

Extração de Regras de Redes Neurais por meio do Algoritmo RX Modificado: Um Exemplo de Aplicação em Modelagem de Dados Meteorológicos.

Eduardo R. Hruschka¹, Nelson F. F. Ebecken²

¹ COPPE/UFRJ / Centro Universitário Positivo

² COPPE/UFRJ

E-mails: erh@ineparnet.com.br, nelson@ntt.ufrj.br

Abstract

The main challenge in using supervised neural networks in data mining applications means to get explicit knowledge from these models. For this purpose, an algorithm for rule extraction from artificial neural networks, based on the hidden units activation values, is developed. This algorithm, denominated Modified RX, was already evaluated in two benchmarks - Iris Plants database and Pima Indians Diabetes database - and the results were published previously. This work deals with the application of this algorithm to a dataset containing 10,000 examples of meteorological observations collected at the International Airport of Rio de Janeiro. Each example is represented by 38 attribute values and one associated class - wet fog or dry fog. Following the data preparation tasks - data representation, data selection and correlation analysis - a neural network is trained to model wet and dry fog conditions, and then the Modified RX Algorithm is used for rule extraction. The results obtained from the rule set provided by the algorithm are compared to those obtained from a classification tree.

1. Introdução

A principal limitação da utilização de redes neurais supervisionadas em aplicações de *data mining* deve à dificuldade de entendimento da representação do conhecimento incorporado por estes modelos. O conhecimento adquirido durante a fase de treinamento de uma rede neural é codificado na arquitetura da rede neural, nas funções de ativação de cada unidade oculta e de saída e nos pesos das conexões. Para superar tal limitação, faz-se uso de técnicas de extração de regras de redes neurais. A tarefa de extração de regras de redes neurais pode ser assim resumida [1]: dada uma rede neural treinada, o conjunto de exemplos de treinamento e um nível de fidelidade entre o conjunto de regras e o modelo de rede neural, deseja-se obter um conjunto de regras que modela a rede neural para um dado nível de fidelidade. A descrição das principais metodologias utilizadas para a extração de regras de redes neurais desenvolvidas até 1995 pode ser encontrada em [2]. Outras metodologias utilizadas para

extração de regras de redes neurais podem ser encontradas em [1-20].

2. Algoritmo implementado

O algoritmo RX [11] é baseado nas ativações das unidades ocultas. Primeiramente, realiza-se o treinamento da rede neural, de modo que se obtenha a taxa de classificação correta desejada. Eliminam-se, então, as conexões redundantes, seguindo-se uma análise dos valores das ativações para a obtenção de regras baseadas nestes valores. Basicamente, pode-se resumir o algoritmo de extração de regras, denominado de RX, aos seguintes passos:

1) Processar os eventos - pertencentes ao conjunto de treinamento - através da rede computando-se os valores de ativação das unidades.

2) Aplicar um algoritmo de agrupamento - clustering - aos valores de ativação das unidades ocultas.

3) Enumerar os valores de ativação discretizados e computar os respectivos valores de saída da rede. Gerar regras que descrevam as saídas da rede em termos dos valores de ativação discretizados em 2).

4) Para cada unidade oculta, enumerar os valores de entrada que conduzem aos respectivos valores de ativação das unidades ocultas, gerando regras que descrevam os valores discretizados das unidades ocultas em termos das entradas da rede neural.

5) Combinar as regras obtidas em 3) e 4) para obter as regras que relacionam as variáveis dependentes às variáveis independentes.

O primeiro passo do algoritmo RX agrupa os valores de ativação das unidades ocultas sem o comprometimento da precisão da rede neural. Após a execução do algoritmo, obtém-se um conjunto de valores de ativação para cada unidade oculta. O algoritmo de agrupamento é aplicado em cada unidade oculta separadamente. Primeiramente, classificam-se os valores de ativação em ordem crescente. Segue-se, então, a procura pelos intervalos correspondentes às ativações obtidas para cada classe, de maneira que os agrupamentos não gerem informações redundantes, isto é, intervalos de ativações iguais gerando classes diferentes. Considerando-se esta metodologia, obtém-se regras da seguinte forma:

Se $(a_1 \theta v_1)$ e $(a_2 \theta v_2)$ e ... e $(a_n \theta v_n)$ então C;

onde:

a_i = atributos

$\theta = \{ =, \leq, \geq, <, >, \neq \}$

v_i = constantes

C = classes.

