

IA APLICADA AO MELHORAMENTO GENÉTICO

Aplicações

Vinícius Quintão Carneiro
Vinicius.carneiro@ufla.br



Aplicação 01 – Classificação de Espécies

THE USE OF MULTIPLE MEASUREMENTS IN
TAXONOMIC PROBLEMS

By R. A. FISHER, Sc.D., F.R.S.



Iris Versicolor



Iris Setosa



Iris Virginica

Aplicação 01 – Classificação de Espécies

<https://archive.ics.uci.edu/ml/datasets/iris>



Machine Learning Repository
Center for Machine Learning and Intelligent Systems

Iris Data Set

Download: [Data Folder](#), [Data Set Description](#)

Abstract: Famous database; from Fisher, 1936



Data Set Characteristics:	Multivariate	Number of Instances:	150	Area:	Life
Attribute Characteristics:	Real	Number of Attributes:	4	Date Donated	1988-07-01
Associated Tasks:	Classification	Missing Values?	No	Number of Web Hits:	2668249

Aplicação 01 – Classificação de Espécies

comprimento_sepala	largura_sepala	comprimento_petala	largura_petala	Espécie
5.1	3.5	1.4	0.2	Iris-setosa
4.9	3	1.4	0.2	Iris-setosa
4.7	3.2	1.3	0.2	Iris-setosa
4.6	3.1	1.5	0.2	Iris-setosa
5	3.6	1.4	0.2	Iris-setosa
5.4	3.9	1.7	0.4	Iris-setosa
4.6	3.4	1.4	0.3	Iris-setosa
5	3.4	1.5	0.2	Iris-setosa
4.4	2.9	1.4	0.2	Iris-setosa
4.9	3.1	1.5	0.1	Iris-setosa
5.4	3.7	1.5	0.2	Iris-setosa
4.8	3.4	1.6	0.2	Iris-setosa
4.8	3	1.4	0.1	Iris-setosa

Aplicação 02 – Classificação de Folhas

<https://archive.ics.uci.edu/ml/datasets/leaf>

A faint, abstract background consisting of a network of light blue lines connecting numerous small yellow circular nodes, resembling a complex web or a neural network.

UCI 

Machine Learning Repository
Center for Machine Learning and Intelligent Systems

Leaf Data Set

Download: [Data Folder](#), [Data Set Description](#)

Abstract: This dataset consists in a collection of shape and texture features extracted from digital images of leaf specimens originating from a total of 40 different plant species.

Data Set Characteristics:	Multivariate	Number of Instances:	340	Area:	Computer
Attribute Characteristics:	Real	Number of Attributes:	16	Date Donated	2014-02-24
Associated Tasks:	Classification	Missing Values?	N/A	Number of Web Hits:	100939

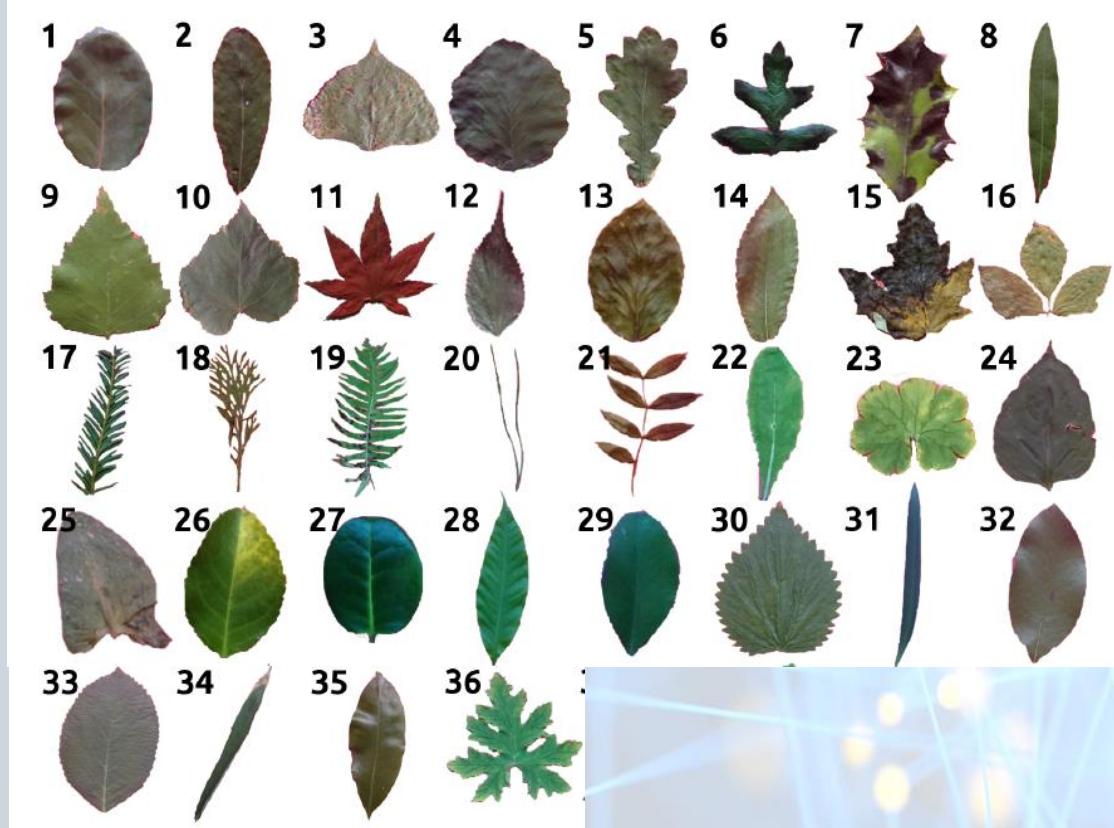
Aplicação 02 – Classificação de Folhas

<https://archive.ics.uci.edu/ml/datasets/leaf>

- | | |
|-------------------------|------------------------------|
| 1. Class (Species) | 9. Maximal Indentation Depth |
| 2. Specimen Number | 10. Lobedness |
| 3. Eccentricity | 11. Average Intensity |
| 4. Aspect Ratio | 12. Average Contrast |
| 5. Elongation | 13. Smoothness |
| 6. Solidity | 14. Third moment |
| 7. Stochastic Convexity | 15. Uniformity |
| 8. Isoperimetric Factor | 16. Entropy |

Aplicação 02 – Classificação de Folhas

<https://archive.ics.uci.edu/ml/datasets/leaf>



Aplicação 03 - Predição de preços de apartamentos



Zobacz mieszkania w najnowszych inwestycjach



ul. 28 lipca 1943 r.
Apartamenty Wola III etap



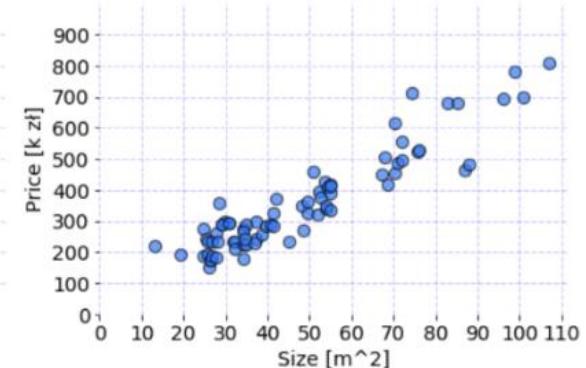
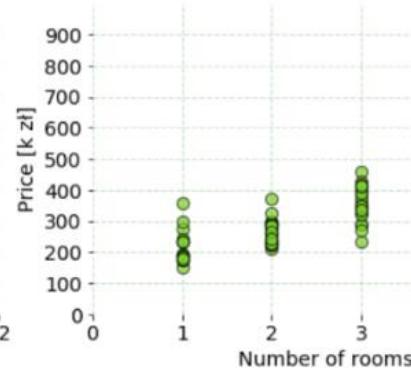
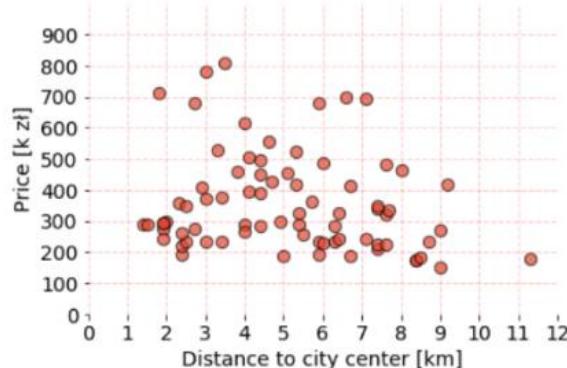
Dominikanów
Park Zaczarowanej Dorożki



ul. Gdańska 12
GDAŃSKA 12



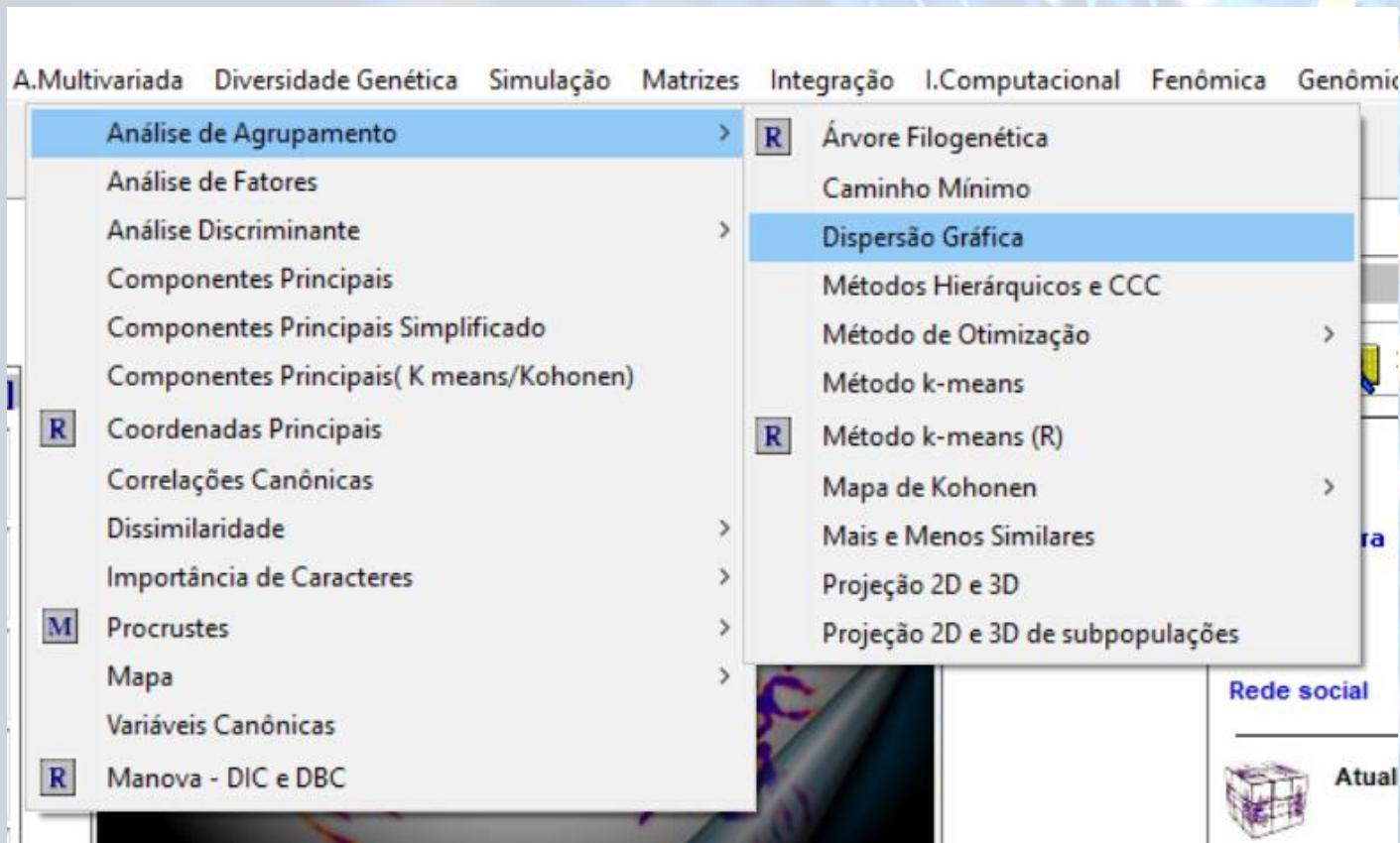
ul. Ułańów 64D
**ul. Lublańska / Ja-
bud. 1**
Zaczarowany Młyn

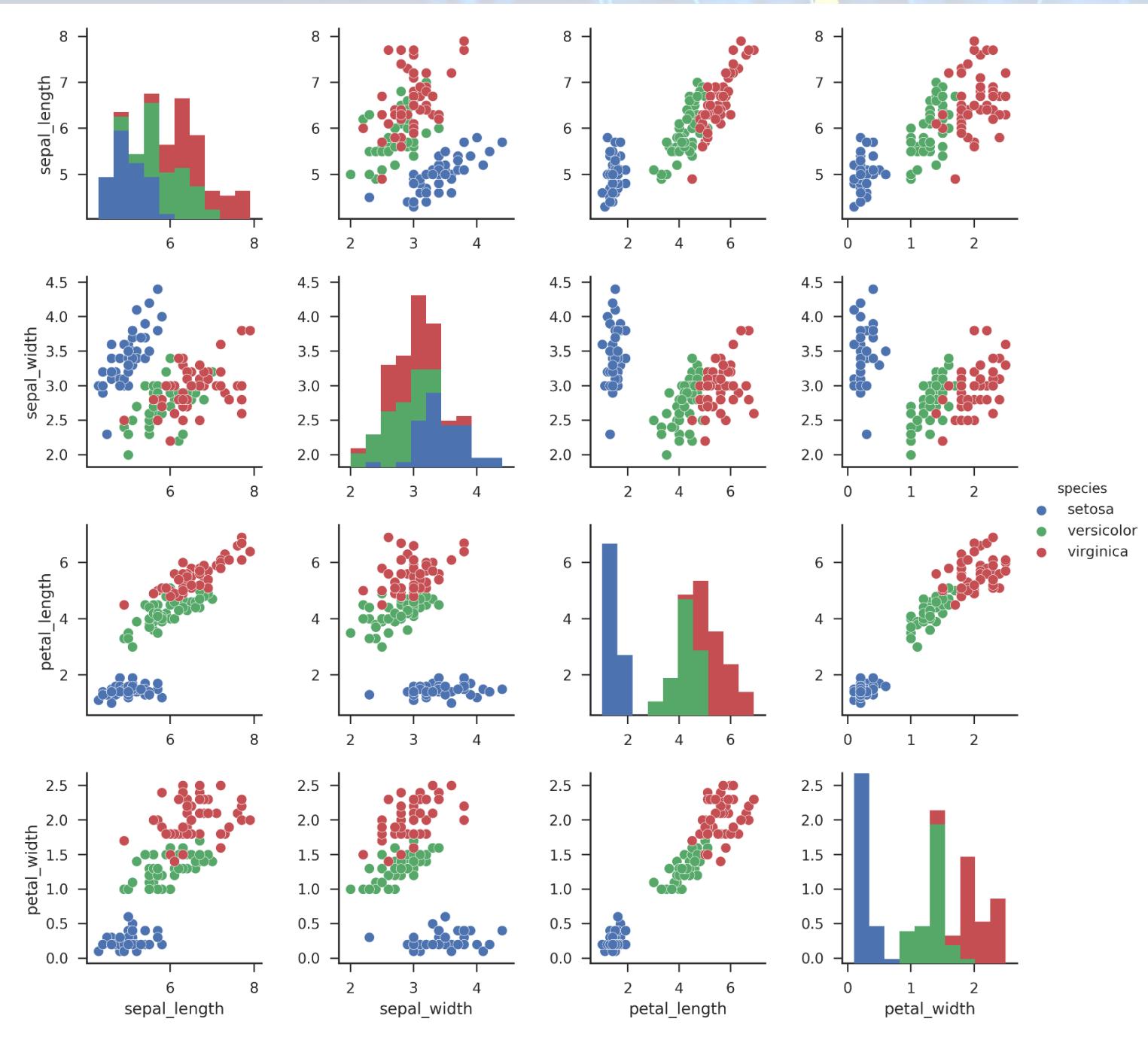


GENES

Dispersão Gráfica

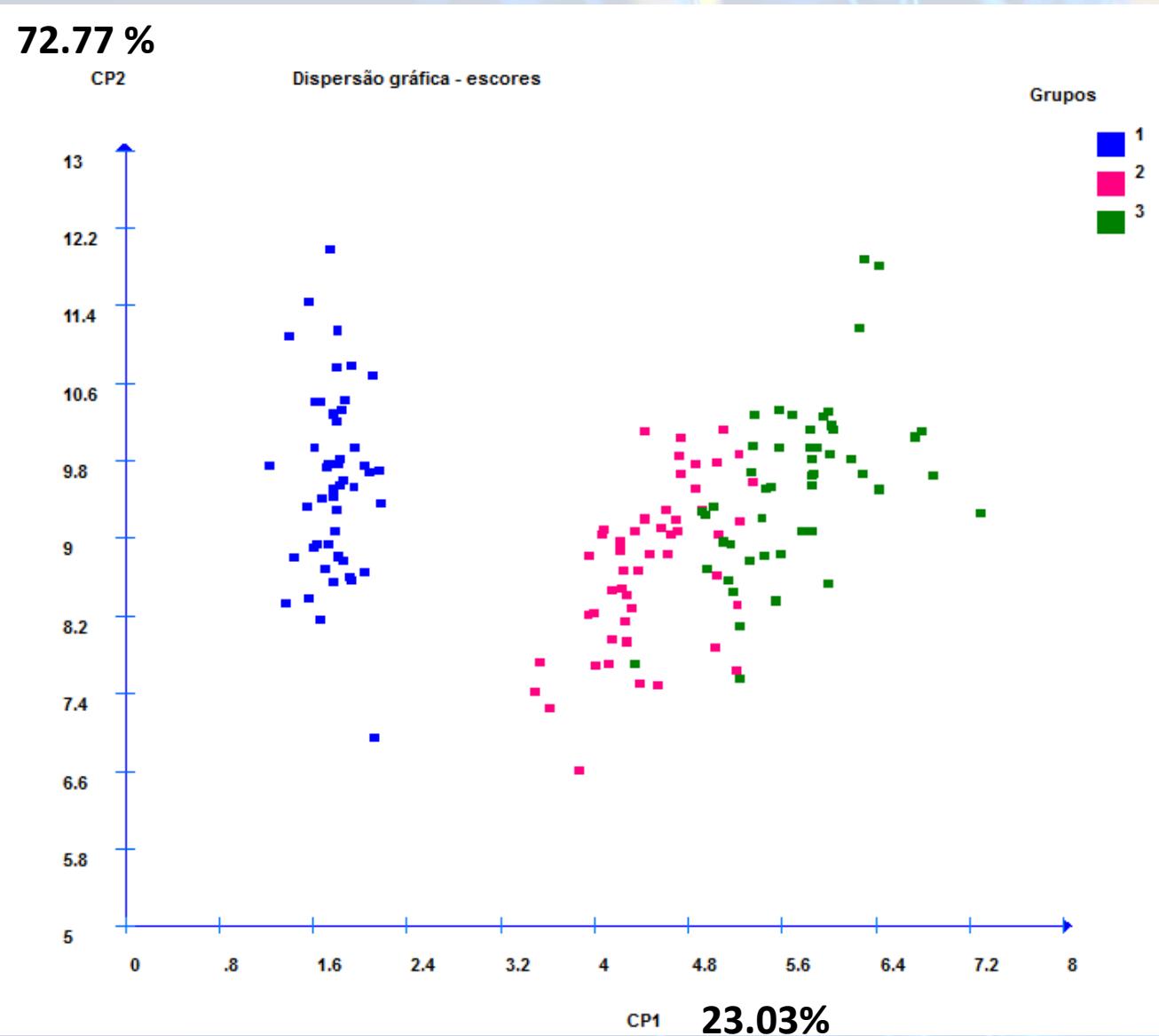
A. Multivariada -> Análise de Agrupamento -> Dispersão Gráfica



