

# Redes Auto-Organizáveis: Mapas de Kohonen

Gustavo Scaloni Vendramini  
Guilherme José Henrique  
Sean Carlisto de Alvarenga  
Vinícius Fernandes de Jesus

10 de setembro de 2013

## Resumo

Este artigo apresenta uma implementação de um modelo de rede neural auto-organizável conhecida como mapas auto-organizáveis de Kohonen ou redes SOM (do inglês *Self-Organizing Maps*).

## 1 Introdução

## 2 Redes SOM

### 2.1 Treinamento

A rede SOM utiliza um algoritmo de aprendizado competitivo e não-supervisionado para o aprendizado da rede. Um padrão de entrada é apresentado a rede, um neurônio vence e inicia a atualização dos pesos do neurônio vencedor e de seus vizinhos (até um raio de vizinhança). Isto repete para cada nova entrada e a taxa de aprendizado e o raio de vizinhança são decrementados durante o processo.[Braga]

Atualização dos pesos do neurônio vencedor e de seus vizinhos:

$$w_{ji}(t+1) = \begin{cases} w_{ji}(t) + \eta(t)(x_i - w_{ji}(t)) & \text{se } j \in \Lambda(t) \\ w_{ji}(t) & \text{caso contrário} \end{cases}$$

Onde  $w_{ji}$  é o peso entre os neurônios  $i$  e  $j$ ,  $\eta(t)$  é a taxa de aprendizado e  $\Lambda$  é a vizinhança.

```
1: Inicializar pesos e parâmetros;
2: repita
3:   para todo padrão de treinamento faça
4:     Definir neurônio vencedor;
5:     Atualizar os pesos deste neurônio e de seus vizinhos;
6:     se o número do ciclo for múltiplo de N então
7:       Então reduzir a taxa de aprendizado;
8:     fim-se
9:   fim-para
10: até que mapa de características não mudar
```

## 3 Métodos

## 4 Resultados

A rede neural auto-organizável implementada nesse trabalho, utilizou o algoritmo Mapa de Kohonen, desenvolvido por Teuvo Kohonen em 1982 [1]. A implementação deste trabalho pode ser vista em <https://github.com/QSF/ia-kohonen>.

Para os resultados práticos, a rede foi treinada com um conjunto de 1728 exemplos de dados reais, fornecidos pelo site *UCI Machine Learning Repository*<sup>1</sup>, variando valores da taxa de aprendizado, pesos iniciais, raio da vizinhança e quantidade de neurônios na rede. O conjunto de exemplos escolhido foi o *Car Evaluation*<sup>2</sup>, onde dados os atributos preço de compra do veículo, custo da manutenção, quantidade de portas do veículo, número de passageiros, tamanho do bagageiro e segurança estimada do veículo, avalia se o veículo é aceitável, inaceitável, bom ou muito bom.

Uma vez que o algoritmo Mapa de Kohonen organiza ou agrupa dimensionalmente os dados, o conjunto de exemplos foi adaptado, apresentando somente três dimensões (preço de compra do veículo, custo da manutenção e segurança estimada do veículo). O primeiro e segundo atributo aceitos os valores low, med, high e vhigh, enquanto que o útil aceita os valores low, med, high.

Com o conjunto de dados coletados, algumas entradas foram testadas para verificar a modificação final da rede. Esses testes serão exibidos a seguir, sendo dividido pela quantidade de neurônios na camada de saída (16, 20, 25 e 36 neurônios).

### 4.1 16 neurônios

Nos conjuntos de teste apresentados por essa seção, a rede possuía 16 neurônios na camada de saída, sendo que sua topologia é da forma de uma grade 4x4.

Para realizar os testes na rede mencionada, foi variado o valor da taxa de aprendizagem e dos pesos iniciais, com raio fixo em 0.

Quatro testes foram executados, onde a configuração da rede para cada caso é ilustrada em 1.

Tabela 1: Configurações da Rede com 16 Neurônios na Saída

Teste	Taxa de Aprendizagem	Pesos Iniciais
1	0.1	todos 1
2	0.1	Random
3	0.3	todos 1
4	0.3	Random

As figuras 1, 2, 3 e 4 representam a rede após a execução dos testes 1, 2, 3 e 4; respectivamente.

### 4.2 20 neurônios

Nesta seção, a rede possuía 20 neurônios na camada de saída, sendo uma grade 4x5.

