Clusterizador de sementes utilizando mapas de Kohonen

Felipe Chabatura Neto, João Paulo K. Castilho, Leonardo Tironi Fassini

[1]Departamento de Ciência da Computação - Universidade Federal da Fronteira Sul (UFFS)
Chapecó – SC – Brasil

{felipechabat, joao.pkc, leehtironi}@gmail.com

Resumo. *Neste trabalho, utiliza-se uma rede de Kohonen, uma rede neural de aprendizado não-supervisionado, para classificação de dados de sementes em três tipos distintos. O trabalho se inicia com uma introdução sobre o método de classificação e depois prossegue descrevendo a abordagem tomada para resolução do problema. Por fim, descreve-se o processo de testes e os resultados obtidos são debatidos.*

Abstract. *The present work utilize a Kohonen network, a unsupervised learning neural network, to classify data about seeds in three different types. The work begins with an introduction about the classification method and after proceeds describing the chosen approach to solve the problem. After that, the test process is described and the results are discussed.*

1. Introdução

O presente trabalho utiliza o método de aprendizado não supervisionado de mapas de *Kohonen* para resolver o problema de classificação de três tipos diferentes de sementes. Inicialmente um estudo sobre o método foi realizado a fim de obter-se o conhecimento necessário para sua aplicação na resolução do problema.

Posteriormente, o método foi implementado na linguagem de programação *python* e então aplicado na classificação dos dados de sementes provenientes de um *dataset*. Por fim, foram comparados os resultados obtidos utilizando diferentes tamanhos para a topologia da *rede neural*.

1.1. Mapas auto-organizáveis (SOM)

Um mapa de *Kohonen* é um tipo de rede neural de aprendizado não-supervisionado, tendo como propósito agrupar os dados da entrada em grupos (*clusters*). A rede possui duas camadas: a camada de entrada e a camada de agrupamento, que serve como uma camada de saída e geralmente é organizada em forma de *grid*. Cada nó de entrada é conectado a todos os nós da camada de agrupamento (Figura 1) [Coppin 2010].

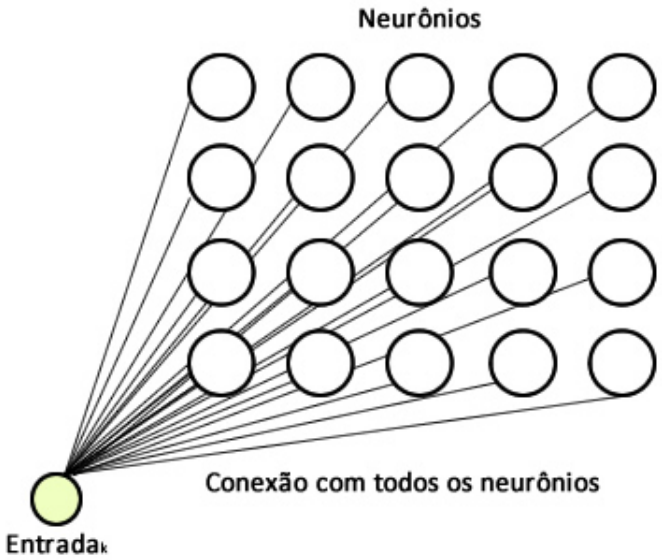


Figura 1. Imagem de um *grid* de um mapa de Kohonen. [Pacheco 2018]

O método de agrupamento é dividido em duas etapas. A primeira, chamada de etapa competitiva, utiliza o algoritmo *vencedor-leva-tudo* para selecionar o neurônio mais apto a fornecer a saída em resposta a uma entrada. A segunda, chamada de etapa cooperativa, consiste do ajuste dos pesos do neurônio da rede, de acordo com a proximidade em relação ao neurônio vencedor.

1.2. *Dataset* utilizado

O *dataset* utilizado consiste em sete atributos e um rótulo (classificação da semente), sendo que o último foi usado somente na avaliação. De acordo com [UCI 2018], os atributos são:

1. Área da semente (A).
2. Perímetro da semente (P).
3. Compacidade da semente, dada pela fórmula $C = \frac{4\pi A}{P^2}$
4. Comprimento do núcleo da semente.
5. Largura do núcleo da semente.
6. Coeficiente de assimetria da semente.
7. Comprimento do sulco[1] do núcleo da semente.