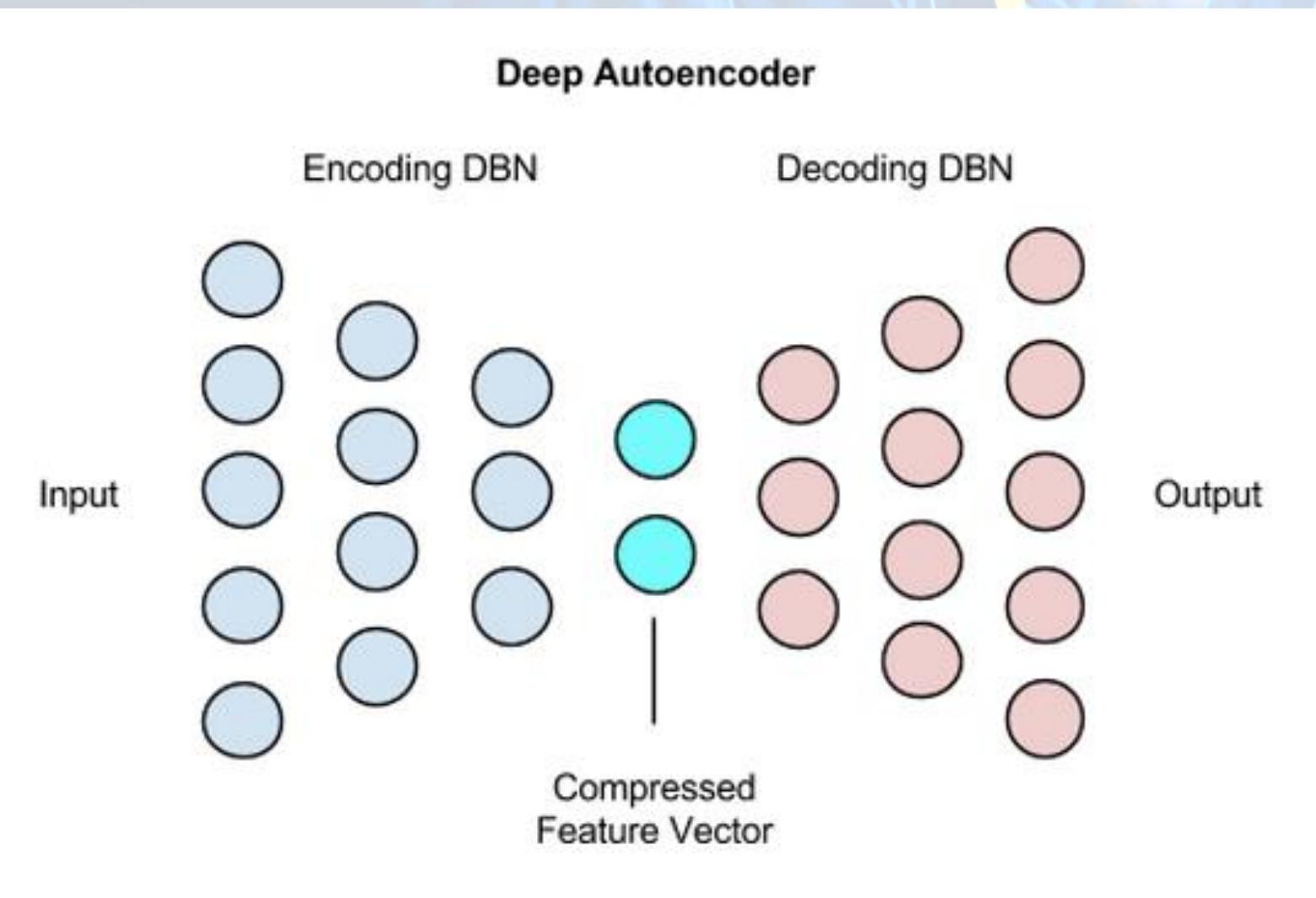


Redução de Dimensionalidade

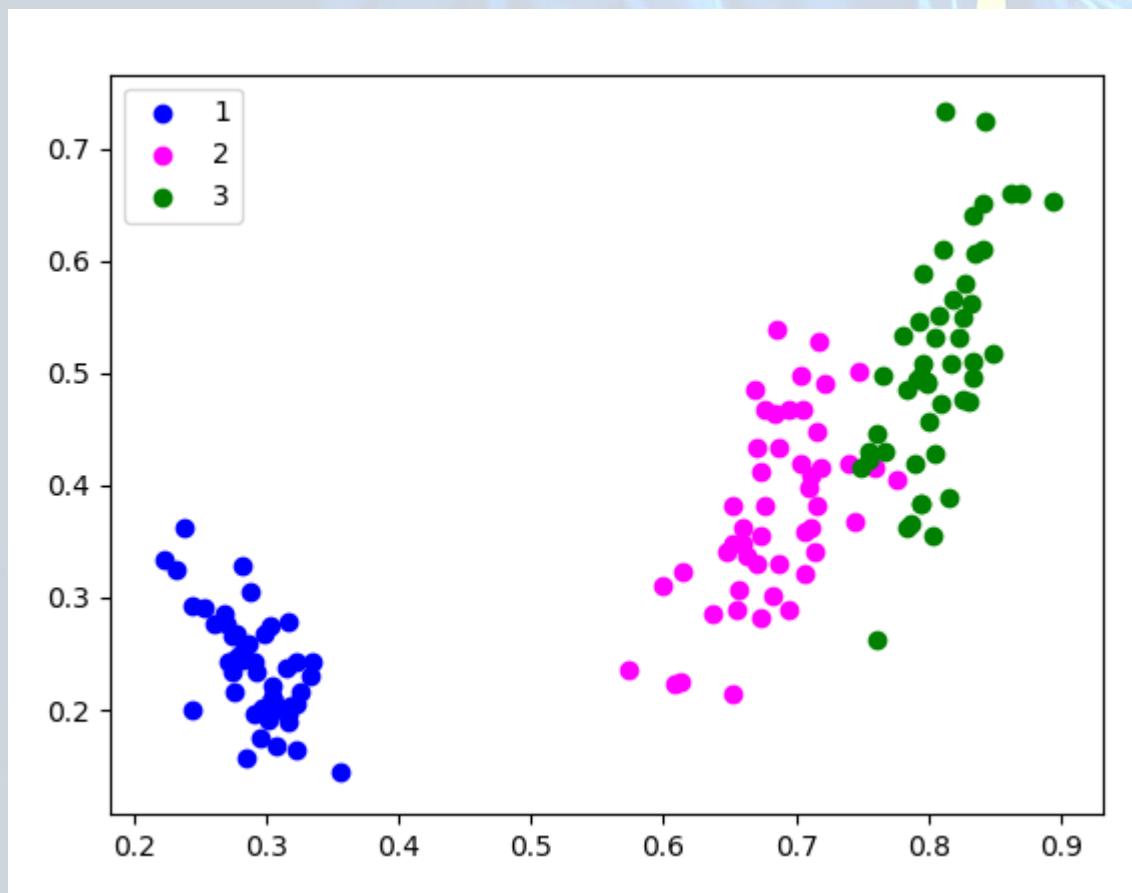
Componentes Principais



Autoencoder



Autoencoder



GENES

Redução de Dimensionalidade

1. A.Multivariada -> Componentes Principais

2. I.Computacional -> Autoencoder

A.Multivariada Diversidade Genética Simulação Matrizes In

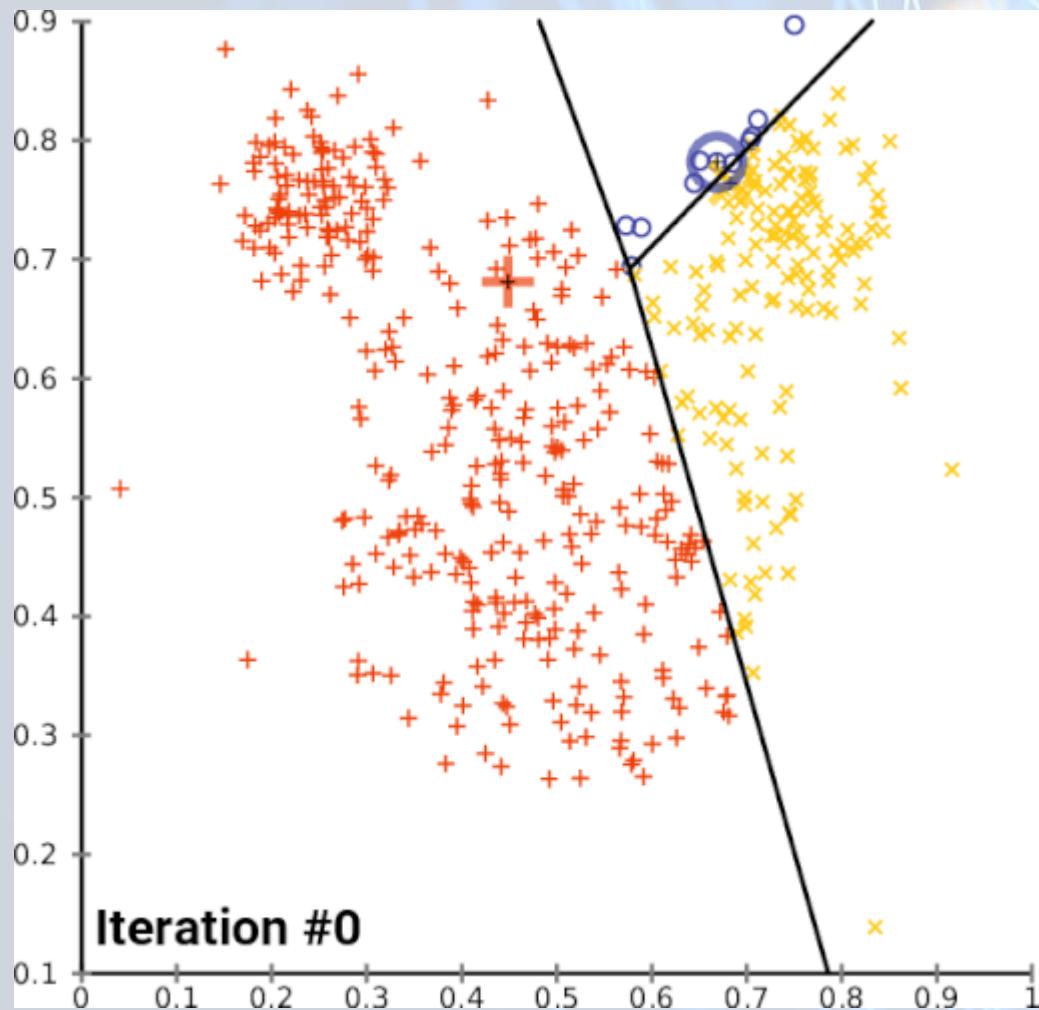
- Análise de Agrupamento >
- Análise de Fatores
- Análise Discriminante >
- Componentes Principais**
- Componentes Principais Simplificado
- Componentes Principais(K means/Kohonen)
- Coordenadas Principais
- Correlações Canônicas
- Dissimilaridade >
- Importância de Caracteres >
- M Procrustes
- Mapa >
- Variáveis Canônicas
- R Manova - DIC e DBC

I.Computacional Fenômica Genômica Meta Análise

- Comparação de conjunto de dados
- Replicação de ampliação de dados
- Partição de arquivo de dados
- M Análise Discriminante >
- R Análise Discriminante >
- M Ajuste de Modelos >
- R Ajuste de Modelos >
- R Árvore Aleatória >
- M GWS >
- R GWS >
- M PMC - Anova >
- R Autoencoder**
- M Análise Temporal >
- M Estabilidade e Adaptabilidade - Eberhart e Russell >
- M Fuzzy - Eberhart e Russell >
- M Fuzzy - Eberhart e Russell / Lin e Binns >
- M Mapa de Kohonen >
- R Mapa de Kohonen >
- Modelos Clássicos >

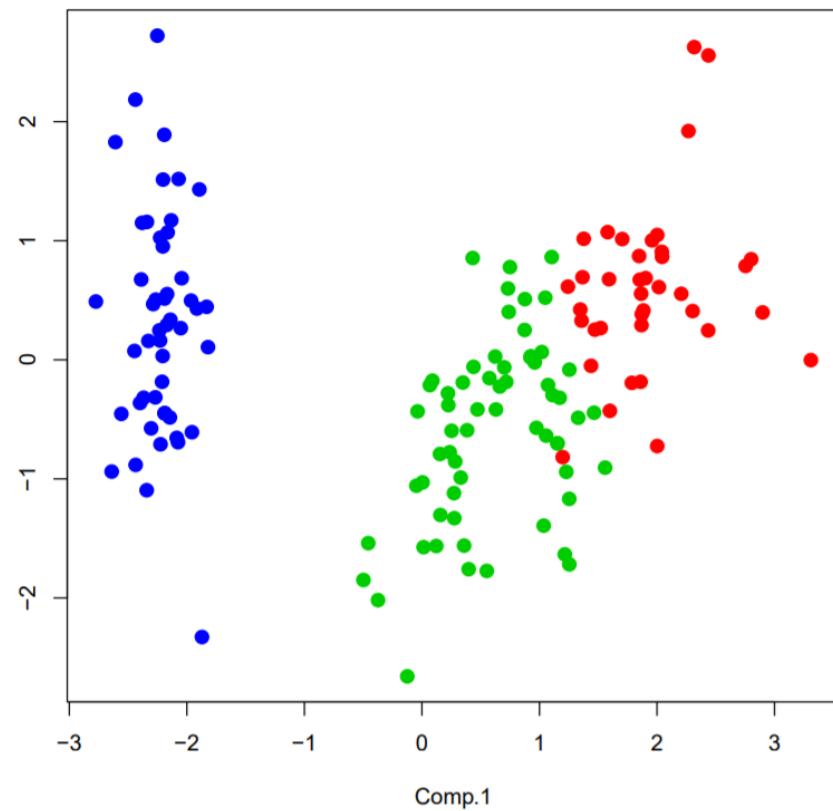
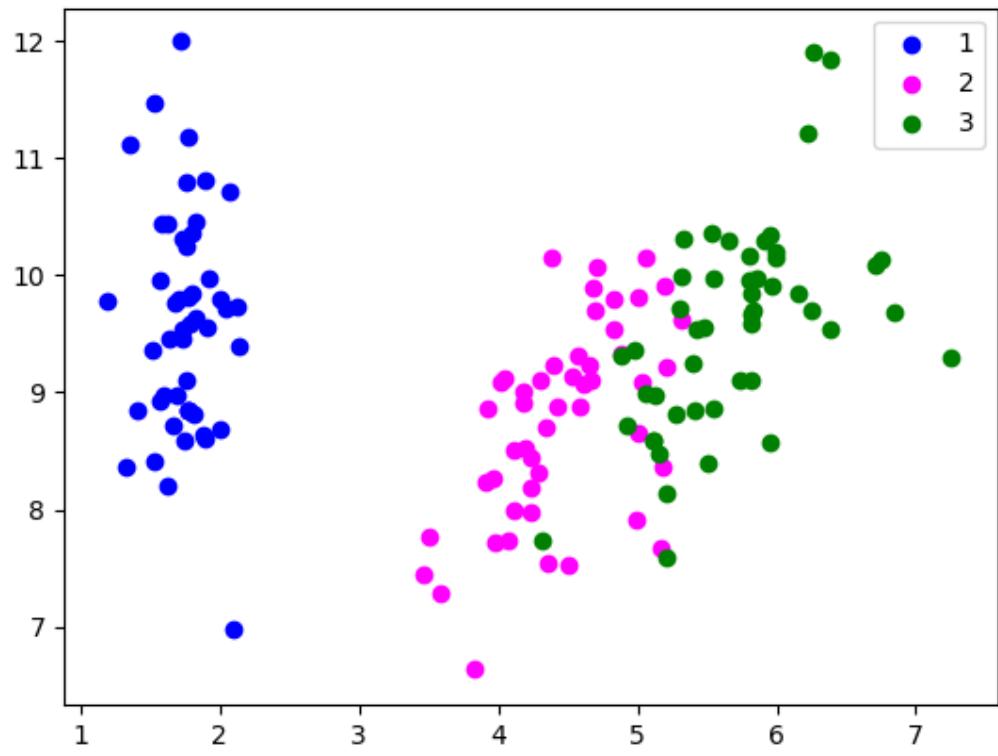
Agrupamento

K Means

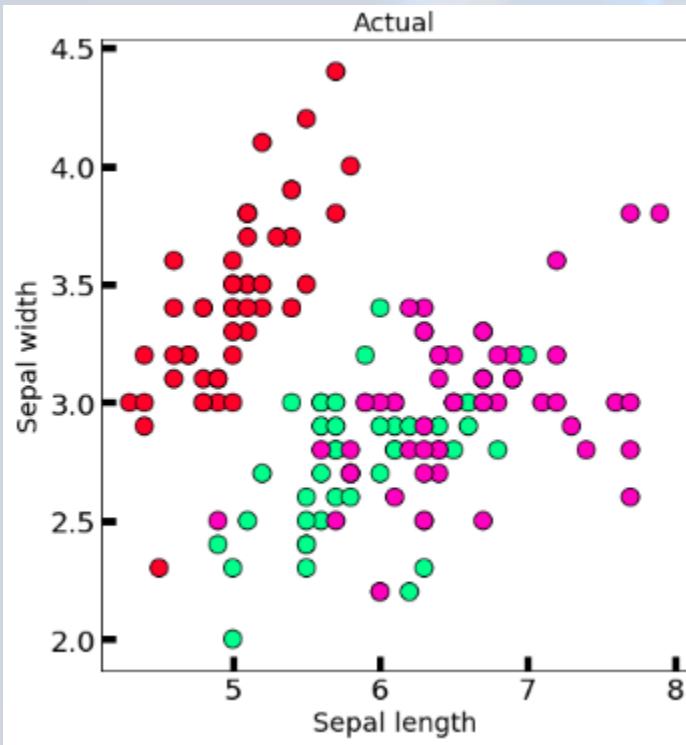


K Means

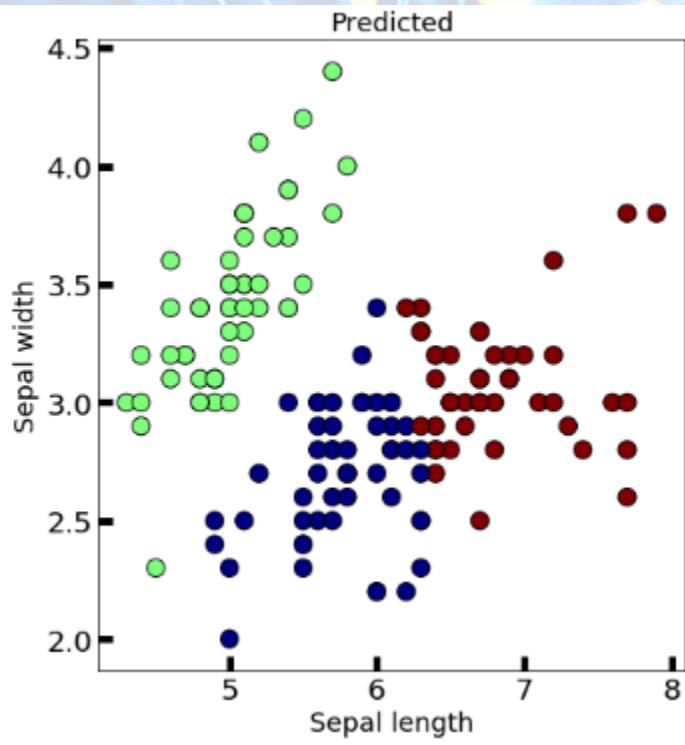
K-Means com k grupos



Original



K-means



K-means

Industrial Crops & Products 108 (2017) 806–813

Contents lists available at ScienceDirect

Industrial Crops & Products

journal homepage: www.elsevier.com/locate/indcrop

ELSEVIER

Research Paper

High-performance prediction of macauba fruit biomass for agricultural and industrial purposes using Artificial Neural Networks

Carla Aparecida de O. Castro^a, Rafael T. Resende^{a,*}, Kacilda N. Kuki^c, Vinícius Q. Carneiro^b, Gustavo E. Marcatti^d, Cosme Damião Cruz^b, Sérgio Y. Motoike^c

CrossMark

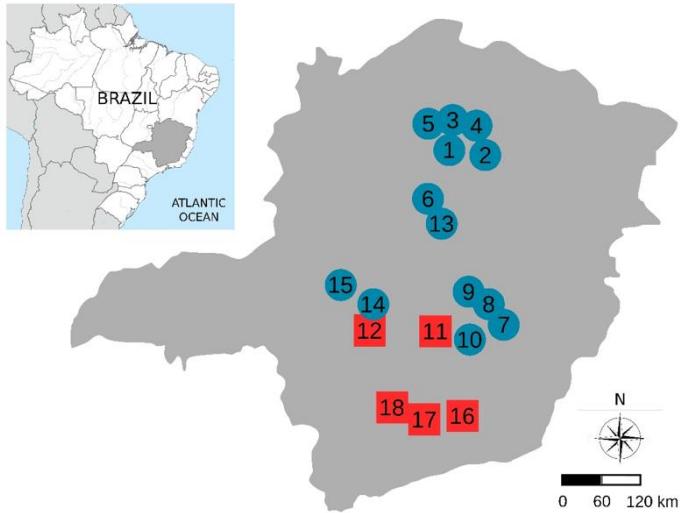


Figura 3. Distribuição das localidades ao longo do território do estado de Minas Gerais (Brasil). As localidades foram agrupadas de acordo com o método k-means e estão relacionadas às coordenadas apresentadas na tabela 1, sendo que os números dentro dos círculos azuis correspondem às localidades do grupo 1 e os números dentro dos quadrados vermelhos são as localidades do grupo 2.

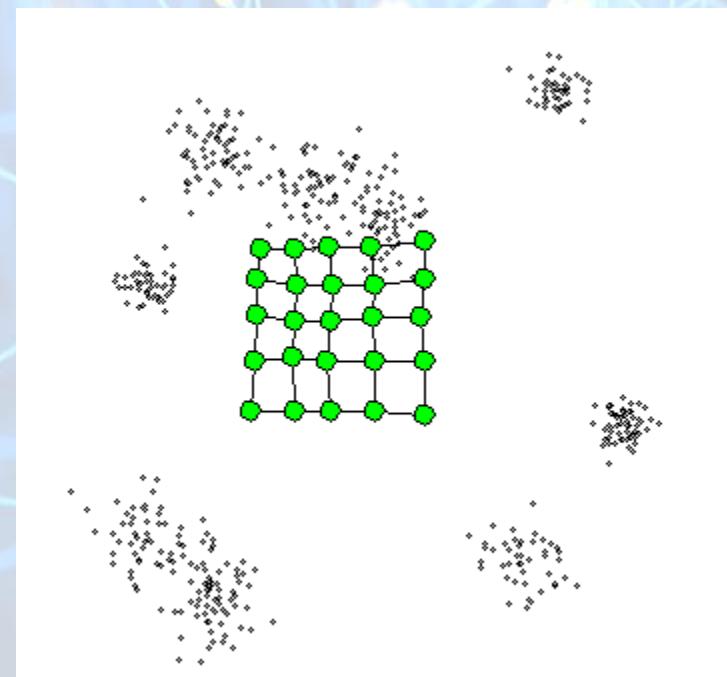
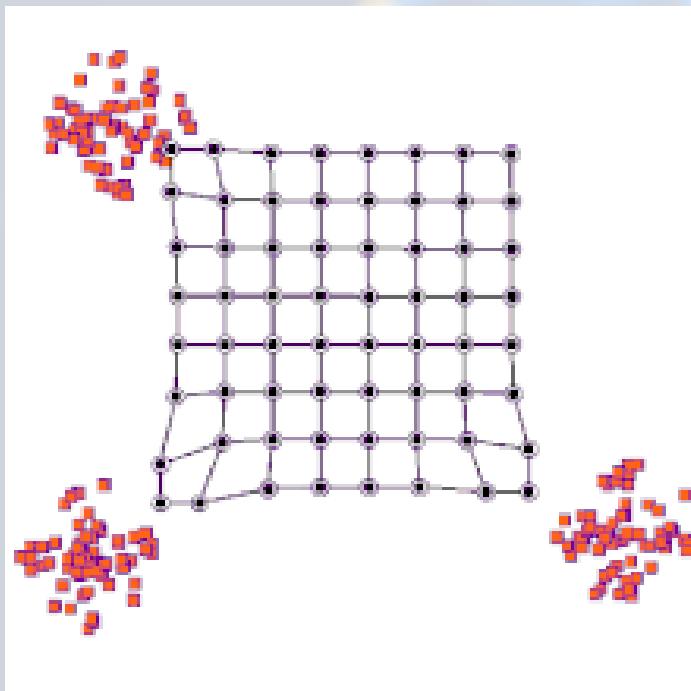
Tabela 4. K-means utilizando as variáveis preditoras. É apresentado informações sobre os grupos formados (G01 e G02) e as médias das variáveis analisadas.

GRUPOS	G01	G02	Total
Médias das variáveis	PFFI	41,62	50,80 43,07
	DRF	44,59	47,19 45,10
	DAF	43,58	46,56 44,08
	EE1	5,74	5,94 5,79
	EE2	4,07	4,32 4,11
Procedências	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 13, 14, 15	11, 12, 16, 17, 18	1 – 18
Número de matrizes	135	37	172
Número de frutos	447	96	543
Médias climáticas	Temperatura (°C)	22,02	20,70 21,65
	Precipitação (mm ³)	1247,31	1459,00 1306,11

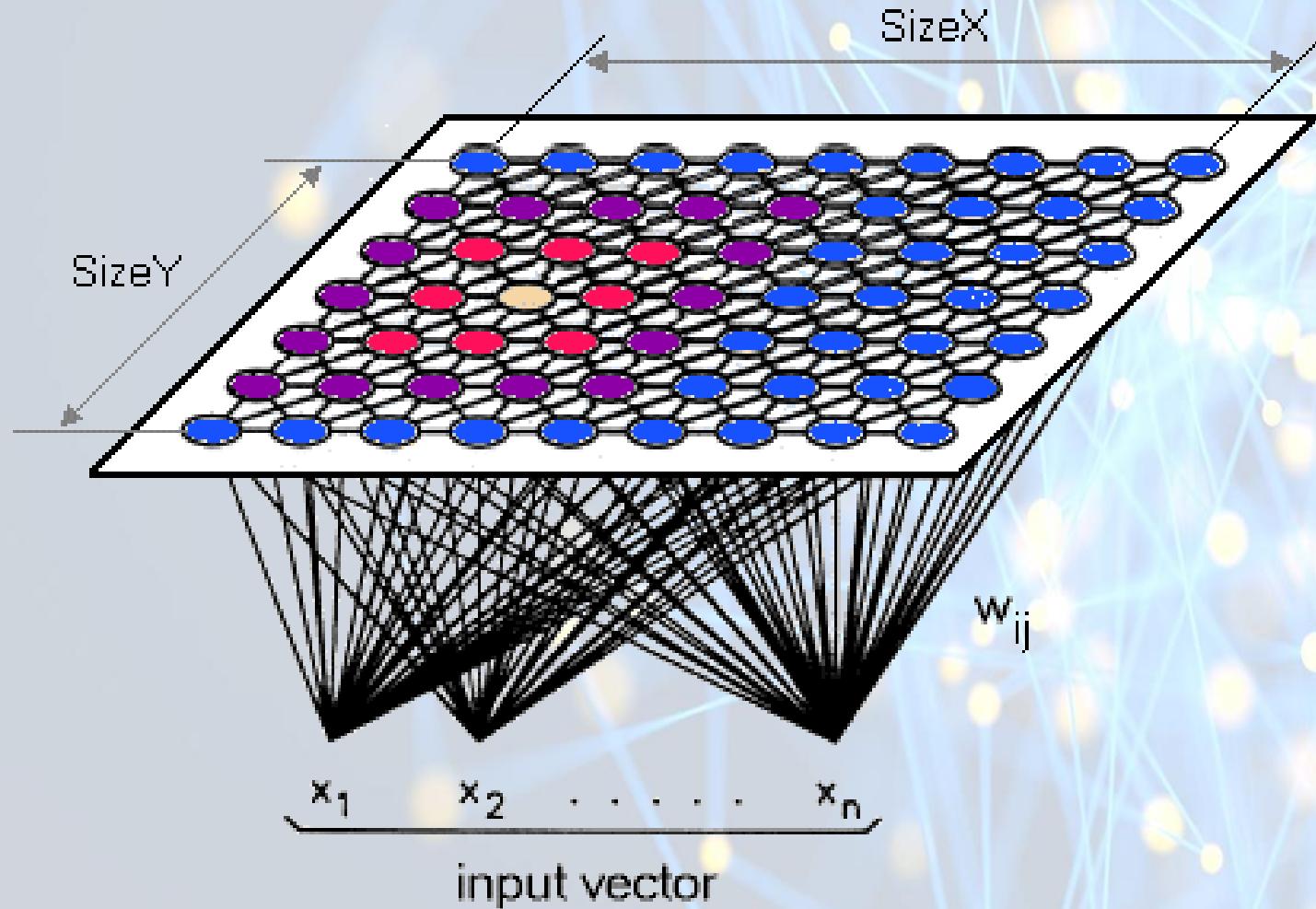
Mapas Auto Organizáveis – Kohonen

Aprendizado Não Supervisionado

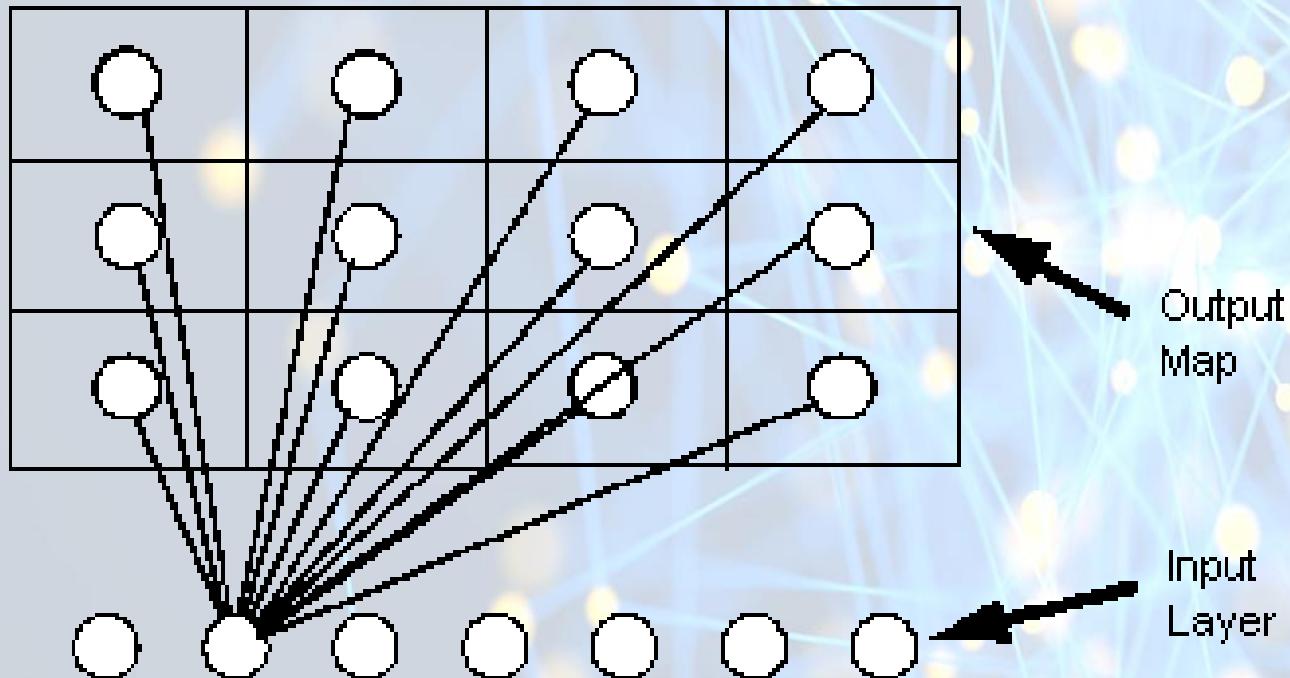
Mapas Auto-Organizáveis



Mapas Auto-Organizáveis



Mapas Auto-Organizáveis



All input neurons are connected to all neurons
in the output map. (Not shown for clarity)

