

FERNANDO ITANO

REDES NEURAS ARTIFICIAIS: UMA ALTERNATIVA PARA A
QUANTIFICAÇÃO E GERENCIAMENTO DO RISCO LEGAL

São Paulo
2013

FERNANDO ITANO

REDES NEURAS ARTIFICIAIS: UMA ALTERNATIVA PARA A
QUANTIFICAÇÃO E GERENCIAMENTO DO RISCO LEGAL

São Paulo
2013

FERNANDO ITANO

REDES NEURAIS ARTIFICIAIS: UMA ALTERNATIVA PARA A
QUANTIFICAÇÃO E GERENCIAMENTO DO RISCO LEGAL

Monografia apresentada à Escola
Politécnica da Universidade de São Paulo
para conclusão do curso de MBA em
Engenharia Financeira.

Área de Concentração: Risco Operacional

Orientador: Prof. Dr. Emilio Del Moral
Hernandez

São Paulo
2013

FERNANDO ITANO

REDES NEURAIS ARTIFICIAIS: UMA ALTERNATIVA PARA A
QUANTIFICAÇÃO E GERENCIAMENTO DO RISCO LEGAL

Monografia apresentada à Escola
Politécnica da Universidade de São Paulo
para conclusão do curso de MBA em
Engenharia Financeira.

Área de Concentração: Risco Operacional

Orientador: Prof. Dr. Emilio Del Moral
Hernandez

São Paulo
2013

AGRADECIMENTOS

À minha esposa pelo incondicional apoio e paciência.

Aos meus familiares e amigos pela compreensão nos momentos de ausência.

Ao meu orientador Prof. Dr. Emilio Del Moral Hernandez pela atenção, apoio, compreensão e incentivo durante todo o trabalho.

RESUMO

A quantificação e gerenciamento do risco legal em instituições financeiras têm requerido cada vez maior precisão, visto que o volume de ações cíveis contrárias é bastante relevante, pequenos desvios podem se transformar em diferenças impactantes no resultado financeiro da instituição. Embora existam diversas técnicas utilizadas atualmente na quantificação e gerenciamento deste risco, a utilização de redes neurais artificiais surge como uma alternativa que busca capturar complexas relações entre as diversas características das ações cíveis contrárias e o valor de sua perda. A maior precisão nos cálculos também propicia a construção de um orçamento mais realista, melhor planejamento financeiro e um melhor cálculo prospectivo dos resultados financeiros. Neste trabalho foram comparadas as técnicas de árvore de decisão, modelos lineares generalizados e redes neurais artificiais quanto à precisão no cálculo da perda esperada em ações cíveis contrárias. O modelo de redes neurais artificiais apresentou desempenho superior aos modelos de árvore de decisão e lineares generalizados na estimação da perda esperada, indicando que a utilização desta classe de modelos pode trazer benefícios na quantificação e gerenciamento do risco legal.

Palavras-chave: Risco Operacional. Risco Legal. Redes Neurais Artificiais. Quantificação de Riscos. Gerenciamento de Riscos.

ABSTRACT

Quantification and management of legal risk at financial institutions have been requiring increasingly more accuracy, since the volume of civil lawsuits is quite relevant and small deviations may cause important changes in financial results. Although several techniques have been used to quantify and manage legal risk, artificial neural networks come as an alternative capable of establish complex relationships between the civil lawsuits characteristics and the amount of loss. More accuracy in the calculations will also be helpful to get a better budget, financial planning and prospective financial results calculations. In this study we compared decision trees, generalized linear models and artificial neural networks regarding their accuracy in the calculation of expected loss with civil lawsuits. Artificial neural networks showed better performance than decision trees and generalized linear models on estimation of expected loss, indicating that the technique of artificial neural networks can bring benefits in quantification and management of legal risk.

Keywords: Operational Risk. Legal Risk. Artificial Neural Networks. Risk Quantification. Risk Management.

LISTA DE ILUSTRAÇÕES

Figura 1 - Método de cálculo simplificado do valor da perda esperada em ações cíveis.	17
Figura 2 - Método de cálculo com explicitação dos componentes do valor da perda esperada em ações cíveis.....	17
Figura 3 - Exemplo de Árvore de Decisão para a classificação de ações cíveis.....	19
Figura 4 - Relação linear entre uma variável explicativa e uma variável resposta	22
Figura 5 - Representação de uma rede neural artificial com 2 entradas, 6 neurônios na camada oculta e 2 saídas. Círculos representam os neurônios e as setas representam os pesos sinápticos.....	23
Figura 7 - Utilização das técnicas de AD, MLG e RNA para obtenção do valor da perda esperada com ações cíveis massificadas.....	27
Figura 8 - Árvore de Decisão obtida para a estimação da Probabilidade de Perda Financeira em ações cíveis massificadas. As regras acima de cada nó se referem ao ramo esquerdo.	30
Figura 9 - Árvore de Decisão obtida para a estimação da Probabilidade de Condenação em ações cíveis massificadas. Como comentado na figura 7, as regras acima de cada nó se referem ao ramo esquerdo.....	33
Figura 10 - Árvore de Decisão obtida para a estimação do Valor da Perda com Acordo em ações cíveis massificadas. Como comentado na figura 7, as regras acima de cada nó se referem ao ramo esquerdo.	36
Figura 11 - Gráficos Quantil-Quantil do Valor da Perda com Acordo e Raiz do Valor da Perda com Acordo com a distribuição Gamma.	37
Figura 12 - Árvore de Decisão obtida para a estimação do Valor da Perda com Condenação em ações cíveis massificadas. Como comentado na figura 7, as regras acima de cada nó se referem ao ramo esquerdo.....	40
Figura 13 - Gráficos Quantil-Quantil do Valor da Perda com Condenação e o logaritmo natural do Valor da Perda com Condenação com a distribuição Normal...	41

LISTA DE TABELAS

Tabela 1 - Funções de ligação canônica em Modelos Lineares Generalizados.....	21
Tabela 2 - Resumo das características das técnicas avaliadas.	26
Tabela 3 - Informações disponíveis para a estimação do valor da perda esperada em ações cíveis massificadas.	28
Tabela 4 - Parâmetros estimados do MLG de Probabilidade de Perda Financeira em ações cíveis massificadas.	31
Tabela 5 - Pesos sinápticos ajustados para a RNA de Probabilidade de Perda Financeira em ações cíveis massificadas.	32
Tabela 6 - Desempenho das técnicas avaliadas na estimação da Probabilidade de Perda Financeira em ações cíveis massificadas.	32
Tabela 7 - Parâmetros estimados do MLG de Probabilidade de Condenação em ações cíveis massificadas.	34
Tabela 8 - Pesos sinápticos ajustados para a RNA de Probabilidade de Condenação em ações cíveis massificadas.	35
Tabela 9 - Desempenho das técnicas avaliadas na estimação da Probabilidade de Condenação em ações cíveis massificadas.	35
Tabela 10 - Parâmetros estimados do MLG de Valor da Perda com Acordo em ações cíveis massificadas.	38
Tabela 11 - Pesos sinápticos ajustados para a RNA de Probabilidade de Condenação em ações cíveis massificadas.	38
Tabela 12 - Desempenho das técnicas avaliadas na estimação do Valor da Perda com Acordo em ações cíveis massificadas.	39
Tabela 13 - Parâmetros estimados do MLG de Valor da Perda com Condenação em ações cíveis massificadas.	41
Tabela 14 - Pesos sinápticos ajustados para a RNA de Valor da Perda com Condenação em ações cíveis massificadas.	42

Tabela 15 - Desempenho das técnicas avaliadas na estimação do Valor da Perda com Condenação em ações cíveis massificadas.42

Tabela 16 - Desempenho das técnicas avaliadas na estimação da Perda Esperada com ações cíveis massificadas.43

LISTA DE SIGLAS

AD	Árvore de Decisão
CSBB	Comitê de Supervisão Bancária de Basileia
MLG	Modelos Lineares Generalizados
RNA	Redes Neurais Artificiais
RO	Risco Operacional

SUMÁRIO

1. Introdução.....	12
1.1. Objetivo	12
1.2. Motivação e Justificativa	12
1.3. Metodologia.....	12
1.4. Estrutura do trabalho.....	13
2. Risco Legal como componente de Risco Operacional	14
2.1. Importância do gerenciamento adequado do Risco Operacional.....	14
2.2. Risco Legal	15
2.2.1. Impacto financeiro pela materialização do Risco Legal	15
2.2.2. Avaliação do Risco Legal	16
3. Metodologias utilizadas na avaliação do risco legal	17
4. Técnicas utilizadas na estimação da perda esperada	19
4.1. Árvores de Decisão (AD).....	19
4.2. Modelos Lineares Generalizados (MLG).....	21
4.3. Redes Neurais Artificiais (RNA)	23
5. Comparação das técnicas de modelagem.....	25
5.1. Relações complexas entre variáveis independentes e dependentes	25
5.2. Interpretação dos parâmetros	25
5.3. Suposições e premissas	26
5.4. Intensidade computacional.....	26
5.5. Comparação das técnicas na estimação da perda esperada.....	27
5.6. Amostra.....	27
5.7. Software utilizado	28
5.8. Estimação do valor da Perda Esperada em ações cíveis massificadas	29

5.8.1. Probabilidade de Perda Financeira.....	29
5.8.2. Probabilidade de Condenação	32
5.8.3. Valor da Perda com Acordo.....	36
5.8.4. Valor da Perda com Condenação.....	39
5.8.5. Consolidação das Estimações.....	42
6. Conclusões	44
REFERÊNCIAS.....	45
APENDICE A – Programas desenvolvidos em R.....	47

1. Introdução

A excelência no gerenciamento do risco legal possibilita às instituições conhecer mais profundamente seus pontos de fragilidade e atuar na melhoria de processos, sistemas e infraestrutura. Melhorias dessas naturezas trazem maior eficiência operacional, segurança perante os *stakeholders*, além de influenciar positivamente a imagem da instituição perante seus atuais e potenciais clientes. Por esses e tantos outros motivos, a busca de novas técnicas e metodologias que permitam a quantificação do risco de forma mais precisa é fundamental. Neste sentido, as redes neurais artificiais são apresentadas como uma alternativa aos consagrados métodos de regressão e árvores de decisão.

