

**Universidade Federal de São Carlos**

**Bruna Zamith  
Diego Roberto Gonçalves de Pontes  
Elaine Cecília Gatto**

Comparação das Redes Neurais Multi-Layer Perceptron,  
Radial Basis Function Network e Self Organizing Maps  
na classificação do dataset Iris

Trabalho apresentado à disciplina de Introdução a Redes Neurais Artificiais do curso de Ciência da Computação do Departamento de Computação da Universidade Federal de São Carlos.

Professor Dr. Ricardo Cerri.

**São Carlos  
Abril  
2018**

# Conteúdo

<b>1</b>	<b>Introdução e objetivos</b>	<b>2</b>
<b>2</b>	<b>Configuração das Redes Neurais e estratégias para classificação</b>	<b>2</b>
2.1	Multi-Layer Perceptron . . . . .	2
2.2	Radial Basis Function Network . . . . .	3
2.3	Self Organizing Maps . . . . .	3
<b>3</b>	<b>Metodologia de Comparação</b>	<b>4</b>
<b>4</b>	<b>Resultados</b>	<b>5</b>
<b>5</b>	<b>Conclusões</b>	<b>6</b>

# 1 Introdução e objetivos

Redes Neurais Artificiais são modelos matemáticos que simulam os neurônios biológicos e suas sinapses. Tais redes são capazes de aprender padrões, classificar e agrupar dados através de treinamento e ajustes matemáticos[1].

Neste relatório serão abordados três tipos de implementações de Redes Neurais: (i) Multi-Layer Perceptron; (ii) Radial Basis Function Network; e (iii) Self Organizing Maps.

Como as redes neurais podem ser utilizadas para classificar padrões, após um treinamento prévio, este trabalho tem como objetivo ajustar cada tipo de implementação para classificar o conjunto de dados Iris<sup>1</sup>, fazendo comparações entre suas acurácias e matrizes de confusão.

O presente trabalho está organizado e apresentado da seguinte forma: na Seção 2 vê-se a configuração e detalhes de implementação de cada rede neural. A Seção 3 aborda a metodologia de comparação utilizada no desenvolvimento deste trabalho. Na Seção 4 é apresentado os resultados obtidos e por fim na Seção 5 estão as conclusões.

## 2 Configuração das Redes Neurais e estratégias para classificação

Para a execução das comparações entre as redes neurais para com o conjunto de dados Iris, foram definidas configurações e estratégias para classificação de cada arquitetura de rede, as quais são descritas nas subseções seguintes.

### 2.1 Multi-Layer Perceptron

Para a configuração da Multi-Layer Perceptron foram utilizados 4 neurônios na camada de entrada, sendo cada um para um atributo do conjunto de dados, 3 para a camada escondida e 3 para saída, um para cada classe do problema; com taxa de aprendizagem de 0.3 e com erro de  $1e^{-2}$ . A configuração é ilustrada na Tabela 1.

Tabela 1: Configuração da Rede Neural Multi-Layer Perceptron

Qtd neurônios por camada				
Entrada	Escondida	Saída	Taxa de Aprendizagem	Erro
4	3	3	0.3	$1e^{-2}$

---

<sup>1</sup><https://archive.ics.uci.edu/ml/datasets/iris>

De acordo com a Tabela 1 pode-se observar que há 3 neurônios de saída, atribuindo cada um para uma determina da classe. Após o treinamento, com os pesos calculados e atualizados, fornecendo uma entrada para a rede, inicia-se a fase de propagação e posterior retropropagação. A classe atribuída ao exemplo é com base no neurônio de saída que retorna a maior probabilidade.

Exemplo: Com uma entrada 4.9, 2.5, 4.5 e 1.7 (classe virginica) na rede e obtendo como saída nos neurônios os valores 0.25, 0.05 e 0.7, respectivamente, atribui-se para essa entrada a classe virginica, pois definiu-se que o primeiro neurônio da saída classifica como setosa, o segundo neurônio como versicolor e o terceiro como virginica.

## 2.2 Radial Basis Function Network

Na configuração da Radial Basis Function Network utilizou-se 4 neurônios na camada de entrada, um para cada atributo do conjunto de dados, 5 neurônios na camada oculta ( $K=5$ ) e 1 neurônio na saída. Sendo o valor de  $\gamma = 1.0$  e a função gaussiana para  $\phi$ . A configuração pode ser observada na Tabela 2.

