

Redes neurais auto-organizáveis (SONN)

Manoel Fernando Tenório
Wei-Tsih Lee

Resumo para a matéria Redes neurais I.
Adriano Martins Moutinho

Estrutura

- ▶ Tem base em um processo iterativo de escolha e aproximação com base em funções do segundo grau.
- ▶ Cada neurônio possui apenas duas entradas e uma saída, com função de transferência de segunda ordem.
- ▶ A rede se modifica, tendo estrutura variável. Podendo se organizar de formas diferentes a cada treinamento.

Funcionamento do algoritmo

- ▶ O algoritmo escolhe aleatoriamente, duas variáveis de entrada e combina as mesmas usando uma equação de segundo grau:

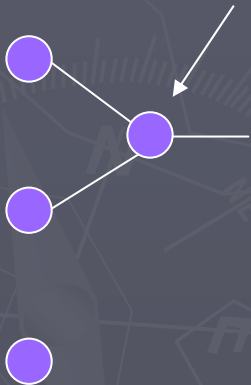
$$\hat{y} = a_0 + a_1x_1 + a_2x_2 + a_3x_1x_2 + a_4x_1^2 + a_5x_2^2$$

- ▶ Os coeficientes da equação são então calculados minimizando o erro em relação à saída, isto é feito segundo a equação acima.

Funcionamento do algoritmo (cont)

- ▶ Assim, a saída do novo neurônio é comparada com a saída desejada e uma função de conformidade é calculada. Caso a conformidade do novo neurônio seja melhor do que dos seus pais, o mesmo é aceito. Caso contrário, o algoritmo continua selecionando outras entradas.
- ▶ A saída do neurônio adicionado permanece e pode ser selecionada novamente pelo algoritmo

Melhores
coeficientes
calculados



Funcionamento do algoritmo (cont)

- ▶ O calculo do melhor conjunto de coeficientes é feita com base na equação abaixo:

$$[Ent] \times [Coef] = [Saida]$$

- ▶ Onde a matriz de entrada é construída da seguinte forma, sendo as entradas z e w , com n amostras:

$$[Ent] = \begin{bmatrix} 1 & 1 & 1 & 1 & \dots & 1 \\ w_1 & w_2 & w_3 & w_4 & \dots & w_N \\ z_1 & z_2 & z_3 & z_4 & \dots & z_N \\ w_1 \cdot z_1 & w_2 \cdot z_2 & w_3 \cdot z_3 & w_4 \cdot z_4 & \dots & w_N \cdot z_N \\ w_1^2 & w_2^2 & w_3^2 & w_4^2 & \dots & w_N^2 \\ z_1^2 & z_2^2 & z_3^2 & z_4^2 & \dots & z_N^2 \end{bmatrix}$$

Funcionamento do algoritmo (cont)

- ▶ Assim, isolando a matriz [coef]:

$$[Ent]^T \cdot [Ent] \cdot [Coef] = [Ent]^T \cdot [Saída]$$

$$([Ent]^T \cdot [Ent])^{-1} \cdot ([Ent]^T \times [Ent]) \cdot [Coef] = ([Ent]^T \cdot [Ent])^{-1} \cdot [Ent]^T \cdot [Saída]$$

$$[Coef] = ([Ent]^T \cdot [Ent])^{-1} \times [Ent]^T \cdot [Saída]$$

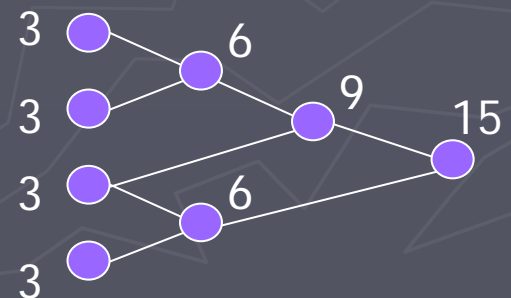
- ▶ Este processo escolhe os coeficientes que minimizam a diferença entre a saída e um polinômio de segundo grau gerado a partir de duas variáveis do sistema.