1.1. Objetivo

Este trabalho tem como objetivo comparar os modelos de redes neurais artificiais as já consagradas técnicas de árvores de decisão e modelos lineares generalizados na quantificação do risco legal.

1.2. Motivação e Justificativa

Por sua complexidade e relevância, o risco legal tem se tornado um dos riscos mais importantes das grandes instituições brasileiras, e a excelência em seu gerenciamento pode significar um considerável aumento na eficiência operacional das instituições e em sua vantagem competitiva.

1.3. Metodologia

São comparados os resultados obtidos pela utilização dos quatro modelos em cada uma das técnicas: árvore de decisão, modelos lineares generalizados e redes neurais artificiais, na quantificação da perda esperada com ações cíveis contrárias massificadas. O primeiro modelo estima a probabilidade de perda, o segundo a probabilidade de condenação, o terceiro o valor da perda em caso de acordo e o quarto e último o valor da perda em caso de condenação.

1.4. Estrutura do trabalho

O texto está distribuído em 6 capítulos, descritos brevemente a seguir:

No capítulo 1 são apresentados o objetivo, motivação e justificativa, e metodologia.

No capítulo 2 é apresentada uma breve introdução ao risco operacional e mais especificamente ao risco legal com informações sobre os impactos financeiros decorrentes deste risco e suas formas de avaliação.

No capítulo 3 são apresentadas duas metodologias para a estimação da perda esperada com ações cíveis massificadas e as vantagens na utilização de cada uma delas.

No capítulo 4 é apresentado um resumo das técnicas avaliadas na quantificação do risco legal associado às ações cíveis massificadas.

No capítulo 5 é apresentada a metodologia e os resultados obtidos na comparação de desempenho das técnicas avaliadas.

No capítulo 6 são apresentadas as conclusões e propostas de continuação deste trabalho.

2. Risco Legal como componente de Risco Operacional

2.1. Importância do gerenciamento adequado do Risco Operacional

O interesse por risco operacional tem aumentado muito nos últimos dez anos, principalmente após diversos eventos de fraude que resultaram na falência de alguns grandes bancos como o Banco Barings que inspirou o filme “A Fraude” (Rogue Trader – 1999). Neste filme são apresentados os fatos que levaram o Banco Bearings, fundado em 1762, ir à falência em 1995 após um funcionário esconder os prejuízos causados em operações mal sucedidas no mercado futuro e acumular um prejuízo de £827 milhões (CRUZ, 2002).

Com o objetivo de tornar o sistema financeiro mundial mais robusto a estes eventos, o Comitê de Supervisão Bancária de Basileia (CSBB), com sede na cidade de Basileia na Suíça, incluiu na segunda versão de seu acordo de capitais, também conhecido como Novo Acordo da Basileia ou simplesmente Basileia II, regras de capital mínimo requerido, orientações para o processo de supervisão dos órgãos reguladores e definições que pautam a disciplina de mercado para o risco operacional¹, definindo-o da seguinte forma: “Risco Operacional é definido como o risco de perdas resultantes de falhas de processos internos, pessoas e sistemas ou eventos externos. Esta definição inclui o risco legal, mas exclui o risco estratégico e reputacional.” (tradução nossa).

Dessa forma, risco operacional está relacionado a erros operacionais que gerem perdas às instituições. No Brasil o Novo Acordo de Basileia foi interpretado pelo Banco Central do Brasil (BACEN), supervisor das instituições financeiras brasileiras, que emitiu em junho de 2006 a resolução 3.380 determinando a implementação de uma estrutura responsável pelo gerenciamento do risco operacional nas instituições autorizadas por ele a funcionar.

¹ Na versão anterior eram abordados apenas os riscos de crédito e mercado.

2.2. Risco Legal

Sendo o risco legal uma componente do risco operacional com relevância considerável no cenário brasileiro, a resolução 3.380 informa no art. 2º parágrafo 1º:

A definição de que trata o caput inclui o risco legal associado à inadequação ou deficiência em contratos firmados pela instituição, bem como a sanções em razão de descumprimento de dispositivos legais e a indenizações por danos a terceiros decorrentes das atividades desenvolvidas pela instituição.

O risco legal têm se apresentado às instituições na forma de ações cíveis movidas, em sua maior parte por clientes, com os motivos mais diversos: questionar a qualidade dos serviços prestados ou produtos vendidos, requerer indenizações por perdas e danos causados por falha da instituição entre outros. Ainda que as causas de origem das ações cíveis sejam fundamentalmente falhas de processos, pessoas, sistemas ou infraestrutura, observa-se no Brasil um volume considerável de ações iniciadas com o objetivo de questionar as condições contratuais e cobrança de taxas e tarifas.

2.2.1. Impacto financeiro pela materialização do Risco Legal

Do ponto de vista financeiro, é relevante destacar que o prazo de encerramento de uma ação cível é incerto e em muitos casos ultrapassa seis meses, sendo prudente que as instituições façam o reconhecimento contábil de eventuais perdas com estas ações no momento em que se iniciam. Essa prática, obrigatória às companhias de capital aberto², visa aumentar a solidez do sistema financeiro fazendo com que as instituições tenham capital suficiente para cumprir suas obrigações, e é regulamentada pela Comissão de Valores Mobiliários (CVM) e o Instituto Brasileiro dos Auditores Independentes (IBRACON), que definem na Instrução de número 489 (CVM 489) e na Norma e Procedimento de Contabilidade número 22 (NPC 22) os critérios de reconhecimento e mensuração aplicáveis a provisões e contingências passivas.

Entre os critérios apresentados para a avaliação do risco legal, a NPC 22 apresenta três parâmetros para a probabilidade de perda com contingências passivas:

² Companhias de capital aberto são aquelas que possuem ações negociadas em bolsas de valores.

Provável, Possível e Remota, em ordem decrescente de probabilidade de materialização do risco, determinando que apenas aqueles avaliados como Provável devam ter provisão. Além disso, informa que o montante reconhecido como provisão deve ser a melhor estimativa do desembolso exigido para liquidar a obrigação.

2.2.2. Avaliação do Risco Legal

Para avaliar adequadamente o risco legal envolvido nas ações contrárias, as instituições, em geral, as dividem em duas categorias:

- **Ações diferenciadas:** as ações são avaliadas individualmente quanto à probabilidade de perda e ao valor em risco caso a perda se materialize. Contempla um pequeno volume de ações que possuem valores elevados, segundo os critérios da própria instituição;
- **Ações massificadas:** as ações são avaliadas quanto à probabilidade de perda e ao valor em risco em grandes grupos, em geral, utilizando-se de modelos estatísticos. Contempla grandes volumes de ações que possuem valores menores do que os das ações diferenciadas.

Sendo a provisão considerada contabilmente como uma despesa, divergências entre a avaliação do risco legal e o valor de desembolso efetivo geram variações indesejadas nos resultados das instituições. Além disso, o excesso de provisão reflete a ineficiência no gerenciamento do risco operacional, pois aumenta as despesas da instituição, reduzindo o capital disponível para a realização de negócios e, portanto sua competitividade.

3. Metodologias utilizadas na avaliação do risco legal

Existem diferentes modelos de avaliação do risco legal em ações massificadas e todos eles têm como objetivo a estimação da perda esperada em seu desfecho. As metodologias mais simples buscam estimar diretamente a perda esperada a partir a média dos valores das perdas em ações cíveis já encerradas. Esta metodologia não permite a identificação de fatores que podem contribuir com a variação da perda esperada, como por exemplo, o aumento no número de acordos ou a redução dos valores pagos em condenações.

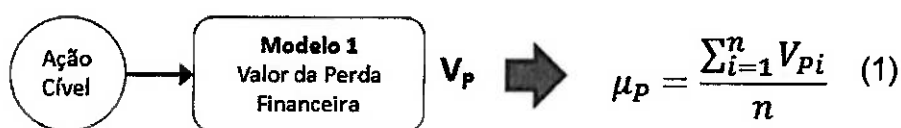


Figura 1 - Método de cálculo simplificado do valor da perda esperada em ações cíveis com apenas 1 modelo.

Onde:

V_{pi} : Valor da Perda da i -ésima ação cível

n : número de ações cíveis utilizadas na estimação da perda esperada

Para permitir que esses fatores sejam compreendidos e utilizados na gestão do risco legal, preferiu-se utilizar uma metodologia que busca a estimação de quatro fatores que influenciam diretamente o valor da perda esperada em ações: probabilidade de perda, probabilidade de acordo ou condenação, valor da perda esperada em acordos e valor da perda esperada em condenações.