2. Metodologia

Nesta seção serão apresentados os procedimentos metodológicos adotados na implementação do método de classificação.

2.1. Tratamento do *dataset*

Como o *dataset* está ordenado por tipo de semente, foi necessário aleatorizá-lo para então dividi-lo em 70% para treino e 30% para teste. Caso os dados não fossem aleatorizados, a entrada de treino teria majoritariamente um tipo de semente.

[1]Marca mais estreita que comprida e mais ou menos profunda, num material; fissura, ranhura.

2.2. Inicialização da rede

A topologia da camada de agrupamento é variável e controlada por parâmetros, a fim de testar diversas configurações e comparar os resultados. Os valores dos pesos de cada neurônio são inicializados com valores aleatórios uniformes (entre -1 e 1).

2.3. Treinamento

O treinamento ocorre por uma quantidade predeterminada de épocas, que é determinada através de um parâmetro. Cada época do treinamento é constituída por duas etapas que se repetem para cada entrada da porção do *dataset* que foi separada para o treinamento. Na primeira, a etapa competitiva, a entrada será comparada com cada neurônio (pela Equação 1[2]) e é declarado o vencedor o neurônio mais parecido com a entrada; i.e., o neurônio com a menor distância euclidiana.

$$d(a, b) = \sqrt{\sum_{i=1}^{k}(x_i - w_i)^2}. \tag{1}$$

Na segunda etapa, a cooperativa, os pesos dos neurônios da rede serão atualizados de acordo com a proximidade de cada neurônio em relação ao neurônio vencedor. Com isso, os neurônios próximos se ativarão para entradas mais similares a que ativou o neurônio vencedor. O objetivo desta etapa é criar áreas de ativação para sementes de mesmo tipo.

Como o neurônio vencedor deve influenciar os neurônios ao seu redor, é necessária uma equação para delimitar a influência dele para todos os outros. Ao mesmo tempo, essa influência deve ser menor a medida que épocas vão passando. Por isso a Equação 2

$$\sigma_t = \sigma_0 \times e^{-\frac{t}{\tau}} \tag{2}$$

define o valor de influência que o neurônio receberá, sendo $\sigma_0 = max(m, n)$, i.e. o máximo entre a quantidade de linhas e a quantidade de colunas, t a época atual e τ é uma constante definida por $\frac{nIter}{\log \sigma_0}$ em que $nIter$ é a quantidade de épocas que serão rodadas o algoritmo.

Tendo σ_t para controlar a área de influência, a Equação 3

$$h_t = e^{\frac{d(n_i, n_v)}{2\sigma(t)}} \tag{3}$$

será usada pra contribuir à alteração do neurônio atual. Sendo n_i o neurônio atual e n_v o neurônio vencedor, então $d(n_i, n_v)$ é a distância[3] entre eles.

Por fim, calcula-se a taxa de aprendizado para a época atual, dada pela Equação 4.

$$\alpha_t = \alpha_0 \times e^{-\frac{t}{\tau}} \tag{4}$$

Essa taxa de aprendizado decairá ao longo do tempo, sendo α_0 a taxa de aprendizado inicial.

[2]Na Equação 1 x_i é a *i*-ésima *feature* da entrada e w_i é o *i*-ésimo peso do neurônio.

[3]Essa é a distância topológica, ou seja, distância entre os dois neurônios no grid.

Por fim, a fórmula para atualização dos pesos de cada neurônio do *grid* é dada pela Equação 5.

$$W_k(t+1) = W_k(t) + \alpha_t \times h_t \times d(X_i, W_k) \tag{5}$$

Sendo $W_k(t)$ os pesos atuais do neurônio k, e $d(X_i, W_k)$ é calculada pela expressão $W_k - X_k$.

2.4. Testes

Após rodar o algoritmo com diferentes topologias (e.g., [3,1], [3,2], ... , [10x10]) e com 100 épocas, é construída uma matriz de ocorrências. Essa matriz é uma matriz $n * m * 3$, onde n é o numero de linhas, m é o número de colunas e 3 é porque existem 3 tipos de sementes. Durante o período de teste, se um neurônio ativar para uma entrada X_n, então a posição $i \times j \times t$ na matriz de ocorrências (onde i e j são, respectivamente, a linha e coluna da posição do neurônio e t é o tipo da semente da entrada) será incrementada.