<sup>1</sup><http://archive.ics.uci.edu/ml/>

<sup>2</sup><http://archive.ics.uci.edu/ml/datasets/Car+Evaluation>

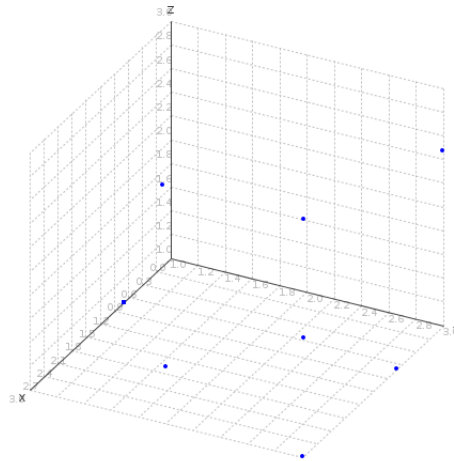


Figura 1: 16 Neurônios - Teste 1

Para realizar os testes na rede mencionada, novamente foi variado o valor da taxa de aprendizagem e dos pesos iniciais, mas com raio fixo em 1.

Também foram executados quatro testes que tem sua configuração ilustrada na tabela 2.

Tabela 2: Configurações da Rede com 20 Neurônios na Saída

Teste	Taxa de Aprendizagem	Pesos Iniciais
1	0.5	todos 1
2	0.5	Random
3	0.75	todos 1
4	0.75	Random

Os testes 1, 2, 3 e 4 são representados pelas imagens 5, 6, 7 e 8; respectivamente.

### 4.3 25 neurônios

Os gráficos a seguir mostram a plotação dos pontos após a execução do algoritmo para alguns valores de parâmetros e para uma rede com 25 neurônios. Apesar de não se poder afirmar se o algoritmo convergiu, vemos que aumentando o raio de vizinhança, o algoritmo consegue “especializar” melhor o conjunto. O efeito da taxa de aprendizado está sobre a velocidade na qual o algoritmo irá convergir. Ela geralmente é dada empiricamente, assim como os pesos iniciais, que segundo

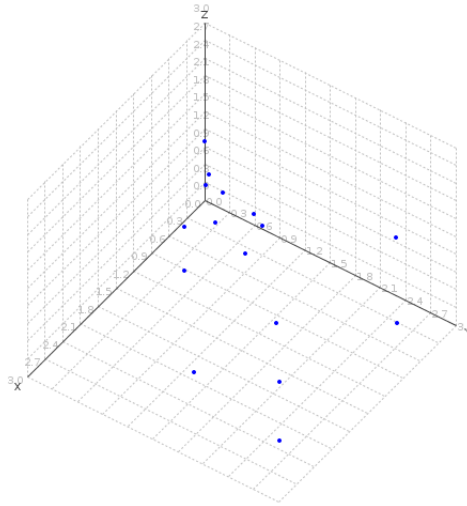


Figura 2: 16 Neurônios - Teste 2

[1], se forem inicializados com valores iguais, faz o algoritmo convergir mais rapidamente.

#### 4.4 36 neurônios

Os gráficos a seguir mostram a plotação dos pontos após a execução do algoritmo para alguns valores de parâmetros e para uma rede com 36 neurônios.