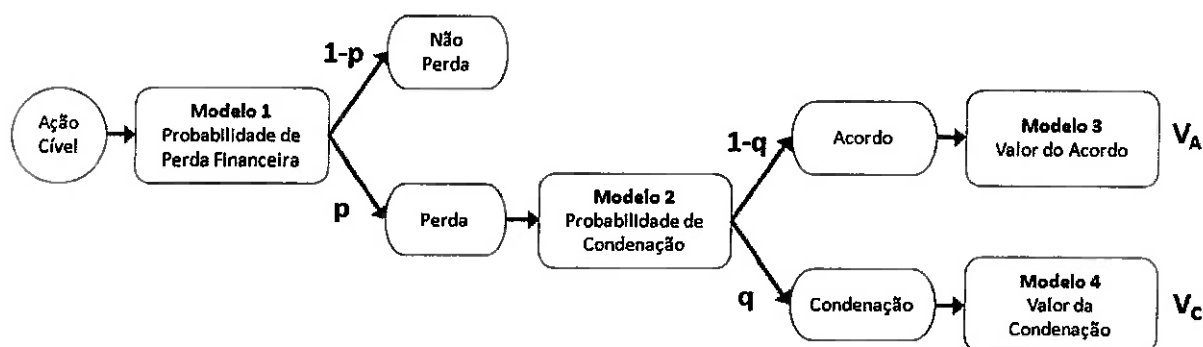


Figura 2 - Método de cálculo com explicitação dos componentes do valor da perda esperada em ações cíveis contendo 4 modelos.

Dessa forma o valor da perda esperada em ações cíveis será:

$$\mu_p = p \cdot [(1 - q) \cdot V_A + q \cdot V_C] \quad (2)$$

Onde:

p: Probabilidade de Perda Financeira

q: Probabilidade de Perda Financeira por Condenação

V_a: Valor da perda com Acordo

V_c: Valor da perda com Condenação

Por permitir uma melhor compreensão de quais fatores tem maior influencia no risco legal incorrido pela instituição e permitir ao gestor a tomada de decisão com maior qualidade para a mitigação deste risco, optou-se pela segunda metodologia.

4. Técnicas utilizadas na estimação da perda esperada

Entre as diversas técnicas de modelagem existentes, foram selecionadas as de Árvores de Decisão, Modelos Lineares Generalizados e Redes Neurais Artificiais por representarem, respectivamente, facilidade na compreensão do modelo, consagração em modelagem estatística e inovação no reconhecimento de padrões.

4.1. Árvores de Decisão (AD)

As Árvores de Decisão (AD) têm como principal característica a estruturação de regras de classificação a partir das variáveis independentes (BREIMAN et al., 1984), ordenadas hierarquicamente de acordo com sua correlação com a variável dependente, de forma a obter grupos semelhantes, quanto à variável dependente. A forma de estruturação dessas regras permite que as AD consigam lidar bem com não linearidades nos fenômenos observados. Essas regras podem ser utilizadas para obter uma maior compreensão do fenômeno observado ou realizar a classificação de uma unidade amostral não observada anteriormente. Na composição das AD, denominam-se nós os pontos de partição das observações em dois ou mais grupos a partir das variáveis independentes, e folhas as terminações que contêm as observações com maior similaridade em relação a variável dependente.

A figura 3 ilustra um exemplo de classificação das ações cíveis em “PERDA” e “NÃO PERDA” utilizando as variáveis: Motivo pelo qual a ação foi movida e UF em que ela está sendo julgada, abordado pela técnica de AD.

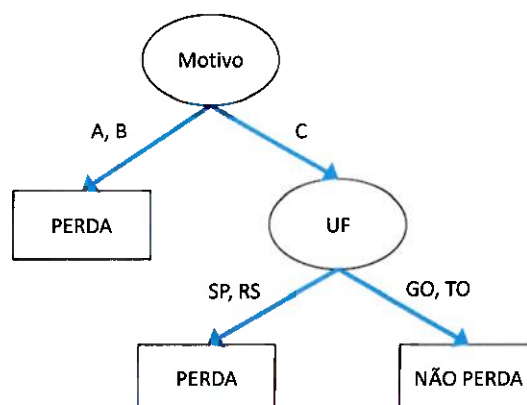


Figura 3 - Exemplo de Árvore de Decisão para a classificação de ações cíveis.

Neste exemplo, os Motivos A e B levam diretamente a uma classificação de “PERDA” enquanto o motivo C considera a variável UF, que classifica SP e RS como “PERDA” e GO e TO como “NÃO PERDA”.

Em problemas de classificação, o resultado obtido com AD é a probabilidade de uma observação ser classificada em um determinado nível da variável dependente a partir das informações das variáveis independentes. No exemplo acima foram consideradas como Perda as ações cíveis que apresentaram probabilidade de perda superior a 50% e como Não Perda aquelas que apresentaram probabilidade de perda inferior a 50%.

Em problemas de regressão, quando a variável dependente é numérica, as folhas da AD indicarão a média dessa variável, lembrando que os algoritmos de construção buscarão criar grupos homogêneos quanto à variável dependente.

Mesmo em problemas que envolvam diversas variáveis independentes, as AD, em geral, apresentam-se de forma bastante clara e de fácil compreensão.

O algoritmo utilizado para a construção das AD deste trabalho foi parametrizado da seguinte forma:

- N° mínimo de observações para realizar o particionamento: 20
- N° mínimo de observações em um nó terminal: 7
- Fator mínimo de melhoria do ajuste para continuação do particionamento: 1%
- N° máximo de particionamentos da raiz até o nó terminal: 30

A definição destes parâmetros, juntamente com o método de validação cruzada, evita que as AD sejam construídas de forma a obter um ajuste próximo ao perfeito para as unidades amostrais utilizadas, mas percam sua capacidade de generalização³.

³ Esse efeito é conhecido também como *Overfitting*.

4.2. Modelos Lineares Generalizados (MLG)

Análises estatísticas clássicas como a Regressão Linear (RL) ou Modelos Lineares Generalizados (MLG) são amplamente utilizados no estudo de relações existentes entre uma variável dependente e uma ou mais variáveis independentes. Os modelos RL estabelecem uma relação linear entre as variáveis independentes e a variável dependente, que torna mais intuitiva a interpretação dos parâmetros, mas nem sempre corresponde à realidade do fenômeno que se estuda. Estes modelos possuem também suposições quanto aos resíduos, que devem ter distribuição Normal com média igual à zero, variância constante e não apresentarem uma relação com a variável dependente (independência) (MORETTIN; BUSSAB, 2011).

Os MLG e os modelos RL possuem como principais diferenças:

- **Distribuições de Probabilidade:** Os modelos de RL são baseados exclusivamente na distribuição de probabilidades Normal, enquanto os MLG podem se utilizar de todas as distribuições de probabilidades da Família Exponencial, composta pelas distribuições Normal, Exponencial, Gamma, Qui-Quadrado, Beta, Binomial, Poisson entre outras.
- **Função de Ligação:** a função de ligação define a relação entre o preditor linear e a média da distribuição de probabilidades utilizada. Nos modelos de RL a função de ligação é a identidade, ou seja, o preditor linear é a própria média da distribuição Normal. Nos MLG as funções de ligação naturais, ou canônicas, estão associadas a cada distribuição de probabilidades.

Tabela 1 - Funções de ligação canônica em Modelos Lineares Generalizados.

Distribuição	Normal	Binomial	Gamma	Normal Inversa
Ligação	$\mu = \eta$	$\log\left(\frac{\mu}{1-\mu}\right) = \eta$	$\mu^{-1} = \eta$	$\mu^{-2} = \eta$

Para garantir a validade dos resultados obtidos com os MLG é necessária a utilização de algumas técnicas de diagnóstico (PAULA, 2012). Estas técnicas visam avaliar a existência de observações influentes, qualidade do ajuste e eventuais violações nas suposições do modelo (MCCULLAGH; NELDER, 1989).

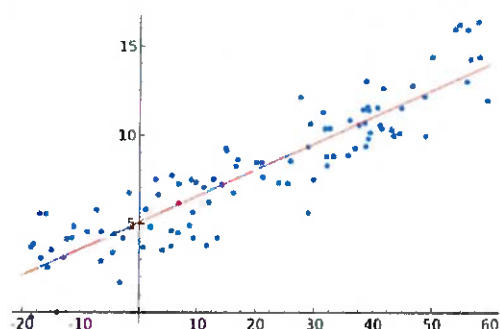


Figura 4 - Relação linear entre uma variável explicativa e uma variável resposta⁴

Para a estimação da Probabilidade de Perda e Probabilidade de Condenação com MLG, em que a variável resposta é binária, foi utilizada a distribuição Binomial com a função de ligação *logit*, modelos estes também conhecidos como Modelos de Regressão Logística.

Na estimação do Valor da Perda com Acordo e Valor da Perda com Condenação foram avaliadas as distribuições Normal, Gamma e Gaussiana Inversa, com transformações da variável dependente para obtenção do melhor ajuste e não violação das suposições dos modelos.

Em ambos os casos foram utilizados apenas os efeitos aditivos e o teste de significância dos parâmetros para selecionar as variáveis independentes relevantes. Foram realizadas também as análises de diagnóstico dos modelos ajustados para avaliar a existência de observações influentes, qualidade do ajuste e a não violação das suposições dos modelos.

⁴ Fonte da figura: http://en.wikipedia.org/wiki/File:Linear_regression.svg (12/01/2013)

4.3. Redes Neurais Artificiais (RNA)

As Redes Neurais Artificiais (RNA) pertencem à classe de ferramentas de Inteligência Artificial que, assim como o sistema de inteligência biológico, tem a capacidade de utilizar informações existentes para aprender, por exemplo, a reconhecer padrões e imitar funções (HAYKIN, 2001). Têm como característica a grande capacidade de identificar complexas relações multidimensionais entre as variáveis independentes (entradas) e as variáveis dependentes (saídas), tendo como aplicações típicas o reconhecimento de padrões, interfaces homem-máquina naturais, agrupamento de dados multidimensionais, previsão de séries temporais e a modelagem de sistemas não lineares.