Ao final do período de testes, cada posição da matriz de ocorrências terá a quantidade de cada tipo de semente da entrada que fez o neurônio correspondente ativar. Posteriormente, esses dados foram usados para realizar os cálculos de precisão. Ademais, uma imagem foi gerada com base nessa matriz de ocorrências, atribuindo uma cor a cada neurônio, sendo a cor vermelho para sementes do tipo 1, verde para sementes do tipo 2, azul para sementes do tipo 3 e preta se o neurônio não foi ativado nenhuma vez. Entretanto, existe a possibilidade de um neurônio ser ativado para mais de um tipo de sementes. Neste caso, o neurônio ativado terá uma cor conforme a quantidade de sementes de cada tipo que ativou aquele neurônio.

Para decidir qual tipo de semente representaria um neurônio que ativou para mais de um tipo, foi usado como critério de desempate qual tipo de semente mais ativou ele, i.e., se três sementes de tipo 1 e duas sementes de tipo 2 ativaram o neurônio, então ele será considerado de tipo 1, colorido de acordo com a quantidade de sementes de cada tipo e todas as sementes de tipo 2 que o ativaram são consideradas erro. Se ocorrer um empate ao definir o tipo de semente que mais vezes ativou um neurônio, então o neurônio é dito como errado, e todas as suas ativações são consideradas como erros no cálculo da precisão.

3. Resultados

A seguir serão mostrados vários resultados obtidos, cada um com uma figura que representa a matriz de ocorrências (onde cada neurônio é um quadrado) e a taxa de acerto para cada tipo de semente:

3.1. Topologia 3×1

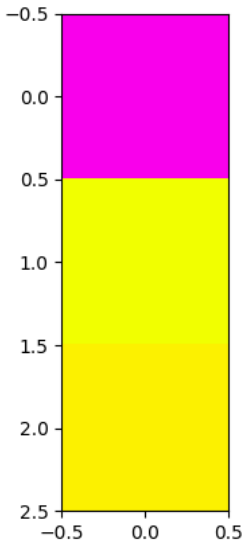


Figura 2. Imagem do gráfico da matriz de ocorrências de tamanho 3×1

- Taxa de acerto para sementes tipo 1: 57.69%.
- Taxa de acerto para sementes tipo 2: 93.75%.
- Taxa de acerto para sementes tipo 3: 100.00%.

3.2. Topologia 3×2

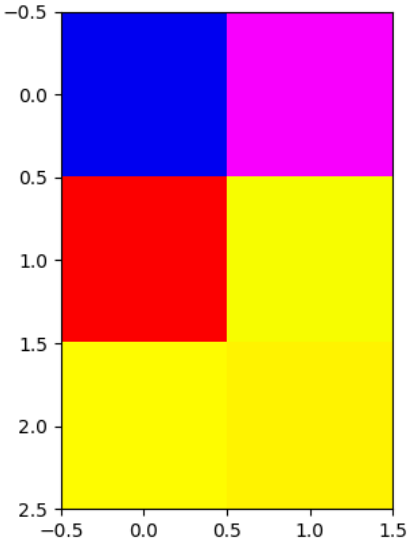
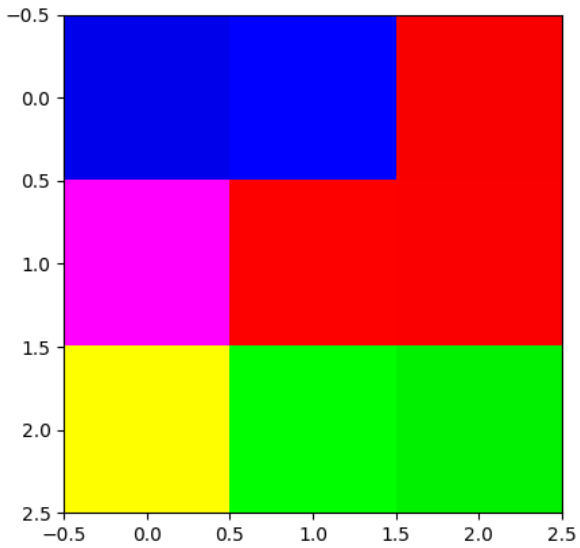


Figura 3. Imagem do gráfico da matriz de ocorrências de tamanho 3×2

- Taxa de acerto para sementes tipo 1: 87.50%.

