

Redes de Elman e Jordan Aplicadas na Solução do Problema XOR

Fabício Pereira Härter
LAC/INPE
fabricao@lac.inpe.br

Haroldo Fraga de Campos Velho
LAC/INPE
haroldo@lac.inpe.br

Resumo

Neste trabalho aplica-se as Redes Neurais conhecidas na literatura como Rede de Elman e Rede de Jordan na solução do problema XOR. Este problema vem sendo empregado por muitos autores para testar a eficiência de Redes Neurais, pois o XOR não pode ser resolvido pelo Perceptron de uma única camada. As redes de Elman e Jordan são particularmente importantes por apresentarem conexões recorrentes. O fato destas Redes Neurais terem tido sucesso na solução deste problema, motiva os autores a aplicá-las à problemas de neurodinâmica, como assimilação de dados.

1. Introdução

Assimilação de dados é uma área das ciências atmosféricas, que dada a sua importância, vem sendo bastante pesquisada. Este problema vem sendo estudado também pelos autores deste trabalho. Pode-se definir assimilação de dados como o processo onde dados observacionais (ruidosos) são inseridos em modelos matemáticos imprecisos com o objetivo de melhorar a qualidade da previsão. A técnica que apresenta bom resultado quando aplicada a este tipo de problema é o Filtro de Kalman (FK), porém esta é uma técnica de alto custo computacional.

Como objetivo de resolver o problema do alto custo computacional do FK, técnicas de Redes Neurais (RN) tem sido propostas à este problema. Em [7] Redes Neurais tipo Perceptron de Múltiplas Camadas (RPMC) foram testadas em assimilação de dados para os modelos de Lorenz e Hénon no estado caótico. Aplicação de RN em assimilação de dados para os modelos de Lorenz e DYNAMO são discutidos [8]. O FK, nas versões estendida e adaptativa, e o PMC mostraram-se boas técnicas para assimilação de dados nestes modelos conceituais, inclusive quando o sistema é não-linear. Em [3] foram testadas diferentes arquiteturas das RPMC e Rede Funções de base Radial (RFBR).

Entretanto, estas redes são aplicadas no sentido de emular o FK, que por sua vez é recorrente, ou seja, as variáveis de estado no tempo n dependem das variáveis no tempo $n-1$. Este fato motiva os autores a testarem redes recorrentes, que são redes que possuem um ou mais laços de realimentação. A RN que vem sendo testadas com sucesso são a Rede de Elman (RE) [2], a Rede de Jordan (RJ) [6] e a Rede de Hopfield (RH) [5]. Neste trabalho se apresenta resultados da RE e da RJ, aplicadas ao problema XOR, com o intuito de testar-se os algoritmos que vêm sendo implementados.

Na seção 2 apresenta-se o problema XOR, na seção 3 são apresentadas a RE e a RJ. A seção 4 é reservada para resultados e considerações finais.

2. O Problema XOR

No Perceptron elementar (de camada única), não há neurônios ocultos. Conseqüentemente, ele não pode separar padrões de entrada que sejam não linearmente separáveis. Entretanto, padrões não linearmente separáveis ocorrem freqüentemente. Esta situação surge no problema OU exclusivo (XOR, *Exclusive OR*), que pode ser visto como o caso especial de um caso mais geral, que é o de classificar pontos no hipercubo unitário. Cada ponto no hipercubo pertence ou a classe 0 ou a classe 1. Entretanto, no caso especial do problema XOR, necessitamos considerar apenas os quatro vértices do quadrado unitário que correspondem aos padrões de entrada (0,0), (0,1), (1,1), e (1,0). O primeiro e o terceiro padrões de entrada pertencem a classe 0 ($0 \oplus 0 = 0$, $1 \oplus 1 = 0$, $\oplus$ representa o operador da função booleana OU exclusivo). Os padrões de entrada (0,0) e (1,1) estão em vértices opostos do quadrado unitário; apesar disso possuem a mesma saída 0. Por outro lado, os padrões de entrada (0,1) e (1,0) estão em vértices opostos do quadrado, mas pertencem a classe 1 ($0 \oplus 1 = 1$, $1 \oplus 0 = 1$), [4].