Com seu funcionamento inspirado no do cérebro humano, estrutura computacional mais complexa conhecida pelo homem, as RNA são compostas de neurônios artificiais, que assim como os biológicos, são conectados uns aos outros formando uma rede que transmite as informações em diferentes níveis e de forma paralela (RIPLEY, 1996).

Uma RNA é composta de três elementos principais:

1. Entrada: características das observações representadas pelas variáveis independentes utilizadas na modelagem.
2. Camadas Ocultas: neurônios artificiais responsáveis por “aprender” a relação entre as variáveis independentes e as variáveis dependentes.
3. Saída: resultado do processamento das variáveis independentes pelos neurônios das camadas ocultas.

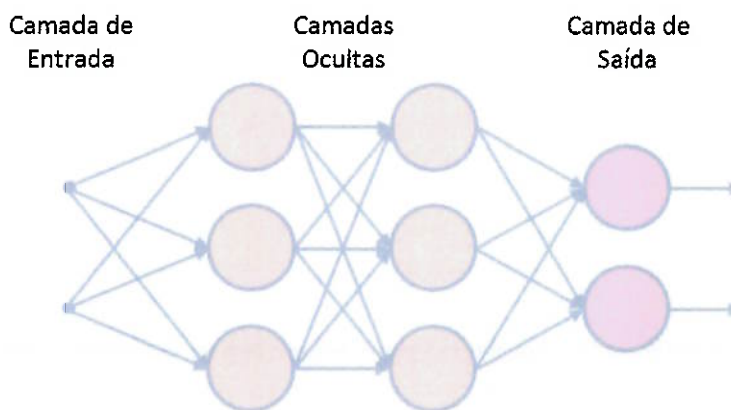


Figura 5 - Representação de uma rede neural artificial com 2 entradas, 6 neurônios na camada oculta e 2 saídas. Círculos representam os neurônios e as setas representam os pesos sinápticos.

A figura 5 ilustra uma RNA do tipo Perceptron de Multicamadas (*Multilayer Perceptron* - MLP) com dois neurônios na camada de entrada, três neurônios na primeira camada oculta, três neurônios na segunda camada oculta e dois neurônios na camada de saída.

Existem diversos métodos de treinamento de uma RNA, o mais conhecido e o que será utilizado neste trabalho é o de aprendizagem supervisionada por retropropagação (*backpropagation*). Neste método os exemplos são fornecidos à RNA, que recebe as informações de entrada, processa os sinais nas camadas ocultas devolve a informação na saída, que é comparada com a informação de saída dos exemplos previamente fornecidos. Nessa comparação são calculadas as diferenças que orientarão como os pesos sinápticos devem ser ajustados de forma que a RNA consiga, a partir das informações de entrada (variáveis independentes), reproduzir as informações de saída (variáveis dependentes).

Ao contrário das AD e MLG os pesos sinápticos não possuem interpretação prática de seus valores, embora seja bastante simples identificar como uma RNA já treinada processa a informação de entrada para produzir a informação de saída, bastando para isso fornecer as informações de entrada de forma sistemática e observar o resultado do processamento pela rede neural artificial.

Foram avaliadas neste trabalho as RNA com estrutura MLP de uma camada oculta. O número de neurônios na camada oculta foi escolhido de forma que a Taxa de Erros de Classificação (TEC) ou a Raiz do Erro Quadrático Médio (REQM) fossem as menores possíveis com o menor número de neurônios na camada oculta. Como o treinamento de uma rede neural utiliza o sorteio aleatório dos pesos sinápticos iniciais, foram realizados até 500 ensaios com sorteios aleatórios dos pesos sinápticos iniciais para evitar que o processo de otimização não fosse atingido devido à existência de mínimos locais.

5. Comparação das técnicas de modelagem

As técnicas consideradas podem ser comparadas com relação a quatro principais características: capacidade de identificar relações complexas entre as variáveis independentes e dependentes, facilidade na interpretação dos parâmetros, existência de suposições e premissas, e intensidade computacional.

5.1. Relações complexas entre variáveis independentes e dependentes

As AD, por sua característica de particionamento hierárquico, possuem a capacidade de identificar relações não lineares. Porém não são capazes de imitar uma função contínua, uma vez que isso requereria infinitos particionamentos. Já os modelos RL e MLG são capazes de imitar funções contínuas, porém limitam-se as funções lineares. Ainda que seja possível transformar as variáveis independentes e/ou dependentes para tornar uma relação não linear em linear e então aplicar esta técnica, muitas vezes as transformações podem distorcer os parâmetros estimados, levando a conclusões equivocadas. As RNA por outro lado são capazes de identificar complexas relações entre as variáveis independentes e dependentes, lineares ou não lineares, de forma contínua naturalmente, sendo, portanto a técnica mais eficaz neste aspecto.

5.2. Interpretação dos parâmetros

As regras de particionamento obtidas nas AD, em geral, são de fácil compreensão. Porém caso a AD apresente muitos particionamentos, a compreensão de todas as regras pode não ser trivial. Os modelos RL e MLG são os que possuem a mais fácil interpretação dos parâmetros do modelo, devido, principalmente, a premissa de que a relação entre as variáveis independentes e dependentes é linear ou possuem uma função bem definida. Os pesos sinápticos de uma RNA não podem ser facilmente interpretados, pois cada um deles representa apenas uma parte de um processamento complexo e paralelo da informação, sendo neste aspecto a técnica que apresenta menor possibilidade de interpretação dos parâmetros.

5.3. Suposições e premissas

O método de particionamento recursivo das AD não exigem suposições ou premissas a respeito das variáveis independentes e/ou dependentes. Os modelos RL e MLG por outro lado possuem diversas suposições como: normalidade, independência e homocedasticidade dos resíduos nos modelos de RL. Violações nessas suposições podem invalidar as estimações e, portanto suas conclusões. A avaliação e eventual correção para não violação das suposições não é trivial e requer muitas análises. As RNA, assim como as AD, não possuem suposições sobre as variáveis independentes e/ou dependentes, sendo, portanto, juntamente com as AD, as técnicas mais robustas neste aspecto.

5.4. Intensidade computacional

A intensidade computacional no particionamento recursivo de uma AD depende bastante da quantidade de observações e particionamentos realizados, mas de uma forma geral, é bastante baixa, pois não requer a utilização de complexos algoritmos de otimização. Os modelos de RL e MLG, nos casos mais complexos, onde a estimação dos parâmetros depende da otimização numérica de uma equação, requer baixo poder computacional, pois ainda que exista um processo de otimização numérica envolvido, este normalmente não possui muitas dimensões. Nas RNA o processo de otimização numérica é intenso, pois uma rede como a apresentada no item 4.3 possui 21 pesos sinápticos para serem ajustados simultaneamente de forma ótima. Neste aspecto, a técnica de RNA é a que apresenta maior necessidade de poder computacional.

Tabela 2 - Resumo das características das técnicas avaliadas.

Técnica	Capacidade de identificar relações complexas	Interpretação dos parâmetros	Suposições e Premissas	Intensidade Computacional
AD	Média	Média	Não	Baixa
MLG	Média	Alta	Sim	Média
RNA	Alta	Baixa	Não	Alta

5.5. Comparação das técnicas na estimação da perda esperada

A estimação da perda esperada em ações cíveis massificadas, como definido no capítulo 3, utilizou a segunda metodologia que consiste na estimação da Probabilidade de Perda Financeira, Probabilidade de Perda Financeira por Condenação, Valor da Perda com Acordo e Valor da Perda Condenação de forma separada. Ao final, os resultados desses quatro modelos foram combinados para estimar a Perda Esperada com ações cíveis massificadas.

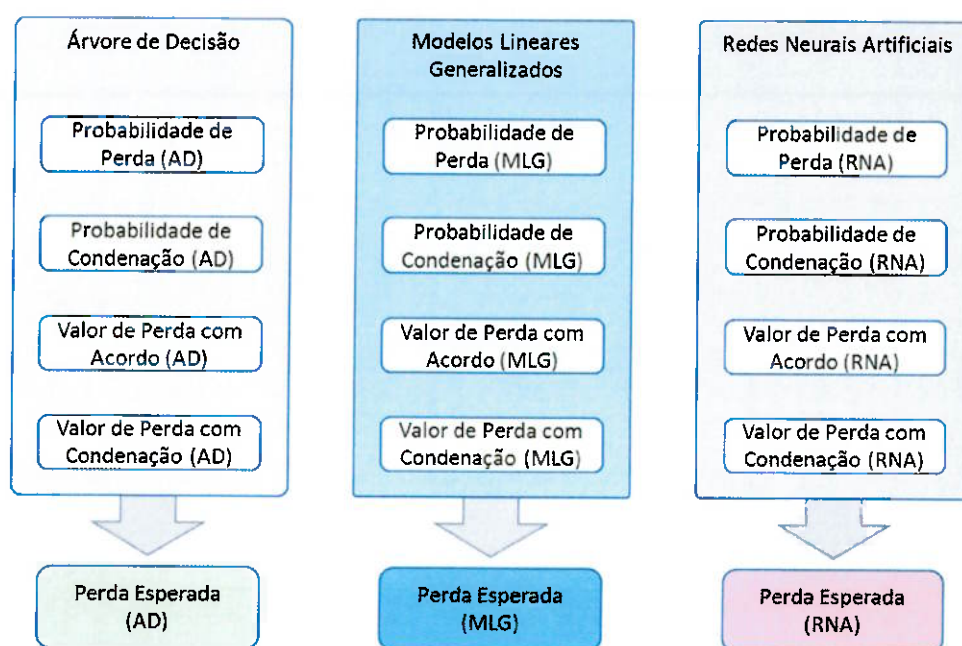


Figura 6 - Utilização das técnicas de AD, MLG e RNA para obtenção do valor da perda esperada com ações cíveis massificadas.

