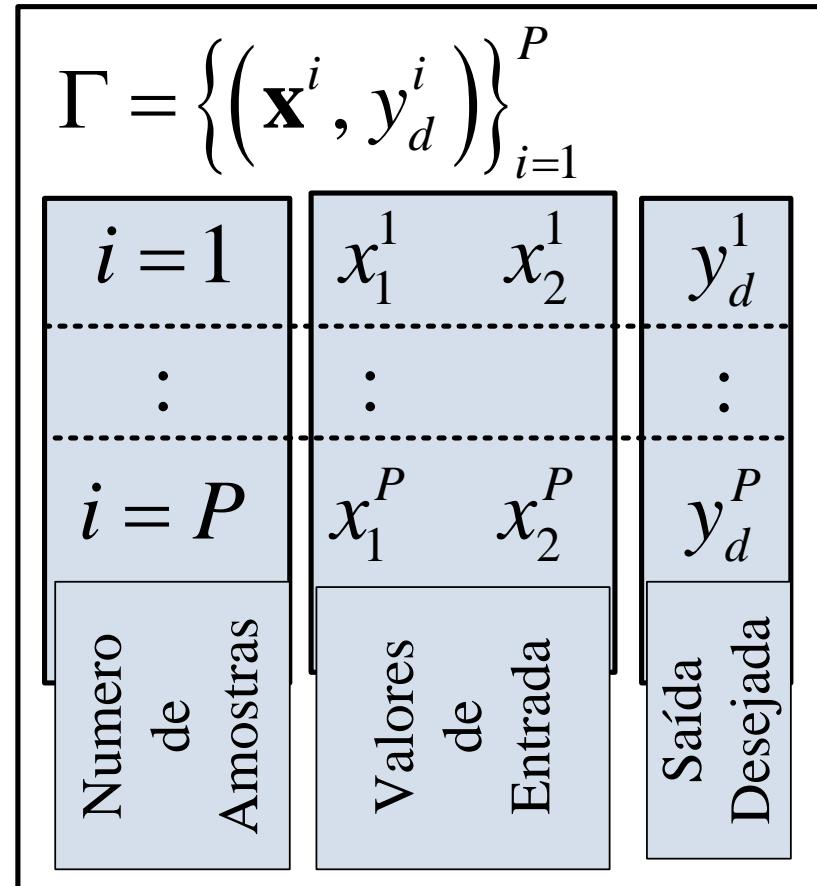


Treinamento ou Aprendizado

- Formalização:
 - Conjunto de treinamento $\rightarrow \Gamma = \{(x^i, y_d^i)\}_{i=1}^P$ **Lista de exercícios com gabarito** ☺
 - i e P é são os índices, e total de indivíduos do conjunto de treinamento, respectivamente.
 - Entrada $\rightarrow x^i$. **Lista de perguntas** ☺
 - Saída desejada $\rightarrow y_d$. **Gabarito** ☺
 - Saída obtida $\rightarrow y$. **Resultado do exercício** ☺
 - Erro $\rightarrow e = y - y_d$. **Conferindo as listas** ☺
 - Erro Individual $\rightarrow e^i = y^i - y_d^i$
 - Erro Médio Quadrático $\rightarrow e_{qm} = \frac{\sum_{i=1}^P (y^i - y_d^i)^2}{P}$
 - Aprendizado $\rightarrow \omega(n+1) = \omega(n) + \eta e x$. **Aprendendo por tentativa e erro, fazendo as listas** ☺
 - Taxa de Aprendizado $\rightarrow \{0 \leq \eta \leq 1\}$, Praxe $\eta=0,05$. **Inteligência** ☺ **Capacidade de aprender novos conceitos.**
 - Desenvolvedor Define \rightarrow Cuidado como valor
 - Pesos Iniciais $\{0 \leq \omega \leq 1\}$. **Conhecimento atual do assunto** ☺