O presente trabalho é baseado nesta metodologia, e o algoritmo aqui desenvolvido é, por isso, denominado de *Algoritmo RX Modificado*. Basicamente, as mudanças se devem ao fato de que há um conjunto particular de valores de ativações para cada classe e, assim, pode-se extrair regras separadamente para cada classe, diferentemente do que é realizado pelo Algoritmo RX. Conceitualmente, entende-se que as ativações das unidades refletem o conhecimento adquirido pela rede neural para cada classe. Neste contexto, exemplos pertencentes a uma classe em particular possuem ativações semelhantes, permitindo pois, que se realize um agrupamento prévio dos mesmos. Além disso, desenvolveu-se um algoritmo de *clustering* diferente daquele empregado pelo Algoritmo RX.

3. Algoritmo de Clustering

O algoritmo implementado é, de certa forma, uma variante do algoritmo conhecido por *Leader Algorithm* [21]. Em se tratando do algoritmo implementado, assume-se como líder a média do conjunto de valores, e a tolerância é dada em função do desvio padrão. A distância utilizada é a *Euclidiana*. A definição do critério de semelhança, representado pela distância máxima tolerada entre dois números T, é inerente ao domínio e difícil de ser quantificada. Por isso, optou-se pela utilização de um critério de semelhança variável em função do desvio padrão do conjunto de dados. Para tal permite-se que o usuário defina seu critério de proximidade em função de uma constante multiplicada pelo desvio padrão. Esta constante poderá variar de 1 a 4. Em relação ao ponto de partida, entende-se ser conveniente a utilização do ponto médio, de maneira que se tenha, no primeiro *cluster*, os valores mais próximos à média. Seguem-se os passos computados pelo algoritmo:

1) Calcule a média e o desvio padrão do conjunto de valores C ;

2) Agrupe os valores v cuja distância Dv seja menor ou igual ao desvio padrão multiplicado pelo fator de tolerância MULTI, onde Dv é a distância entre o valor v e a média.

3) Retire, do conjunto de valores C, os valores agrupados no *cluster* ;

4) Se C = { ϕ } então pare.

Senão vá para o passo 1)

O número de *clusters* encontrados é variável, dependendo de cada aplicação e sendo calculado automaticamente pelo algoritmo. Além disso, convém citar que o algoritmo implementado permite a visualização de uma quantidade de *clusters* dado pela seguinte expressão:

$NC = (5 - FATOR)$ onde:

NC = número de clusters impressos no arquivo de saída;

FATOR = fator de multiplicação de desvio padrão {1,2,3,4}.

Esta sistemática envolve o conceito relativo ao número de *clusters* significativos. Por exemplo, ao considerar uma tolerância igual ao desvio padrão multiplicado por quatro (MULTI=4), somente um *cluster* seria significativo. Em contrapartida, considera-se 4 *clusters* para a tolerância igual a uma unidade de desvio padrão.

Inicialmente foram efetuados diversos testes com *benchmarks* considerados clássicos, obtendo-se excelentes resultados [16,17]. Neste trabalho estuda-se o caso prático de um problema de previsão meteorológica (nevoeiros em aeroportos).

4. Aplicação

O conjunto de dados utilizado neste exemplo é constituído de uma série de medições realizadas de hora em hora, durante dez anos (1951 a 1960), no Aeroporto Internacional do Rio de Janeiro. O conjunto de dados contém 88.000 exemplos de observações meteorológicas constituídas dos seguintes atributos: mês; data; hora; total de nuvens; direção e velocidade do vento de superfície; visibilidade horizontal; tempo presente (Nada a relatar, Precipitação à vista, Névoa seca / fumaça, Areia/poeira, Névoa úmida/nevoeiro, Chuvisco, Chuva, Neve, Trovoada / relâmpago e Granizo); quantidade, tipo e altura das nuvens para as camadas 1, 2, 3 e 4; ajuste do altímetro; temperatura do ponto de orvalho; velocidade de rajada; pressão atmosférica ao nível do mar e da pista; direção das nuvens nas camadas 1, 2, 3 e 4; variação da pressão em tendência e na diferença de 3 horas; temperatura medida no bulbo úmido e no bulbo seco; precipitação e umidade relativa do ar.