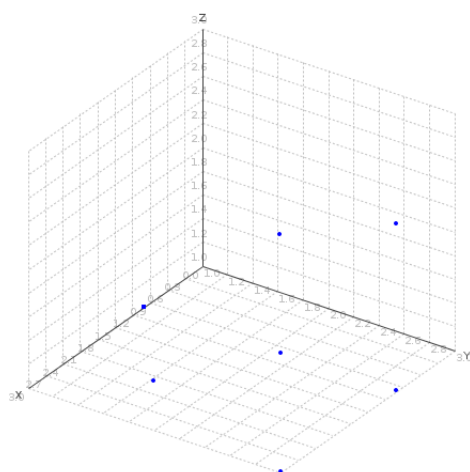


Figura 3: 16 Neurônios - Teste 3

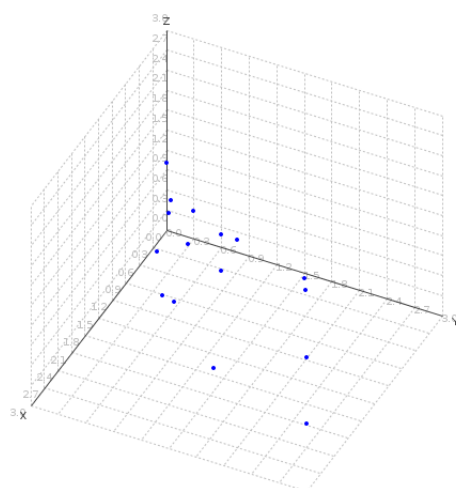


Figura 4: 16 Neurônios - Teste 4

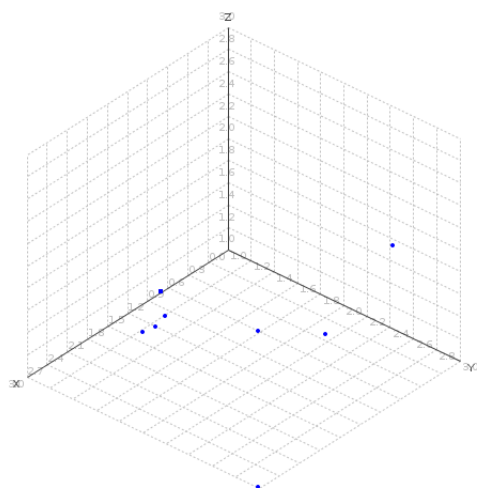


Figura 5: 20 Neurônios - Teste 1

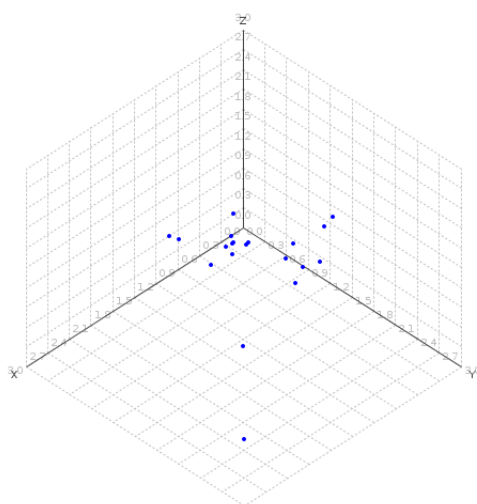


Figura 6: 20 Neurônios - Teste 2

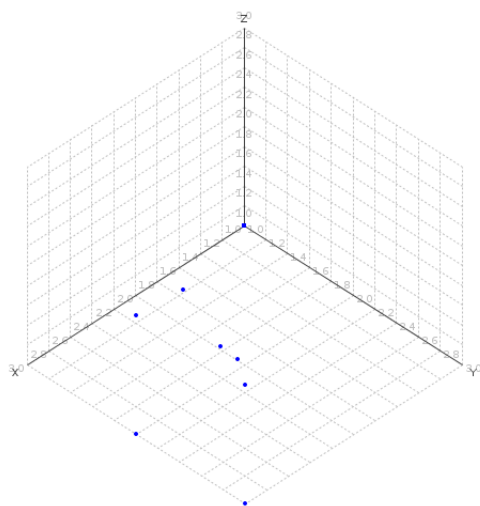


Figura 7: 20 Neurônios - Teste 3

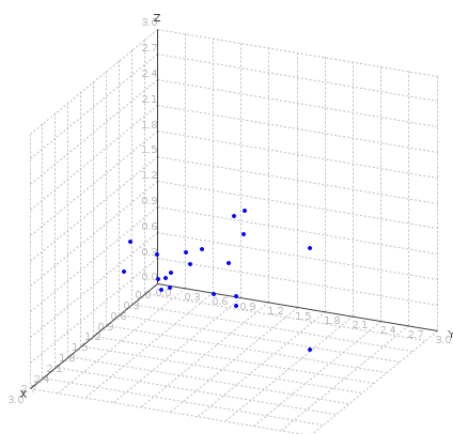


Figura 8: 20 Neurônios - Teste 4



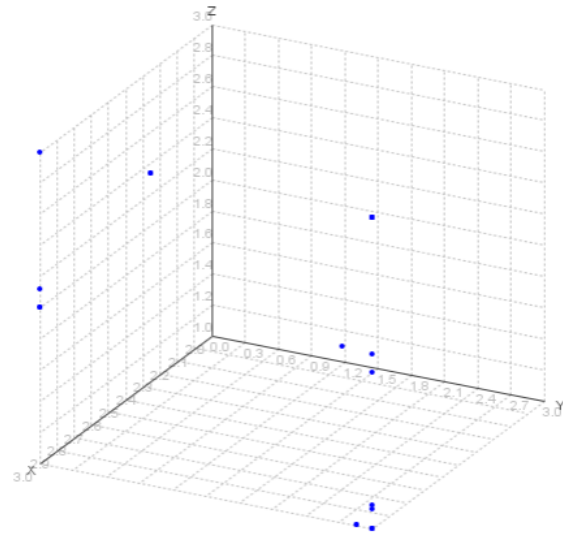


