Proceedings of the V Brazilian Conference on Neural Networks - V Congresso Brasileiro de Redes Neurais
pp. 115–120, April 2–5, 2001 - Rio de Janeiro - RJ - Brazil

Arquiteturas de Redes Neurais Aplicadas a Data Mining no Mercado Financeiro
Uma Aplicação para a Geração de Credit Ratings

Daniel Arraes, Robinson Semolini, Ronaldo Picinini
InfoMiner – Tecnologia do Conhecimento
E-mails: daniel.arraes@uol.com.br, semolini@bestway.com.br, rppici@hotmail.com

Abstract

Credit Ratings play a central role in Credit Risk Management for Financial Institutions. Credit Risk Managers are interested in calculating Risk Spreads, in order to balance the expected losses on a portfolio of customers or credit operations, what is much easier to do using Credit Ratings. In this paper, some alternative approaches are proposed to obtain Credit Ratings, derived from Statistical Theory and Neural Networks Architectures, instead of the usual Linear Logistic Regression Model: Additive Logistic Regression, Multi Layer Perceptron Neural Networks and Bayesian Networks. A comparison of the methods is presented and some recommendations are indicated.

1. O Processo de Data Mining

Data Mining pode ser definido como um processo para descoberta de padrões e tendências em grandes conjuntos de dados, objetivando formar conhecimento para tomadas de decisão. O objetivo principal do Data Mining é formar novos conhecimentos sobre um fenômeno que está sendo estudado. Esse processo de descoberta é realizado através da aplicação sistemática de algumas metodologias de análise de dados, entre elas: Análise Estatística Elementar (medidas de posição e dispersão, gráficos), Amostragem e Inferência Estatística, Análise de Regressão (linear e logística), Análise de Cluster, Análise de Componentes Principais e Análise Fatorial, Análise de Correspondência, Conjoint Analysis, Análise de Associação (Market Basket Analysis), Árvores de Classificação (Algoritmos CHAID, CART) e Redes Neurais (MLP, RBF, Redes Bayesianas, Kohonen)

O processo de Data Mining tem sido utilizado em grande número de áreas de negócio: Bancos, Financeiras, Administradoras de Cartões de Crédito, Comércio Varejista, Empresas de telecomunicações, Seguradoras e Administradoras de Planos de Saúde, Grandes Indústrias (Farmacêuticas, Alimentícias).

As três primeiras áreas citadas utilizam mais intensamente o processo de Data Mining para construção de novos conhecimentos sobre seu negócio, o mercado e o ambiente em que a empresa está inserida. A área financeira tem sido a pioneira na expansão do conceito de Data Mining como suporte para praticamente todos os seus processos de decisão, incluindo o problema de gestão de risco de crédito, garantindo agilidade e segurança em suas operações.

2. Spread de Risco e obtenção de Ratings

De forma simplificada, pode-se considerar a taxa de juros de uma operação de crédito como sendo formada por três componentes, conforme o gráfico abaixo:

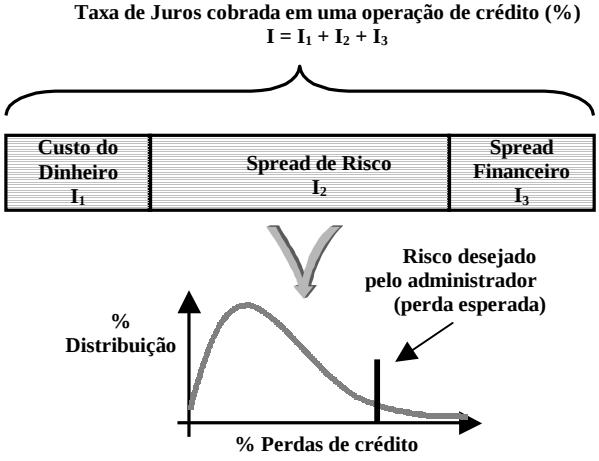

Figura 1

O Spread de Risco oferece ao administrador uma proteção contra perdas de crédito esperadas. Para seu cálculo adequado é necessário estimar a distribuição de perdas das carteiras de crédito, utilizando para tal cálculo uma expectativa ou perda esperada sobre cada uma das operações.