Tabela 2: Configuração da Radial Basis Function Network

Qtd neurônios por camada				
Entrada	Escondida (K)	Saída	Gama	RBF
4	5	1	1.0	Gaussiana

De acordo com a Tabela 2, optou-se por utilizar apenas um neurônio na camada de saída. A lógica para predição de uma das três classes dispostas no conjunto de dados (setosa, versicolor e virginica) utilizando-se apenas um neurônio deu-se da seguinte maneira: Primeiramente transformou-se cada classe em um fator numérico, sendo 1 para setosa, 2 para versicolor e 3 para virginica. Tendo que a saída da rede RBF é linear e utilizando apenas um neurônio, o valor obtido por este é comparado por proximidade ao fator numérico atribuído para a respectiva classe.

Exemplo: Considerando que a saída foi de 1.9 e o valor da classe versicolor é 2, então a saída da rede está mais próxima da classe 2, ou seja, a versicolor. Ou ainda, se a saída da rede for 3.2, por aproximação, a classe predita será da classe 3, a virginica.

## 2.3 Self Organizing Maps

E por fim para o SOM foi utilizada uma taxa de aprendizagem inicial de 0.1 com uma vizinhança igual a 8, sendo os valores de  $\text{Tau1}(\tau_1)$  e  $\text{Tau2}(\tau_2)$

$1000/\log(8)$  e 1000 respectivamente, o valor de 30 para o número máximo de interações e 8 para a dimensão do mapa. A configuração geral do SOM está descrita na Tabela 3.

Tabela 3: Configuração do Self Organizing Maps

Taxa de aprend. inicial	Largura Vizinhança	$\tau_1$	$\tau_2$	Nro. max interações	Dimensão Mapa
0.1	8	$1000/\log(8)$	1000	30	8

Sabendo-se que o SOM organiza dimensionalmente dados de entrada em *clusters* e o objetivo deste trabalho é usar o modelo para classificar uma entrada ou conjunto de entradas, tomou-se da seguinte estratégia para essa possível classificação: (i) o modelo SOM retorna, depois de treinado, um vetor de pesos e um mapa; (ii) o mapa permite a visualização da organização em clusters de acordo com as características de entrada; (iii) para que o SOM possa funcionar como um classificador, criou-se uma função que recebe o mapa com seus vetores de pesos já treinados e uma entrada, retornando a predição da classe; e (iv) a predição é feita de maneira análoga ao treinamento, porém manteve-se os valores dos pesos pós treinamento, como dito no item (iii), e fornecendo uma entrada é possível encontrar o neurônio de menor distância, verificando-se no mapa qual a classe correspondente a posição deste neurônio, deste modo tendo a predição final.

Exemplo: Avaliando que após o treinamento a classe setosa ficou, hipoteticamente, agrupada nas posições [1][1], [1][2], [1][3], [2][1], [2][2], [2][3] e a entrada fornecida ficou o mais próximo dessas posições, então será classificada como setosa.

### 3 Metodologia de Comparação

Considerando o objetivo de comparar as classificações dadas por cada rede neural para o conjunto de dados Iris, utilizou-se a matriz de confusão, calculando acurácia entre a classificação desejada e a classificação obtida. Além disso, implementamos a estratégia de *K-Fold Cross Validation*, com  $K=10$ .

Nas subseções seguintes serão ilustradas as matrizes de confusão obtidas após o treinamento dos três tipos de redes neurais abordadas por este trabalho.

## 4 Resultados

Para sumarização dos testes gerou-se matrizes de confusão, obtendo a acurácia média de cada rede neural. No Listing 1 é ilustrada a matriz de confusão da rede *Multi-Layer Perceptron*.

```

1 Confusion Matrix and Statistics
2
3           Reference
4 Prediction  setosa versicolor virginica
5   setosa      50           0           0
6 versicolor    0          47           2
7   virginica    0           3          48
8
9 Overall Statistics
10
11           Accuracy : 0.9667
12           95% CI : (0.9239, 0.9891)
13   No Information Rate : 0.3333
14   P-Value [Acc > NIR] : < 2.2e-16
15
16           Kappa : 0.95
17   Mcnemar's Test P-Value : NA
18
19 Statistics by Class:
20
21           Class: setosa Class: versicolor Class: virginica
22 Sensitivity           1.0000           0.9400           0.9600
23 Specificity           1.0000           0.9800           0.9700
24 Pos Pred Value        1.0000           0.9592           0.9412
25 Neg Pred Value        1.0000           0.9703           0.9798
26 Prevalence            0.3333           0.3333           0.3333
27 Detection Rate         0.3333           0.3133           0.3200
28 Detection Prevalence   0.3333           0.3267           0.3400
29 Balanced Accuracy      1.0000           0.9600           0.9650