No concernente à aplicação, deseja-se mostrar a possibilidade de se obter algum conhecimento baseando-se somente no conjunto de dados, haja vista que se dispõe apenas de um conhecimento bastante superficial do domínio em questão e que não se pretende elucidar os fenômenos complexos da Meteorologia.

Uma condição que se mostra bastante interessante diz respeito à presença de névoa úmida ou névoa seca, dado que estes fenômenos são importantes para a operação de aeroportos. Neste aspecto, surgiu o interesse de se descobrir quais as variáveis importantes no processo e a relação entre as mesmas. Para tal, extraiu-se do conjunto original de dados todos os exemplos relativos à névoa úmida e névoa seca. Cumpre citar que, deste subconjunto, muitos exemplos contém variáveis cujos valores são desconhecidos, sendo, portanto, inadequados para a obtenção de conhecimento. Sob este aspecto, deve-se definir critérios para a eliminação dos exemplos que apresentem campos cujos valores são desconhecidos. Para tal, torna-se importante a disponibilidade de

algum conhecimento de domínio. Neste sentido, cumpre citar que o conhecimento do fenômeno de névoa úmida parece ser mais importante para a operação de aeroportos. Assim sendo, optou-se por focalizar as tarefas de preparação dos dados neste fenômeno.

BYERS [22] sustenta que a névoa úmida é uma espécie de estrato, que se forma próximo ao solo, afetando seriamente a visibilidade da superfície. Estratos são nuvens características da parte mais baixa da atmosfera, que se formam quando acontece uma inversão de temperatura ou quando existe uma camada isotérmica. Cita, ainda, que a investigação de tais fenômenos basicamente se reduz à determinação das circunstâncias nas quais ocorrem resfriamento das massas de ar na superfície, em presença de suficiente umidade. Em outros casos, o fenômeno pode ocorrer devido ao aumento da quantidade de vapor de água, sem que ocorra resfriamento da massa de ar. Do ponto de vista físico, a neblina se forma pela diminuição da temperatura do ponto de orvalho ou pela adição de vapor de água, até que a temperatura real seja igual à temperatura do ponto de orvalho [22].

Considerando-se, então, os conceitos previamente expostos, optou-se por eliminar todos os exemplos para os quais os valores de umidade relativa do ar e precipitação são desconhecidos. Em seguida, eliminou-se as variáveis relativas às nuvens da camada 4 (quantidade, tipo e altura), pois estas variáveis não apresentam nenhum valor - provavelmente valores nulos - para névoa úmida, do mesmo modo que se eliminou as variáveis relativas à camada 3, para a qual se dispõe de 52 exemplos que apresentam valores não nulos. Estas duas camadas provavelmente não exercem influência no fenômeno de nevoeiro, pois são as camadas mais altas de nuvens. A variável rajada também foi eliminada pois apresentava somente um valor conhecido. A variável relativa à direção das nuvens nas camadas 1, 2, 3 e 4 também foi eliminada, dada a pequena quantidade de valores conhecidos. A variável pressão ao nível da pista também foi eliminada por estar diretamente relacionada à pressão ao nível do mar e por conter muitos valores desconhecidos.

Ainda em relação à preparação dos dados, optou-se pela eliminação de algumas variáveis muito correlacionadas, pois a rede neural aprende tanto melhor quanto menor a correlação entre as variáveis independentes [23,24]. Desta maneira, desconsiderou-se a variável de ajuste do altímetro, cujo coeficiente de correlação linear relativo à pressão ao nível do mar é igual a 1,00. Desconsiderou-se, ainda, a temperatura do bulbo úmido, correlacionada à temperatura do ponto de orvalho em 0,96 e correlacionada em relação à temperatura do bulbo seco em 0,95. Estas duas últimas também estão correlacionadas em 0,86, mas optou-se pela manutenção de ambas, pois o conceito de névoa úmida se baseia nestas duas variáveis.