Essa perda esperada é expressa geralmente por uma classificação de risco ou Credit Rating atribuída a cada operação. O Credit Rating é construído através da observação do histórico de perdas da instituição financeira, obtendo-se um modelo matemático que relacione a probabilidade de perda com características conhecidas da operação e/ou do cliente.

A abordagem matemática mais utilizada pra a obtenção dos Credit Ratings envolve a estimativa de parâmetros de um modelo de regressão:

$$E(t) = f(\mathbf{x}) + \varepsilon \qquad (1)$$

- **X** é um vetor de variáveis explanatórias ou independentes;
- ε é um erro (ou desvio) sobre o qual geralmente se assume uma distribuição de probabilidades;
- E(.) é o operador esperança matemática.

A variável *t* é geralmente binária, onde:

$$t = \begin{cases} 0\text{, se houve perda de crédito na operação} \\ 1\text{, se não houve perda de crédito na operação} \end{cases} \qquad (2)$$

O evento inadimplência é observado durante o tempo de duração da operação, ou durante um horizonte fixo de tempo quando as operações não possuem um prazo limitado (p. ex. operações de crédito rotativo: cheque especial, cartão de crédito).

3. A obtenção de Credit Ratings

Estaremos trabalhando com o problema de obtenção de Credit Ratings em operações de crédito bancário. O gráfico abaixo resume o delineamento adotado para o modelo de Rating desenvolvido.

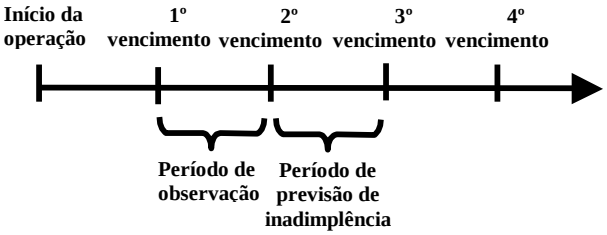

Figura 2

A operação é renovada a cada 3 (três) meses, e os períodos de previsão de inadimplência se tornam períodos de observação no vencimento seguinte. Foram selecionadas, aleatoriamente, 8.000 operações de crédito, divididas em 3 amostras, conforme tabela abaixo:

Tabela 1: Amostras de operações

	Amostras		
	Treinamento (50%)	Validação (25%)	Teste (25%)
Não Inadimplentes	2.000	1.000	1.000
Inadimplentes	2.000	1.000	1.000
Total	4.000	2.000	2.000

Serão utilizadas como variáveis explanatórias algumas características da operação e do cliente: informações cadastrais, informações sobre o comportamento da operação analisada, informações sobre outras operações de crédito do mesmo cliente, informações sobre recursos financeiros (aplicações) do cliente e informações externas (mercado); totalizando 22 variáveis.

Estaremos propondo algumas abordagens de solução para um problema de classificação em duas classes C_1 e C_2, onde C_1 representa a classe de operações inadimplentes no período de observação (conf. Figura 2). Nosso interesse será conhecer a probabilidade de uma operação pertencer a determinada classe dado seu vetor de variáveis explanatórias ***X***. As seguintes abordagens serão consideradas: Regressão Logística Linear (RLG), Redes Multi-layer Perceptron (MLP), Redes Bayesianas (BAY), e Regressão Logística Aditiva (Modelos Aditivos Generalizados - GAM).