- Taxa de acerto para sementes tipo 2: 85.00%.
- Taxa de acerto para sementes tipo 3: 84.21%.

3.3. Topologia 3×3



.

Figura 4. Imagem do gráfico da matriz de ocorrências de tamanho 3×3

- Taxa de acerto para sementes tipo 1: 88.89%.
- Taxa de acerto para sementes tipo 2: 100.00%.
- Taxa de acerto para sementes tipo 3: 100.00%.

3.4. Topologia 3×4

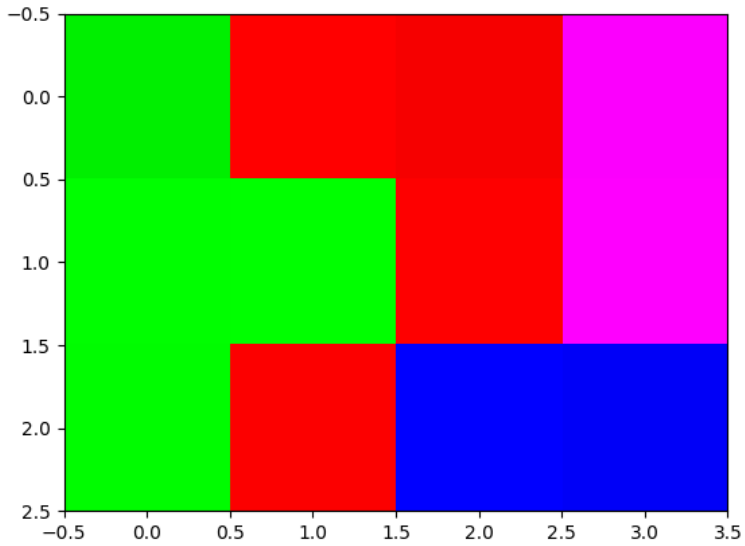


Figura 5. Imagem do gráfico da matriz de ocorrências de tamanho 3×4

- Taxa de acerto para sementes tipo 1: 91.67%.
- Taxa de acerto para sementes tipo 2: 100.00%.
- Taxa de acerto para sementes tipo 3: 66.67%.

3.5. Topologia 4×4

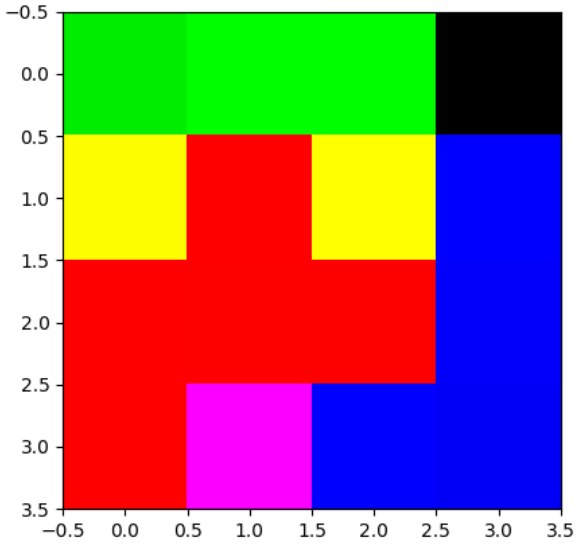


Figura 6. Imagem do gráfico da matriz de ocorrências de tamanho 4×4

- Taxa de acerto para sementes tipo 1: 88.24%.

- Taxa de acerto para sementes tipo 2: 96.30%.
- Taxa de acerto para sementes tipo 3: 94.74%.