Funcionamento do algoritmo (cont)

- ▶ No entanto, é preciso também minimizar a estrutura utilizada na rede.
- ▶ Deseja-se não só minimizar o MSE mas também manter a estrutura o mais simples possível. Para isto, utiliza-se uma função de conformidade chamada SEC.
- ▶ Para tornar isto possível, as ligações buscam minimizar a função SEC abaixo, onde N é o número de padrões e k é gerado recursivamente somando os valores de k dos pais, sendo os nós primários inicializados com $k_0=3$ ou $k_0=1$:

$$SEC = \frac{1}{2} \cdot N \cdot \log(MSE) + \frac{1}{2} \cdot k \log N$$

Valores recursivos de K



Funcionamento do algoritmo (cont)

- ▶ Porém, é possível que uma determinada ligação entre duas variáveis não possua um SEC menor que o atual, mas isso poderia permitir uma melhor ligação no futuro.
- ▶ Para possibilitar melhores ligações futuras, utiliza-se um método baseado no processo de cura de metais.
- ▶ No processo de cura de metais, um determinado metal é esquentado até uma temperatura T e esfriado lentamente. Este processo permite que as ligações fortes entre as moléculas sejam formadas, embora inicialmente ocorram ligações fracas.

Funcionamento do algoritmo (cont)

- ▶ Seguindo o mesmo método, uma variável temperatura é inicializada com um valor alto, decrescendo com o passar do tempo. Dessa forma, enquanto a temperatura estiver alta, haverá maior probabilidade de aceitar um nó filho mesmo que não possua menor SEC que seus pais.
- ▶ Para isso, gera-se um número aleatório, que se for menor que $e^{(\text{delta}/\text{Temp})}$, o nó filho é aceito. Delta é a diferença entre o valor da menor energia e a energia do nó em questão. Temp é a temperatura do momento atual.

Funcionamento do algoritmo (cont)

- ▶ Como o número aleatório gerado está entre zero e um, caso a energia atual esteja próxima à energia mínima de todos os nós da rede, o nó filho tem muita chance de ser aceito. A probabilidade de aceitação também diminui conforme a temperatura decresce.
- ▶ Isso poderá possibilitar a criações de ligações mais fortes no futuro.
- ▶ Caso esta ligação não permita ligações mais fortes, a mesma será eliminada futuramente.