3. Redes de Elman e Jordan

A seguir descreve-se as RE e RJ, conforme [1]. Nas RE, Figura 1, além das unidades de entrada, intermediária e de saída há também unidades de contexto, como nas redes recorrentes em geral. As unidades de entrada e de saída interagem com o ambiente externo, enquanto as unidades intermediárias e de contexto não o fazem. As unidades de entrada são apenas unidades de armazenamento (*buffer*) que passam os sinais sem modificá-los. As unidades de saída são unidades lineares que somam os sinais que recebem. Neste trabalho as unidades intermediárias tem funções de ativação não lineares, embora segundo a teoria possam ser lineares. As unidades de contexto são usadas apenas para memorizar as ativações anteriores das unidades intermediárias e podem ser consideradas como atraso no tempo em um passo. As conexões *feedforward* são modificáveis, e as conexões recorrentes são fixas, motivo pelo qual a RE é apenas parcialmente recorrente.

Em um intervalo de tempo específico k , as ativações das unidades intermediárias (em $k-1$) e as entradas correntes (em k) são utilizadas como entradas da rede. Em um primeiro estágio, *feedforward*, estas entradas são propagadas para frente para produzir as saídas. Posteriormente, a rede é treinada como o algoritmo de aprendizagem *back-propagation* [7] padrão. Após este passo de treinamento, as ativações das unidades intermediárias no tempo k são introduzidas, através das ligações recorrentes nas unidades de contexto, sendo salvas nestas unidades para o próximo passo de treinamento ($k+1$).

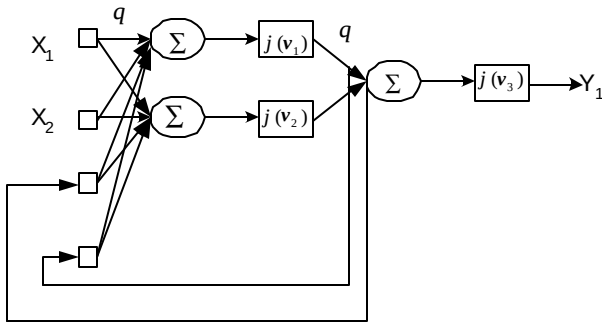


Figura 1 – Rede de Elman

Na RJ, Figura 2, a saída da rede é copiada para a unidade de contexto. Além disso, as unidades de contexto são localmente recorrentes. A grande diferença em termos de topologia entre as duas redes é que a recorrência na RE é feita da camada escondida para as entradas, enquanto na RJ a recorrência é feita das saídas para as entradas.

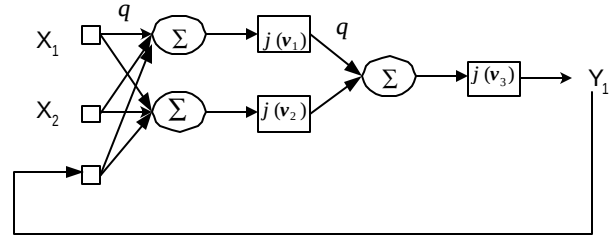


Figura 2 – Rede de Jordan

4. Resultados e Considerações Finais

Como pode-se observar nas figuras 1 e 2, são apresentados duas entradas as redes, que tem 2 neurônios na camada escondida e um neurônio na saída. A função de ativação na camada escondida é uma tangente hiperbólica e na saída a função de ativação aplicada é linear. Utilizou-se taxa de aprendizagem 0,01. Devido a arquitetura, a RE tem duas unidades de contexto, enquanto a RJ tem uma unidade de contexto.

Nas tabelas a seguir, pode-se observar como foi inicializado o elemento recorrente, o número de épocas necessárias para a rede atingir o erro pré-estabelecido, o erro e o resultado após a ativação da rede neural.