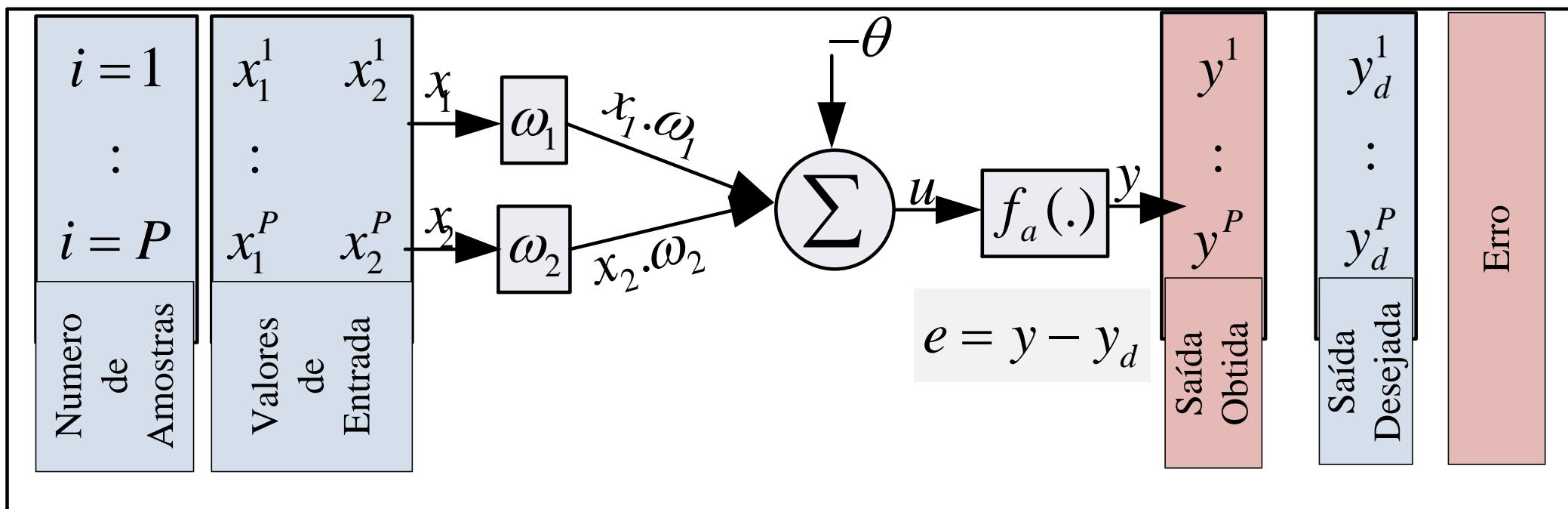
Treinamento ou Aprendizado

- Conjunto de treinamento $\rightarrow \Gamma = \{(x^i, y_d^i)\}_{i=1}^P$ **Lista de exercícios com gabarito ☺**
 - i e P é são os indices, e total de indivíduos do conjunto de treinamento, respectivamente.
- Entrada $\rightarrow x^i$. **Lista de perguntas ☺**
- Saída desejada $\rightarrow y_d$. **Gabarito ☺**