5.6. Amostra

Para a estimação da Probabilidade de Perda Financeira, Probabilidade de Perda Financeira por Condenação, Valor da Perda com Acordo e Valor da Perda com Condenação foi realizada uma amostragem aleatória simples (BOLFARINE; BUSSAB, 2006) da base de dados de ações cíveis massificadas encerradas de uma grande instituição financeira brasileira contendo 1.800 observações, quatro variáveis independentes, sendo três delas categóricas e uma numérica, e três variáveis dependentes, sendo duas delas indicadoras e uma numérica.

As variáveis independentes categóricas tiveram suas classes mascaradas a fim de preservar o sigilo das informações. Nas variáveis numéricas, dependentes e independentes, foi aplicada uma transformação linear também para preservar o sigilo das informações.

Tabela 3 - Informações disponíveis para a estimação do valor da perda esperada em ações cíveis massificadas.

Variável	Descrição	Domínio
INPUT_A	Classificação da ação cível pelo departamento jurídico segundo o atributo A	Classes: A01 e A02
INPUT_B	Classificação da ação cível pelo departamento jurídico segundo o atributo B	Classes: B01, B02 e B03
INPUT_C	Classificação da ação cível pelo departamento jurídico segundo o atributo C	Classes: C01, C02 e C03
INPUT_D	Tempo dispendido entre o início e encerramento da ação	Valor Contínuo
TARGET_A	Variável indicadora de ocorrência de perda na ação cível	1: Perda 0: Não Perda
TARGET_B	Variável indicadora de ocorrência de condenação na ação cível	1: Condenação 0: Acordo
TARGET_C	Valor total da perda com a ação cível	Valor

5.7. Software utilizado

O trabalho de modelagem foi realizado por meio do *software* R, um ambiente computacional estatístico que oferece como principais vantagens:

- Grande quantidade de bibliotecas desenvolvidas e validadas por inúmeros membros colaboradores. Os códigos fontes das funções existentes nas bibliotecas são também disponibilizados para avaliação.
- Construção de gráficos de alta qualidade e grande capacidade de customização.
- Linguagem de programação amigável e similar a outros *softwares* estatísticos e científicos (S-plus, Matlab, Octave entre outros).
- Distribuição gratuita sob os termos da Licença Geral Pública da Free Software Foundation (GPL).

Além das bibliotecas incluídas no sistema base, foram utilizadas também:

- *rpart*: Desenvolvimento de modelos de árvore de decisão
- *glm*: Ajuste e diagnóstico de modelos lineares generalizados
- *nnet*: Treinamento e aplicação de redes neurais artificiais

- car: Construção de gráficos QQ-Plot para diagnóstico de adequação de distribuições
- nortest: Testes de normalidade

O software e as bibliotecas podem ser encontrados no sítio <http://www.r-project.org>.

5.8. Estimação do valor da Perda Esperada em ações cíveis massificadas

Cada um dos doze modelos foi ajustado individualmente, buscando sempre o melhor ajuste segundo cada uma das técnicas avaliadas. Para avaliação da capacidade preditiva dos modelos, foi utilizada a técnica de validação cruzada com 70% da amostra inicial utilizada para o treinamento do modelo e os 30% restantes utilizados em sua validação.

A comparação de desempenho dos modelos para a estimação da Probabilidade de Perda e Probabilidade de Condenação foi realizada considerando a Taxa de Erros de Classificação (TEC) e o índice ROC. A TEC corresponde ao percentual de observações que foram classificadas incorretamente pelo modelo. O índice ROC corresponde à área sob a curva definida pela relação entre sensibilidade (verdadeiros positivos) e 1-especificidade (falsos positivos) e informa a probabilidade do modelo classificar corretamente uma observação selecionada aleatoriamente.

Nos modelos de estimação do Valor da Perda com Acordo e Valor da Perda com Condenação, a comparação foi realizada por meio da média da diferença entre o valor observado e o valor estimado e da Raiz do Erro Quadrático Médio (REQM).

5.8.1. Probabilidade de Perda Financeira

Estes modelos têm como objetivo estimar a probabilidade de uma ação cível resultar em perda financeira a partir das variáveis independentes INPUT_A, INPUT_B, INPUT_C e o fator tempo dispendido entre o início e o fim da ação (INPUT_D).

Foram utilizadas as 1.800 observações da amostra, uma vez que a informação de Perda / Não Perda está disponível para todas elas. Dessas, 1.260 (70%) foram utilizadas para o treinamento dos modelos e as 540 (30%) restantes foram utilizadas na validação cruzada.

5.8.1.1. Árvore de Decisão

A AD que apresentou o melhor ajuste utilizou as quatro variáveis independentes disponíveis, resultando em nove nós com dez folhas⁵, conforme a figura 7.

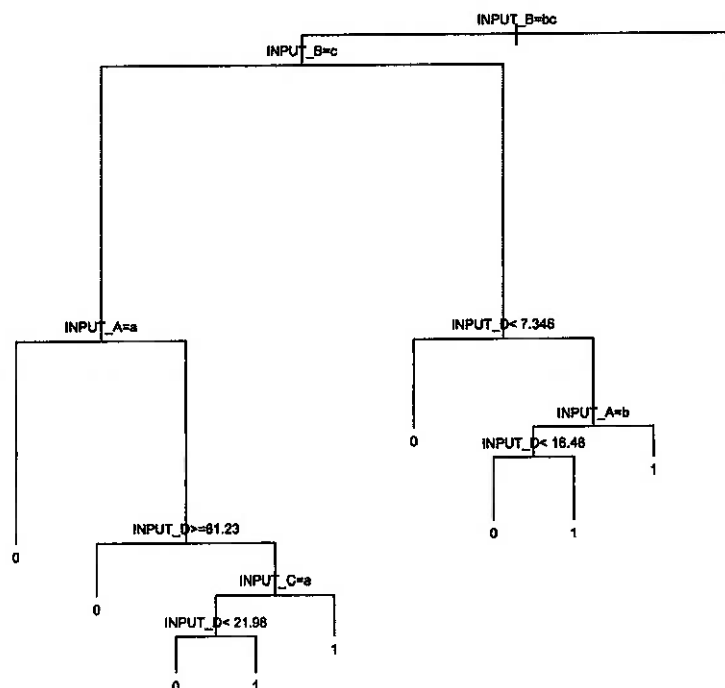


Figura 7 - Árvore de Decisão obtida para a estimativa da Probabilidade de Perda Financeira em ações cíveis massificadas. As regras acima de cada nó se referem ao ramo esquerdo.

5.8.1.2. Modelos Lineares Generalizados

O MLG que apresentou o melhor desempenho na estimativa da Probabilidade de Perda Financeira utilizou apenas as variáveis independentes INPUT_B e INPUT_C, resultando no seguinte modelo, com os parâmetros estimados na tabela 4:

$$\log \left(\frac{\pi(x)}{1 - \pi(x)} \right) = \alpha + \beta_1 INPUT_B_{B02} + \beta_2 INPUT_B_{B03} + \beta_3 INPUT_C_{C02} + \beta_4 INPUT_C_{C03} + \varepsilon \quad (3)$$

⁵ O número de folhas indica uma maior variabilidade da variável resposta e, por consequência, uma maior complexidade do modelo.

Tabela 4 - Parâmetros estimados do MLG de Probabilidade de Perda Financeira em ações cíveis massificadas.

Parâmetro	Valor Estimado	Erro Padrão	Nível Descritivo ⁶
Constante	2,2202	0,2059	< 0,001
INPUT_BB02	-1,9294	0,2172	< 0,001
INPUT_BB03	-2,8744	0,2158	< 0,001
INPUT_CC02	0,3716	0,1635	0,023
INPUT_CC03	0,8035	0,1662	< 0,001

De forma que:

$\frac{e^\alpha}{1+e^\alpha} = 90\%$: Probabilidade de Perda Financeira em ações cíveis massificadas classificadas como B01 e C01.

$e^{\beta^1} = 0,15$: Razão de chances de Perda Financeira entre ações cíveis massificadas classificadas como B02 e aquelas classificadas como B01, ou seja, as ações classificadas como B02 têm 0,15 da chance de Perda Financeira de ações classificadas como B01.

$e^{\beta^2} = 0,06$: Razão de chances de Perda Financeira entre ações cíveis massificadas classificadas como B03 e aquelas classificadas como B01. Da mesma forma, as ações classificadas como B03 têm 0,06 da chance de Perda Financeira de ações classificadas como B01.

$e^{\beta^3} = 1,45$: Razão de chances de Perda Financeira entre ações cíveis massificadas classificadas como C02 e aquelas classificadas como C01.

$e^{\beta^4} = 2,23$: Razão de chances de Perda Financeira entre ações cíveis massificadas classificadas como C03 e aquelas classificadas como C01.

5.8.1.3. Redes Neurais Artificiais

A RNA que apresentou o melhor ajuste as utilizou as quatro variáveis independentes como informações de entrada e cinco neurônios na camada oculta. Como três das variáveis independentes são categóricas, foram criadas variáveis indicadoras, resultando em seis entradas da RNA.