Tabela 1: Rede de Elman

Inicialização do Elemento Recorrente	Épocas de Treinamento	Erro Atingido no Treinamento	Ativação (resultado)
Aleatório Gaussiano	15077	$1,000 \times 10^{-8}$	2.415×10^{-5} 9.998×10^{-1} 1.824×10^{-4} 9.999×10^{-1}
0.0	27934	$1,000 \times 10^{-8}$	4.367×10^{-5} 9.998×10^{-1} 1.827×10^{-5} 9.993×10^{-1}
1.0	6380	$1,000 \times 10^{-8}$	7.552×10^{-5} 9.998×10^{-1} 1.436×10^{-4} 9.998×10^{-1}

Tabela 1: Rede de Jordan

Inicialização do Elemento Recorrente	Épocas de Treinamento	Erro Atingido no Treinamento	Ativação (resultado)
Aleatório Gaussiano	38553	$1,000 \times 10^{-8}$	4.451×10^{-6} 9.996×10^{-1} 7.006×10^{-4} 9.954×10^{-1}
0.0	32242	$1,000 \times 10^{-8}$	1.140×10^{-4} 9.999×10^{-1} 2.524×10^{-5} 9.994×10^{-1}
1.0	3877	$1,000 \times 10^{-8}$	1.123×10^{-4} 1.0000 1.774×10^{-4} 9.997×10^{-1}

Tanto a RE como a RJ resolveram com boa precisão o problema XOR. Portanto, é importante notar que a inicialização do elemento recorrente é importante para

que a convergência seja rápida. A RE atingiu o erro de $1,000 \times 10^{-8}$ em 15077, 27934 e 6380 épocas de treinamento para os elementos recorrentes inicializados respectivamente com números que seguem uma distribuição gaussiana, 0,0 e 1,0. Já a RJ precisou de 38553, 32242 e 3877 épocas de treinamento para atingir o mesmo erro para as três maneiras de inicialização dos elementos recorrentes.

Os autores deste trabalho já obtiveram bons resultados (ainda não publicados) com estas RN aplicadas a problemas de assimilação de dados com o modelo de Lorenz. Estão em desenvolvimento trabalhos nos quais se aplica estas redes no modelo de Água Rasa, o que significa aumentar a dimensão do problema a ser resolvido.

Reconhecimento: Este trabalho foi financiado pela FAPESP.

6. Referências

- [1] Braga, A. P., Ludermit, T. B., Carvalho, A. C. L. Redes Neurais Artificiais – Teoria e Aplicações, Livros Técnicos e Científicos, Rio de Janeiro (RJ), 2000.
- [2] Elman, J.L.; Find Structure in Time, Cognitive Science, 1990, **14**, 179-211.
- [3] Härter, F. P.; Campos Velho H. F. Diferentes Topologias de Redes Neurais em Assimilação de Dados, Anais do XII Congresso Brasileiro de Meteorologia, Foz do Iguaçu (PR), 2002, 3642-3652.
- [4] Haykin, S. Redes Neurais: Princípios e Prática, Bookman, Porto Alegre, 2001.
- [5] Hopfield, J. J. Neural Networks and Physical Systems with Emergent Collective Computational Abilities, Proceedings of the national Academy of Sciences, USA, 1982, **79**, 2554-2558.
- [6] Jordan, M. I., Attractor Dynamics and Parallelism in a Connectionist Sequential Machine. In Proceedings of the Eight Annual Conference of the Cognitive Science Society, 1986, 531-546.
- [7] Nowosad, A; Rios Neto, A; Campos Velho, H. F. de. Data Assimilation in Chaotic Dynamics Using Neural Networks. In: International Conference on Nonlinear Dynamics, Chaos, Control and Their Applications in Engineering Sciences, Campos do Jordão. Proceedings. Rio de Janeiro: Associação Brasileira de Ciências Mecânica, 2000b, 212-221.
- [2] Nowosad, A.; Campos Velho, H. F. de; Rios Neto, A. Neural Network as a New Approach for Data Assimilation. In: A Meteorologia Brasileira Além do Ano 2000b: XI Congresso Brasileiro de Meteorologia, Campos do Jordão, SP, 2000c, 3078-3086.