Figura 9: Taxa de aprendizado: 0.8662815917277779; Raio: 1; Pesos iniciais: A

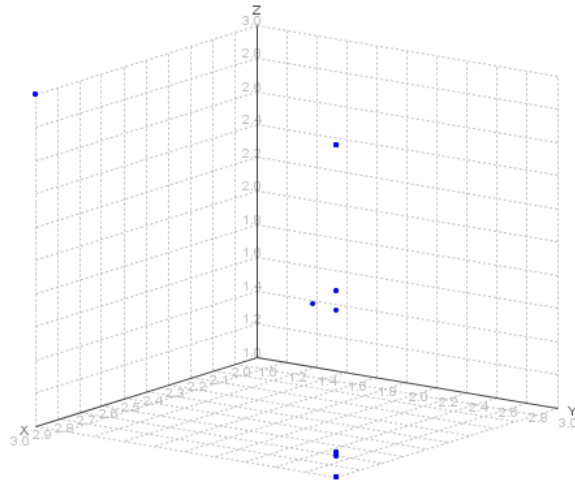


Figura 10: Taxa de aprendizado: 0.8662815917277779; Raio: 2; Pesos iniciais: A

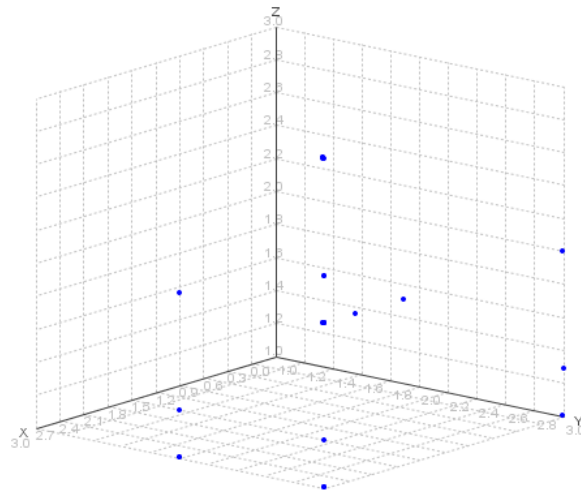


Figura 11: Taxa de aprendizado: 0.5980629675758204; Raio: 1; Pesos iniciais: B

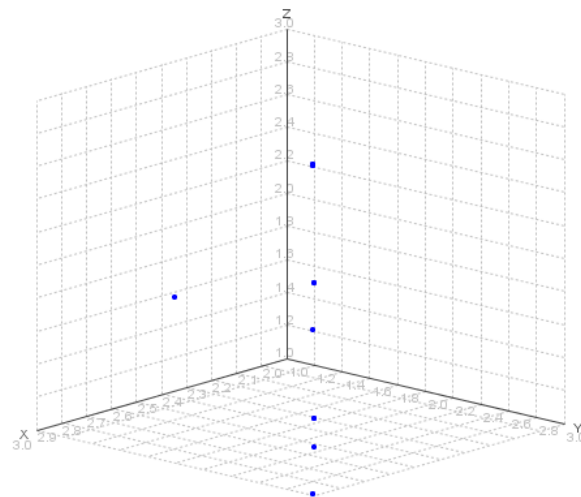


Figura 12: Taxa de aprendizado: 0.5980629675758204; Raio: 2; Pesos iniciais: B