⁶ Teste z que avalia a significância dos parâmetros estimados.

Tabela 5 - Pesos sinápticos ajustados para a RNA de Probabilidade de Perda Financeira em ações cíveis massificadas.

Entrada	Bias	Neurônios da Camada Oculta				
		1	2	3	4	5
Bias		-0,4732	-58,3552	57,4407	-43,1745	4,8154
1		0,2399	42,3571	113,2754	-58,5601	152,2081
2		1,4807	57,6463	6,2656	-144,2458	-35,1115
3		2,2794	56,2677	-171,7356	-39,3590	-333,0111
4		0,0484	1,4384	61,5982	2,4785	-4,7258
5		0,2834	-6,1978	100,7358	185,6073	-157,2657
6		-14,0282	0,0397	0,1035	22,5077	43,6711
Saída	-44,0445	0,6827	-2,0203	2,8401	2,6343	41,1463

5.8.1.4. Comparação de desempenho

Comparando o desempenho dos modelos selecionados na validação, observa-se que a RNA apresentou 6,0% menos erros de classificação que a AD e 20,7% menos erros de classificação que o MLG. A estimativa pontual do índice ROC também indica que a RNA possui desempenho superior aos demais, porém, considerando o intervalo de confiança de 95% deste índice, os três modelos apresentam um desempenho semelhante na estimação da Probabilidade de Perda Financeira em ações cíveis massificadas.

Tabela 6 - Desempenho das técnicas avaliadas na estimação da Probabilidade de Perda Financeira em ações cíveis massificadas.

Modelo	Taxa de Erros de Classificação		Índice ROC	
	Treino	Validação	Treino	Validação
Árvore de Decisão	20,7%	23,3%	0,815 [0,790 ; 0,840]	0,792 [0,754 ; 0,831]
Modelos Lineares Generalizados	27,1%	27,6%	0,774 [0,748 ; 0,800]	0,760 [0,720 ; 0,801]
Redes Neurais Artificiais	21,6%	21,9%	0,822 [0,798 ; 0,846]	0,808 [0,770 ; 0,846]

5.8.2. Probabilidade de Condenação

Estes modelos têm como objetivo estimar a probabilidade de uma ação cível se encerrar por condenação a partir das variáveis independentes INPUT_A, INPUT_B, INPUT_C e o fator tempo dispendido entre o início e o fim da ação (INPUT_D).

Das 1.800 observações amostradas inicialmente, foram mantidas apenas 1.222 observações que resultaram em perda financeira e, portanto contém a informação de que a perda ocorreu por condenação ou acordo. Dessas, 856 observações (70%)

foram utilizadas para o treinamento dos modelos e as 366 observações restantes (30%) foram utilizadas na validação cruzada.

5.8.2.1. Árvore de Decisão

A AD que apresentou o melhor ajuste utilizou as quatro variáveis independentes disponíveis, resultando em nove nós com dez folhas, conforme a figura 8.

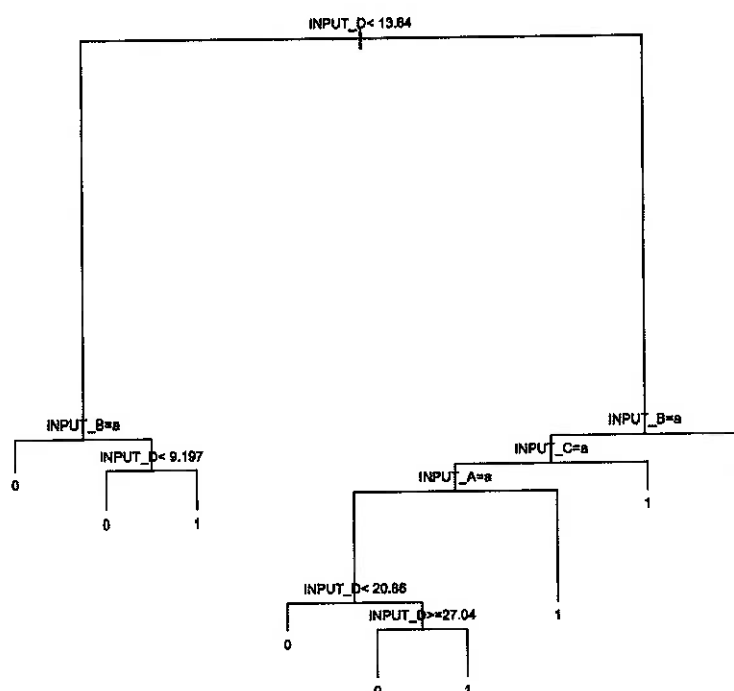


Figura 8 - Árvore de Decisão obtida para a estimativa da Probabilidade de Condenação em ações cíveis massificadas. Como comentado na figura 7, as regras acima de cada nó se referem ao ramo esquerdo.

5.8.2.2. Modelos Lineares Generalizados

O MLG que apresentou o melhor desempenho na estimativa da Probabilidade de Perda Financeira utilizou as quatro variáveis independentes disponíveis, resultando no seguinte modelo, com os parâmetros estimados na tabela 7:

$$\log\left(\frac{\pi(x)}{1-\pi(x)}\right) = \alpha + \beta_1 INPUT_A_{A02} + \beta_2 INPUT_B_{B02} + \beta_3 INPUT_B_{B03} + \beta_4 INPUT_C_{C02} + \beta_5 INPUT_C_{C03} + \beta_6 INPUT_D + \varepsilon \quad (4)$$

Tabela 7 - Parâmetros estimados do MLG de Probabilidade de Condenação em ações cíveis massificadas.

Parâmetro	Valor Estimado	Erro Padrão	Nível Descritivo ⁷
α	-1,3561	0,2639	< 0,001
β_1	0,4079	0,1899	0,032
β_2	1,1926	0,2213	< 0,001
β_3	1,1769	0,2644	< 0,001
β_4	0,0762	0,2134	0,721
β_5	0,5225	0,2260	0,021
β_6	0,0577	0,0073	< 0,001

De forma que:

$\frac{e^\alpha}{1+e^\alpha} = 20\%$: Probabilidade de Condenação em ações cíveis massificadas classificadas como A01, B01 e C01 recém-iniciadas.

$e^{\beta_1} = 1,50$: Razão de chances de Perda Financeira entre ações cíveis massificadas classificadas como A02 e aquelas classificadas como A01.

$e^{\beta_2} = 3,30$: Razão de chances de Perda Financeira entre ações cíveis massificadas classificadas como B02 e aquelas classificadas como B01.

$e^{\beta_3} = 3,24$: Razão de chances de Perda Financeira entre ações cíveis massificadas classificadas como B03 e aquelas classificadas como B01.

$e^{\beta_4} = 1,08$: Razão de chances de Perda Financeira entre ações cíveis massificadas classificadas como C02 e aquelas classificadas como C01.

$e^{\beta_5} = 1,69$: Razão de chances de Perda Financeira entre ações cíveis massificadas classificadas como C03 e aquelas classificadas como C01.

$e^{\beta_6} = 1,06$: Razão de chances de Perda Financeira entre ações cíveis massificadas com uma unidade de tempo dispendido entre o início e o fim de diferença, ou seja, para cada unidade de tempo dispendido a partir do início a chance de condenação aumenta cerca de 6%.

O nível descritivo do parâmetro β_4 maior do que 5% indica que não existe diferença significativa entre as classes C02 e C01 quanto à probabilidade de condenação, porém, como a unificação dessas classes não proporcionará maior desempenho do modelo, apenas maior simplicidade, optou-se pela sua manutenção.

⁷ Teste z que avalia a significância dos parâmetros estimados.

5.8.2.3. Redes Neurais Artificiais

A RNA que apresentou o melhor ajuste as utilizou as quatro variáveis independentes como informações de entrada e quatro neurônios na camada oculta. De forma análoga à RNA de Probabilidade de Perda, foram criadas variáveis indicadoras, resultando em seis entradas da RNA.

Tabela 8 - Pesos sinápticos ajustados para a RNA de Probabilidade de Condenação em ações cíveis massificadas.

Entrada	Bias	Neurônios da Camada Oculta			
		1	2	3	4
Bias		2,2610	1,4141	118,2633	0,4476
1		0,4875	0,9345	26,8341	0,2701
2		1,8378	0,3649	193,3362	0,1085
3		0,5705	-0,1061	-35,6912	0,1835
4		0,2922	-0,2055	391,1485	-0,6333
5		0,2448	0,4567	174,1996	0,2098
6		0,0658	7,7994	-72,6543	-3,8898
Saida	-53,9286	109,2104	-52,7939	-3,8049	-0,2836

5.8.2.4. Comparação de desempenho

Comparando o desempenho dos modelos selecionados na validação, observa-se que a RNA apresentou 10,6% menos erros de classificação que a AD e o MLG, que apresentaram desempenho similar. A estimativa pontual do índice ROC indica que o MLG possui desempenho superior aos demais, porém, considerando o intervalo de confiança de 95% deste índice, os três modelos apresentam um desempenho semelhante na estimação da Probabilidade de Condenação em ações cíveis massificadas.

Tabela 9 - Desempenho das técnicas avaliadas na estimação da Probabilidade de Condenação em ações cíveis massificadas.