Mapas Auto-Organizáveis

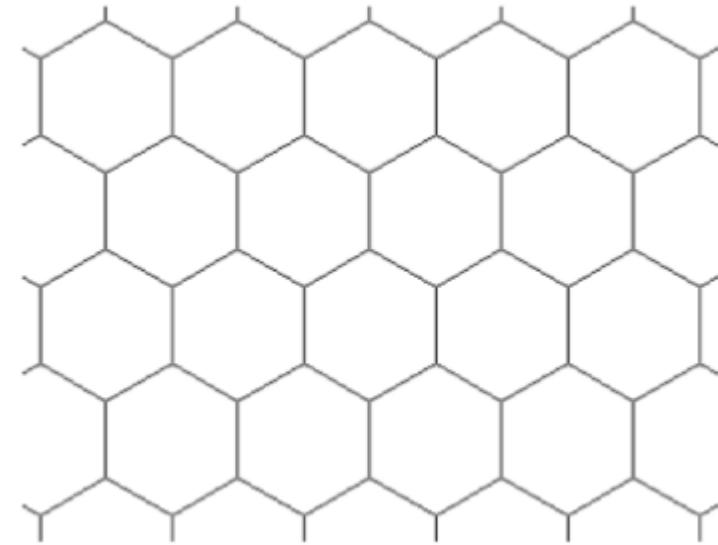
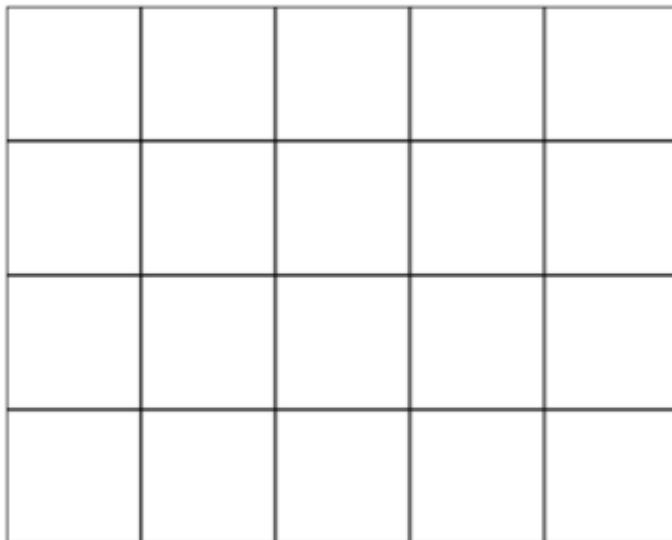
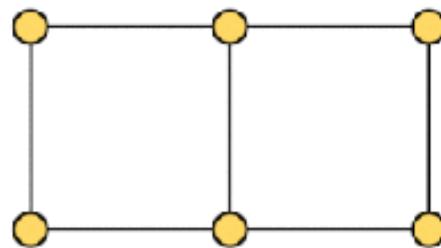


Figure 1. SOM grids can either be rectangular or hexagonal. Each square or hexagon is a neuron.

Mapas Auto-Organizáveis



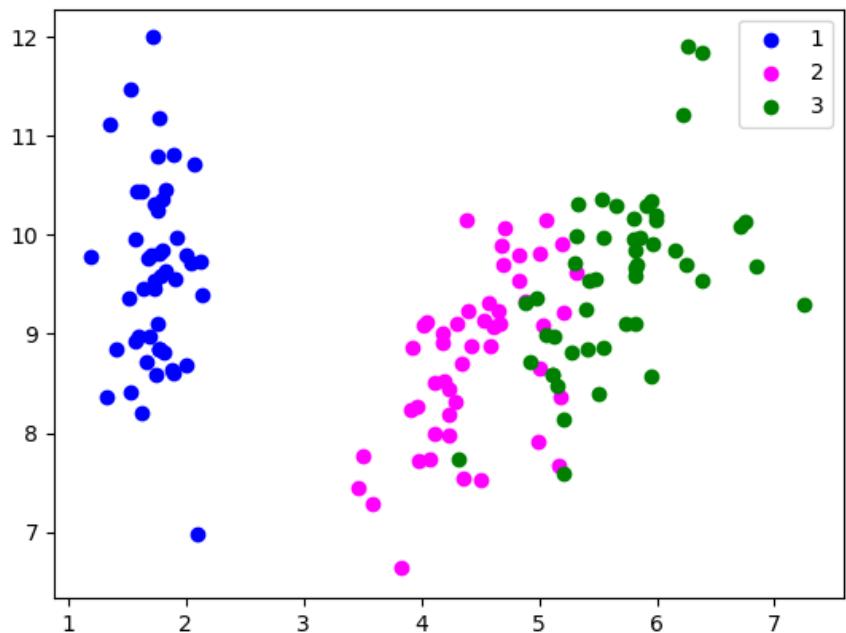
Step 0: Position neurons (orange) in data space.

Mapas Auto-Organizáveis

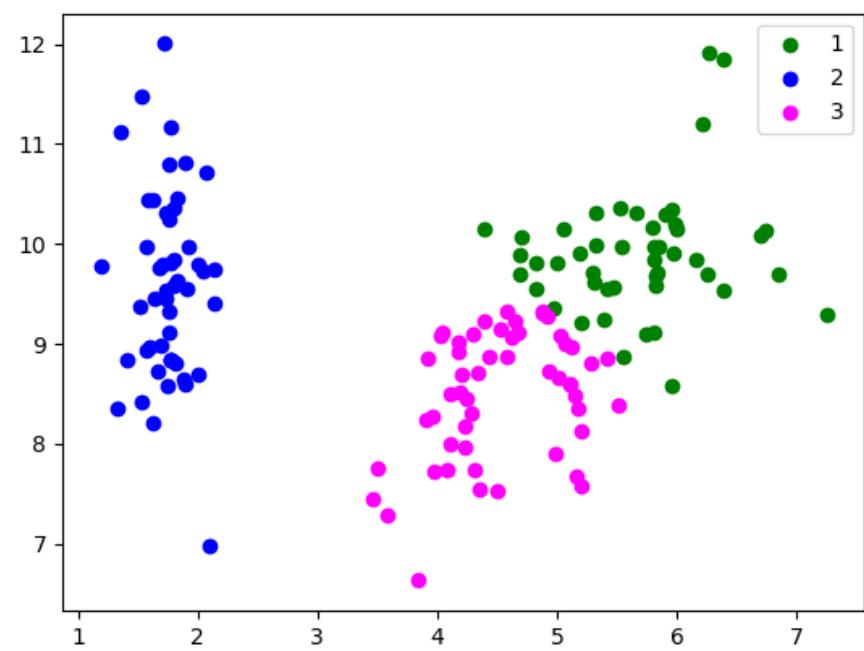


PCA

Original

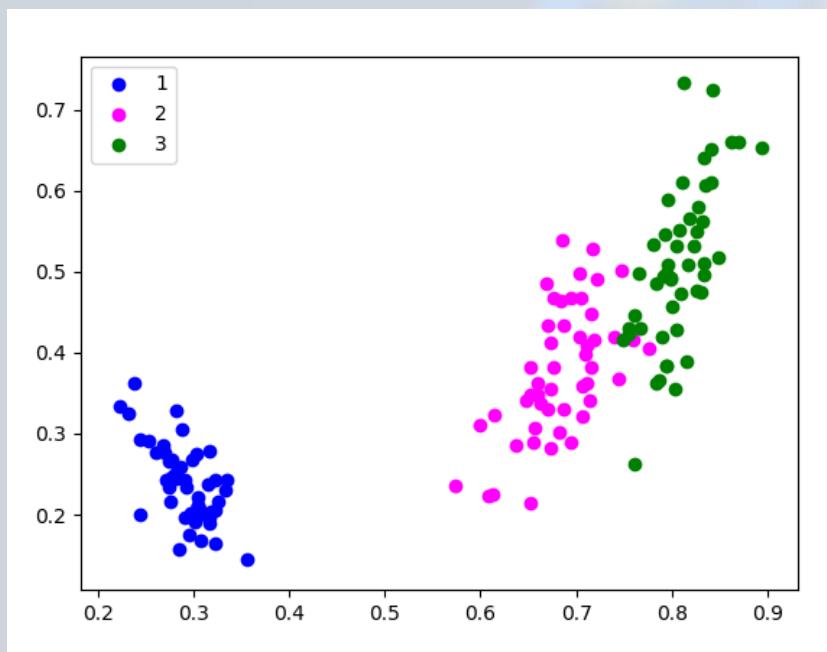


SOM

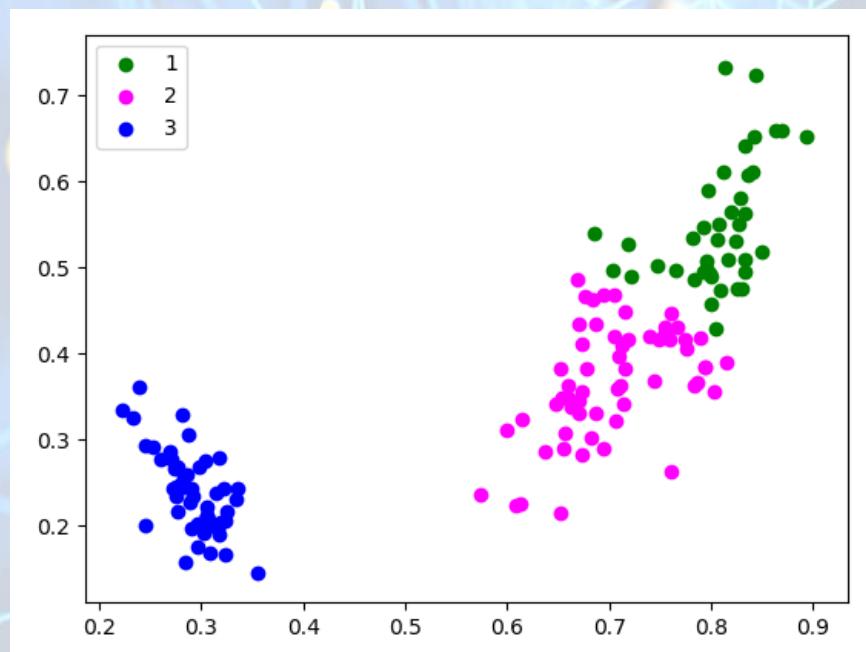


Autoencoder

Original



SOM



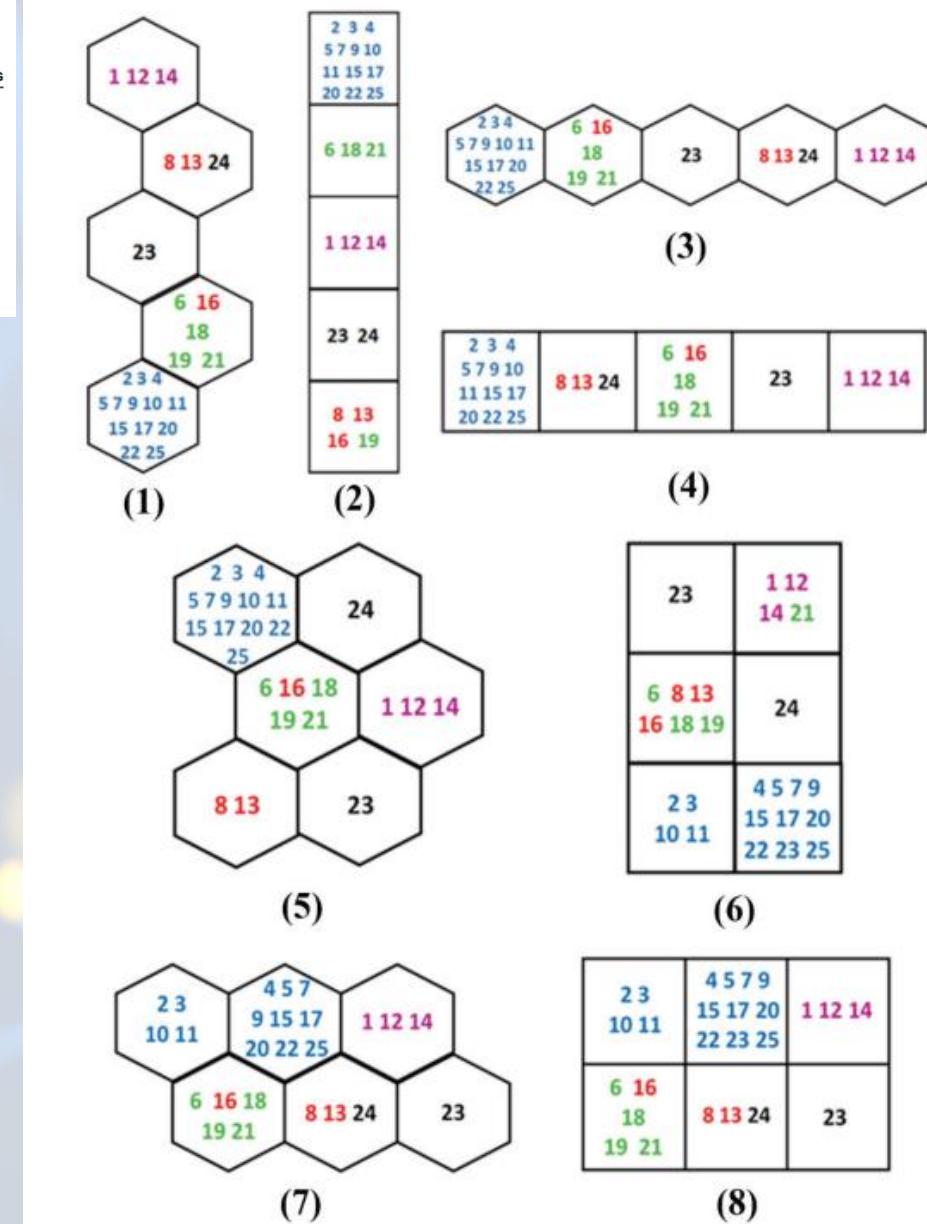
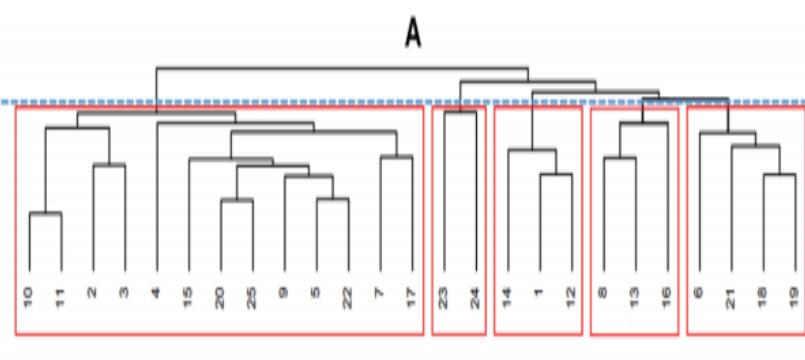


Self-organizing maps in the study of genetic diversity among irrigated rice genotypes

Iara Gonçalves dos Santos^{1*}, Vinícius Quintão Carneiro¹, Antônio Carlos da Silva Junior¹, Cosme Damião Cruz¹ and Plínio César Soares²

¹Laboratório de Bioinformática, Universidade Federal de Viçosa, Av. PH Rolfs, s/n, 36570-000, Campus Universitário, Viçosa, Minas Gerais, Brazil.

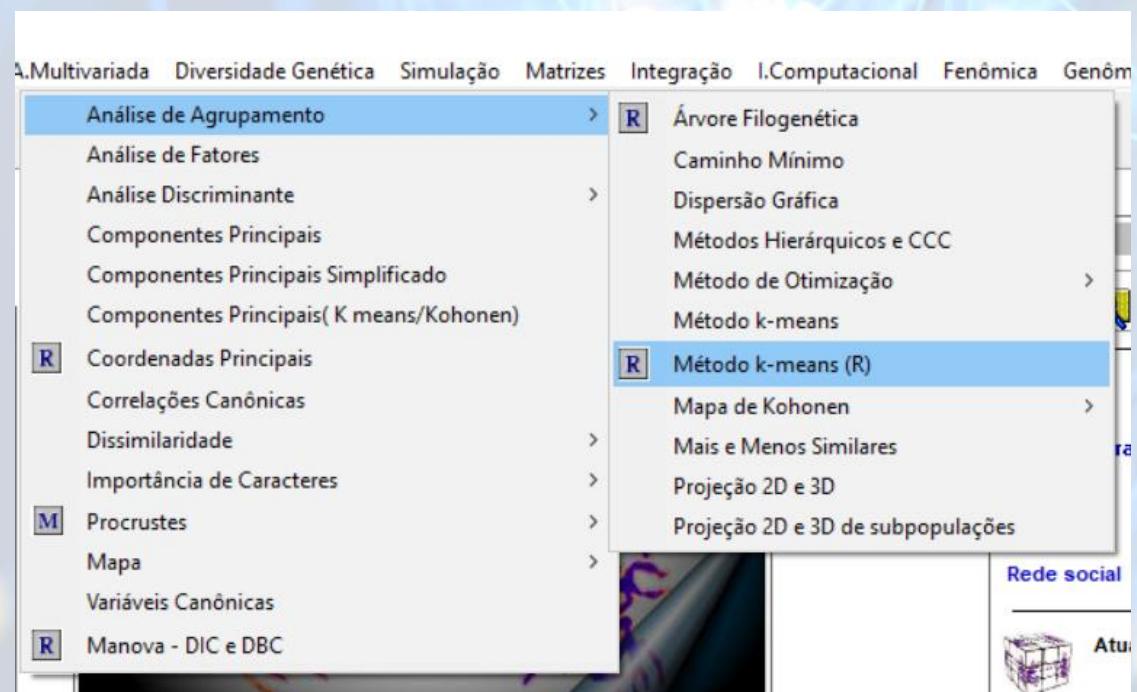
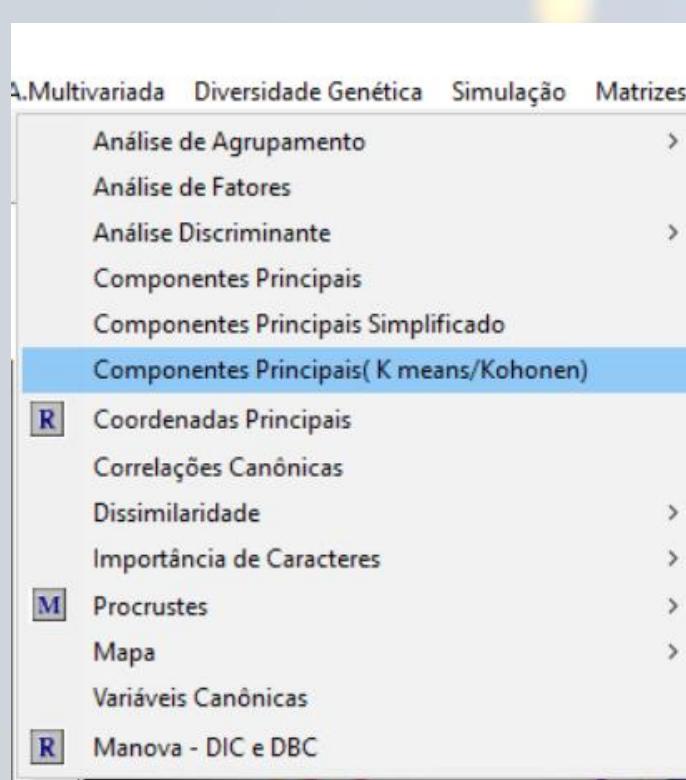
²Empresa de Pesquisa Agropecuária de Minas Gerais, Viçosa, Minas Gerais, Brazil. *Author for correspondence. E-mail: iara.santos@ufv.br



GENES

Agrupamento

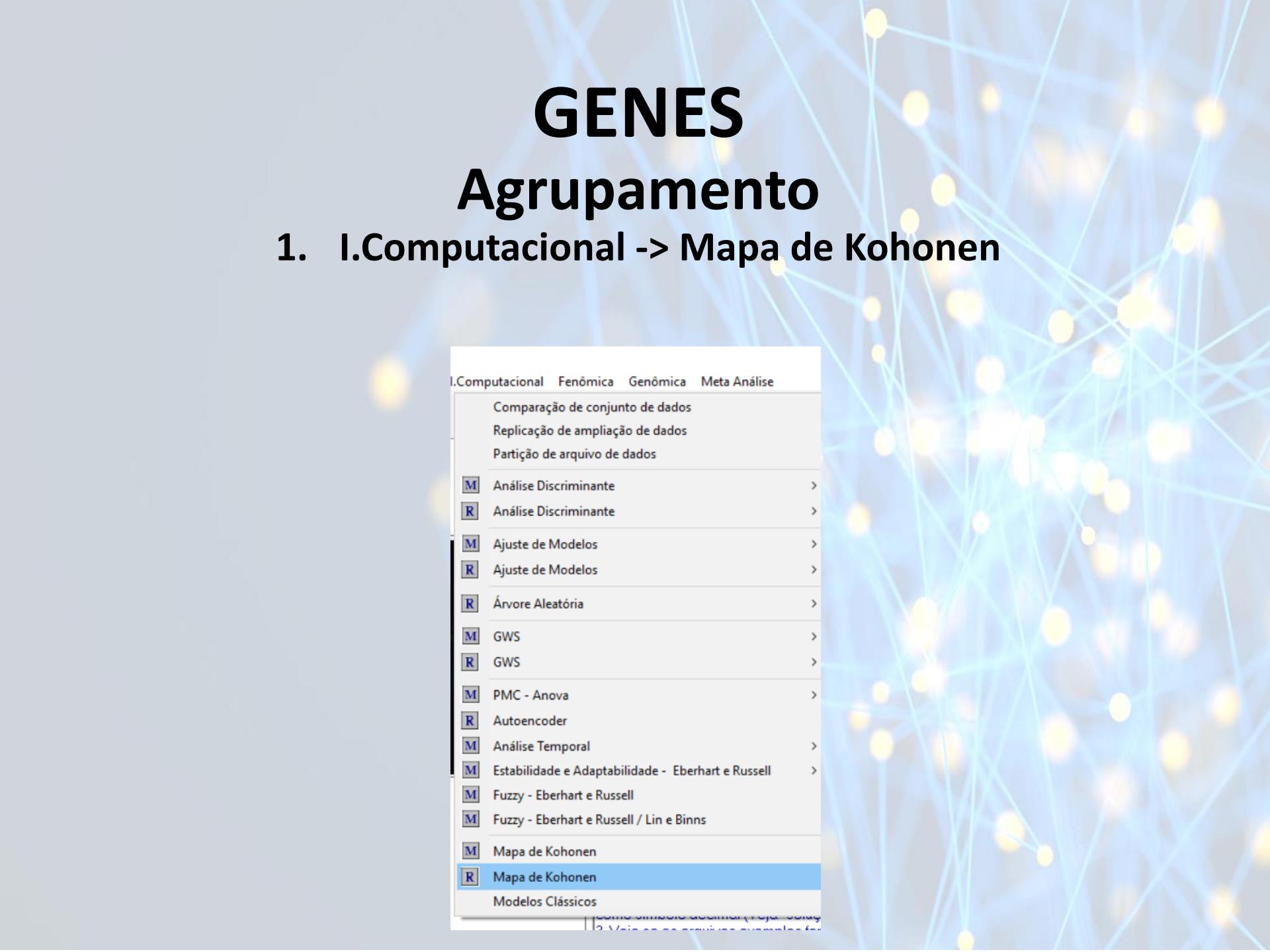
1. A.Multivariada -> Componentes Principais (K means/Kohonen)
2. A.Multivariada -> Análise de Agrupamento -> Método k-means