Obtém-se, assim, 1509 exemplos (1200 de névoa úmida e 309 de névoa seca) considerados como válidos para a obtenção de conhecimento. Cite-se, ainda, que

não se realizou nenhum tipo de análise de consistência mais elaborada do conjunto de dados.

Utilizou-se o simulador *NeuralWorks Predict* [25] para realizar o treinamento da rede neural para o problema de classificação. Considerou-se que os dados contém ruído, pois não se realizou uma análise de consistência adequada e, ainda, as medições dos valores das variáveis nem sempre são confiáveis. Utilizou-se do algoritmo de gradiente adaptativo e de somente um conjunto para treinamento. Não se utilizou de conjuntos de teste e de validação devido ao pequeno número de exemplos disponíveis, relativamente ao número de variáveis empregadas no modelo. Considerando-se a seleção exaustiva de variáveis e a busca exaustiva da solução, o algoritmo genético implementado no simulador de redes neurais selecionou as seguintes variáveis:

- x₁ = mês
- x₂ = direção do vento
- x₃ = visibilidade
- x₄ = altura da segunda camada
- x₅ = temperatura de orvalho
- x₆ = umidade relativa

O modelo de rede neural que proporcionou os melhores resultados conta com 2 unidades ocultas e duas unidades de saída e obteve uma taxa de classificação correta de 89%, para 76 épocas, tanto para névoa seca quanto para névoa úmida. Observa-se que a amostra é não representativa [23]. Obtiveram-se, então, por meio do Algoritmo RX Modificado, as seguintes regras, considerando-se uma unidade de desvio padrão:

Se $\{(-7796 \leq 21.x_1 + 32.x_2 + x_4 - 14.x_5 - 84.x_6 \leq -5880)$
e $(-10460 \leq 27.x_1 + x_4 - 25.x_5 - 107.x_6 \leq -8060)\}$
então névoa úmida.

Se $\{(-5884 \leq 21.x_1 + 32.x_2 + x_4 - 14.x_5 - 84.x_6 \leq -2614)$
e $(-7920 \leq 27.x_1 + x_4 - 25.x_5 - 107.x_6 \leq -4549)\}$
então névoa seca.

A precisão de classificação alcançada por meio das regras, cuja medida de complexidade [26] é igual a 2, foi de 80%, sendo 83,5% para nevoeiro e 60% para névoa seca. Cite-se ainda que, em princípio, as regras refletem mais um conjunto de observações para cada um dos fenômenos do que um conjunto de condições para que tais fenômenos ocorram. Assim sendo, torna-se necessária uma avaliação destas regras por especialistas do domínio em questão. Observa-se, ainda, que a variável x₃ (visibilidade) não está representada nas regras devido aos pequenos valores dos pesos associados à unidade de entrada que a representa no modelo de rede neural.

A diferença de precisão de classificação nas regras, entre névoa úmida e seca, pode ser explicada por se tratar de uma amostra não representativa. Conquanto a rede neural classifique com precisão idêntica as duas classes - contrariamente ao senso comum - as regras

são afetadas na medida em que se tenta agrupar as ativações, formando conceitos. Neste aspecto, poder-se-ia repetir, com o auxílio de um especialista do domínio em questão, os exemplos mais importantes, de modo que as duas classes tivessem a mesma importância no treinamento.

Cumpre, novamente, citar que o conhecimento foi adquirido por meio de uma única fonte, especificamente um conjunto de dados, necessitando, portanto de validação por um especialista. Deseja-se mostrar, unicamente, que a metodologia aqui desenvolvida pode ser útil para o esclarecimento do fenômeno, mas não se tem a pretensão de elucidá-lo.