Modelo	Taxa de Erros de Classificação		Índice ROC	
	Treino	Validação	Treino	Validação
Árvore de Decisão	14,7%	18,9%	0,813 [0,776 ; 0,849]	0,788 [0,734 ; 0,841]
Modelos Lineares Generalizados	18,2%	18,9%	0,812 [0,777 ; 0,846]	0,854 [0,812 ; 0,896]
Redes Neurais Artificiais	17,3%	16,9%	0,815 [0,779 ; 0,852]	0,845 [0,798 ; 0,893]

5.8.3. Valor da Perda com Acordo

Estes modelos têm como objetivo estimar o valor da perda financeira em uma ação cível que se encerre por acordo a partir das variáveis independentes INPUT_A, INPUT_B, INPUT_C e o fator tempo dispendido entre o início e o fim da ação (INPUT_D).

Das 1.800 observações amostradas inicialmente, foram mantidas apenas 313 observações que resultaram em perda financeira por acordo. Dessas, 220 observações (70%) foram utilizadas para o treinamento dos modelos e as 93 observações restantes (30%) foram utilizadas na validação cruzada.

5.8.3.1. Árvore de Decisão

A AD que apresentou o melhor ajuste utilizou as quatro variáveis independentes disponíveis, resultando em seis nós com sete folhas, conforme a figura 9.

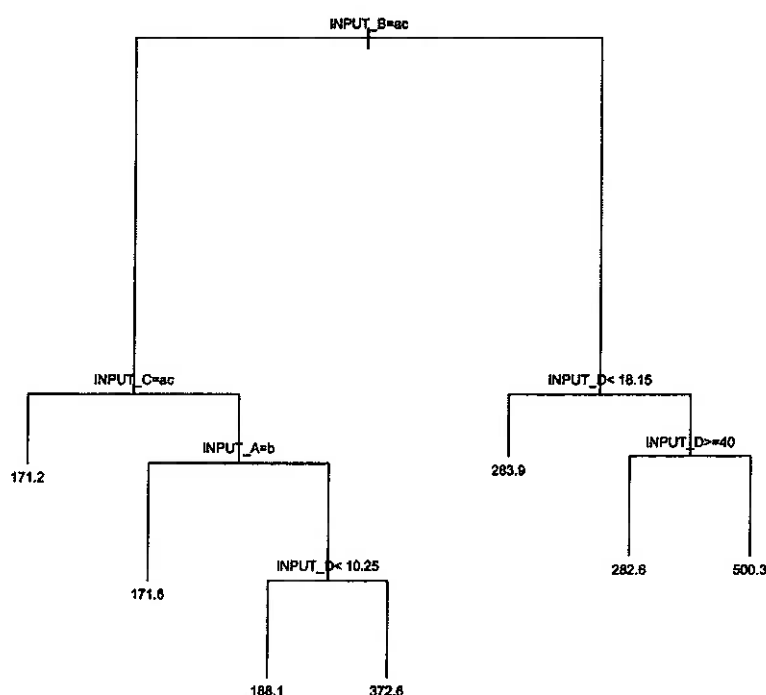


Figura 9 - Árvore de Decisão obtida para a estimativa do Valor da Perda com Acordo em ações cíveis massificadas. Como comentado na figura 7, as regras acima de cada nó se referem ao ramo esquerdo.

5.8.3.2. Modelos Lineares Generalizados

O MLG que apresentou o melhor desempenho na estimação do Valor da Perda com Acordo utilizou as variáveis independentes INPUT_B e INPUT_C, a distribuição Gamma com a função de ligação inversa como base e a variável dependente transformada pela raiz quadrada, resultando no seguinte modelo, com os parâmetros estimados na tabela 10:

$$\sqrt{\text{Valor Perda Acordo}} = \alpha + \beta_1 \text{INPUT_B}_{B02} + \beta_2 \text{INPUT_B}_{B03} + \beta_3 \text{INPUT_C}_{C02} + \beta_4 \text{INPUT_C}_{C03} + \varepsilon \quad (5)$$

O nível descritivo do teste de Kolmogorov-Smirnov (KS) para avaliar a aderência da variável dependente à distribuição Gamma aumentou de 0,16 para 0,49 após a aplicação da raiz quadrada, indicando que essa transformação traz benefícios ao ajuste do modelo, conforme apresentado na figura 10.

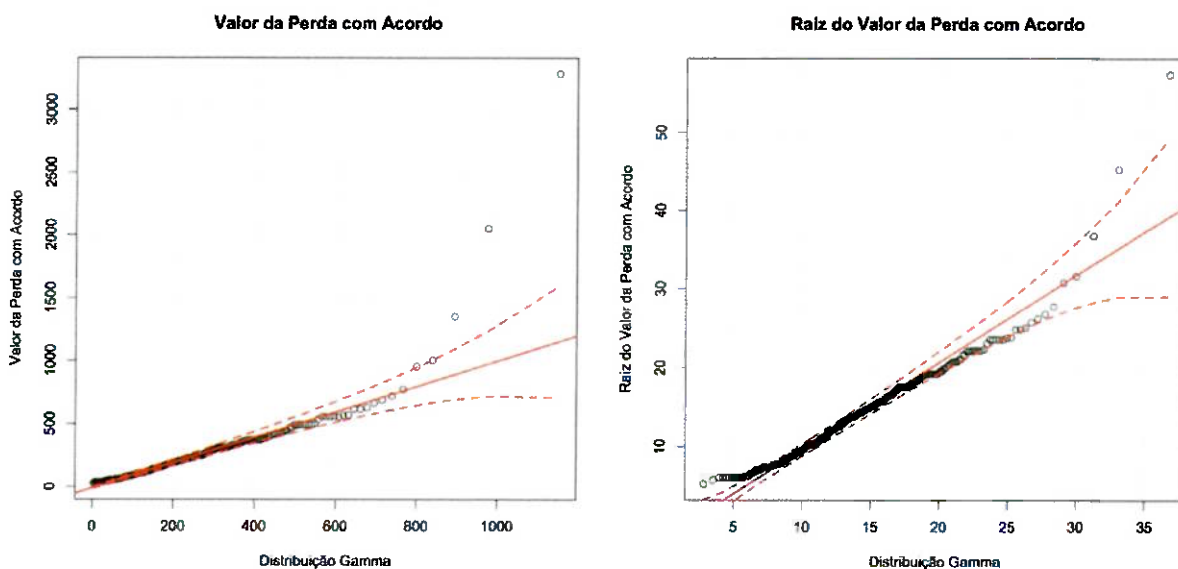


Figura 10 - Gráficos Quantil-Quantil do Valor da Perda com Acordo e Raiz do Valor da Perda com Acordo com a distribuição Gamma.

Quanto mais próximos os pontos estiverem da reta inclinada, mais aderente é a variável dependente da distribuição Gamma. As linhas tracejadas correspondem ao intervalo de confiança de 95%.

Tabela 10 - Parâmetros estimados do MLG de Valor da Perda com Acordo em ações cíveis massificadas.

Parâmetro	Valor Estimado	Erro Padrão	Nível Descritivo ⁸
α	0,0791	0,0033	< 0,001
β_1	-0,0212	0,0041	< 0,001
β_2	-0,0049	0,0065	0,459
β_3	-0,0076	0,0041	0,067
β_4	0,0056	0,0053	0,294

Ao contrário do MLG de Probabilidade de Perda e Probabilidade de Condenação, os parâmetros não possuem uma interpretação direta, principalmente após a transformação da variável dependente para aumentar a aderência à distribuição Gamma.

Assim como no MLG de Probabilidade de Condenação, o nível descritivo dos parâmetros β_2 , β_3 e β_4 maiores do que 5% indicam que não existe diferença significativa entre as classes B03 e B01, e C03, C02 e C01 quanto ao Valor da Perda com Acordo, porém, como a unificação dessas classes não proporcionará maior desempenho do modelo, apenas maior simplicidade, optou-se pela sua manutenção.

5.8.3.3. Redes Neurais Artificiais

A RNA que apresentou o melhor ajuste as utilizou as quatro variáveis independentes como informações de entrada e quatro neurônios na camada oculta. De forma análoga à RNA de Probabilidade de Condenação, foram criadas variáveis indicadoras, resultando em seis entradas da RNA.

Tabela 11 - Pesos sinápticos ajustados para a RNA de Probabilidade de Condenação em ações cíveis massificadas.

Entrada	Bias	Neurônios da Camada Oculta			
		1	2	3	4
Bias		2,2610	1,4141	118,2633	0,4476
1		0,4875	0,9345	26,8341	0,2701
2		1,8378	0,3649	193,3362	0,1085
3		0,5705	-0,1061	-35,6912	0,1835
4		0,2922	-0,2055	391,1485	-0,6333
5		0,2448	0,4567	174,1996	0,2098
6		0,0658	7,7994	-72,6543	-3,8898
Saída	-53,9286	109,2104	-52,7939	-3,8049	-0,2836

⁸ Teste t que avalia a significância dos parâmetros estimados.

5.8.3.4. Comparação de desempenho

Comparando o desempenho dos modelos selecionados na validação, observa-se que a RNA apresentou um REQM 2,9% menor que a AD e 8,2% menor que o MLG. A estimativa pontual da média dos erros indica que a AD possui desempenho superior aos demais, porém, considerando o intervalo de confiança de 95%, os três modelos apresentam um desempenho semelhante na estimação do Valor da Perda com Acordo em ações cíveis massificadas.

Tabela 12 - Desempenho das técnicas avaliadas na estimação do Valor da Perda com Acordo em ações cíveis massificadas.