GENES

Agrupamento

1. I.Computacional -> Mapa de Kohonen



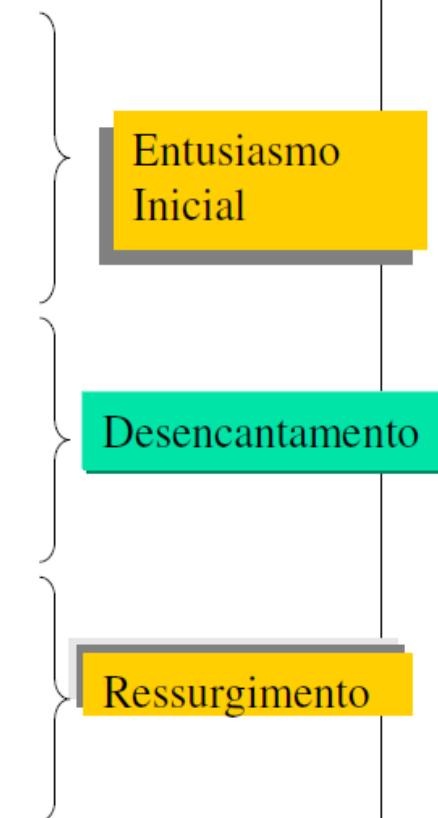
I.Computacional Fenômica Genômica Meta Análise

- Comparação de conjunto de dados
- Replicação de ampliação de dados
- Partição de arquivo de dados
- M** Análise Discriminante >
- R** Análise Discriminante >
- M** Ajuste de Modelos >
- R** Ajuste de Modelos >
- R** Árvore Aleatória >
- M** GWS >
- R** GWS >
- M** PMC - Anova >
- R** Autoencoder >
- M** Análise Temporal >
- M** Estabilidade e Adaptabilidade - Eberhart e Russell >
- M** Fuzzy - Eberhart e Russell >
- M** Fuzzy - Eberhart e Russell / Lin e Binns >
- M** Mapa de Kohonen >
- R** Mapa de Kohonen >
- Modelos Clássicos

Redes Neurais Artificiais

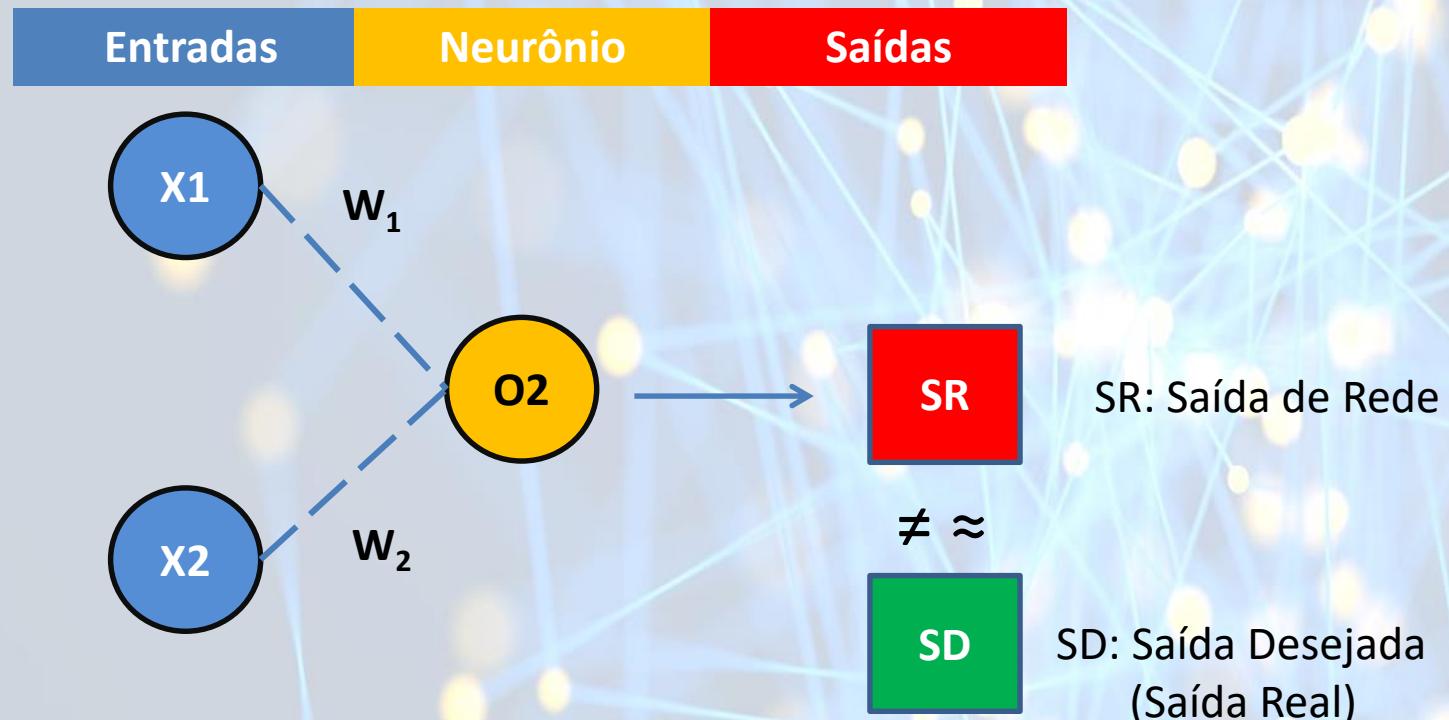
Histórico

- 1943 McCulloch-Pitts Neurônio Booleano
- 1957 Rosenblatt Perceptron
- 1960 Widrow-Hoff ADALINE
- 1969 Minsky-Papert *Perceptrons*
- 1986 Rumelhart, Hinton & Williams (MIT)
Backpropagation p/ Perceptron Multicamadas (MLP)



Redes Neurais Artificiais

Modelo de McCulloch e Pitts (1943)

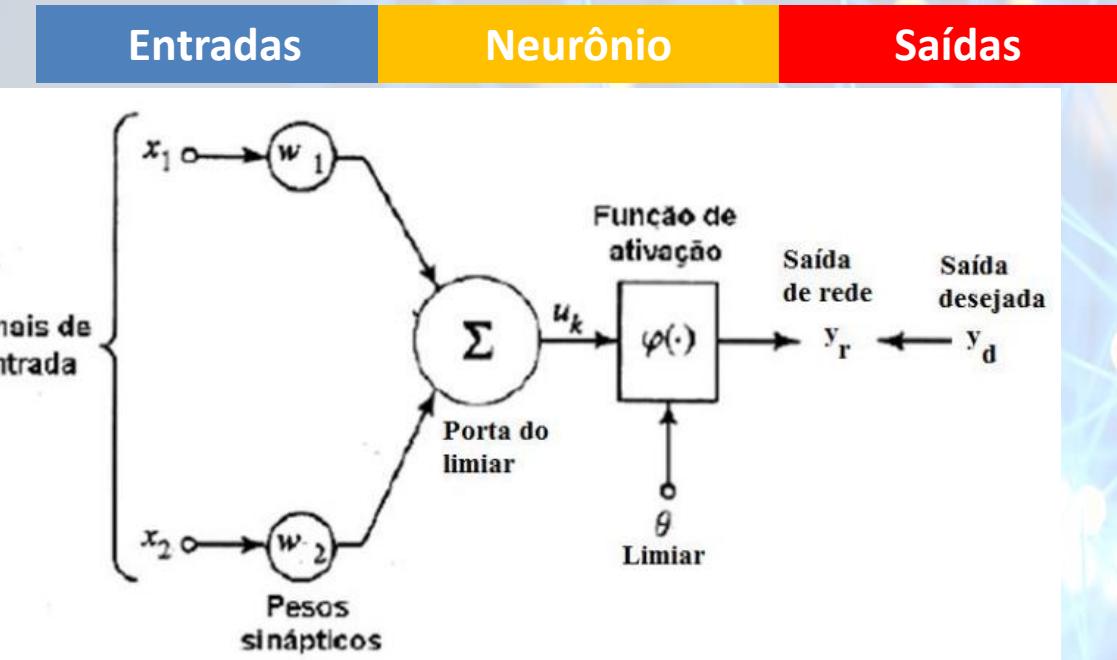


Sinais de Entrada: Medidas advindas do meio externo (variáveis de um problema);

Neurônios: Unidade básica de processamento;

Saídas: Respostas emitidas pelos neurônios.

Modelo de McCulloch e Pitts (1943)



Sinais de Entrada (X_i): Medidas advindas do meio externo (variáveis de um problema)

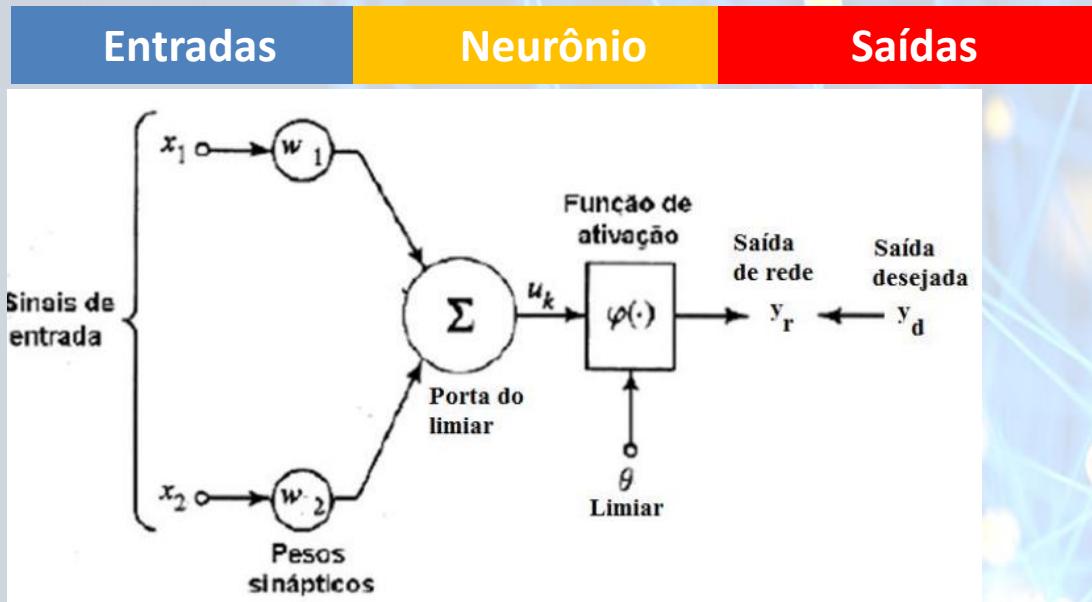
$$X_i = [X_1, X_2]$$

Pesos (W_i): São os valores que irão ponderar cada uma das variáveis de entrada;

$$W_i = [W_1, W_2]$$

Porta do limiar (L): função para agregar todos os sinais de entrada que foram ponderados;

$$L = \sum_{i=1}^p w_i x_i = W_1 X_1 + W_2 X_2$$



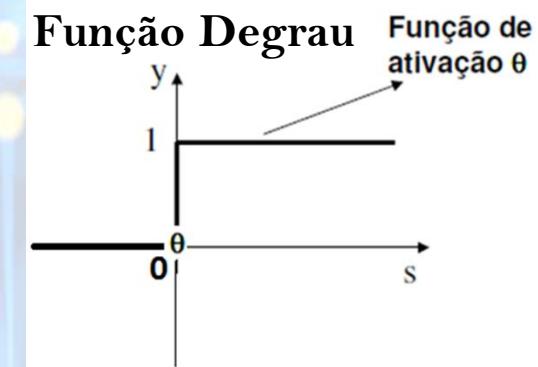
Porta do Limiar

$$L = \sum_{i=1}^p w_i x_i = w_1 x_1 + w_2 x_2$$

Limiar de ativação (θ): Variável (valor) que regula a resposta do neurônio (“Patamar”), segundo uma **função de ativação**.

Função de Ativação ($f(L)$)

- $Y = f(L)$
- 1, se $L \geq \theta$ → Limiar
 - 0, se $L < \theta$



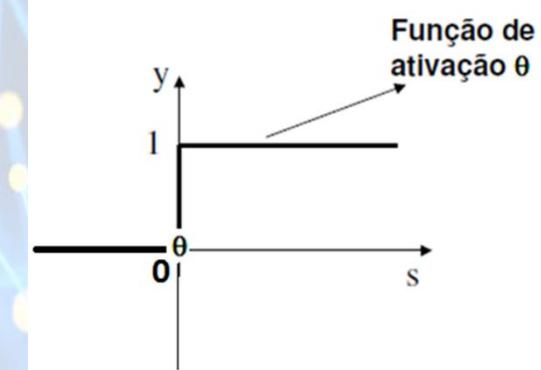
Modelo de McCulloch e Pitts

Porta do Limiar (L)

$$L = \sum_{i=1}^p w_i x_i = w_1 x_1 + w_2 x_2$$

Função de Ativação ($f(L)$)

- $Y = f(L)$
- 1, se $L \geq \theta$ → Limiar
 - 0, se $L < \theta$



Função de ativação (f): Função que regula a saída do neurônio, conforme o **limiar de ativação**.

$$L = W_1 X_1 + W_2 X_2 \geq ou < \theta$$

$$u = W_1 X_1 + W_2 X_2 - \theta \geq ou < 0$$

Potencial de ativação (u): Resultado da diferença entre os valores de **porta do limiar** e **limiar de ativação**.

$$u = \sum_{i=1}^p w_i x_i - \theta$$

Modelo de McCulloch e Pitts

Potencial de ativação (u): Resultado da diferença entre os valores da **porta do limiar** e **limiar de ativação**.

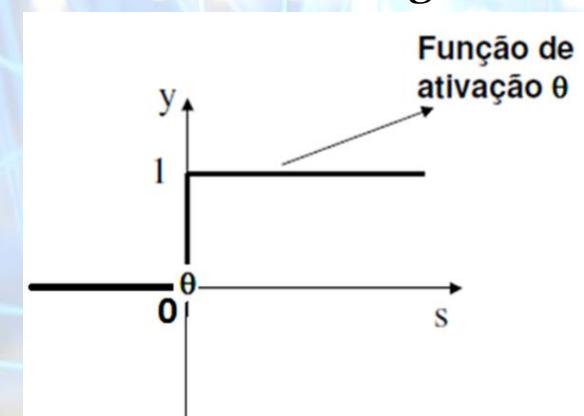
$$u = \sum_i^p W_i X_i - \Theta$$

Função de ativação ($f(u)$): Função que regula a saída do neurônio, conforme o **limiar de ativação**.

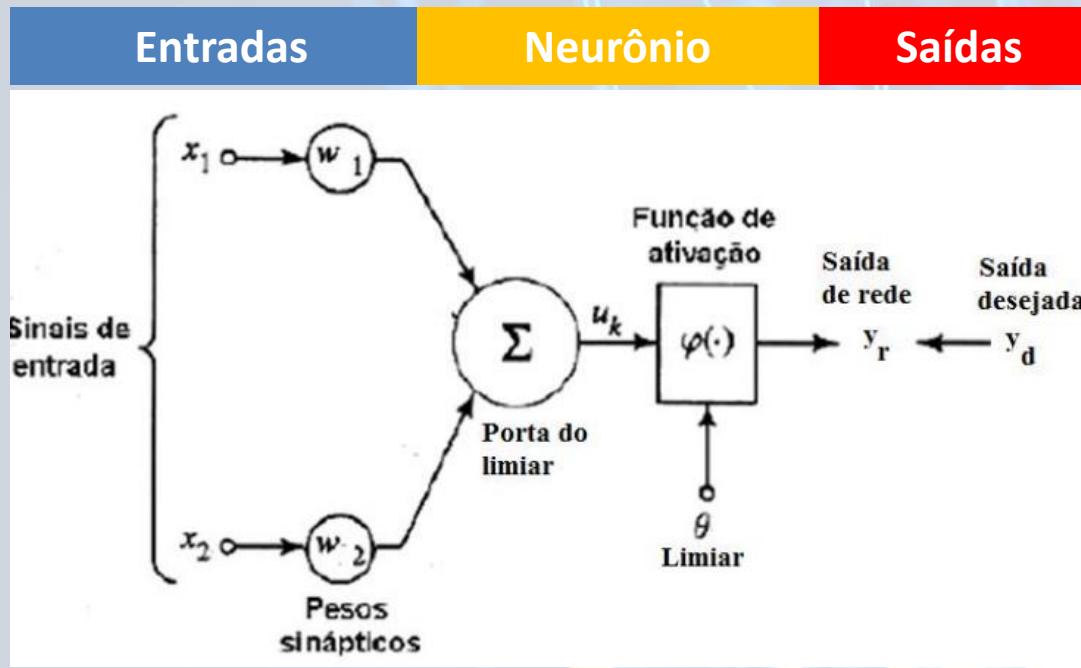
Função de Ativação ($f(u)$)

- $$Y = f(u) \quad \left\{ \begin{array}{l} \bullet \quad 1, \text{ se } u \geq 0 \\ \bullet \quad 0, \text{ se } u < 0 \end{array} \right.$$

Função Degrau



Sinal de Saída (y): Valor final produzido pelo neurônio em relação a um determinado conjunto de sinais de entrada.



Sinais de Entrada (X_i): Medidas advindas do meio externo (variáveis de um problema)

Pesos (W_i): São os valores que irão ponderar cada uma das variáveis de entrada;

Porta do limiar (L): função para agregar todos os sinais de entrada que foram ponderados;

Limiar de ativação (θ): Variável (valor) que regula a resposta do neurônio (“Patamar”).

Potencial de ativação (u): Resultado da diferença entre os valores de **porta do limiar** e **limiar de ativação**.

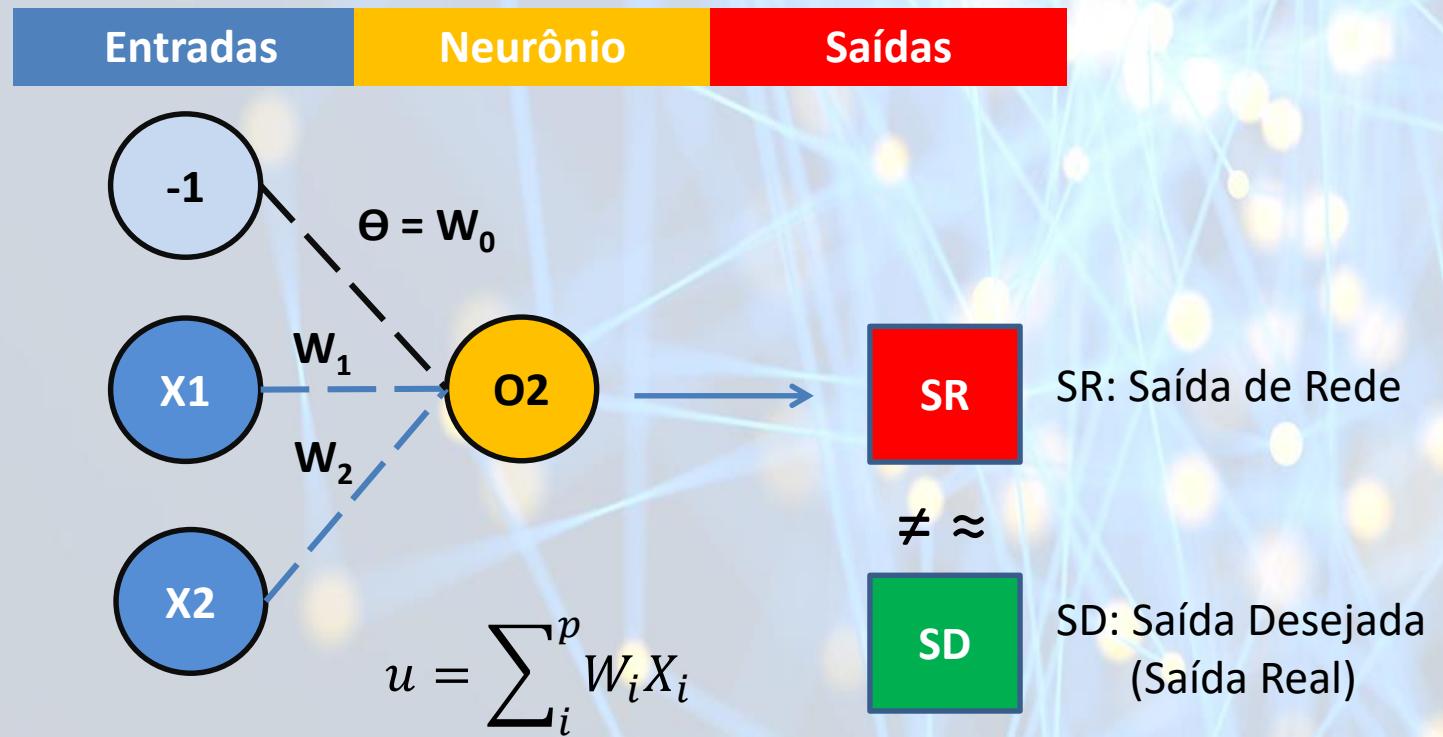
Função de ativação ($f(u)$): Função que limita a saída do neurônio, conforme o **limiar de ativação**.

Sinal de Saída (y): Valor final produzido pelo neurônio em relação a um determinado conjunto de sinais de entrada.

Modelo de McCulloch e Pitts (1943)

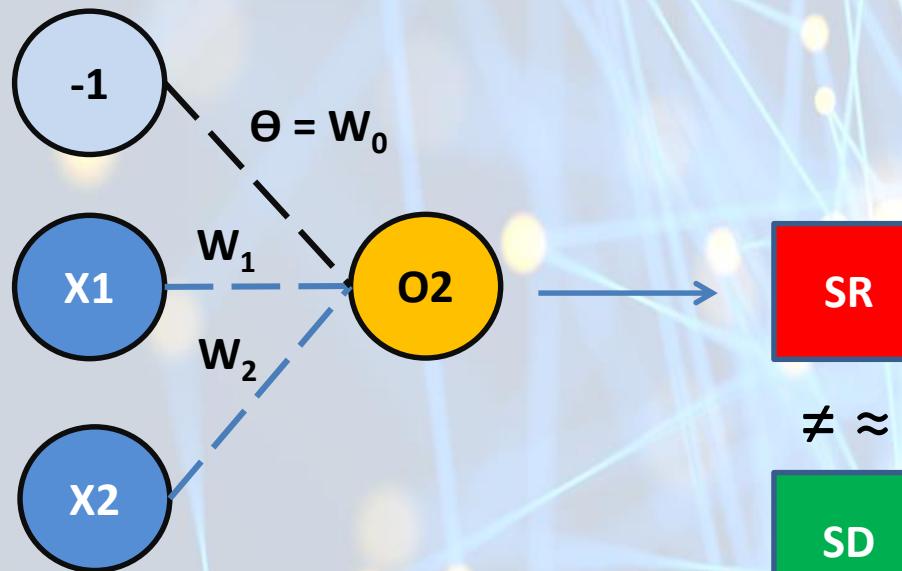
$$u = W_1X_1 + W_2X_2 - 1(\Theta)$$

$$u = \sum_i^p W_i X_i - \Theta$$





COMO ESTIMAR OS PESOS?