4. Pré-processamento dos dados

4.1. Verificação de valores extremos

Valores extremos, ou *outliers*, são valores anômalos observados em um conjunto de dados, e que podem afetar o processo de estimação de modelos. Os outliers podem ocorrer devido a erros de medida ou pela observação de valores reais, porém muito raros. Nas duas situações, é vantajoso eliminar alguns desses valores para reduzir o subespaço em que o problema estará sendo modelado, tornando-o assim uma região mais compacta.

Os outliers desse conjunto de dados foram eliminados através de um exame visual da distribuição de cada variável, realizado com o auxílio de técnicas gráficas como box-plot e histogramas. Ao final do processo, o conjunto de dados disponível foi reduzido para 7.339 observações, sendo que a proporção entre operações adimplentes e inadimplentes não foi significativamente alterada.

4.2. Padronização de variáveis

De acordo com vários autores, entre eles Haykin (1), uma significativa melhora na performance dos algoritmos de aprendizado das redes neurais artificiais pode ser alcançado se certas transformações nos dados de entrada da estrutura de rede forem realizadas, objetivando:

- Obter média próxima a zero para todas as variáveis;
- Obter dados de entrada não correlacionados;
- Obter variáveis de entrada com variância aproximadamente igual.

O primeiro e terceiro objetivos serão atingidos mediante a padronização das variáveis de entrada, através da seguinte transformação (j=1,...,22):

$$Z_j = \frac{X_j - \overline{X_j}}{S_{X_j}} \qquad (3)$$

Na equação (3), $\overline{X_j}$ é a média amostral e S_{X_j} é o desvio padrão amostral da variável X_j.

5. Proposição de Modelos

5.1. Regressão Logística Linear (RLG)

O modelo de regressão logística é geralmente aplicado em problemas cuja variável resposta é dicotômica, e o interesse é saber qual a probabilidade de um indivíduo pertencer a um determinado grupo. O modelo estabelece uma relação linear entre as variáveis explicativas e uma transformação da variável resposta, denominada logito.

Este modelo é descrito pela seguinte relação:

$$g(\mathbf{X}) = \ln\left(\frac{\pi}{1-\pi}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p \quad (4)$$

onde $\ln(\pi / 1 - \pi)$ é o logito (ou transformação logística) e π é a probabilidade de ocorrência da resposta 1 (classe C_1), que é definida pela seguinte função :

$$\pi(\mathbf{X}) = \frac{e^{(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p)}}{1 + e^{(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p)}} \qquad (5)$$

onde β_0, ..., β_k os coeficientes estimados.

O ajuste dos parâmetros será realizado através do método de máxima verossimilhança. Utilizaremos o algoritmo de Newton-Raphson with ridging para encontrarmos o máximo da função de verossimilhança, já que o número de parâmetros a serem estimados não é tão grande e, portanto, o cálculo da matriz Hessiana não exige tanto esforço computacional. Para problemas maiores (acima de 400 parâmetros) é mais indicado utilizar-se um algoritmo Gradiente Conjugado. Maiores detalhes sobre os modelos de regressão logística linear podem ser encontrados em Hosmer and Lemeshow (2).

5.2. Regressão Logística Aditiva (GAM)

O modelo de regressão logística aditiva pode ser visto como uma extensão do modelo de regressão logística linear, onde o preditor ao invés de linear é aditivo, isto é:

$$g(\mathbf{X}) = \ln\left(\frac{\pi}{1-\pi}\right) = \alpha + \sum_{j=1}^{p} f_j(X_j) + \varepsilon \qquad (6)$$

onde as funções fj's são estimadas através de métodos não paramétricos, sendo estas funções arbitrárias univariadas suaves, uma para cada variável explanatória (v. Hastie, T. J. e Tibshirani [3]).