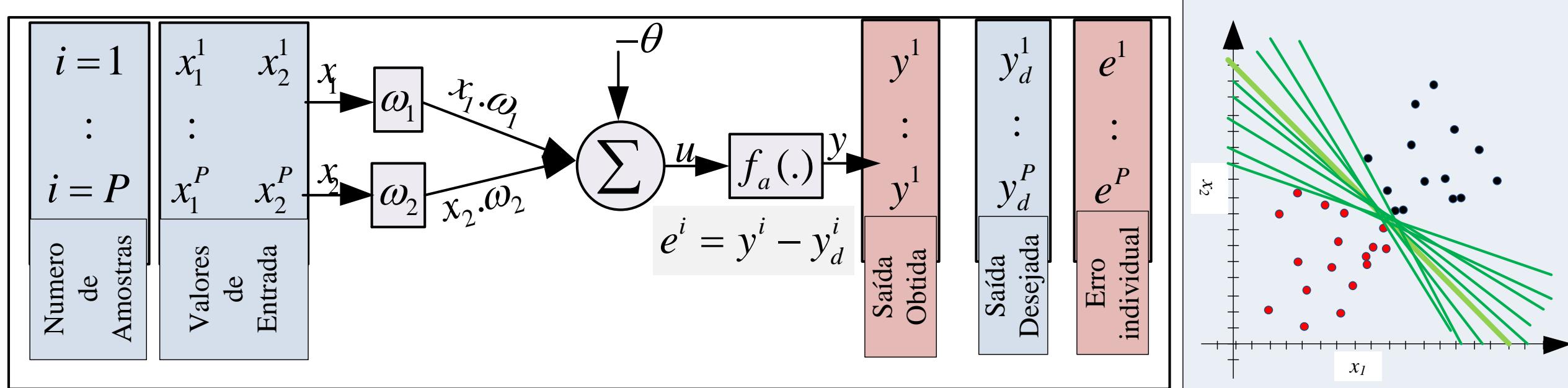
Treinamento ou Aprendizado

- Erro $\rightarrow e = y - y_d$. Conferindo as listas
 - Saída obtida $\rightarrow y$. **Resposta Obtida**
 - Saída desejada $\rightarrow y_d$. **Gabarito**



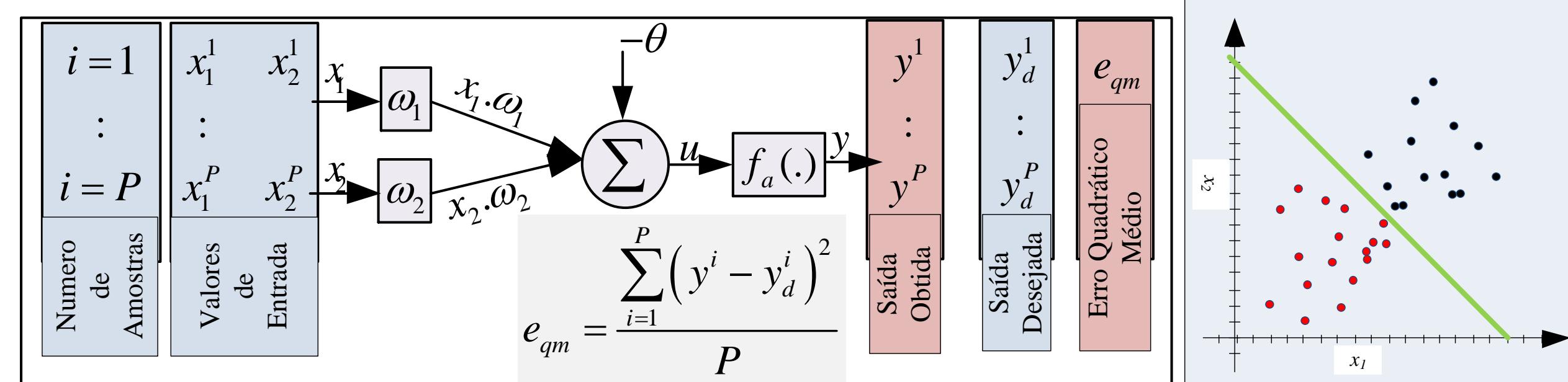
Treinamento ou Aprendizado

- Erro Individual $\rightarrow e^i = y^i - y_d^i$. Fazendo as listas e verificando se acertou cada exercício para finalizar os estudos. ☺
 - Cálculo individual do e de cada individuo do conjunto de amostras de treinamento, Γ .
 - Aprendizado com e^i pode gerar infinitos hiperplanos para um mesmo Γ , todos satisfazem a saída.