$\neq \approx$



SR: Saída de Rede

SD: Saída Desejada
(Saída Real)

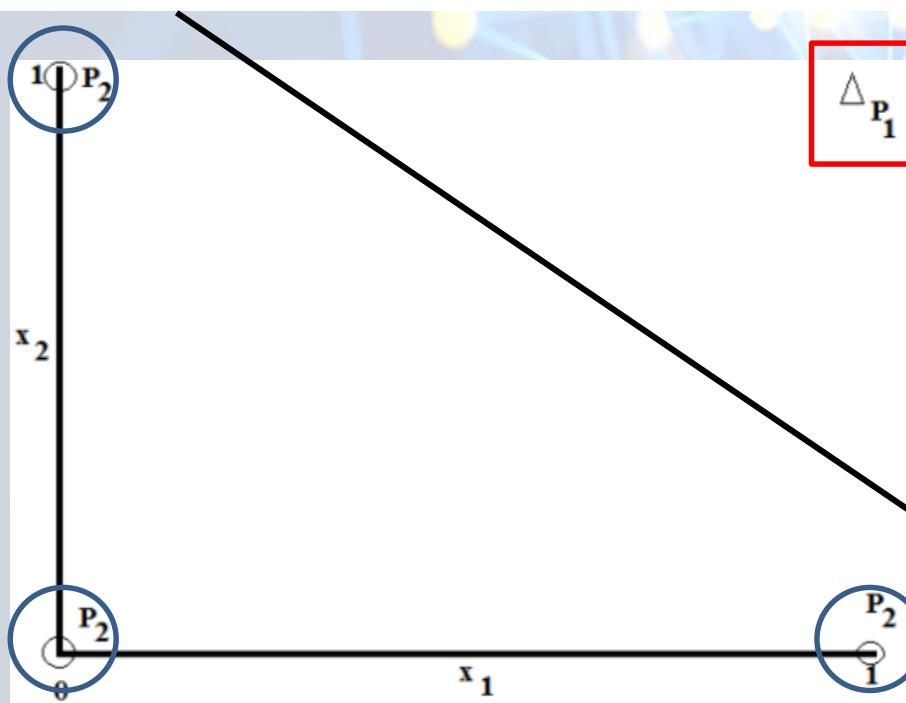
Modelo de McCulloch e Pitts (1943)

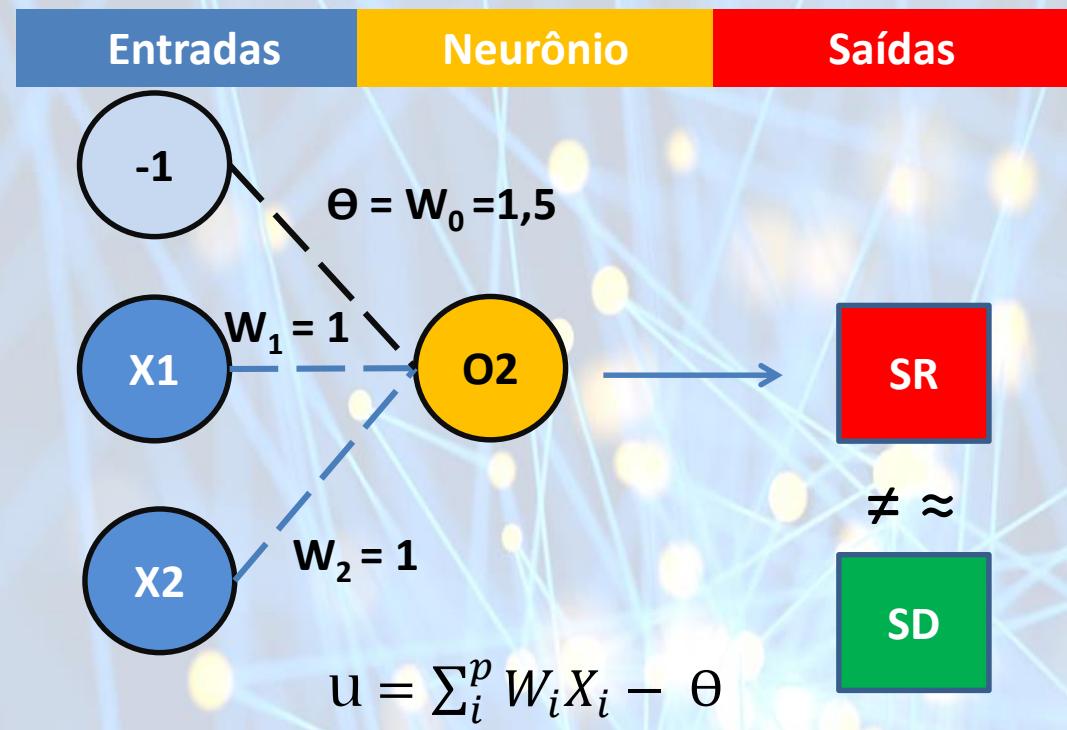
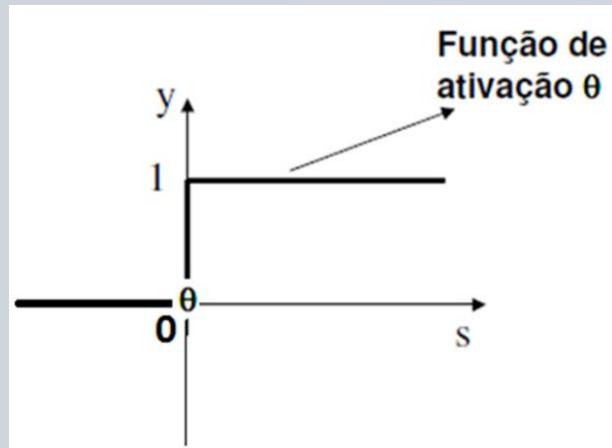
Treinamento: Etapa que consiste em estimar pesos

Época (iteração): Cada vez em que são estimados novos pesos



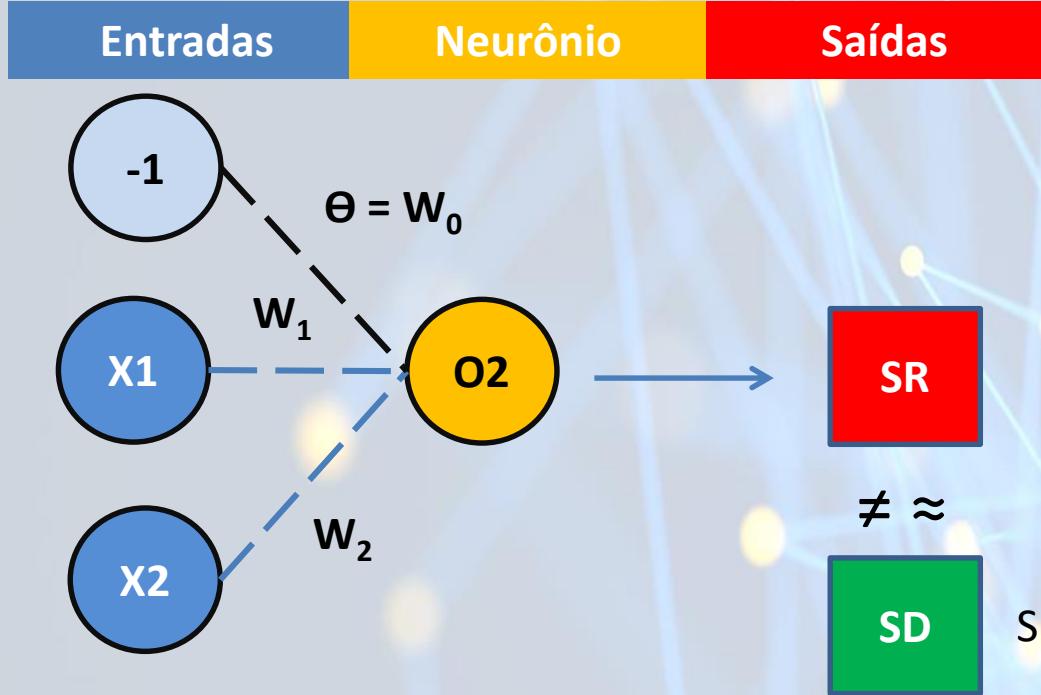
Entradas	Obs1	Obs2	Obs3	Obs4
x_1	1	1	0	0
x_2	1	0	1	0
Saída desejada				
y_d	1 (P_1)	0 (P_2)	0 (P_2)	0 (P_2)





x_1	1	1	0	0	Entrada
x_2	1	0	1	0	Entrada
$L = \sum w_i x_i$	2	1	1	0	Porta do limiar
y_r	1	0	0	0	Saída de Rede
y_d	1	0	0	0	Saída desejada
$\varepsilon = y - y_r$	0	0	0	0	Erros

Modelo Perceptron (1958)



SR: Saída de Rede

SD: Saída Desejada
(Saída Real)

Incorporação de Regras de Aprendizado
Processos iterativos

- Pesos iniciais - aleatórios

$$\vec{w}(t+1) = \vec{w}(t) + \eta \cdot e \cdot \vec{x}$$

$$\vec{X} = [-1; X_1; X_2 \dots X_n]$$

$$\vec{W} = [W_0; W_1; W_2 \dots W_n]$$

$$e = y_d - y_r$$

Taxa de aprendizado: Proporção do erro que contribuirá para o ajuste dos pesos.

Modelo Perceptron (1958)

1. Pesos Aleatórios $\vec{w}(t)$

2. Uso dos Pesos Ajustados

$$\vec{w}(t + 1)$$

- Processamento das Entradas

- Potencial de ativação (u)

$$u = \sum_i^p W_i X_i - \Theta$$

Entradas

Execução

Erro
 $(y_d - y_r)$

Saída de
Rede (y_r)

- Ajuste dos pesos

$$\vec{w}(t + 1) = \vec{w}(t) + \eta \cdot e \cdot \vec{x}$$

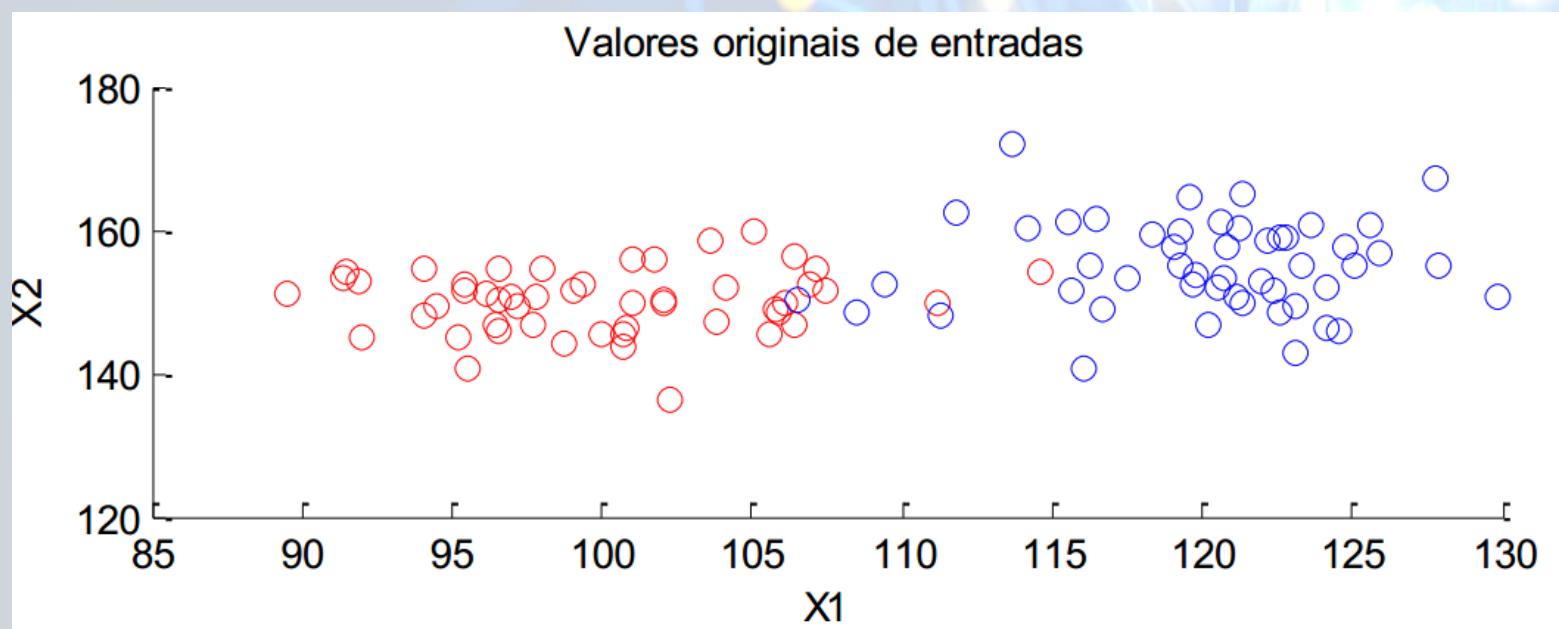
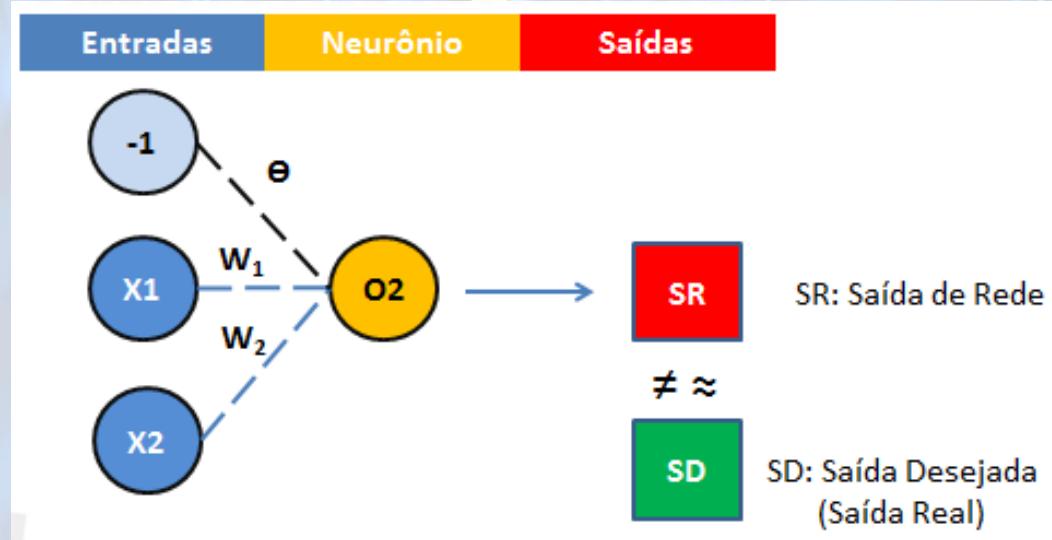
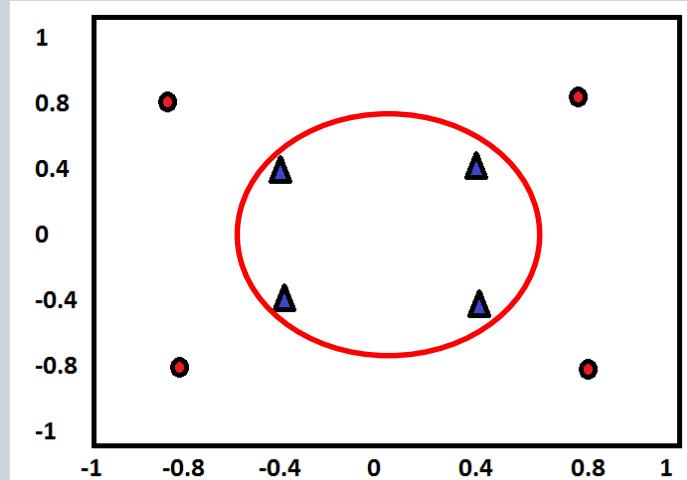
- Obtenção da saída de Rede ($y_r = f(u)$)

Erro: $e = y_d - y_r$

Treinamento: Etapa que consiste em estimar os pesos

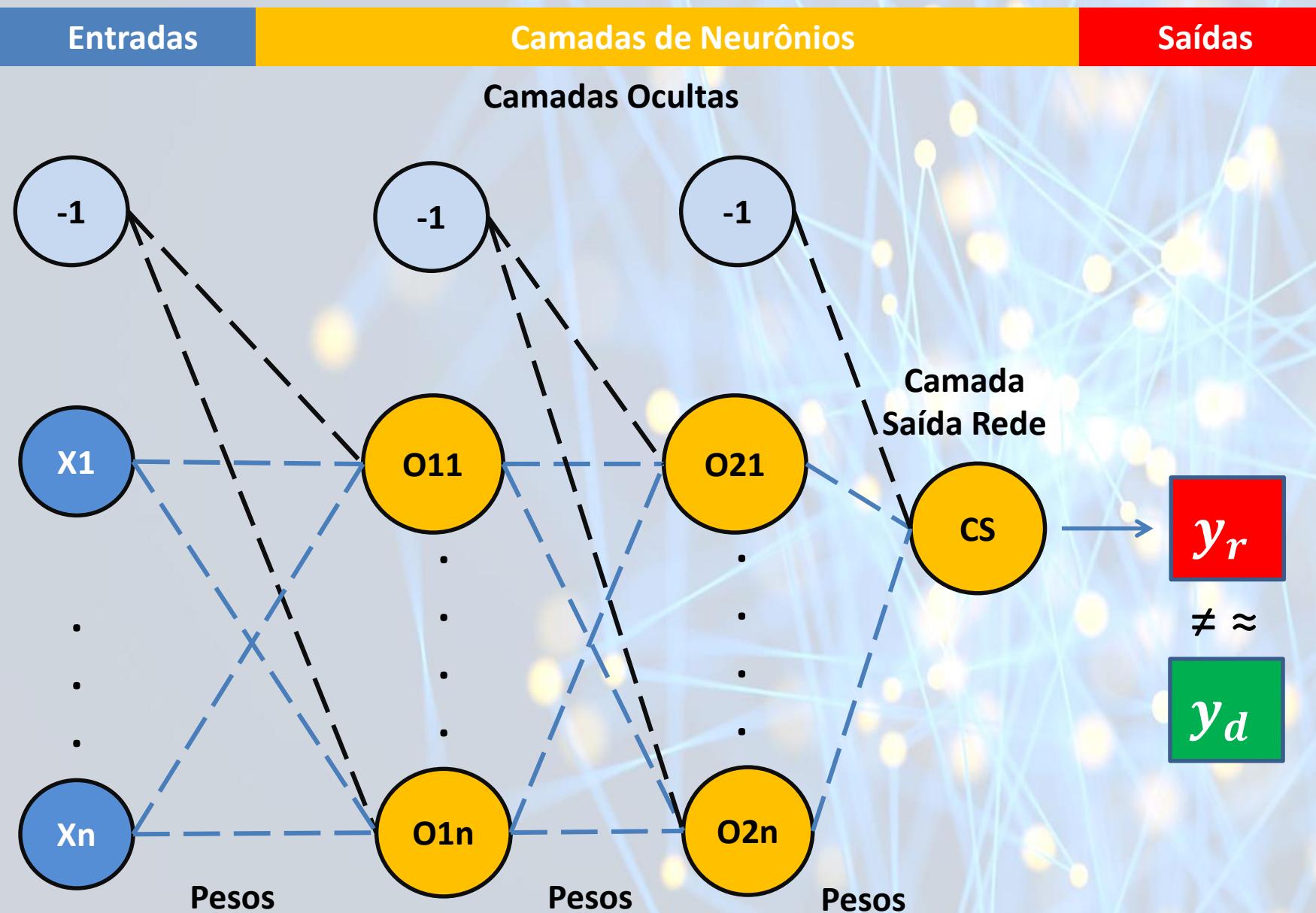
Época (iteração): Cada vez em que são estimados novos pesos

Exemplos

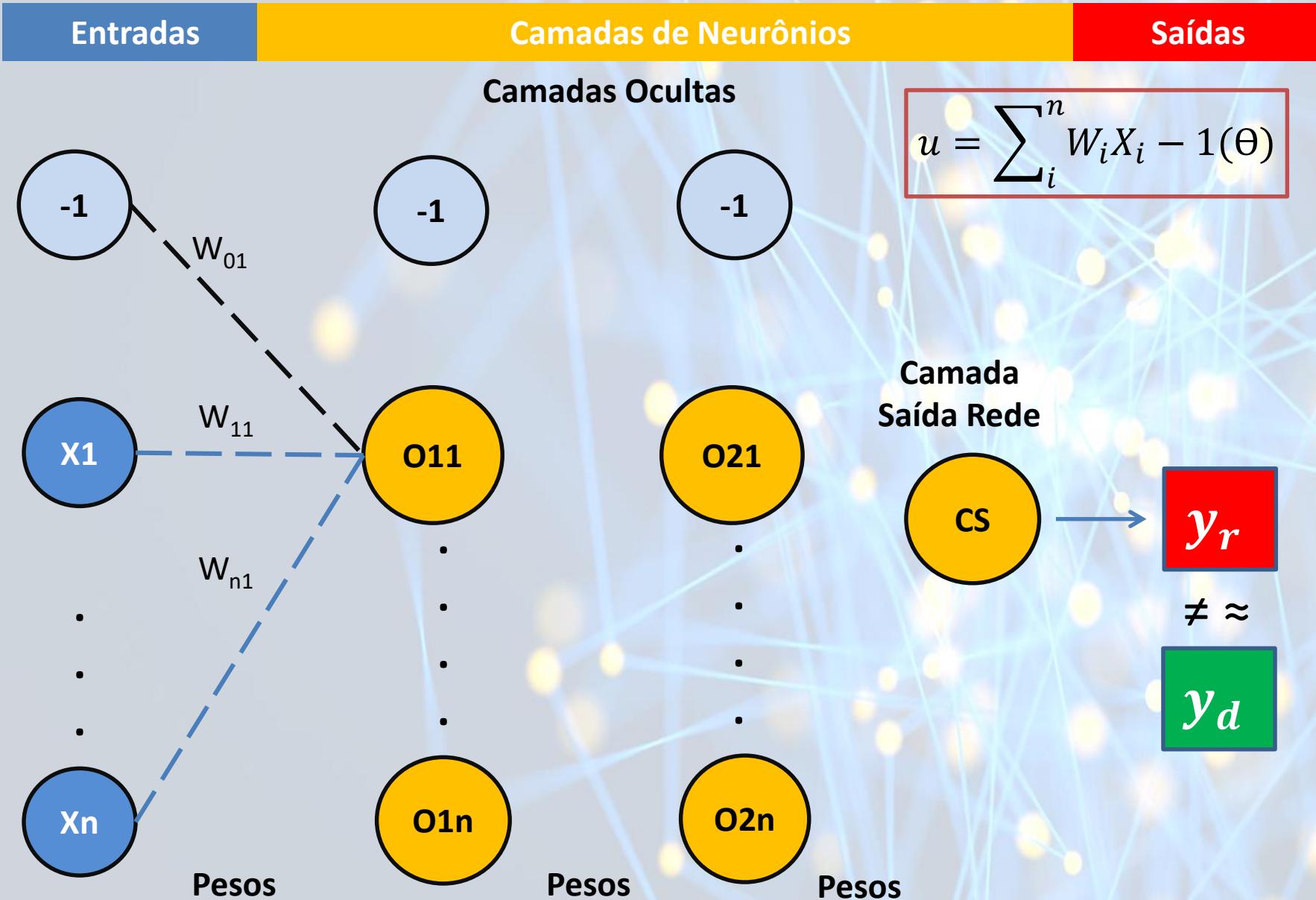


Rede Neural Perceptron Multiplas Camadas

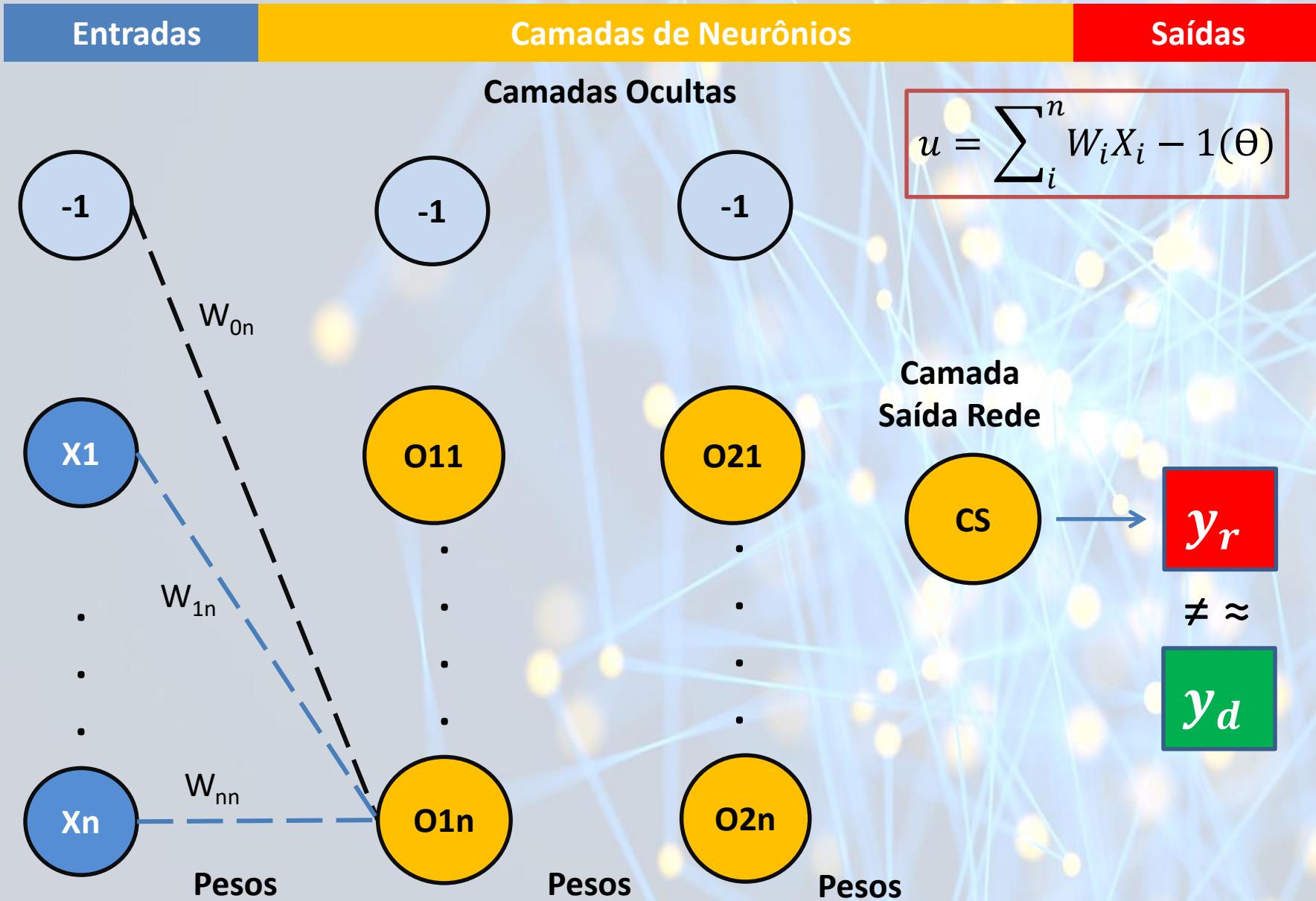
Multilayer Perceptron (MLP)