5.3. Redes Multi Layer Perceptron (MLP)

As redes MLP são um caso especial de arquitetura de rede neural artificial dentro da classe das redes Feed Forward. Redes MLP se caracterizam por possuírem uma camada de neurônios de entrada (input layer), uma ou mais camadas escondidas contendo um ou mais neurônios (hidden layers) e uma camada de saída (output layer). A figura 3 apresenta a arquitetura de rede MLP que será utilizada nesse projeto:

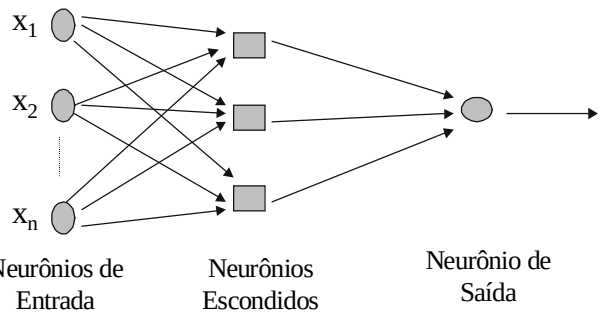

Figura 3

Alguns autores indicam a função logística (sigmoidal) como uma função de ativação com propriedades importantes para um problema de classificação (v. Bishop [4]). Visando enriquecer as comparações efetuadas nesse artigo, treinaremos redes MLP com funções de ativação logística e tangente hiperbólica, dadas pelas expressões abaixo, onde $g = \sum_{j=1}^{p} w_j x_j$ é a soma ponderada das entradas do neurônio.

Função Logística: $\frac{1}{1+e^{(-g)}}$

Função Tangente Hiperbólica: $1 - \frac{2}{1+e^{(2g)}}$

O treinamento da rede consiste em encontrar o vetor de pesos **w** que minimiza uma função de erro. Em nosso trabalho consideraremos o critério de máxima verossimilhança para estimativa do vetor **w**. Se considerarmos nosso problema de classificação com a variável resposta na forma dada por (2), de acordo com Bishop (4) poderemos considerar o valor de saída da rede como uma probabilidade condicional $y_i = P(C_1/\mathbf{x}_i)$,

onde $P(C_1/\mathbf{x}_i)$ é a probabilidade condicional da observação i pertencer à classe 1, dado o vetor de variáveis explanatórias $\mathbf{x}_i$, i=1,2,...,7339. Dessa forma, por termos apenas duas classes, podemos concluir facilmente que $1-y_i = P(C_2/\mathbf{x})$. Com esse delineamento podemos escrever a distribuição condicional de probabilidades da variável t como:

$$P(t_i / \mathbf{x}_i) = y_i^{t_i} (1 - y_i)^{1-t_i} \qquad (7)$$

A expressão (7) é conhecida como distribuição de probabilidade de Bernoulli. A função de verossimilhança é definida como a probabilidade conjunta de ocorrência de todas as observações na amostra, e é uma função do vetor **w**, dada por:

$$V = \prod_{i=1}^{7339} P(t_i / \mathbf{x}_i) = \prod_{i=1}^{7339} y_i^{t_i} (1 - y_i)^{1-t_i} \qquad (8)$$

A expressão (8) é obtida considerando-se que as observações do conjunto de dados são independentes, e representa a probabilidade conjunta de ocorrência do conjunto de dados obtido. Por questões de estabilidade numérica, o usual é trabalharmos com o logaritmo negativo de (8), o que nos permite obter a seguinte função de erro:

$$E = -\sum_{i=1}^{7339} [t_i \ln y_i + (1 - t_i) \ln(1 - y_i)] \qquad (9)$$

O aprendizado supervisionado, nesse caso, é simplesmente um problema de otimização irrestrito, onde a função objetivo é dada por (9) e a minimização é realizada sobre o vetor de parâmetros **w**. Utilizaremos os métodos do Gradiente Conjugado e de Levenbrerg-Marquardt para solução desse problema de minimização.

Visando garantir a capacidade de generalização da rede, estaremos efetuando o treinamento com um método de validação dos resultados conhecido por cross-validation (v. Bishop [4]), que consiste em tentarmos obter o menor valor de (9) para o conjunto de dados de validação.