3.6. Topologia 5×5

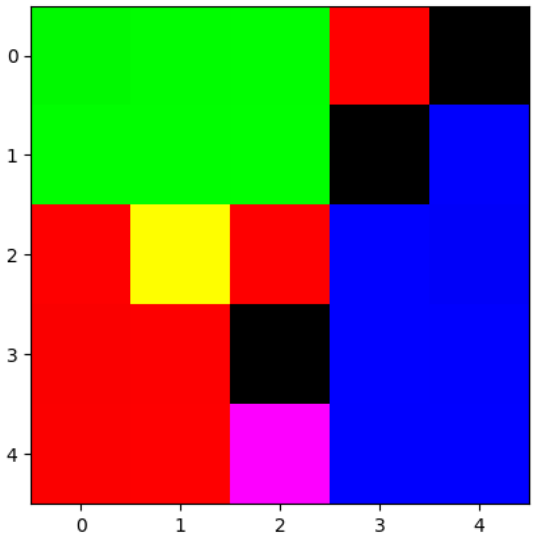


Figura 7. Imagem do gráfico da matriz de ocorrências de tamanho 5×5

- Taxa de acerto para sementes tipo 1: 92.31%.
- Taxa de acerto para sementes tipo 2: 89.47%.
- Taxa de acerto para sementes tipo 3: 94.44%.

3.7. Topologia 6×6

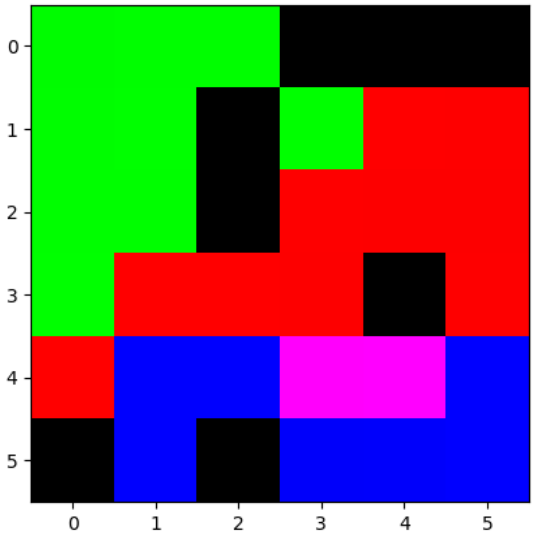


Figura 8. Imagem do gráfico da matriz de ocorrências de tamanho 6×6

- Taxa de acerto para sementes tipo 1: 90.48%.
- Taxa de acerto para sementes tipo 2: 100.00%.
- Taxa de acerto para sementes tipo 3: 95.83%.

3.8. Topologia 10×10

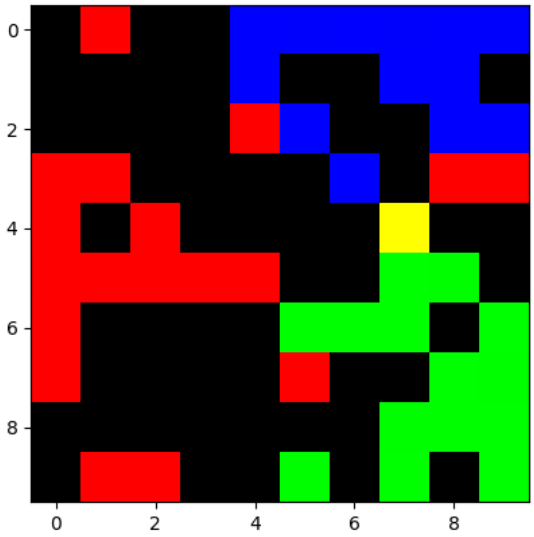


Figura 9. Imagem do gráfico da matriz de ocorrências de tamanho 10×10

- Taxa de acerto para sementes tipo 1: 95.00%.

- Taxa de acerto para sementes tipo 2: 95.45%.
- Taxa de acerto para sementes tipo 3: 100.00%.

3.9. Topologia 20×20

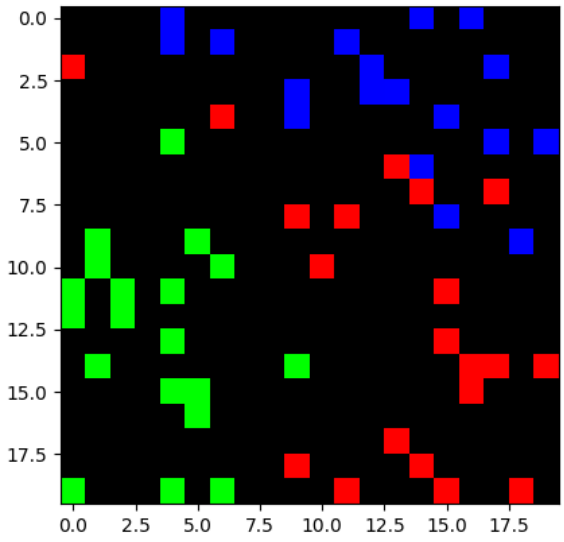


Figura 10. Imagem do gráfico da matriz de ocorrências de tamanho 20×20

- Taxa de acerto para sementes tipo 1: 100%.
- Taxa de acerto para sementes tipo 2: 100%.
- Taxa de acerto para sementes tipo 3: 100%.