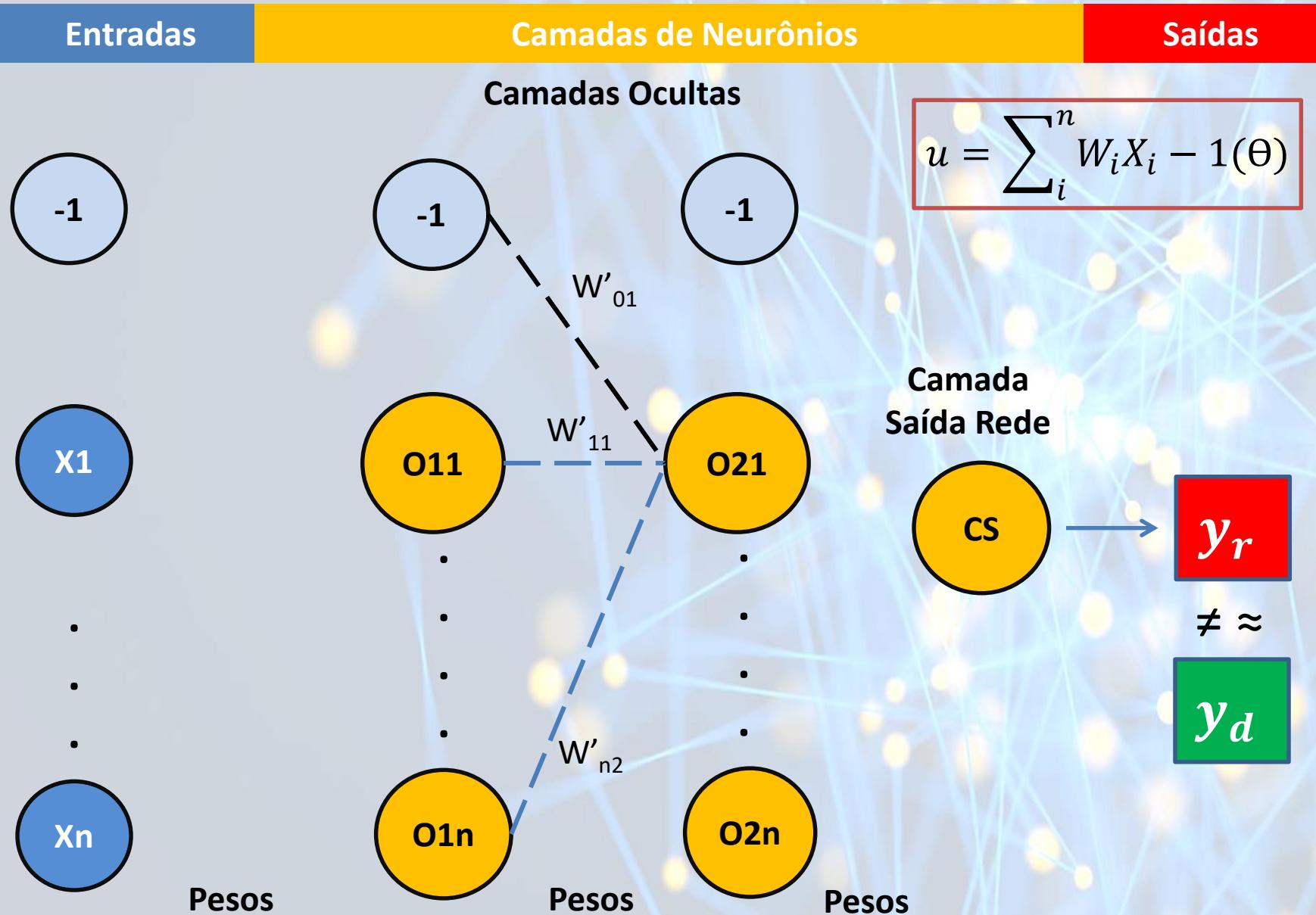
Multilayer Perceptron (MLP)



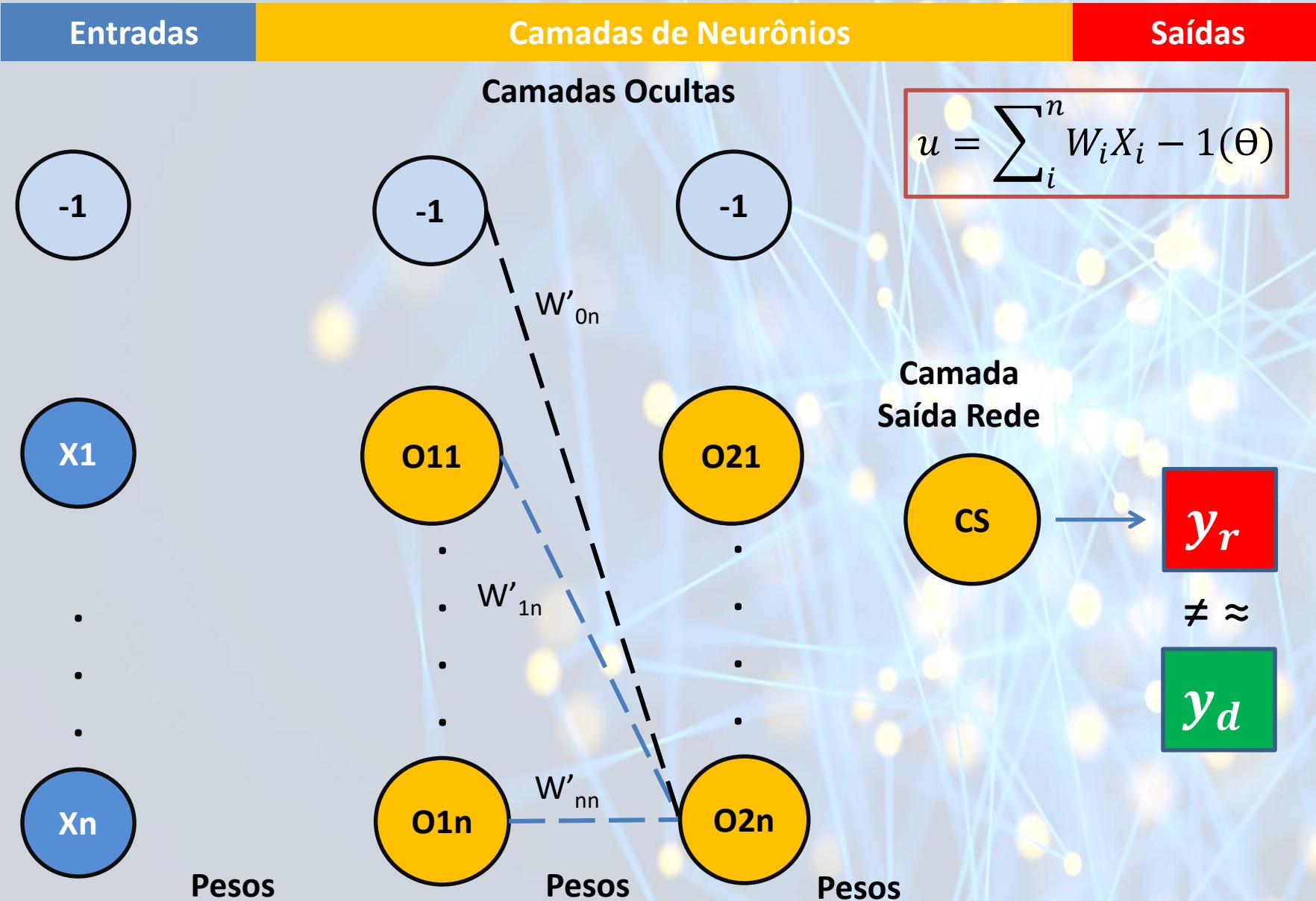
Multilayer Perceptron (MLP)



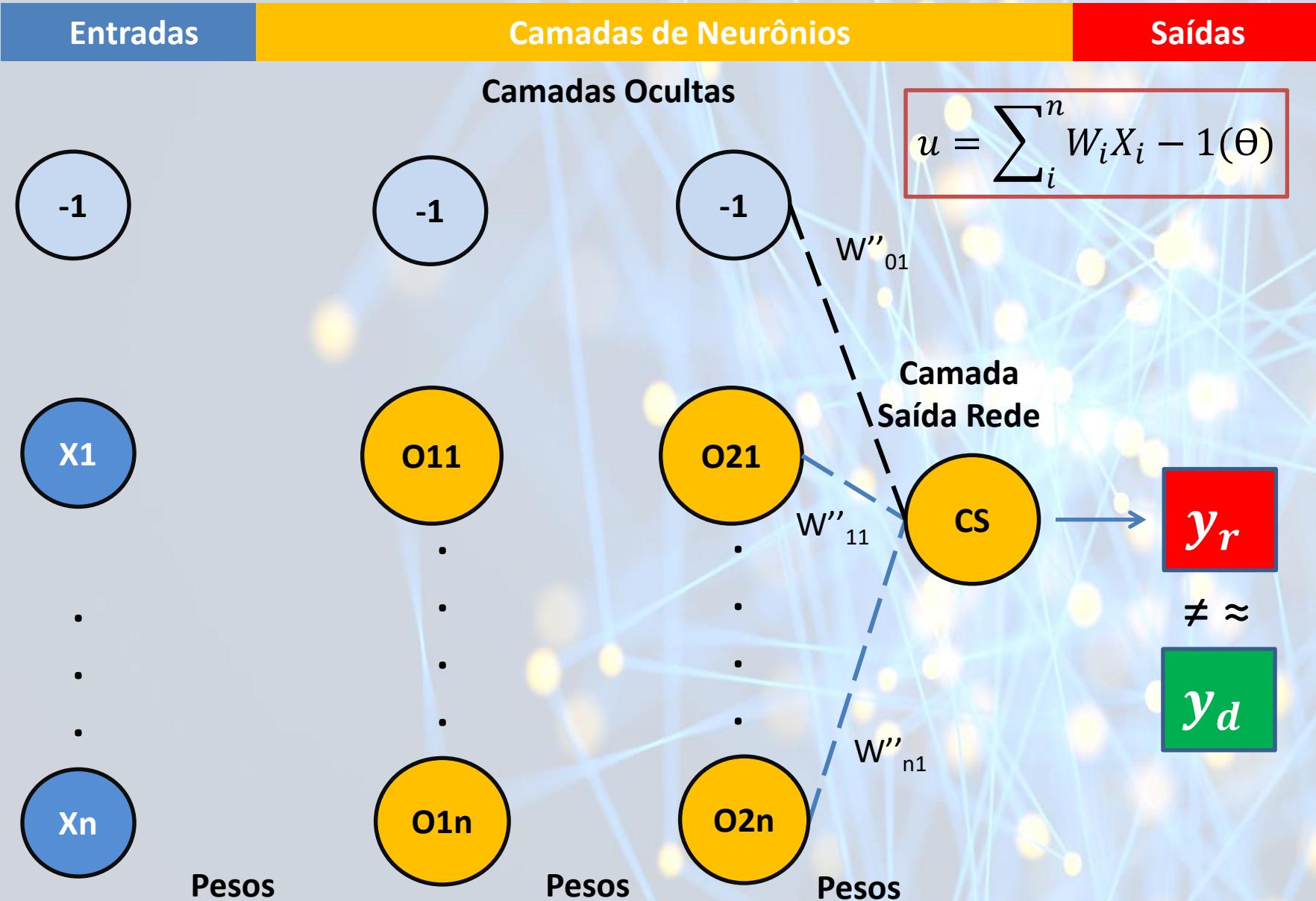
Multilayer Perceptron (MLP)



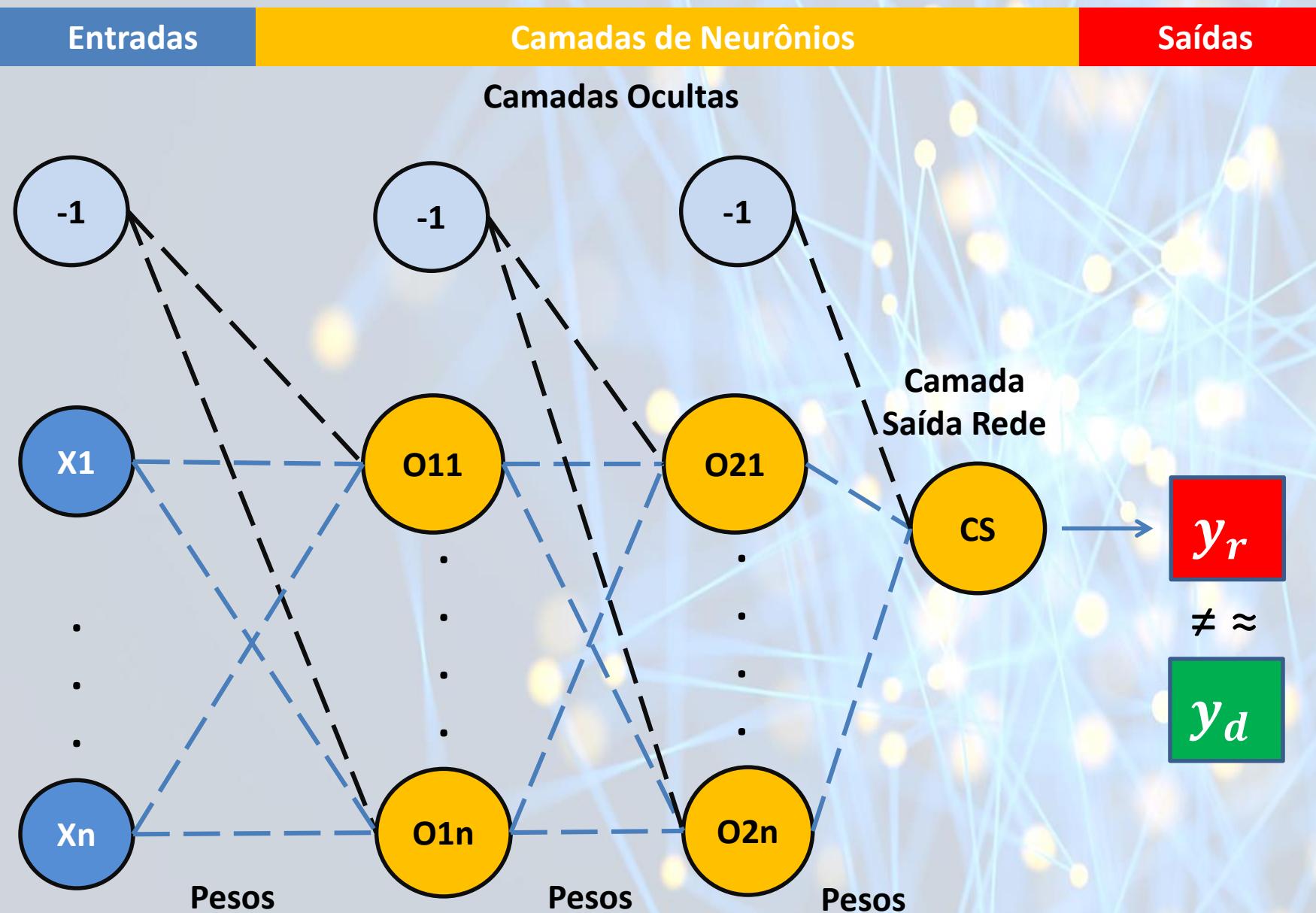
Multilayer Perceptron (MLP)



Multilayer Perceptron (MLP)

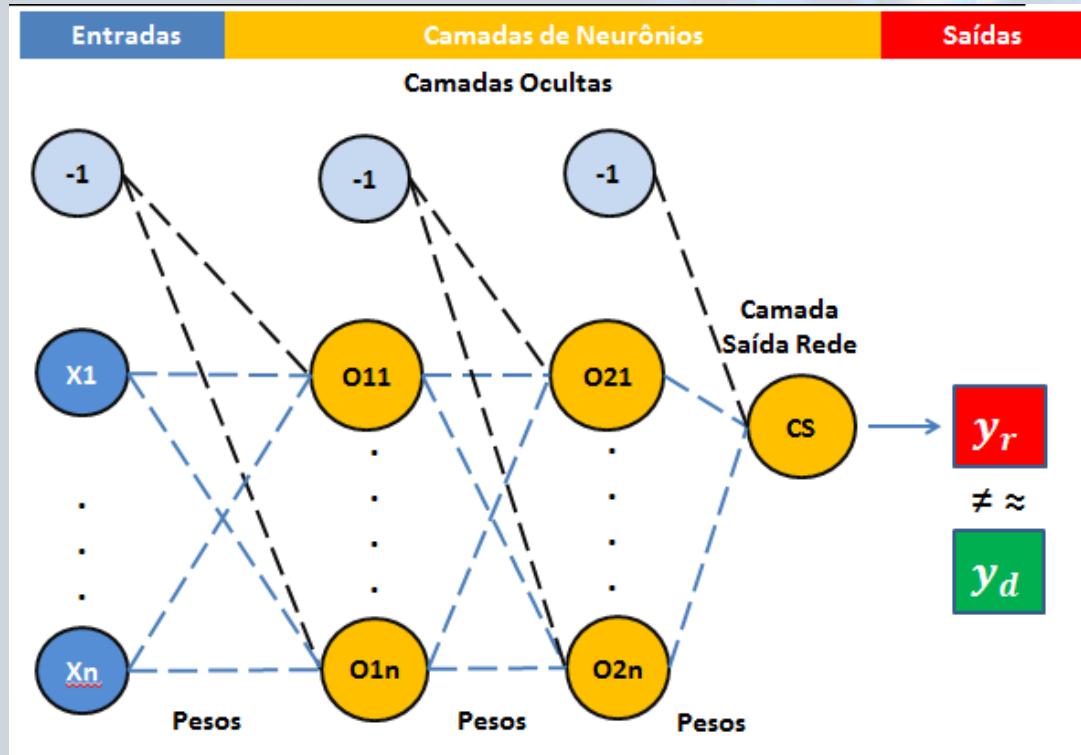


Multilayer Perceptron (MLP)



Algoritmo Backpropagation

Fase Forward



Fase Backward

$$\vec{w}(t+1) = \vec{w}(t) + \eta \cdot e \cdot \vec{x}$$

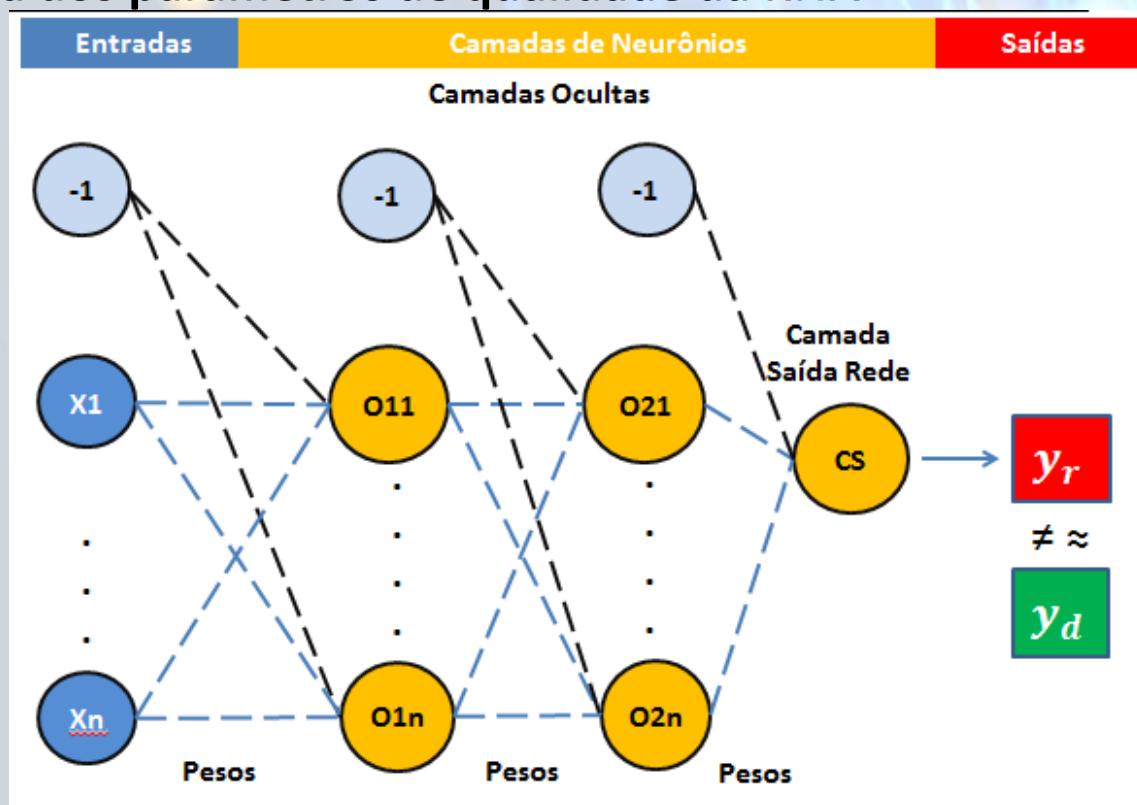
➤ Treinamento

Obs	Entradas	Saída Real
1	[20;500]	1
...
1000	[40;1000]	4



Etapas de Desenvolvimento de uma RNA (MLP)

- ✓ Estudo do problema -> Dados
- ✓ Escolha da arquitetura
 - ❖ Número de camadas ocultas
 - ❖ Número de neurônios
 - ❖ Tipos de funções de ativação
- ✓ Escolha do algoritmo de aprendizado
- ✓ Escolha dos parâmetros de qualidade da RNA



Estudo do Problema - DADOS

➤ Treinamento

Observação	Entradas	Saída Real
1	[20;500]	1
...
800	[40;1000]	4

➤ Validação

Observação	Entradas	Saída Real
1	[10;600]	2
...
200	[60;300]	3

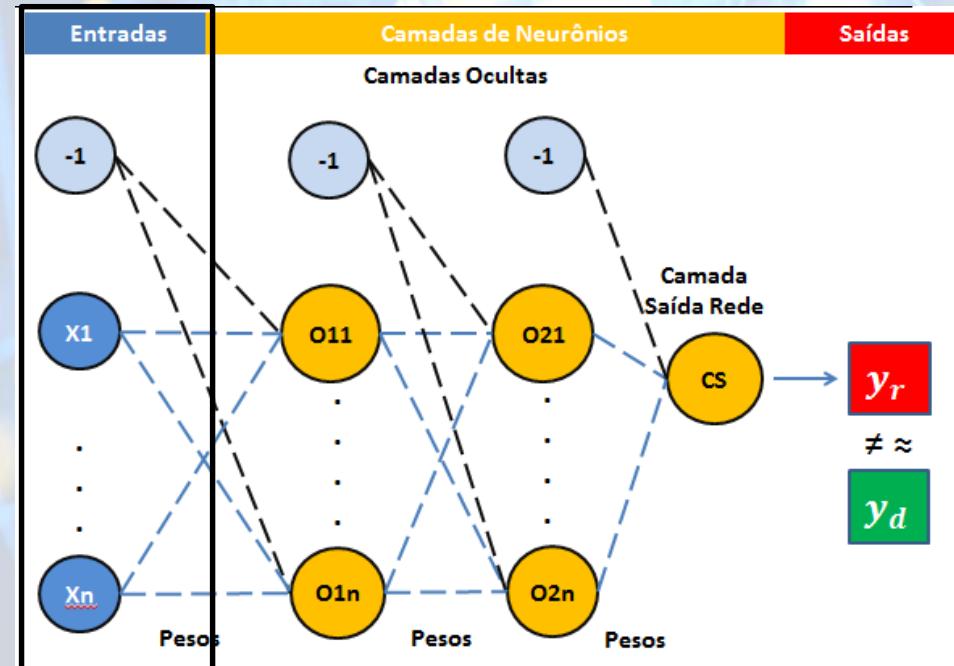
❖ Conjuntos de Dados

❖ Treinamento: 80 %

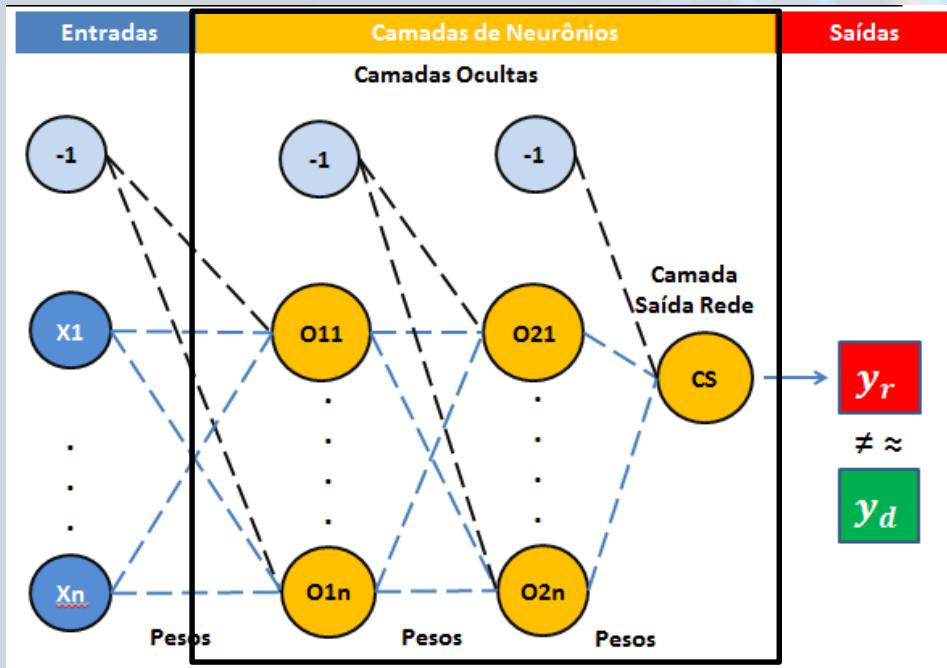
❖ Validação: 20%

➤ Teste

Observação	Entradas	Saída Real
1	[35;1000]	?
...	...	?
100	[20;300]	?