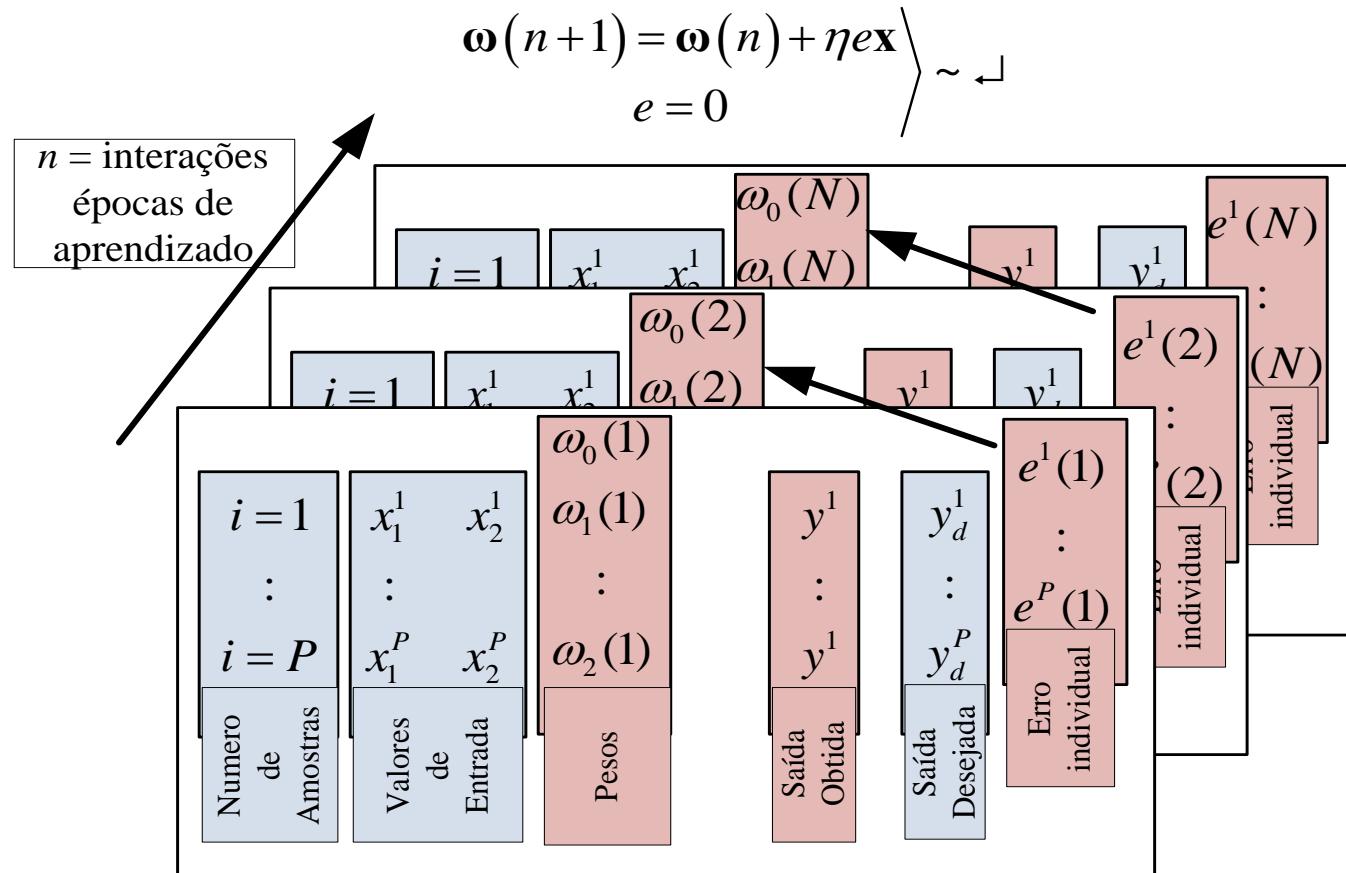
Treinamento ou Aprendizado

- Erro Médio Quadrático $\rightarrow e_{qm} = \frac{\sum_{i=1}^P (y^i - y_d^i)^2}{P}$. Fazendo as listas e verificando a média de acertos da lista toda para finalizar os estudos. ☺
 - Cálculo do EQM do conjunto de amostras de treinamento, Γ .
 - Aprendizado com e_{qm} gera um hiperplanos ótimo para Γ .