A título de ilustração, segue-se o conjunto de regras obtido por meio da árvore de classificação elaborada pelo programa *IBM Intelligent Miner for AIX* [27] :

Se $H_r > 79,5$ então Nevoeiro;
 Se $\{H_r \leq 79,5 \text{ e } T_s \leq 239,5 \text{ e } M \leq 8,5 \text{ e } T_2 \leq 2,5 \text{ e } H \leq 12\}$
 então Nevoeiro;
 Se $\{H_r \leq 79,5 \text{ e } T_s \leq 239,5 \text{ e } M \leq 8,5 \text{ e } T_2 \leq 2,5 \text{ e } H > 12\}$
 então Névoa Seca;
 Se $\{H_r \leq 79,5 \text{ e } T_s \leq 239,5 \text{ e } M \leq 8,5 \text{ e } T_2 > 2,5\}$
 então Nevoeiro;
 Se $\{H_r \leq 79,5 \text{ e } T_s \leq 203 \text{ e } M > 8,5\}$
 então Nevoeiro;
 Se $\{H_r \leq 79,5 \text{ e } (203 < T_s \leq 239,5) \text{ e } M > 8,5\}$
 então Névoa Seca;
 Se $\{H_r \leq 74,5 \text{ e } T_s > 239,5\}$ então Névoa Seca;
 Se $\{(74,5 < H_r \leq 79,5) \text{ e } T_s > 239,5 \text{ e } T_2 \leq 3,5 \text{ e } H \leq 6\}$
 então Nevoeiro;
 Se $\{(74,5 < H_r \leq 79,5) \text{ e } T_s > 239,5 \text{ e } T_2 \leq 3,5 \text{ e } H > 6\}$
 então Névoa Seca;
 Se $\{(74,5 < H_r \leq 79,5) \text{ e } (239,5 < T_s \leq 244,5) \text{ e } T_2 > 3,5\}$
 então Névoa Seca;
 Se $\{(74,5 < H_r \leq 79,5) \text{ e } T_s > 244,5 \text{ e } T_2 > 3,5\}$
 então Nevoeiro;

onde :

H_r = umidade relativa;

T_s = temperatura do bulbo seco;

M = mês;

H = hora;

T_2 = tipo de nuvem na segunda camada.

Este conjunto de regras classifica 94% dos exemplos corretamente, proporcionando uma medida de complexidade [26] igual a 8,3. Assim sendo, observa-se que o conjunto de regras extraído do modelo de rede neural proporciona uma taxa de classificação correta menor - igual a 80% - mas com uma medida de complexidade igual a 2,0.

5. Conclusões

Redes neurais supervisionadas não são amplamente utilizadas em aplicações de *data mining* devido à dificuldade de interpretação de seus modelos. A interpretabilidade de um modelo de rede neural depende, fundamentalmente, da aplicação de

algoritmos eficazes de extração de regras. Neste sentido constata-se que o Algoritmo RX Modificado proporcionou, para a aplicação presente, um conjunto de regras cuja medida de complexidade [26] é inferior à medida de complexidade do conjunto de regras obtido por meio do algoritmo de árvore de classificação [27], mas com uma taxa de classificação correta relativamente menor. A avaliação da complexidade das regras é um assunto muito importante, haja vista que as regras obtidas estão sendo avaliadas e confrontadas com os modelos normalmente utilizados pelos meteorologistas.