3.10. Topologia 30×30

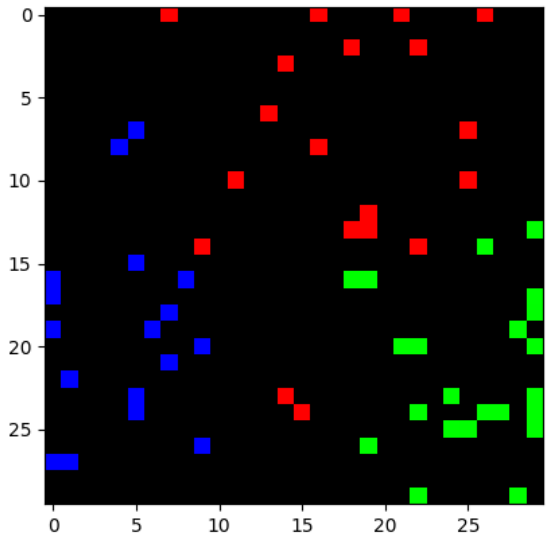


Figura 11. Imagem do gráfico da matriz de ocorrências de tamanho 30×30

- Taxa de acerto para sementes tipo 1: 100%.
- Taxa de acerto para sementes tipo 2: 100%.
- Taxa de acerto para sementes tipo 3: 100%.

3.11. Topologia 40×40

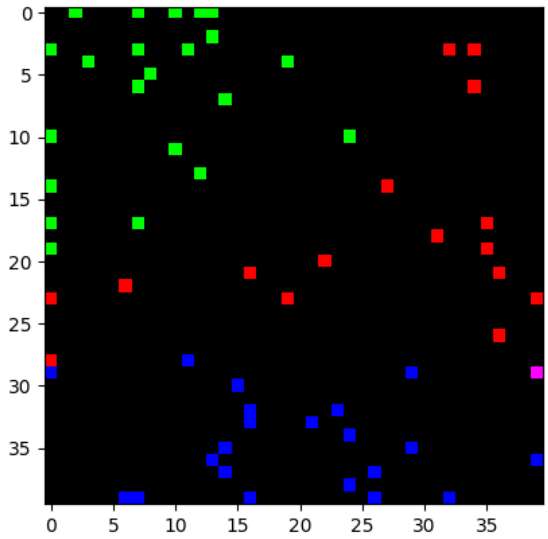


Figura 12. Imagem do gráfico da matriz de ocorrências de tamanho 40×40

- Taxa de acerto para sementes tipo 1: 94.44%.

- Taxa de acerto para sementes tipo 2: 100%.
- Taxa de acerto para sementes tipo 3: 95.45%.

4. Conclusões

Analisando os gráficos percebe-se que se aumentar a topologia, a taxa de acerto também aumenta. Entretanto, o tempo de processamento necessário para executar o método e a quantidade de neurônios não ativados também aumentam. Portanto, os *grids* de tamanhos 5×5, 6×6 ou 10×10 são considerados ideais, pois possuem uma taxa de acerto alta e não requerem muito tempo de processamento.

Referências

Coppin, B. (2010). Inteligência artificial/ben coppin; tradução e revisão técnica jorge duarte pires valério. *Rio de Janeiro: LTC*.

Pacheco, A. (2018). Mapas auto-organizáveis – som. Disponível em: ¡`http://www.computacaointeligente.com.br/algoritmos/mapas-auto-organizaveis-som/`¿. Acesso em: 22/06/2018.

UCI (2018). seeds data set. Disponível em: ¡`https://archive.ics.uci.edu/ml/datasets/seeds`¿. Acesso em: 21/06/2018.