```

Listing 1: Matriz de Confusão MLP

De acordo com o Listing 1, a rede MLP teve uma acurácia média próxima de 97%, conforme pode ser visto na linha 11. Já a rede *Radial Basis Function* teve uma acurácia média próxima de 79% de acordo com o Listing 2.

```

1 Confusion Matrix and Statistics
2
3           Reference
4 Prediction  setosa versicolor virginica
5   setosa      46           0           0
6 versicolor    4          41          19
7   virginica    0           9          31
8
9 Overall Statistics
10
11           Accuracy : 0.7867
12           95% CI : (0.7124, 0.8493)
13   No Information Rate : 0.3333
14   P-Value [Acc > NIR] : < 2.2e-16
15
16           Kappa : 0.68
17   Mcnemar's Test P-Value : NA
18
19 Statistics by Class:
20
21           Class: setosa Class: versicolor Class: virginica
22 Sensitivity           0.9200           0.8200           0.6200
23 Specificity           1.0000           0.7700           0.9100
24 Pos Pred Value        1.0000           0.6406           0.7750
25 Neg Pred Value        0.9615           0.8953           0.8273
26 Prevalence            0.3333           0.3333           0.3333
27 Detection Rate         0.3067           0.2733           0.2067
28 Detection Prevalence   0.3067           0.4267           0.2667
29 Balanced Accuracy      0.9600           0.7950           0.7650

```

Listing 2: Matriz de Confusão RBF

E por fim o Listing 3 mostra uma acurácia média de 44% para *Self Organizing Maps*.

```

1 Confusion Matrix and Statistics
2
3           Reference
4 Prediction  setosa versicolor virginica
5   setosa      25          9         12
6 versicolor   11         24         21
7   virginica   14         17         17
8
9 Overall Statistics
10
11           Accuracy : 0.44
12           95% CI : (0.3591, 0.5233)
13           No Information Rate : 0.3333
14           P-Value [Acc > NIR] : 0.004184
15
16           Kappa : 0.16
17   Mcnemar's Test P-Value : 0.855461
18
19 Statistics by Class:
20
21           Class: setosa Class: versicolor Class: virginica
22 Sensitivity           0.5000           0.4800           0.3400
23 Specificity           0.7900           0.6800           0.6900
24 Pos Pred Value        0.5435           0.4286           0.3542
25 Neg Pred Value        0.7596           0.7234           0.6765
26 Prevalence            0.3333           0.3333           0.3333
27 Detection Rate         0.1667           0.1600           0.1133
28 Detection Prevalence   0.3067           0.3733           0.3200
29 Balanced Accuracy      0.6450           0.5800           0.5150

```

Listing 3: Matriz de Confusão SOM

Em conformidade com as matrizes de confusão a Tabela 4 expõe uma visão geral da acurácia média de cada rede neural.

Tabela 4: Acurácia média obtidas

Rede	Acurácia Média
MLP	0.9666667 + / - 0.04714045
RBF	0.7866667 + / - 0.156505
SOM	0.44 + / - 0.2089362

## 5 Conclusões

Neste trabalho foram abordadas três tipos de redes neurais: A Multi-Layer Perceptron (MLP), Radial Basis Function Network (RBF) e Self Organizing Maps (SOM). A MLP torna possível classificar dados onde é necessário mais de uma reta para separá-los, ou seja, mais de uma classe. A RBF, onde os dados não são linearmente separáveis, é capaz de aumentar a dimensionalidade do conjunto de dados através de funções de ativação base, que neste trabalho foi utilizado a gaussiana, transformando os dados em linearmente separáveis e por fim o SOM, este que apresenta agrupamento das informações em clusters em um mapa, visualmente, onde cada cluster pode ser uma classe

ou uma característica específica. Com os resultados obtidos através das matrizes de confusão, descritas na seção 4, a MLP mostrou-se mais precisa na classificação do conjunto de dados Iris.

Este trabalho também mostrou que é possível ajustar certas redes neurais com estratégias de classificação com o propósito de se adequar as entradas fornecidas e resultando em valores que são plausíveis de interpretação para classificação.



## Referências

- [1] Z. L. Kovács. *Redes neurais artificiais*. Editora Livraria da Física, 2002.

ℒ<sub>T</sub>EX