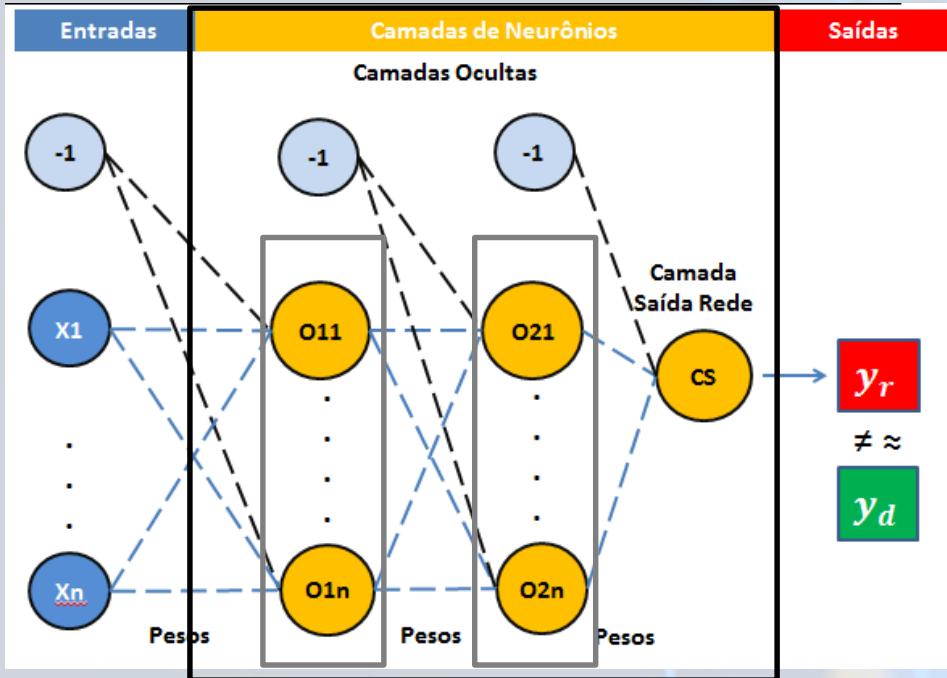
Arquitetura das RNA's - Camadas



- **OBSERVAÇÕES**
- Maior número de camadas ocultas acarreta em:
 - Maior o número de pesos
 - Maior demanda computacional
- **OBSERVAÇÕES**
- Uma única camada de saída da rede neural

- Número de camadas depende do tipo de problema
 - ✓ Problemas classificatórios
 - ❖ Linearmente Separáveis: 1 camada
 - ❖ Não Linearmente Separáveis: pelo menos uma
 - ✓ Problemas de Ajuste de modelos
 - ❖ Simples: 1 Camada
 - ❖ Complexos: 2 ou mais camadas

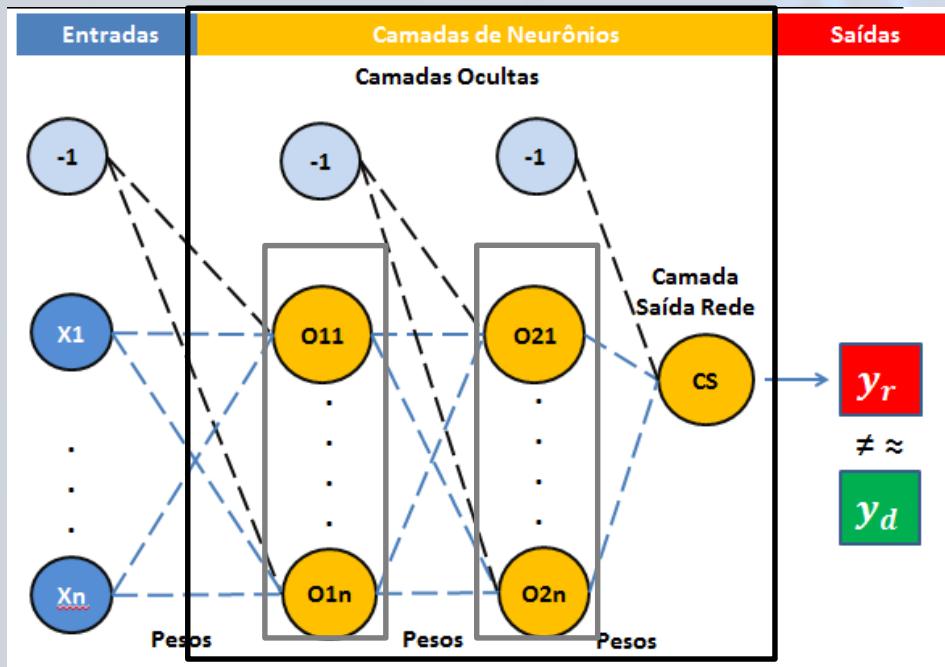
Arquitetura das RNA's - Neurônios



- **OBSERVAÇÕES**
- Maior número de neurônios acarreta em:
 - Maior o número de pesos
 - Maior demanda computacional
- **OBSERVAÇÕES**
- Número de neurônios na camada de saída:
 - 1 ou mais (dependendo do problema)

- O número de neurônios nas camadas ocultas são definidos de forma específica, ou seja, este número pode variar de uma camada para outra.
- Definição do número de neurônios das camadas ocultas:
 - ✓ Forma empírica
 - ❖ Teste utilizando várias combinações (GENES)

Funções de Ativação

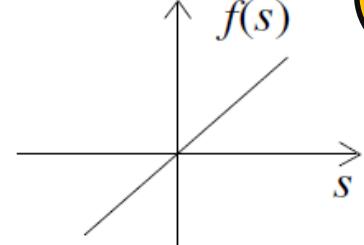
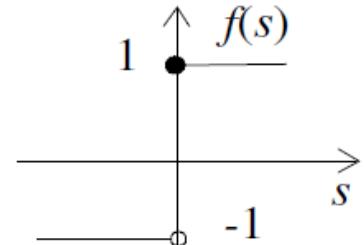
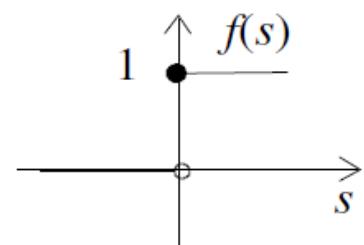


- Todos os neurônios de uma camada oculta apresentam a mesma função de ativação (FA). Porém, o tipo de FA pode variar de uma camada oculta para outra.
- Definição do tipo de função de ativação:
 - ✓ Forma empírica
 - ❖ Teste utilizando várias combinações (GENES)
 - ✓ Consenso
 - ✓ Problemas simples: linear (*purelin*)
 - ✓ Problemas Complexos: logística (*logsig*) ou tangente hiperbólica (*tansig*)

- OBSERVAÇÕES
- O tipo de função de ativação determina:
 - a demanda computacional
- OBSERVAÇÕES
- Função de ativação na camada de saída:
 - Linear (*Purelin*)

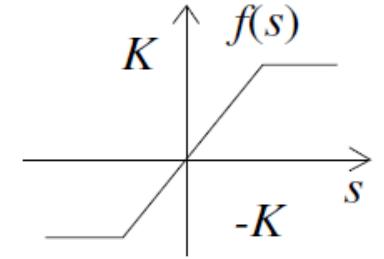
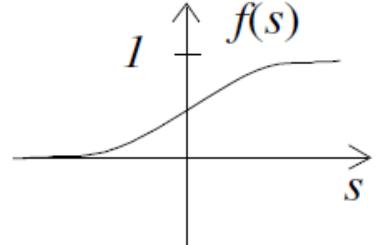
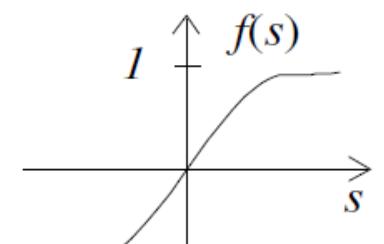
Funções de Ativação

A função de ativação incorporada numa estrutura de RNA dá a cada neurônio que a constitui a capacidade de extrair informações e a potencialidade do aprendizado da rede. O seu papel é determinar a forma e a intensidade de alteração dos valores transmitidos de um neurônio a outro.

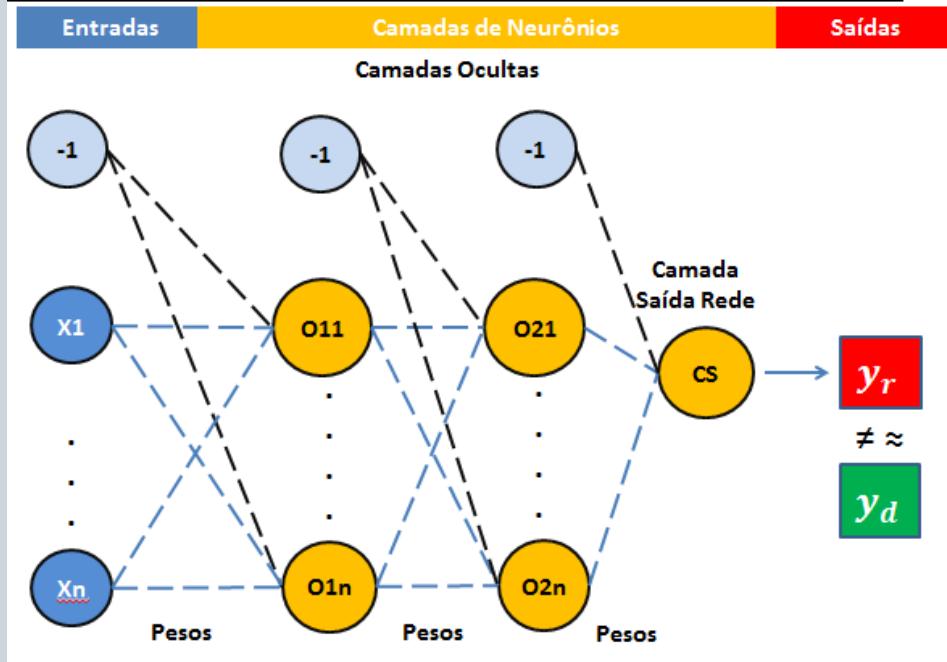
Linear	$f(s) = s$ $u = s = \sum_i^p W_i X_i - \theta$	purelin	
Sinal	$f(s) = \begin{cases} +1 & \text{se } s \geq 0 \\ -1 & \text{se } s < 0 \end{cases}$	hardlims	
Degrau	$f(s) = \begin{cases} +1 & \text{se } s \geq 0 \\ 0 & \text{se } s < 0 \end{cases}$	hardlim	

N

Funções de Ativação

BSB Limiar Lógico	ou $f(s) = \begin{cases} -K & \text{se } s \leq -K \\ s & \text{se } -K < s < +K \\ +K & \text{se } s \geq +K \end{cases}$	satlin satlins	
Logística	$f(s) = \frac{1}{1 + e^{-s}}$	logsig	
Tangente Hiperbólica	$f(s) = \tanh(s) = \frac{1 - e^{-2s}}{1 + e^{-2s}}$	tansig	

Algoritmo de Aprendizado



- OBSERVAÇÕES
- Existem diversos algoritmos baseados no princípio Backpropagation

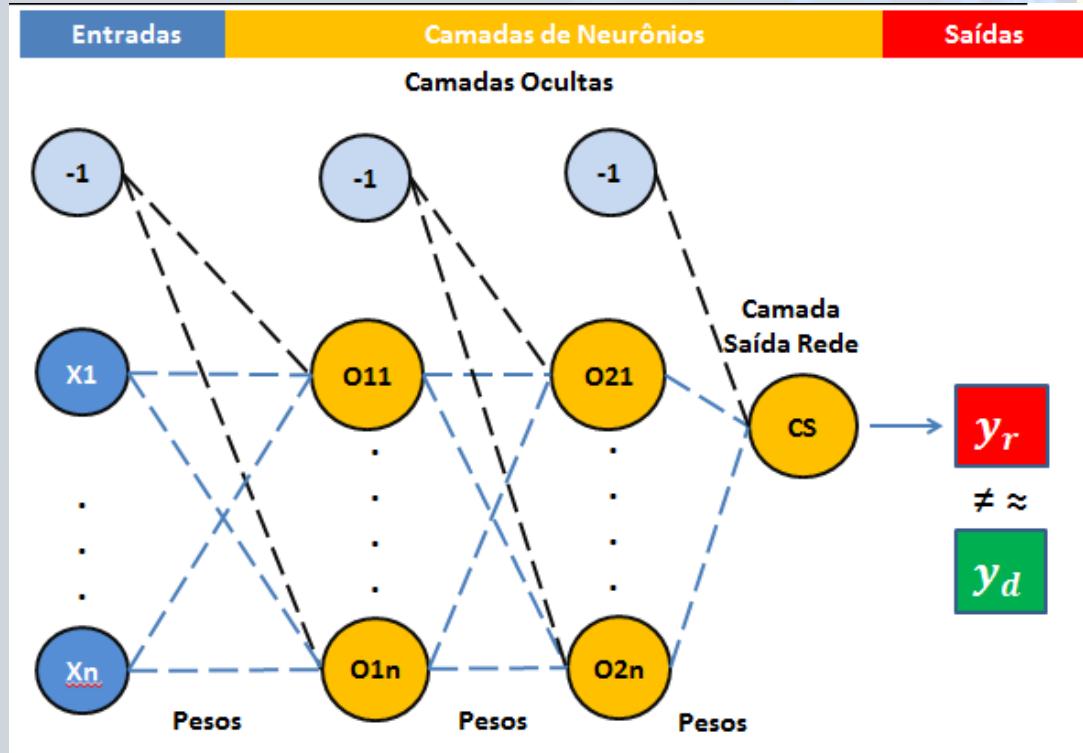
- OBSERVAÇÕES
- O tipo de algoritmo de aprendizado determina:
 - a demanda computacional

- A cada treinamento de uma RNA somente um algoritmo de aprendizado pode ser utilizado.

- Definição do algoritmo de aprendizado:
 - ✓ Forma empírica
 - ✓ Consenso
 - ✓ Problemas simples: Levenberg–Marquardt (*trainlm*), rprop, sag.
 - ✓ Problemas Complexos: Regularização Bayesiana (*trainbr*)

Algoritmo Backpropagation

Fase Forward



Fase Backward

➤ Treinamento

Obs	Entradas	Saída Real
1	[20;500]	1
...
800	[40;1000]	4

Entradas

Execução

Erro
($y_d - y_r$)

Saída de
Rede (y_r)

Parâmetros das RNA's - Treinamento

1. **Taxa de aprendizado:** Proporção do erro que contribuirá para o ajuste dos pesos a cada iteração.

$$\vec{w}(t+1) = \vec{w}(t) + \eta \cdot e \cdot \vec{x}$$

- ❖ Uma taxa de aprendizado **muito baixa** torna o aprendizado da rede muito lento
- ❖ Uma taxa de aprendizado **muito alta** provoca oscilações no treinamento e impede a convergência do processo de aprendizado.
- ❖ Geralmente seu valor varia de 0.1 a 1.0.
- ❖ Sugestão: valor igual a 0.4.

Algoritmo de Aprendizado

1. Iterações: número de vezes que os pesos serão reajustados durante o treinamento de uma RNA.

- Número máximo de iterações (épocas ou ciclos) é determinado pela complexidade do problema.
- Maior número de iterações requer maior demanda computacional.
- Definição do número de iterações:
 - ✓ Forma empírica
 - ✓ Consenso
 - ✓ 500 a 3000 iterações

➤ OBSERVAÇÕES

- Cuidados para determinar número de iterações:
 - Overfitting – Excesso de número de iterações
 - Underfitting – Poucas iterações

2. Erro Quadrático Médio: Medida que expressa em média a qualidade do treinamento e validação de uma RNA.

$$EQM = \frac{1}{p} \sum_{i=1}^p (y_{d(i)} - y_{r(i)})^2$$

- Sugestão é estabelecer um valor de 0.01 no primeiro treinamento e depois ajustá-lo em função do resultado

Resumo – RNA MLP

1. Pesos Aleatórios $\vec{w}(t)$

2. Uso dos Pesos Ajustados

$$\vec{w}(t + 1)$$



- Processamento das Entradas

- Potencial de ativação (u)

$$u = \sum_i^p W_i X_i - \Theta$$

- Ajuste dos pesos

$$\vec{w}(t + 1) = \vec{w}(t) + \eta \cdot e \cdot \vec{x}$$

- Obtenção da saída de Rede ($y_r = f(u)$)

Erro: $e = y_d - y_r$

Aplicações

<https://playground.tensorflow.org/#activation=tanh&batchSize=10&dataset=circle®Dataset=reg-plane&learningRate=0.03®ularizationRate=0&noise=0&networkShape=4,2&seed=0.23530&showTestData=false&discretize=false&percTrainData=50&x=true&y=true&xTimesY=false&xSquared=false&ySquared=false&cosX=false&sinX=false&cosY=false&sinY=false&collectStats=false&problem=classification&initZero=false&hideText=false>

Classificação

Aplicação 01

THE USE OF MULTIPLE MEASUREMENTS IN
TAXONOMIC PROBLEMS

By R. A. FISHER, Sc.D., F.R.S.



Iris Versicolor



Iris Setosa



Iris Virginica

GENES

Classificação

1. A. Multivariada -> Análise Discriminante -> Anderson
2. I.Computacional -> Análise Discriminante -> Neural net

A.Multivariada Diversidade Genética Simulação Matrizes Integração I.Computacional Fenômica Genômica

- Análise de Agrupamento
- Análise de Fatores
- Análise Discriminante**
- Componentes Principais
- Componentes Principais Simplificado
- Componentes Principais(K means/Kohonen)
- R Coordenadas Principais
- Correlações Canônicas
- Dissimilaridade
- Importância de Caracteres
- M Procrustes
- Mapa
- Variáveis Canônicas
- R Manova - DIC e DBC

Rede social Atualizar

I.Computacional Fenômica Genômica Meta Análise

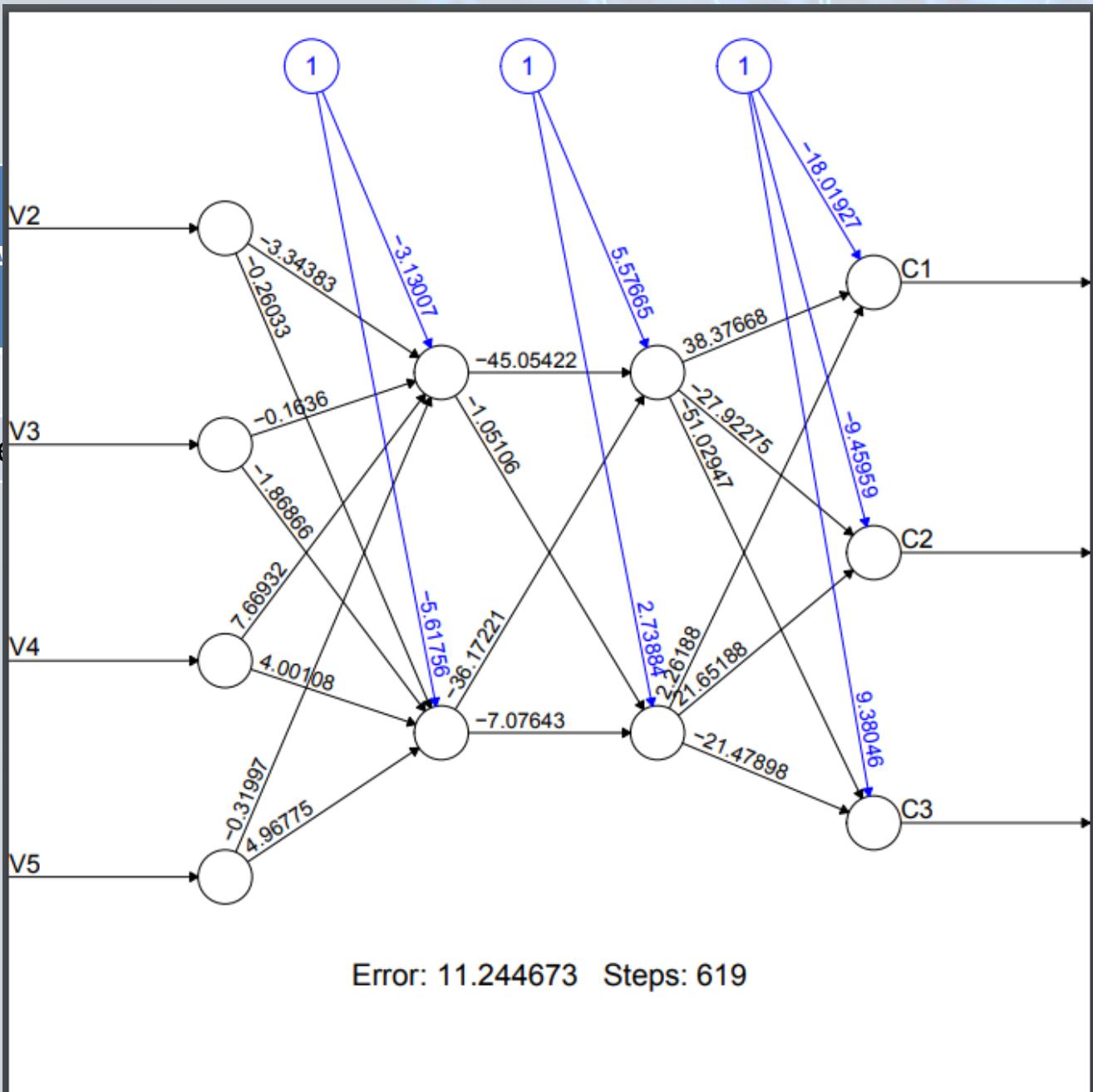
- Comparação de conjunto de dados
- Replicação de ampliação de dados
- Partição de arquivo de dados
- M Análise Discriminante
- R Análise Discriminante**
- M Ajuste de Modelos
- R Ajuste de Modelos
- R Árvore Aleatória
- M GWS
- R GWS
- M PMC - Anova
- R Autoencoder
- M Análise Temporal
- M Estabilidade e Adaptabilidade - Eberhart e Russell
- M Fuzzy - Eberhart e Russell
- M Fuzzy - Eberhart e Russell / Lin e Binns
- M Mapa de Kohonen
- R Mapa de Kohonen
- Modelos Clássicos

[x] [y] Matriz Gene...
...as Mais Soluções e...
...2019.3
...ackage for analysis in exp...
...cientiarum. v.35, n.3, p.2...
...tended and integrated wi...
...n.4, p.547-552, 2016
...mas temos algumas...
...dados
...or está configurado par...
...es e configurações")

RNA MLP

RNA
MELHOR
A.D. Ande

1	VAL	8,37
-		-



RNA MLP

Treinamento

Classe	1	2	3
1	40	0	0
2	0	41	1
3	0	1	37

Validação

Classe	1	2	3
1	10	0	0
2	0	8	0
3	0	0	12

Aplicação 02

<https://archive.ics.uci.edu/ml/datasets/leaf>



Machine Learning Repository
Center for Machine Learning and Intelligent Systems

Leaf Data Set

Download: [Data Folder](#), [Data Set Description](#)

Abstract: This dataset consists in a collection of shape and texture features extracted from digital images of leaf specimens originating from a total of 40 different plant species.

Data Set Characteristics:	Multivariate	Number of Instances:	340	Area:	Computer
Attribute Characteristics:	Real	Number of Attributes:	16	Date Donated	2014-02-24
Associated Tasks:	Classification	Missing Values?	N/A	Number of Web Hits:	100939

Aplicação 02

<https://archive.ics.uci.edu/ml/datasets/leaf>

- | | |
|-------------------------|------------------------------|
| 1. Class (Species) | 9. Maximal Indentation Depth |
| 2. Specimen Number | 10. Lobedness |
| 3. Eccentricity | 11. Average Intensity |
| 4. Aspect Ratio | 12. Average Contrast |
| 5. Elongation | 13. Smoothness |
| 6. Solidity | 14. Third moment |
| 7. Stochastic Convexity | 15. Uniformity |
| 8. Isoperimetric Factor | 16. Entropy |

Aplicação 02



RNA's MLP



Artificial neural networks as auxiliary tools for the improvement of bean plant architecture

V.Q. Carneiro^{1,4*}, G.N. Silva^{2,4*}, C.D. Cruz^{1,2,4}, P.C.S Carneiro^{1,5},
M. Nascimento^{2,4} and J.E.S. Carneiro^{3,5}

Taxas de erro aparente para as RNA's utilizando o diâmetro do hipocótilo individualmente ou em conjunto com altura de plantas.