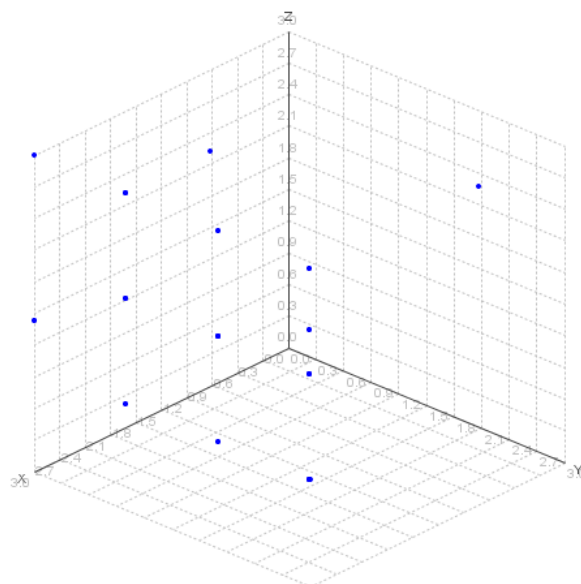
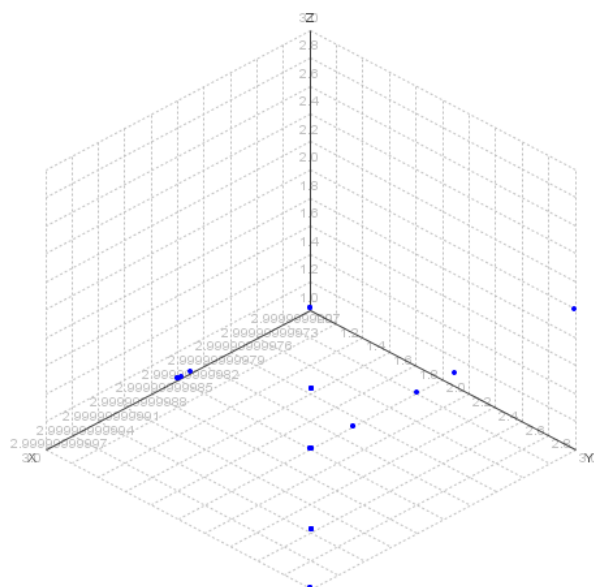


Figura 14: Taxa de aprendizado: 0.2625656059885696; Raio: 2; Pesos iniciais:  
D



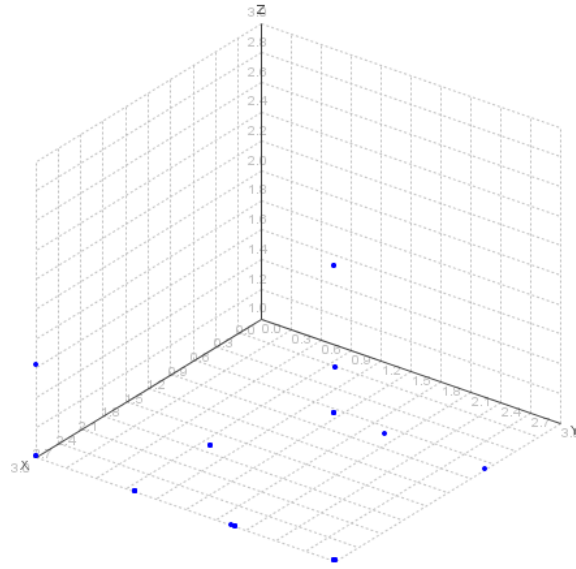


Figura 15: Taxa de aprendizado: 0.5517820314689292; Raio: 1; Pesos iniciais: E

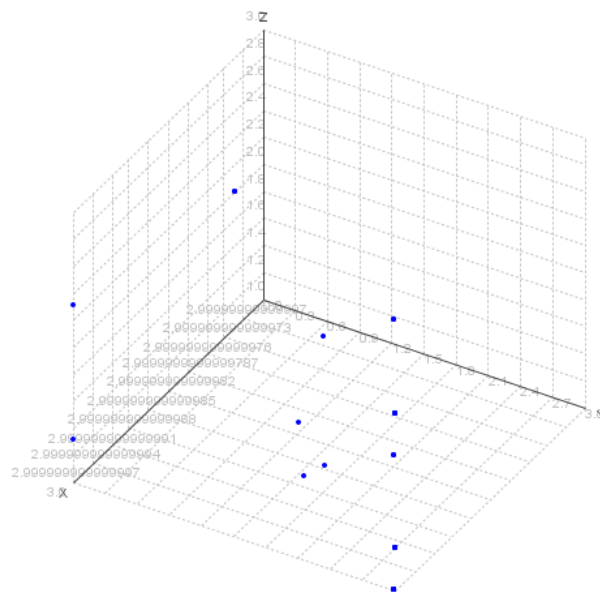


Figura 16: Taxa de aprendizado: 0.5517820314689292; Raio: 2; Pesos iniciais: E

## 5 Conclusão

### Referências

- [1] Teuvo Kohonen. *Self-Organizing Maps*. Springer, 2000.