5.4. Treinamento preliminar

Como já citado, a função (9) será minimizada pela aplicação dos métodos do Gradiente Conjugado e de Levenberg-Marquardt. Como quaisquer outros métodos numéricos para otimização, não existe garantia de que o mínimo global da função (9) será encontrado. Na maioria das situações, o método encontrará um ponto de mínimo local, e o que estamos interessados é em garantir que esse ponto seja um "bom" mínimo local, no sentido de possuir valor baixo para a função de erro. Buscando evitar a ocorrência de mínimos locais com valores altos da função de erro, vamos utilizar um método de treinamento preliminar da rede. O algoritmo de treinamento preliminar é o seguinte:

- Definir um valor M, que representa o número de vezes em que o treinamento preliminar será realizado, onde 5 < M < 20;
- Tomar uma amostra aleatória (pequena) dos dados (aproximadamente 20% dos dados originais);
- Para k = 1 até M:
  - Treinar a rede MLP por um número pequeno de iterações (máximo 20);
  - Armazenar o vetor de pesos final e o valor da f.o.
- Treinar a rede MLP com todos os dados, considerando como pesos iniciais o vetor **w** que corresponde ao menor valor da f.o. obtido no loop anterior.

Nesse artigo, o treinamento preliminar foi realizado para M=5, com 2.000 observações do conjunto de dados original e com número de iterações igual a 20.

5.5. Redes Bayesianas (BAY)

Para a estimativa dos parâmetros da rede bayesiana, utilizaremos a mesma função de erro dada pela equação (7). A diferença entre as duas abordagens consiste no fato de que a aprendizagem bayesiana considera uma distribuição a *priori* para os pesos. Isto proporciona à rede a capacidade de generalização sem precisar usar conjunto de dados de validação.

Neste trabalho tanto a distribuição a priori dos pesos (p(w)) quanto a distribuição condicional do vetor **X** ($p(X/C_1)$ e $p(X/C_2)$) serão consideradas Gaussianas, com a condição adicional de que a matriz de variância e covariância será igual para as duas classes. Com essas hipóteses, podemos obter a distribuição a posteriori dos pesos e a probabilidade a posteriori de uma observação pertencer à classe C_1 (v. Bishop [4]).

6. Comparação de Resultados

Utilizaremos três critérios para comparação entre os diversos modelos desenvolvidos:

a) % de classificação correta (considerando como ponto de corte p = 0.50);
b) estatística de Kolmogorov-Smirnov (KS);
c) construção do rating através do valor predito (probabilidade de inadimplência).

A estatística não paramétrica KS é definida como a maior distância entre duas distribuições de

probabilidade acumuladas, sendo utilizada para testar se a distribuição da variável de interesse é a mesma dentro de grupos diferentes (inadimplentes e não inadimplentes).

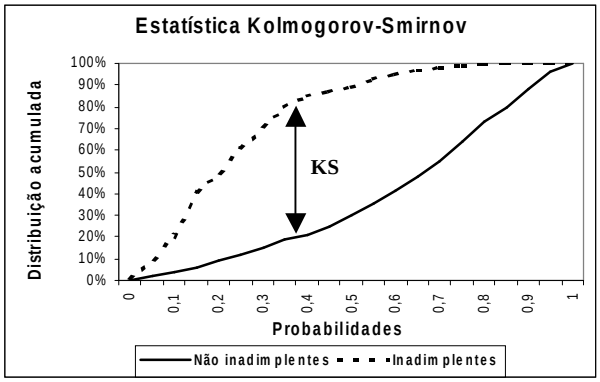

Figura 4

Nosso interesse será mensurar a capacidade de generalização dos modelos; logo os critérios de comparação serão medidos sobre a amostra de teste, por se tratar de um conjunto de dados ainda não apresentado a cada uma das técnicas utilizadas. As tabelas 2, 3 e 4 resumem os resultados da estatística KS e da porcentagem de classificação correta para os modelos desenvolvidos.