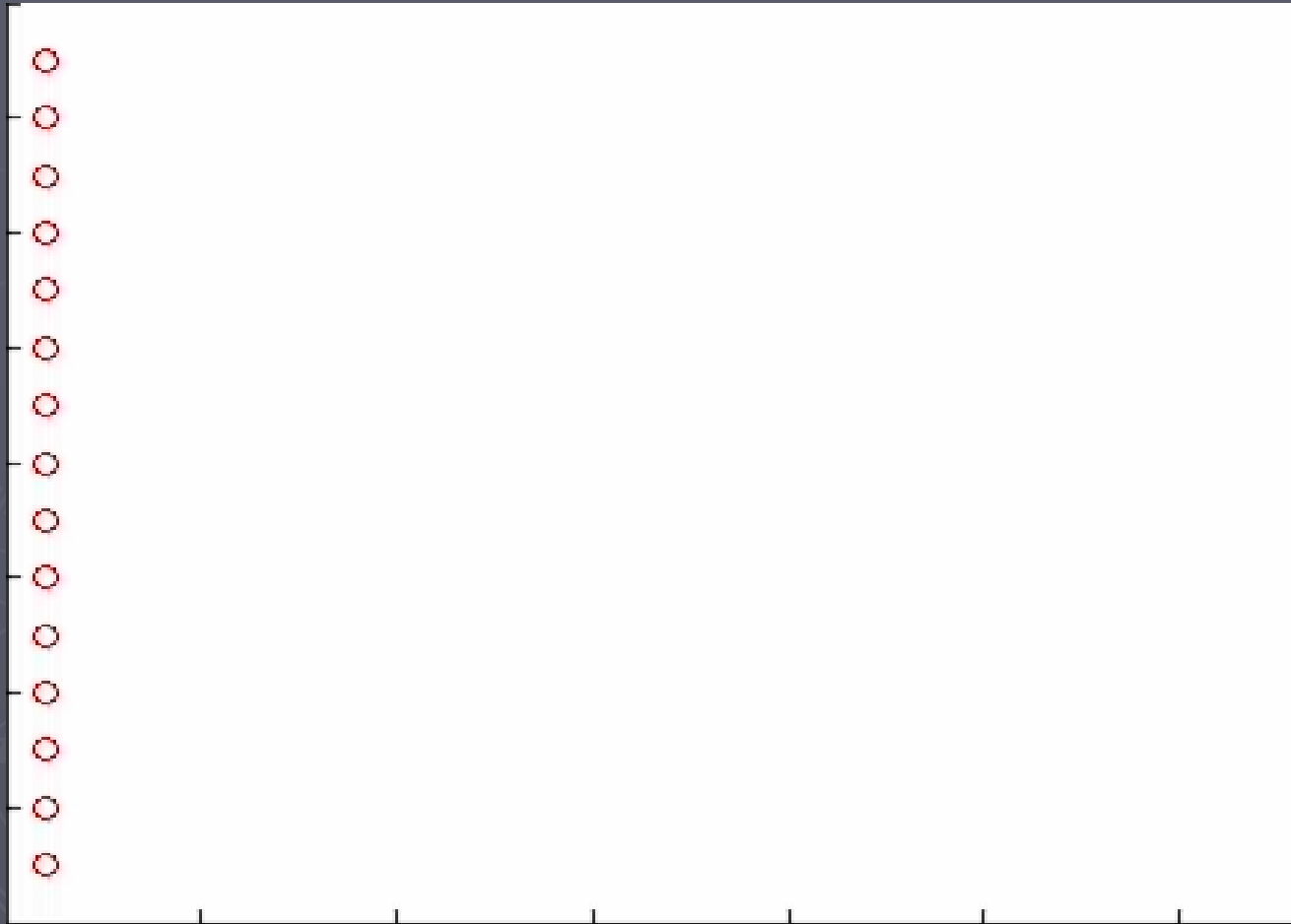
Resumo do algoritmo

- 1) Um neurônio para cada entrada é iniciado.
- 2) O algoritmo escolhe dois pares de entradas aleatoriamente.
- 3) Calcula-se os coeficientes para a aproximação polinomial de segunda ordem.
 - 1) Caso SEC desta aproximação seja melhor que os dois pais, este nó é adicionado.
 - 2) Caso não seja melhor, calcula-se um número aleatório, que se for menor que $e^{(\text{delta}/\text{Temp})}$ o nó é aceito mesmo assim.

Resumo do algoritmo (cont.)

- 4) Caso nenhum nó seja aceito, se $N/(N-1)$ nós já sejam tiverem sido escolhidos, baixa-se a temperatura de um valor estabelecido.
- 5) Caso nenhum nó seja aceito, volta-se para 2 sem baixar a temperatura.
- 6) Caso algum nó seja aceito por SEC, volta-se para 2 baixando-se a temperatura de um valor estabelecido.

Simulação da organização



IND: [2 3 4 7 11 18 19 31 33 34 35]

Resultados e testes

- ▶ Para testar a rede neural SONN, foi utilizado um problema de aproximação de funções cujo objetivo é estimar o fechamento da ação da Petrobrás no Ibovespa para três dias futuros, tendo-se o fechamento, máximo, mínimo, abertura e volume negociados nos dias anteriores.
- ▶ O modelo proposto SONN foi comparado com os tradicionais modelos de rede neural como MLP e a recorrente ELMAN.

Redes MLP e Elman

- ▶ Foram utilizados como entrada os fechamentos, volumes negociados, máximos, mínimos e aberturas de 7 dias anteriores como entrada da rede neural para as redes Elman e MLP.
- ▶ Utilizou-se, nos testes, apenas uma rede para prever os três dias seguintes. Foram treinadas 16 redes, com todas as combinações de número de neurônios abaixo, bem como variações no limiar de componentes principais.
- ▶ Arquiteturas: 50-3 , 30-3 , 50-30-3 , 50-40-3.
- ▶ Limiares PCA: 10^{-3} , 10^{-4} , 10^{-5} , 10^{-6} .