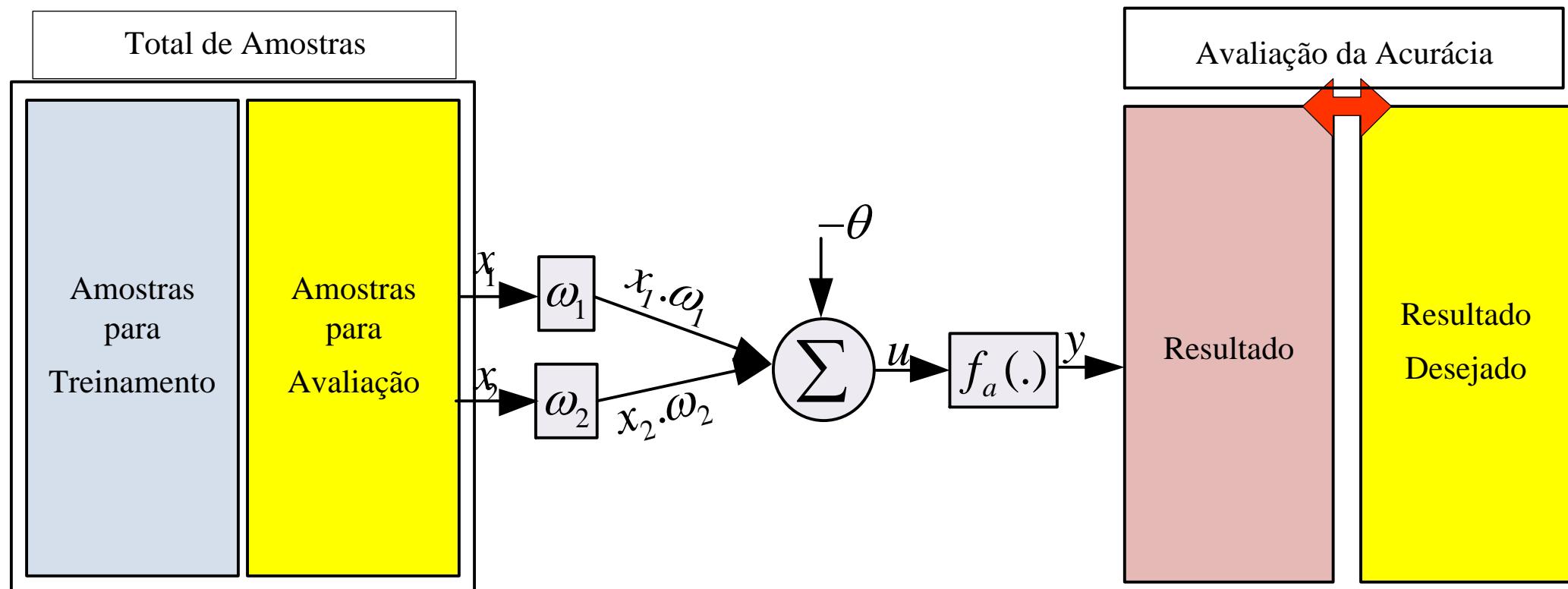
Treinamento ou Aprendizado

- Processo de Aprendizado $\rightarrow \omega(n+1) = \omega(n) + \eta e x$. Aprendendo por tentativa e erro, fazendo as listas 😊
- Taxa de Aprendizado $\rightarrow \{0 \leq \eta \leq 1\}$. Inteligência 😊 Capacidade de aprender novos conceitos.
- Pesos Iniciais $\{0 \leq \omega \leq 1\}$. Conhecimento atual do assunto 😊