Procedimentos	Ano de Predição	Taxa de Erro Aparente – TEA (%)		
		Treinamento	Validação	Predição
RNA (DH)	2009	10,33	15,79	15,79
RNA (DH)	2007	12,83	14,04	15,79
RNA (DH + ALT)	2009	6,83	12,28	5,26
RNA (DH + ALT)	2007	1,83	3,51	0,00

RNA's MLP

Pesq. agropec. bras. vol.52 no.3 Brasília Mar. 2017

<http://dx.doi.org/10.1590/s0100-204x2017000300009>

Artificial neural networks compared with Bayesian generalized linear regression for leaf rust resistance prediction in Arabica coffee

Gabi Nunes Silva⁽¹⁾, Moysés Nascimento⁽¹⁾, Isabela de Castro Sant'Anna⁽²⁾, Cosme Damião Cruz⁽²⁾, Eveline Teixeira Caixeta⁽³⁾, Pedro Crescêncio Souza Carneiro⁽²⁾, Renato Domiciano Silva Rosado⁽²⁾, Kátia Nogueira Pestana⁽⁴⁾, Dênia Pires de Almeida⁽⁵⁾ and Marciane da Silva Oliveira⁽²⁾

Table 1. Prediction error rates obtained with artificial neural network and Bayesian generalized linear regression (BGLR) in training and validation populations, and percentage of agreement between the two methodologies.

Methodology	Classification error (%)	
	Training	Validation
Artificial neural network	0.00	1.600
Bayesian generalized linear regression	0.00	2.400
Concordance (%)	-	81.37

Regressão

Aplicações

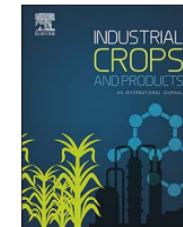
Industrial Crops & Products 108 (2017) 806–813



Contents lists available at ScienceDirect

Industrial Crops & Products

journal homepage: www.elsevier.com/locate/indcrop



Research Paper

High-performance prediction of macauba fruit biomass for agricultural and industrial purposes using Artificial Neural Networks



Carla Aparecida de O. Castro^a, Rafael T. Resende^{a,*}, Kacilda N. Kuki^c, Vinícius Q. Carneiro^b, Gustavo E. Marcatti^d, Cosme Damião Cruz^b, Sérgio Y. Motoike^c

Table 5
Results of prediction accuracy (r) and bias estimation (Bias) of the semi-destructive and non-destructive procedures.

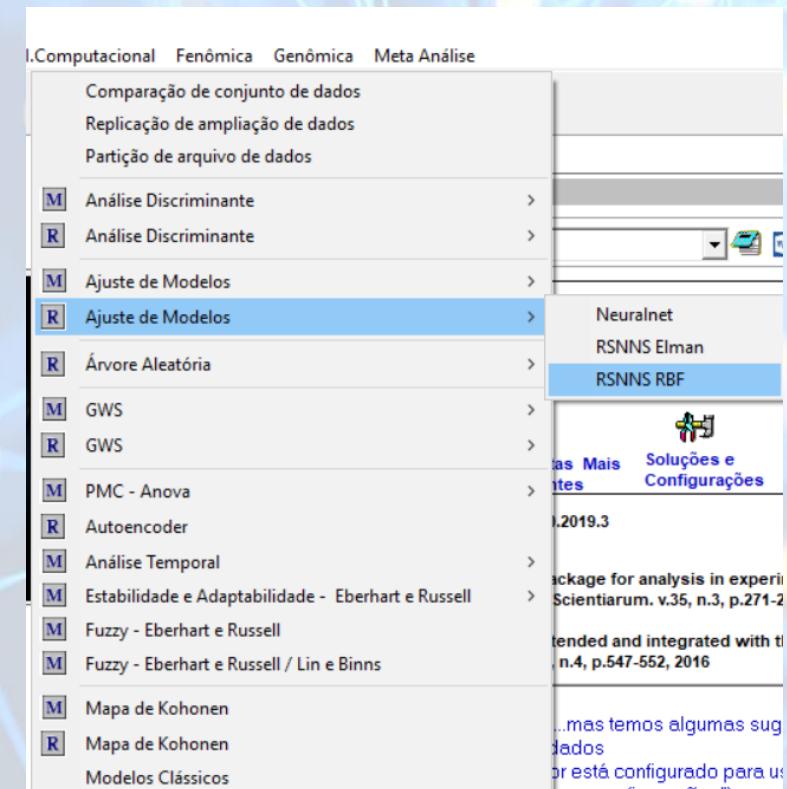
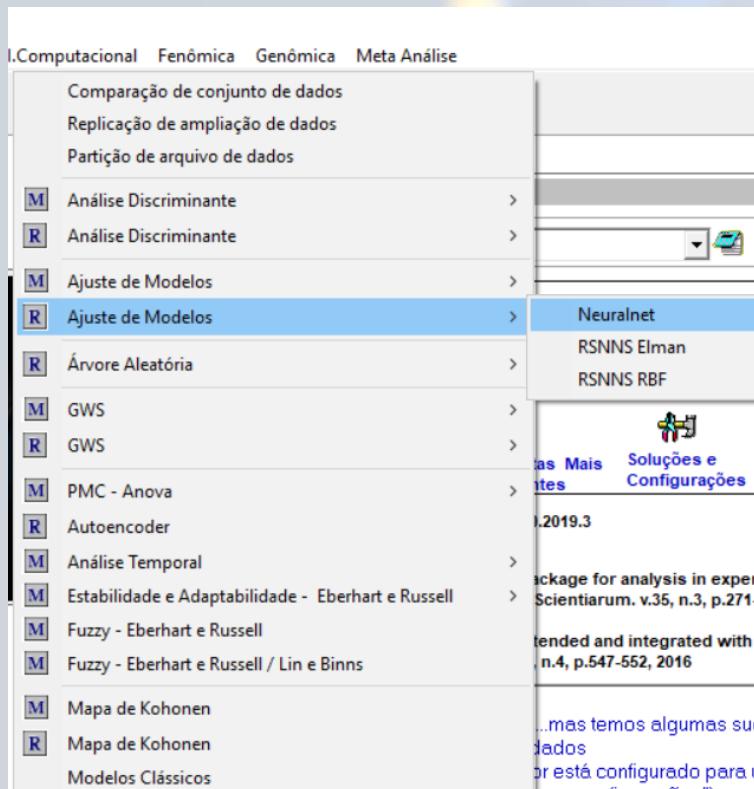
Procedure	Fruit Variable	RS		G1 → G2		G2 → G1		
		ANN		MLM		ANN		MLM
		r	Bias (%)	r	r	Bias (%)	r	r
Semi-destructive	KDW	0.93	8.73	0.56	0.97	6.56	0.50	0.92
	PDW	0.98	4.46	0.73	0.98	4.64	0.80	0.97
	HDW	0.97	6.00	0.81	0.98	4.82	0.79	0.96
	EDW	0.95	5.80	0.66	0.98	5.07	0.62	0.96
	KOC	0.96	0.94	0.16	0.96	0.74	0.02	0.95
	POC	0.89	7.23	0.41	0.90	3.97	0.59	0.89
Non-destructive	KDW	0.96	8.17	0.50	0.97	6.53	0.50	0.90
	PDW	0.97	5.63	0.78	0.94	7.58	0.76	0.94
	HDW	0.96	6.37	0.82	0.95	5.79	0.78	0.94
	EDW	0.95	6.56	0.63	0.97	5.38	0.59	0.90
	KOC	0.86	1.83	0.19	0.94	1.43	0.02	0.79
	POC	0.75	10.70	0.45	0.76	7.95	0.62	0.73

RS: Random Sampling; G1 → G2: Training on G1 group and validation on G2 group; G2 → G1: Training on G2 group and validation on G1 group; ANN: Artificial Neural Network; MLM: Multivariate Linear Model. Fruit variables acronyms, KDW: Kernel or seed dry weight; EDW: Endocarp Dry Weight; PDW: Pulp or Mesocarp dry Weight; HDW: Husk or Exocarp Dry Weight; KOC: Oil Content in the Kernel; POC: Oil Content in the Pulp.

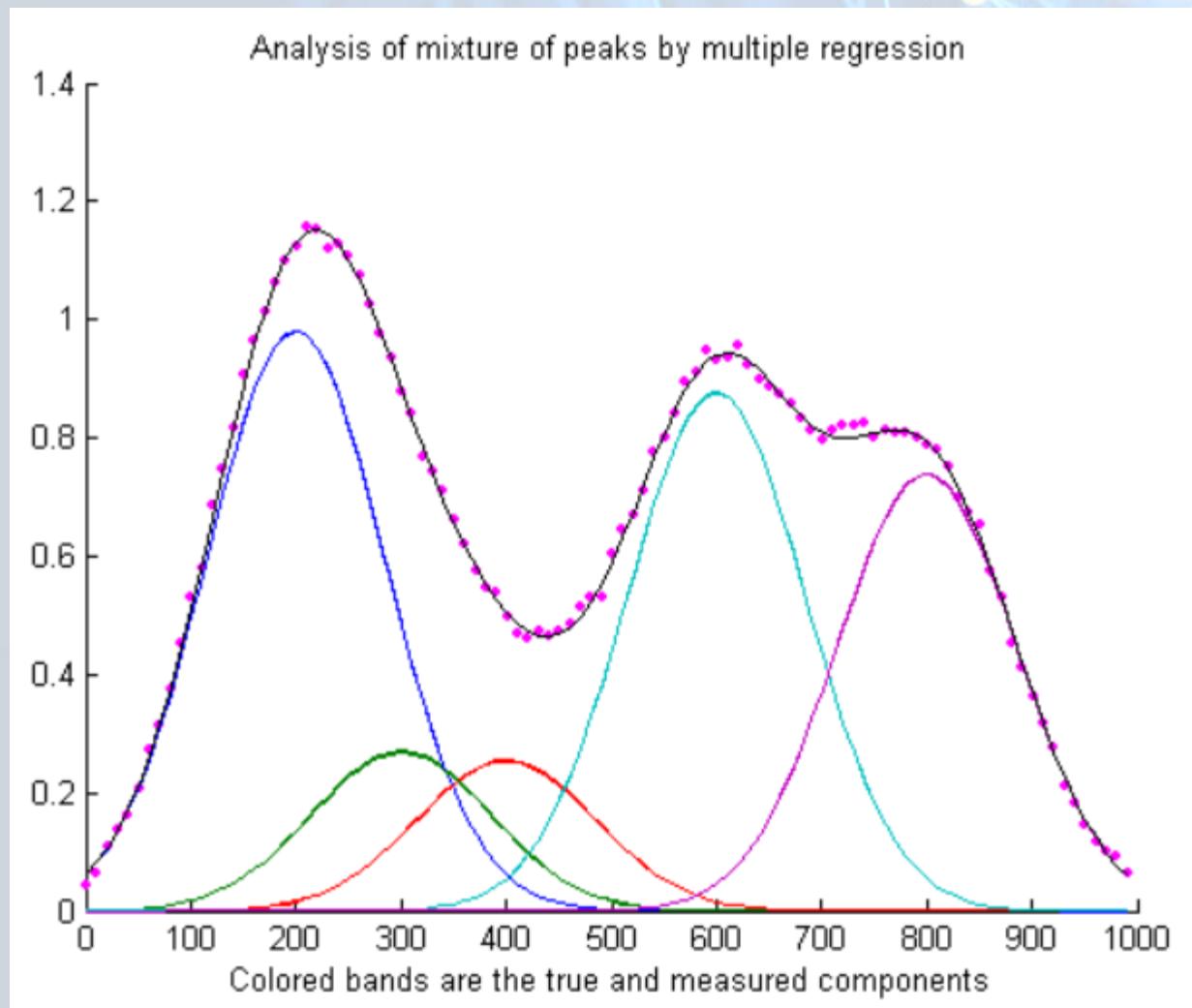
GENES

Regressão

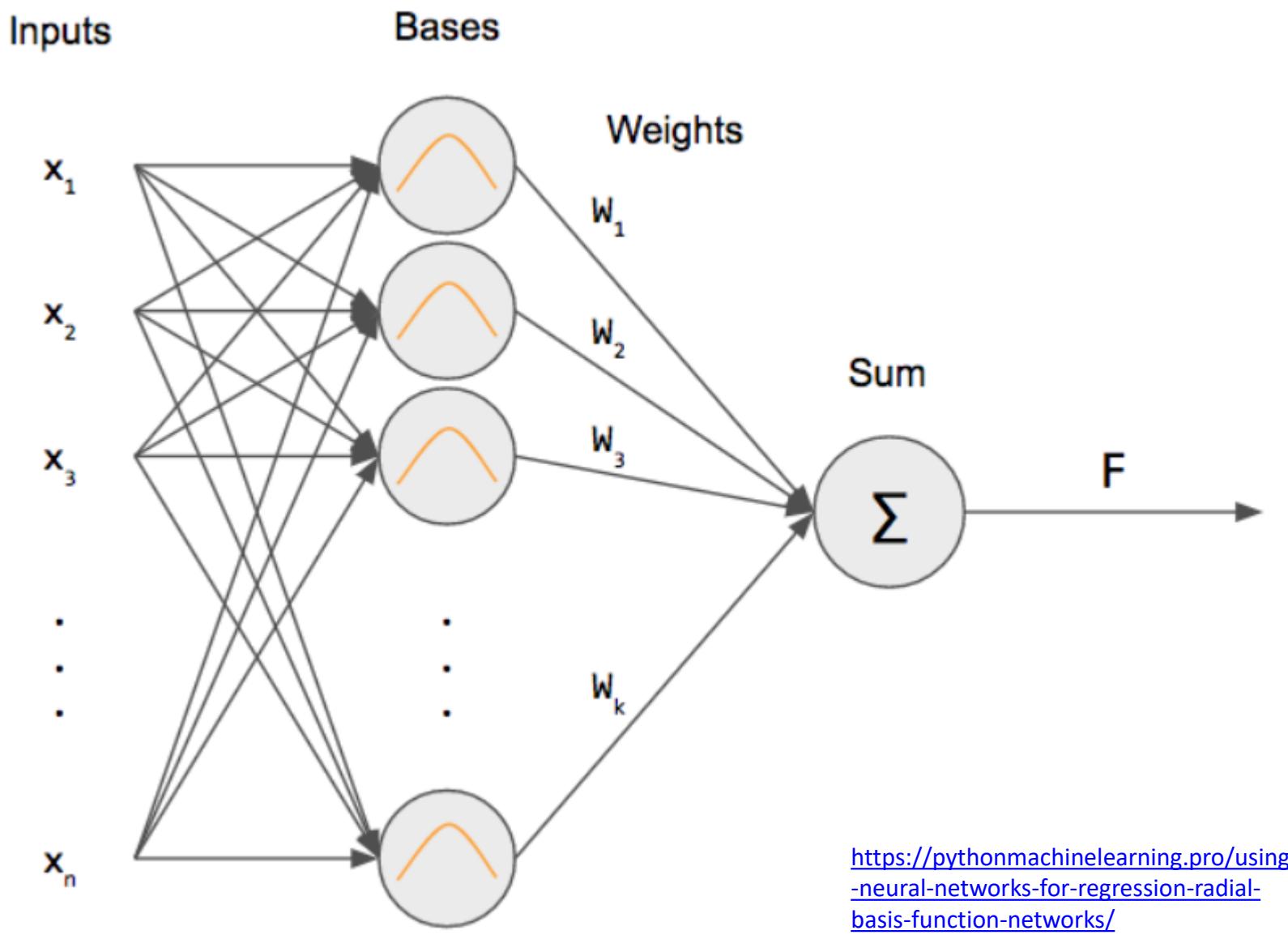
1. I.Computacional -> Ajuste de Modelos -> Neuralnet
2. I.Computacional -> Ajuste de Modelos -> RSNNS RBF



Rede Neural de Base Radial Regressão

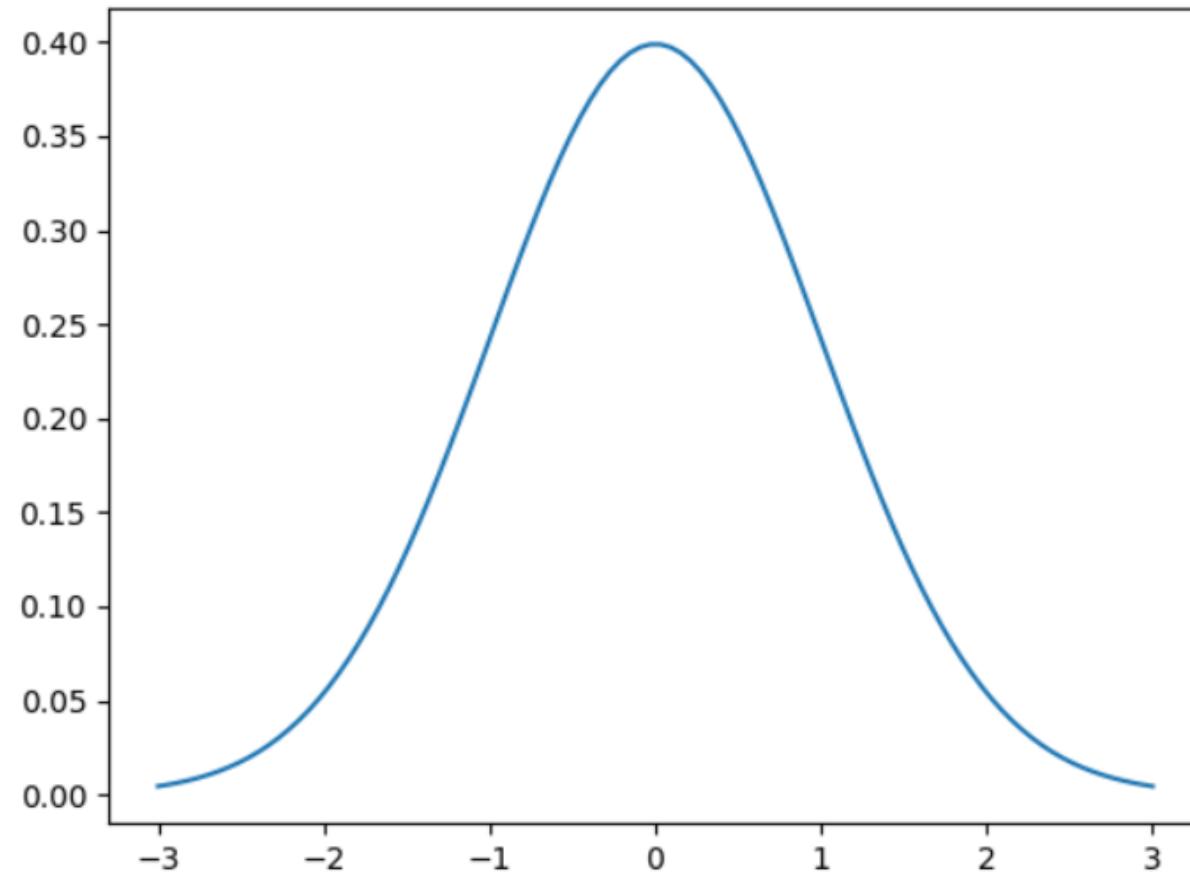


Rede Neural de Base Radial (RBF)



Função Gaussiana

$$\mathcal{N}(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x - \mu)^2}{2\sigma^2}}$$



Aplicação 03

Predição de preços de apartamentos

Zobacz mieszkania w najnowszych inwestycjach



ul. 28 lipca 1943 r.

Apartamenty Wola III etap



Dominikanów

Park Zaczarowanej Dorożki



ul. Gdańską 12

GDAŃSKA 12

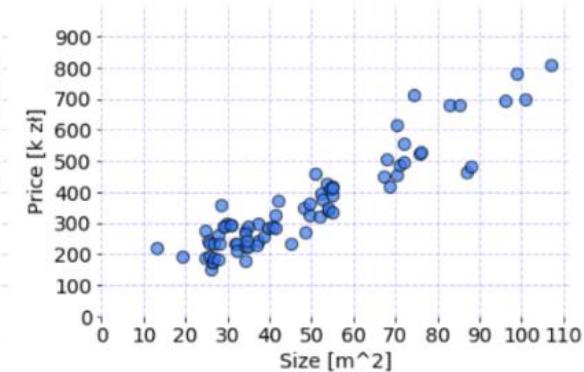
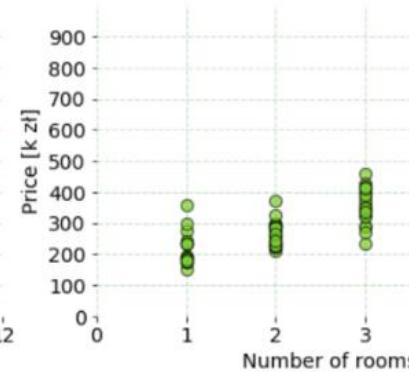
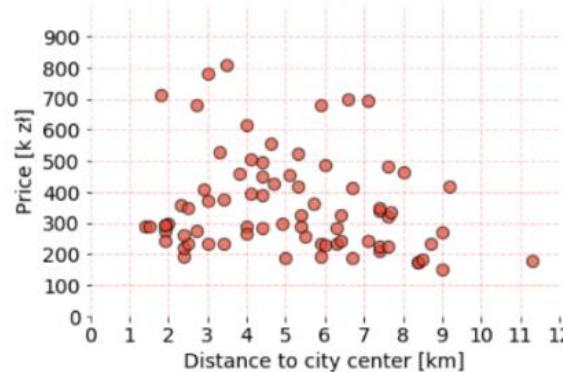


ul. Ułańów 64D



ul. Lublańska / J. bud. 1

Zaczarowany Młyn





ELSEVIER

ECOLOGICAL
MODELLING

Ecological Modelling 154 (2002) 135–150

www.elsevier.com/locate/ecolmodel

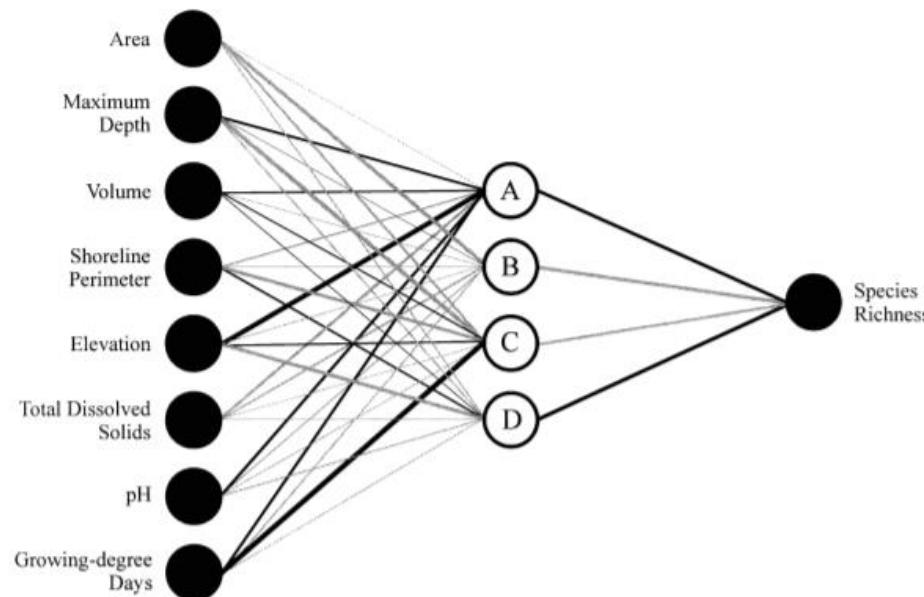
Illuminating the “black box”: a randomization approach for understanding variable contributions in artificial neural networks

Julian D. Olden *, Donald A. Jackson

Department of Zoology, University of Toronto, 25 Harbord Street, Toronto, Ontario, Canada M5S 3G5

Rec

02

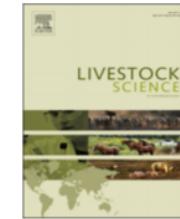




Contents lists available at [ScienceDirect](#)

Livestock Science

journal homepage: www.elsevier.com/locate/livsci



Short communication

Accessing marker effects and heritability estimates from genome prediction by Bayesian regularized neural networks



Leonardo Siqueira Glória ^{a,c}, Cosme Damião Cruz ^b, Ricardo Augusto Mendonça Vieira ^c,
Marcos Deon Vilela de Resende ^d, Paulo Sávio Lopes ^a, Otávio H.G.B. Dias de Siqueira ^a,
Fabiano Fonseca e Silva ^{a,*}

^a Department of Animal Science, Universidade Federal de Viçosa, Av. P.H. Holfs, 36570-000 Viçosa, Brazil

^b Department of General Biology, Universidade Federal de Viçosa, Av. P.H. Holfs, 36570-000 Viçosa, Brazil

^c Laboratory of Animal Science, Universidade Estadual do Norte Fluminense, 28013-602 Campos dos Goytacazes, Brazil

^d Embrapa Florestas/UFV, Estrada da Ribeira, km 111, 83411-000 Colombo, Brazil



Gianola et al. BMC Genetics 2011, 12:87
<http://www.biomedcentral.com/1471-2156/12/87>



METHODOLOGY ARTICLE

Open Access

Predicting complex quantitative traits with Bayesian neural networks: a case study with Jersey cows and wheat

Daniel Gianola^{1,2,3}, Hayrettin Okut^{1,4*}, Kent A Weigel² and Guilherme JM Rosa^{1,3}

González-Camacho et al. BMC Genomics (2016) 17:208
DOI 10.1186/s12864-016-2553-1

BMC Genomics

RESEARCH ARTICLE

Open Access

Genome-enabled prediction using probabilistic neural network classifiers

Juan Manuel González-Camacho¹, José Crossa^{2*}, Paulino Pérez-Rodríguez¹, Leonardo Ornella³ and Daniel Gianola⁴



Conhecimento nunca é demais



“Não importa o quanto devagar você vá, desde que você não pare” (Confúcio)