Resultados e testes (MLP)

MSE (MLP)		
Três dias	Dois dias	Um dia
0.048226	0.041824	0.019905
0.040887	0.033223	0.020943
0.054066	0.037466	0.03836
0.051453	0.04156	0.015988
0.042699	0.035581	0.024981
0.068195	0.044101	0.030539
0.055124	0.033703	0.028558
0.051975	0.030898	0.017736
0.061804	0.049583	0.029405
0.055558	0.040392	0.030278
Media (16 redes)	Media (16 redes)	Media (16 redes)
0.06572	0.04512	0.03259

Resultados e testes (ELMAN)

MSE (ELMAN)		
Três dias	Dois dias	Um dia
0.047621	0.029378	0.021429
0.044724	0.027754	0.021564
0.03596	0.025407	0.017005
0.045063	0.034718	0.023135
0.040045	0.034424	0.025156
0.053318	0.039425	0.023397
0.044726	0.030707	0.023217
0.04389	0.032547	0.017709
0.046389	0.030483	0.021519
0.041549	0.029807	0.020426
Media (16 redes)	Media (16 redes)	Media (16 redes)
0.048346	0.03489	0.026571

Resultados e testes SONN (1 dia)

<i>Dia</i>	<i>MSE</i>	<i>Entradas usadas</i>	<i>Tempo de convergência</i>	<i>Temp. inicial</i>	<i>Tx. Temp.</i>	<i>Error Goal</i>
1	0.007444	1 4 10 12 16 17 18 25	3min46s	10	1	0.0001
1	0.0089035	1 4 10 16 17	5s	10	6	1.00E-05
1	0.008685	3 4 10 16 17	5s	10	6	1.00E-06
1	0.007444	1 4 10 12 17 18 19 27	4min49s	10	1	1.00E-06
1	0.0089035	1 4 10 17 18	7s	10	5	1.00E-05
1	0.0067785	1 4 18	7s	10	5	1.00E-06
1	0.0089035	1 4 10 16 17	7s	10	6	0.0001
1	0.0067785	1 4 17	5s	10	6	1.00E-05
1	0.0089035	1 4 10 16 17	10s	10	6	1.00E-06
1	0.0067785	1 4 17	5s	10	6	0.0001
1	0.00745	1 4 12 16 17	5s	10	6	1.00E-05
1	0.008555	4 6 10 16 17	5s	10	6	1.00E-06
Melhor rede neural: MLP 0.019905 ELMAN:0.017709						

Resultados e testes SONN (2 dias)

<i>Dia</i>	<i>MSE</i>	<i>Entradas usadas</i>	<i>Tempo de convergência</i>	<i>Temp. inicial</i>	<i>Tx. Temp.</i>	<i>Error Goal</i>
2	0.0152045	3 4 5 6 16 17 18	7s	10	5	0.0001
2	0.0161465	1 3 4 5 16 17 18	7s	10	5	1.00E-06
2	0.0183855	4 5 6 8 16 17 29	4min31s	10	1	0.0001
2	0.018834	3 4 5 6 10 16 19 21 28	1min60s	10	1	1.00E-06
2	0.0173115	1 4 5 7 16 17 18	7s	10	5	0.0001
2	0.0173115	1 4 5 7 16 17 18	9s	10	5	1.00E-05
2	0.0173115	1 4 5 7 16 17 18	7s	10	5	1.00E-06
2	0.0149405	3 4 6 16 17	5s	10	6	0.0001
2	0.016145	1 3 4 5 6 16 18 21 27	3min14s	10	1	1.00E-06
2	0.0175145	4 5 6 7 16 17 18	7s	10	5	0.0001
2	0.0158625	3 4 5 6 16 17 18	7s	10	5	1.00E-05

Melhor rede neural: MLP 0.030898 ELMAN: 0.025407

Resultados e testes SONN (3 dias)