Tabela 2: Resultados das Regressões Logísticas

Regressão Logistica	% Classificação Correta	KS
Linear	84.76	70.36
Aditiva	85.46	71.64

Tabela 3 – Resultados das Redes Neurais MLP

Função de Ativação	Técnicas de Treinamento	Neurônios internos	Numero Épocas	% Classif. Correta	KS
Sigmoidal	LM	2	32	85,42	71,64
Sigmoidal	LM	3	54	84,51	69,68
Sigmoidal	LM	4	94	85,42	71,25
Sigmoidal	LM	5	82	84,51	70,06
Sigmoidal	GC	2	78	84,81	70,67
Sigmoidal	GC	3	78	85,26	71,64
Sigmoidal	GC	4	104	85,26	70,62
Sigmoidal	GC	5	65	84,81	70,98
Tg Hiperb.	LM	2	27	85,32	70,71
Tg Hiperb.	LM	3	83	84,86	71,14
Tg Hiperb.	LM	4	35	84,66	70,24
Tg Hiperb.	LM	5	97	84,15	68,68
Tg Hiperb.	GC	2	82	84,86	71,16
Tg Hiperb.	GC	3	55	84,99	70,09
Tg Hiperb.	GC	4	99	85,11	70,54
Tg Hiperb.	GC	5	105	85,11	70,21

LM = Levemberg-Marquardt GC = Gradiente Conjugado

Tabela 4: Resultado das Redes Neurais BAY

Nós na camada escondida	% Classificação Correta	KS
2	85,12	71,12
3	84,81	69,92
4	83,65	67,90
5	83,25	67,80

Podemos notar que praticamente todos os modelos testados conduzem a resultados bastante próximos nos primeiros dois critérios de comparação. Nos modelos MLP e BAY observa-se que o aumento do número de neurônios implica em uma menor capacidade de generalização da rede, fato que pode ser concluído pelo comportamento da estatística KS.

Para comparação entre as diversas abordagens utilizaremos a melhor rede treinada em cada uma das arquiteturas utilizadas:

- Rede MLP: 2 neurônios, ativação logística, otimização por Levenberg-Marquardt;
- Rede Bayesiana: 2 neurônios.

A tabela 5 compara os modelos RLG e GAM com as melhores redes.

Tabela 5: Resultados dos melhores modelos

Modelo Proposto	% Classificação Correta	KS
RLG	84,76	70,36
GAM	85,46	71,64
MLP	85,42	71,64
BAY	85,12	71,12

Novamente pode-se observar grande proximidade entre os resultados obtidos, porém existe uma desvantagem entre a RLG e as outras técnicas. Os resultados das técnicas MLP e GAM são praticamente iguais.

Como um terceiro critério de comparação, construímos um credit rating a partir do valor predito (probabilidade a posteriori de inadimplência) de cada modelo. A probabilidade a posteriori foi dividida em 10 categorias nomeadas com letras, conforme tabela 6.

Tabela 6 – Credit Rating

Rating	E(T)	Índice de Inadimplência			
		RLG	GAM	MLP	BAY
J	0.0 - 0.1	0,98	0,98	0,96	0,96
I	0.1 - 0.2	0,84	0,82	0,85	0,81
H	0.2 - 0.3	0,66	0,68	0,65	0,66
G	0.3 - 0.4	0,67	0,62	0,55	0,65
F	0.4 - 0.5	0,48	0,56	0,53	0,56
E	0.5 - 0.6	0,49	0,41	0,42	0,48
D	0.6 - 0.7	0,32	0,30	0,32	0,42
C	0.7 - 0.8	0,32	0,33	0,27	0,36
B	0.8 - 0.9	0,17	0,17	0,23	0,13
A	0.9 - 1.0	0,04	0,03	0,05	0,05

O índice de inadimplência em cada rating é calculado pela expressão:

$$I_i = \frac{QI_i}{QT_i} \qquad (10)$$

onde QI_i é a quantidade de operações inadimplentes no rating i e QT_i é a quantidade total de operações no rating i.