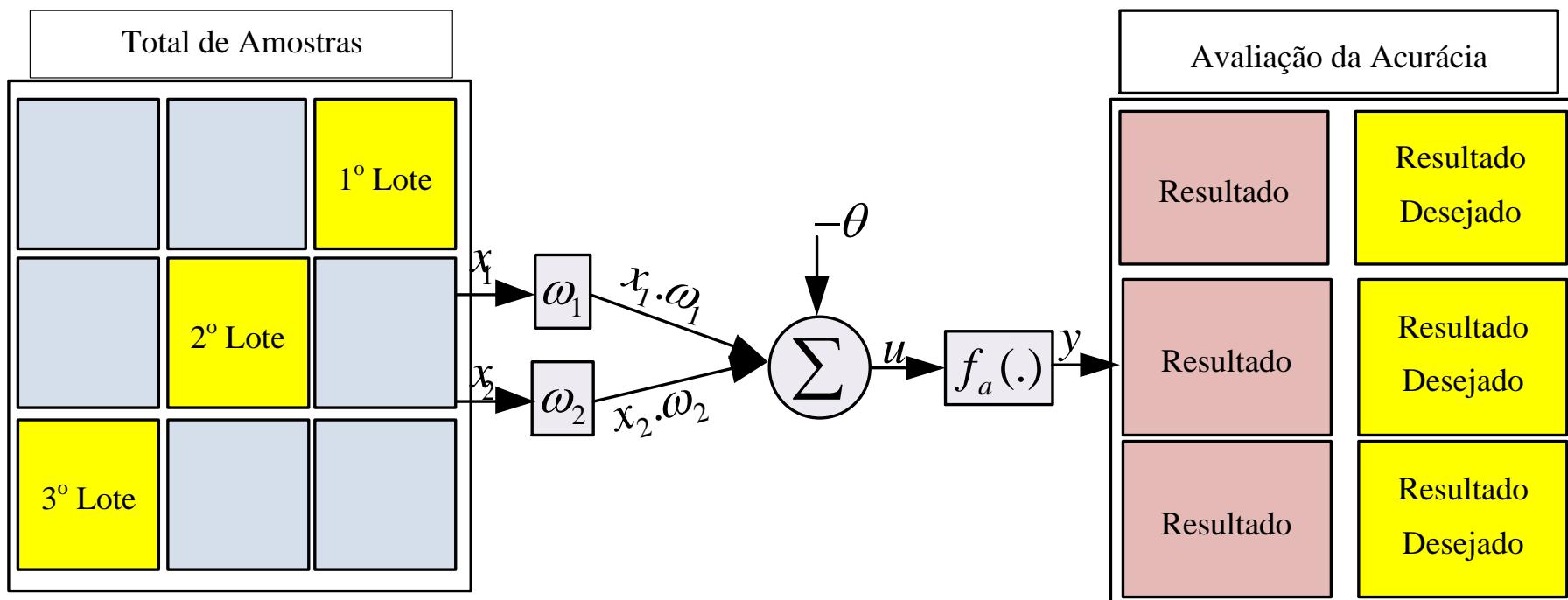
Avaliação: Visão Geral

- Avaliar Acurácia da rede em classificar.
- Separar total de amostras em 2
 - Treinamento
 - Avaliação



Avaliação

- Validação Cruzada “*k-fold cross validation*”
- O total de amostras é particionado, k amostras para Avaliação, e ($total - k$) amostras para Treinamento.
- Para aproveitar todas as amostras, o particionamento ocorre em etapas cruzadas e acumulativas de avaliação.
- Exemplo $k = 1/3$, divididas em 3 etapas de avaliação.



Lab Redes Neurais