<i>Dia</i>	<i>MSE</i>	<i>Entradas usadas</i>	<i>Tempo de convergência</i>	<i>Temp. inicial</i>	<i>Tx. Temp.</i>	<i>Error Goal</i>
3	0.023887	3 4 5 6 15 16 17 18 20	8min10s	10	1	1.00E-06
3	0.0238095	2 4 5 16 17	5s	10	6	1.00E-05
3	0.0238095	2 4 5 16 17	5s	10	6	1.00E-06
3	0.023841	3 4 5 6 15 16 18 19 24	7min31s	10	1	1.00E-05
3	0.02385	3 4 5 6 17 18 19 26	4min15s	10	1	1.00E-06
3	0.0238095	2 4 5 16 17	5s	10	6	0.0001
3	0.0248865	3 4 5 16 17	5s	10	6	1.00E-06
3	0.0242465	3 4 5 13 17 18 30	7min18s	10	1	1.00E-05
3	0.0241905	3 4 5 11 15 16 17 22 32	9min56s	10	1	1.00E-06
3	0.0248865	3 4 5 16 17	12s	10	5	0.0001
3	0.024203	2 3 4 5 16 17 18	7s	10	5	1.00E-05
Melhor rede neural: MLP 0.042699 ELMAN: 0.03596						

Resultados:

- ▶ Os resultados mostram uma melhor adaptação da rede SONN em relação às tradicionais MLP e ELMAN.
- ▶ No entanto, para fins comerciais, um sistema de predição de valores de ações deveria ser capaz, acima de tudo, de prever corretamente quando uma ação terá queda, ou quando será valorizada.
- ▶ Desta forma, uma simples contagem do número de vezes em que a rede indicou queda ou aumento e realmente houve queda ou aumento é um melhor parâmetro de performance que o MSE.

Estimativa de primeira derivada:

- ▶ Para possibilitar esta comparação, conta-se o número de vezes em que o produto das primeiras derivadas do alvo e o indicado pelo modelo é positivo. Dessa forma tem-se o número de vezes em que o modelo indicou o correto futuro do título ao investidor.

número de pontos em que

$$Est'(t) \cdot Des'(t) = 0$$

Número de estimativas corretas (MLP)

Um dia (200 pontos)	Dois dias (200 pontos)	Três dias (200 pontos)
97	105	106
112	109	103
99	99	113
102	115	94
94	99	104
104	97	98
97	96	95
105	111	98
97	108	92
102	110	98
103	108	93
95	110	100
101	100	99
Média: 98	Média: 104	Média: 99

Número de estimativas corretas (ELMAN)

Um dia (200 pontos)	Dois dias (200 pontos)	Três dia (200 pontos)
114	92	103
106	100	109
111	102	104
109	94	105
98	110	88
99	93	109
106	110	95
102	113	118
105	102	116
101	97	109
106	112	107
104	107	96
104	100	98
Média: 103	Média: 104	Média: 103

Número de estimativas corretas (SONN)

Um dia (200 pontos)	Dois dias (200 pontos)	Três dia (200 pontos)
109	113	102
108	108	104
110	106	102
110	106	103
108	110	101
107	112	102
108	106	103
108	106	102
110	112	102
108	106	103
109	110	102
112	107	102
108	113	104
Média: 107.9259259	Média: 108.8888889	Média: 101.7037037
Melhor MLP: 112	Melhor MLP: 115	Melhor MLP: 106
Melhor ELMAN: 112	Melhor ELMAN : 115	Melhor ELMAN : 106

Conclusões

- ▶ O modelo SONN mostrou-se bastante rápido e eficiente, sendo mais rápido que a MLP e a ELMAN em todos os casos.
- ▶ Embora o novo modelo tenha sido mais eficiente na aproximação da função do problema, ele ainda falha em estimar as quedas e aumentos nas ações da bolsa. Havendo ainda um certo deslocamento (delay) entre a função desejada e a obtida.
- ▶ A rede SONN mostrou-se uma alternativa viável e prática para substituir o modelo neural padrão (MLP) em alguns casos.