Pela distribuição acima podemos notar que os modelos neurais (MLP e BAY) foram os únicos que geraram taxas de inadimplências sempre decrescentes ao longo do valor predito, o que é importante na construção de um rating. A seguir comparamos graficamente as taxas de inadimplência entre os modelos.

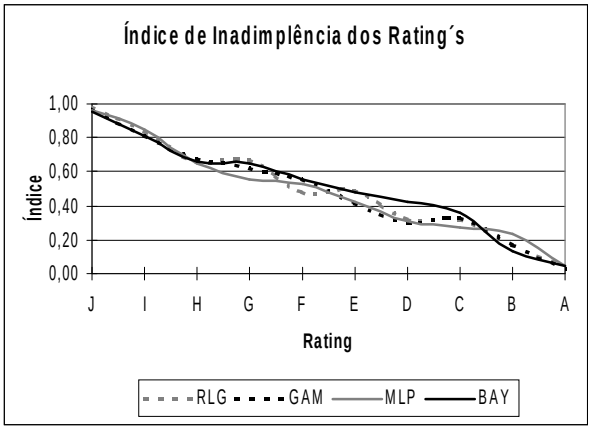

Figura 5

O modelo MLP apresenta um comportamento mais suave da taxa de inadimplência, o que também é uma qualidade desejável num modelo de rating. O modelo RLG é o que apresenta comportamento menos suave da taxa de inadimplência.

7. Conclusões

Em termos práticos nenhum dos modelos apresentou resultados significativamente melhores em comparação com os demais. Porém, o modelo de regressão logística linear (RLG) apresentou resultados inferiores aos demais em todos os critérios de comparação. O modelo de regressão logística aditiva (GAM) apresenta-se como uma evolução do modelo linear, com resultados bastante semelhantes à rede MLP. A rede bayesiana (BAY) ocupa posição intermediária, mas recomendaríamos seu uso somente quando poucos dados estiverem disponíveis para treinamento e não houver possibilidade de utilização de um conjunto de dados para validação. Os modelos MLP e GAM são os que apresentaram melhores resultados, ainda que com pouca diferença para os demais.

Como extensão desse estudo, sugerimos:

a) Utilização de redes auto construtivas, onde a arquitetura neural é construída a partir dos dados de treinamento por um método supervisionado de aprendizado;
b) Utilização de redes com aprendizado através de algoritmos genéticos;
c) Trabalhar com uma estrutura diferente para a variável resposta, onde seriam definidas diversas categorias de inadimplência, em lugar de uma variável binária, que foi o caso estudado nesse trabalho;
d) Particionar o conjunto de dados em segmentos distintos, ajustando uma rede MLP a cada segmento, o que possibilitaria a obtenção de regiões mais compactas, melhorando a capacidade de acerto das redes.

Referências

[1] Haykin, Simon; *Neural Networks – a comprehensive foundation*; 2ª edição; Prentice-Hall; 1999

[2] Hosmer, David W. e Lemeshow, Stanley ; Applied Logistic Regression ; John Wiley & Sons ; 1989

[3] Hastie, T. J. e Tibshirani; R. J. – *Generalized Additive Models*; 2ª edição; Chapman and Hall; 1990

[4] Bishop, Christopher M.; *Neural Networks for Pattern Recognition*; Oxford University Press; 1995

[5] Dorffner, Georg; *A generalized view on learning in feedforward neural networks*; Neural Computing Surveys 1, 1-47; 1997

[6] Jonhson, Richard A. e Wichern, Dean W.; Applied Multivariate Statistical Analysis; 3ª edição; Prentice-Hall; 1992