Referências Bibliográficas

- [1] SHAVLIK, J. "An Overview of Research at Wisconsin on Knowledge-Based Neural Networks", *Proceedings of the International Conference on Neural Networks*, 1996.
- [2] ANDREWS, R., DIEDERICH, J., TICKLE, A. B. "A Survey and Critique of Techniques for Extracting Rules from Trained Artificial Neural Networks", **Knowledge Based Systems**, 8(6), pp. 373- 389, 1995.
- [3] NARAZAKI, H., SHIGAKI, I., WATANABE, T. "A Method for Extracting Approximate Rules from Neural Network", **IEEE**, 1995.
- [4] KANE, R., MILGRAM, M. "Financial Forecasting and Rules Extraction from Trained Networks", **IEEE**, pp. 3190-3195, 1994.
- [5] NARAZAKI, H., WATANABE, T., YAMAMOTO, M., "Reorganizing Knowledge in Neural Networks: An Explanatory Mechanism for Neural Networks in Data Classification Problems", **IEEE Transactions on Systems, Man, and Cybernetics**, v.26, n. 1, pág. 107-117, Feb., 1996.
- [6] SETHI, I, YOO, J., "Multivalued Logic Mapping of Neurons in Feedforward Networks", **Engineering Intelligent Systems**, v.4, n.4, pp. 243-253, 1996.
- [7] CRAVEN, M. W., SHAVLIK, J. W., "Extracting Tree-Structured Representations of Trained Networks", **Advances in Neural Information Processing Systems**, v.8, MIT Press, Cambridge, MA, 1996.
- [8] TAHA, I., GHOSH, J. "Symbolic Interpretation of Artificial Neural Networks", **IEEE Transactions on Knowledge and Data Engineering**, Sep. 1996.
- [9] TAHA, I., GHOSH, J. "Three Techniques for Extracting Rules from Feedforward Networks", **Intelligent Engineering Systems Through Artificial Neural Networks**, v.6, ASME Press, Nov. 1996.
- [10] TAHA, I., GHOSH, J., "A Hybrid Intelligent Architecture and Its Application to Water Reservoir Control", **Journal of Smart Engineering Systems**, October 1996.
- [11] LU, H., SETIONO, R., LIU, H. "Effective Data Mining Using Neural Networks", **IEEE Transactions on Knowledge and Data Engineering**, v. 8, n. 6, pp. 957-961, 1996.
- [12] OLIVEIRA, J. P., ULIANA, P. B., LIMA, W. C., "Algoritmos de Extração de Regras de Redes Neurais Artificiais". In: Anais do III Congresso Brasileiro de Redes Neurais, pág 173-177, Florianópolis, Julho de 1997.
- [13] BENÍTEZ, J.M., CASTRO, J.L., REUQUENA, I. "Are Artificial N.N. Black Boxes?", **IEEE Transactions on Neural Networks**, v.8, n.5, Sep. 1997.

- [14] SETIONO, R., LIU, H. "NeuroLinear: From neural networks to oblique decision rules", **Neurocomputing**, v.16, n. 3, Sep. 1997, pp. 225-235.
- [15] SETIONO, R. "Extracting Rules from Neural Networks by Pruning and Hidden-unit Splitting" **Neural Computation**, v.9, n.1, pp. 205-225, Jan. 1997.
- [16] HRUSCHKA, E. R. "Um Estudo sobre a Extração de Regras de Redes Neurais em Aplicações de Data Mining", Tese de Mestrado, COPPE/UFRJ, 1998.
- [17] HRUSCHKA, E.R., EBECKEN, N. F. F. Rule Extraction from Neural Networks in Data Mining Applications, In: **Data Mining**, pp. 289-301, Computational Mechanics Publications, 1998, Southampton.
- [18] WEIJTERS, T., BOSH, A., HERIK, J., "Interpretable Neural Networks with BP-SOM", *10th European Conference on Machine Learning - ECML 98*, April 1998, Chemnitz, Germany.
- [19] DAS, S., MOZER, M. , "Dynamic on-line clustering and state extraction: an approach to symbolic learning " , **Neural Networks**, n.11, pp. 53-64, 1998.
- [20] CRAVEN, M. W., SHAVLIK, J. W., "Using Neural Networks for Data Mining", **Future Generation Computer Systems**, accepted to appear, 1997.
- [21] HARTIGAN, J. A., Clustering Algorithms, 1 ed., USA, John Wiley & Sons Inc., 1975.
- [22] [60] BYERS, H. R., General Meteorology, 3^a ed., New York, McGraw Hill, 1959.
- [23] SMITH, M., Neural Networks for Statistical Modeling, 1 ed. USA, International Thomson Computer Press, 1996.
- [24] YILDIZ, N., "Correlation Structure of Training Data and the Fitting Ability of Back Propagation Networks: Some Experimental Results " , **Neural Computing & Applications**, v.5, pág. 14-19, 1997.
- [25] NeuralWare, Inc - Technical Publications Group, documentação do programa *NeuralWorks Predict*, 1995.
- [26] GAINES, B. R. "Transforming Rules and Trees into Comprehensible Knowledge Structures ". In: **Advances in Knowledge Discovery and Data Mining**, MIT Press, pp. 205-229, 1996.
- [27] IBM Intelligent Miner for AIX - User's Guide - Version 1-Release1,1996