Modelo	Raiz do Erro Quadrático Médio (REQM)		Intervalo de Confiança 95% da Média dos Erros	
	Treino	Validação	Treino	Validação
Árvore de Decisão	184,96	343,72	0,00 [-362,51 ; 362,51]	13,24 [-660,44 ; 686,92]
Modelos Lineares Generalizados	198,05	363,48	27,35 [-360,81 ; 415,52]	36,76 [-675,65 ; 749,16]
Redes Neurais Artificiais	178,78	333,83	0,00 [-350,41 ; 350,41]	13,87 [-640,43 ; 668,16]

5.8.4. Valor da Perda com Condenação

Estes modelos têm como objetivo estimar o valor da perda financeira em uma ação cível que se encerre por condenação a partir das variáveis independentes INPUT_A, INPUT_B, INPUT_C e o fator tempo dispendido entre o início e o fim da ação (INPUT_D).

Das 1.800 observações amostradas inicialmente, foram mantidas apenas 909 observações que resultaram em perda financeira por condenação. Dessas, 637 observações (70%) foram utilizadas para o treinamento dos modelos e as 272 observações restantes (30%) foram utilizadas na validação cruzada.

5.8.4.1. Árvore de Decisão

A AD que apresentou o melhor ajuste utilizou as quatro variáveis independentes disponíveis, resultando em doze nós com treze folhas, conforme a figura 11.

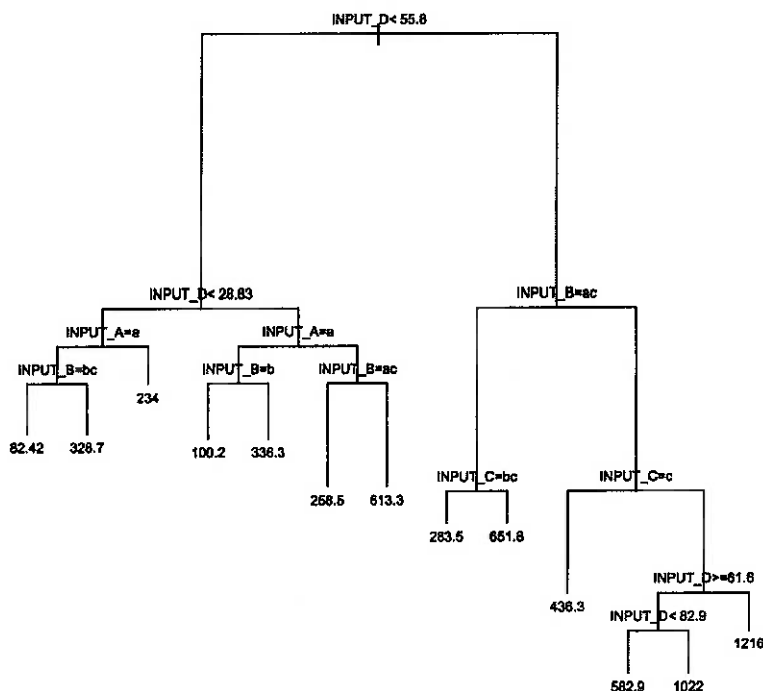


Figura 11 - Árvore de Decisão obtida para a estimativa do Valor da Perda com Condenação em ações cíveis massificadas. Como comentado na figura 7, as regras acima de cada nó se referem ao ramo esquerdo.

5.8.4.2. Modelos Lineares Generalizados

O MLG que apresentou o melhor desempenho na estimativa do Valor da Perda com Condenação utilizou as variáveis independentes INPUT_A, INPUT_B e INPUT_D, a distribuição Normal com a função de ligação identidade como base e a variável dependente transformada pelo logaritmo natural, resultando no seguinte modelo, com os parâmetros estimados na tabela 13:

$$\ln(\text{Valor Perda Acordo}) = \alpha + \beta_1 \text{INPUT_A}_{A02} + \beta_2 \text{INPUT_B}_{B02} + \beta_3 \text{INPUT_B}_{B03} + \beta_4 \text{INPUT_D} + \varepsilon \quad (6)$$

O nível descritivo do teste de Kolmogorov-Smirnov (KS) para avaliar a aderência da variável dependente à distribuição Normal aumentou de menor que 0,01 para 0,04 após a aplicação do logaritmo natural, indicando que essa transformação traz benefícios ao ajuste do modelo, conforme apresentado na figura 12.

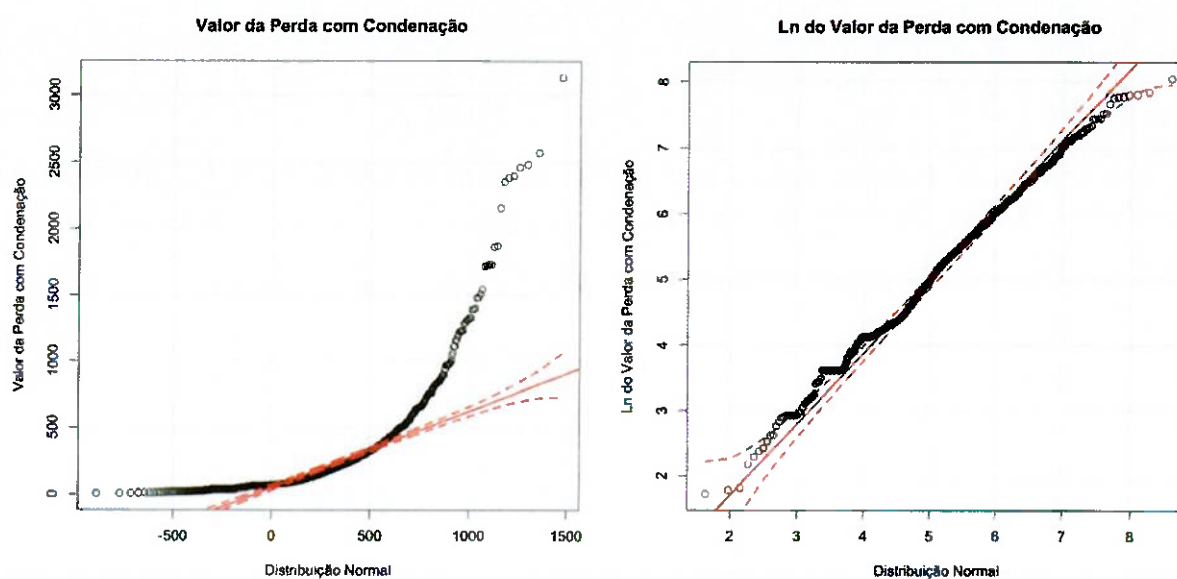


Figura 12 - Gráficos Quantil-Quantil do Valor da Perda com Condenação e o logaritmo natural do Valor da Perda com Condenação com a distribuição Normal.

Como informado na figura 10, quanto mais próximos os pontos estiverem da reta inclinada, mais aderente é a variável dependente da distribuição Normal. As linhas tracejadas correspondem ao intervalo de confiança de 95%.

Tabela 13 - Parâmetros estimados do MLG de Valor da Perda com Condenação em ações cíveis massificadas.

Parâmetro	Valor Estimado	Erro Padrão	Nível Descritivo ⁹
α	4,6992	0,0945	< 0,001
β_1	0,5165	0,0827	< 0,001
β_2	-0,4349	0,0922	< 0,001
β_3	-0,5671	0,1048	< 0,001
β_4	0,0120	0,0014	< 0,001

Ao contrário do MLG de Probabilidade de Perda e Probabilidade de Condenação, os parâmetros não possuem uma interpretação direta, principalmente após a transformação da variável dependente para aumentar a aderência à distribuição Normal.

5.8.4.3. Redes Neurais Artificiais

A RNA que apresentou o melhor ajuste as utilizou as quatro variáveis independentes como informações de entrada e três neurônios na camada oculta. De forma análoga

⁹ Teste t que avalia a significância dos parâmetros estimados.

à RNA de Valor de Perda com Acordo, foram criadas variáveis indicadoras, resultando em seis entradas da RNA.

Tabela 14 - Pesos sinápticos ajustados para a RNA de Valor da Perda com Condenação em ações cíveis massificadas.

Entrada	Bias	Neurônios da Camada Oculta		
		1	2	3
Bias		21,1924	0,6631	-0,0972
1		-5,9104	4,3080	-11,8328
2		-7,4201	-7,0418	14,4654
3		-0,0643	6,4848	-1,3801
4		-15,9473	2,2281	-9,2333
5		-12,6512	2,4698	-6,5391
6		0,2486	0,0259	-0,0634
Saída	0,5417	0,2747	-0,7344	-0,7634

5.8.4.4. Comparação de desempenho

Comparando o desempenho dos modelos selecionados na validação, observa-se que a RNA apresentou um REQM 16,4% menor do que a AD e 21,3% menor que o MLG. A estimativa pontual da média dos erros indica que a RNA possui desempenho superior aos demais, porém, considerando o intervalo de confiança de 95%, os três modelos apresentam um desempenho semelhante na estimação do Valor da Perda com Condenação em ações cíveis massificadas.

Tabela 15 - Desempenho das técnicas avaliadas na estimação do Valor da Perda com Condenação em ações cíveis massificadas.

Modelo	Raiz do Erro Quadrático Médio (REQM)		Intervalo de Confiança 95% da Média dos Erros	
	Treino	Validação	Treino	Validação
Árvore de Decisão	304,39	330,44	0,00 [-596,60 ; 596,60]	-40,56 [-688,22 ; 607,10]
Modelos Lineares Generalizados	371,33	351,21	108,18 [-619,61 ; 835,97]	81,85 [-606,50 ; 770,21]
Redes Neurais Artificiais	298,17	276,35	-0,02 [-584,42 ; 584,38]	-29,20 [-570,83 ; 512,44]

5.8.5. Consolidação das Estimações

Utilizando a equação 2 e as estimativas da Probabilidade de Perda, Probabilidade de Condenação, Valor da Perda com Acordo e Valor da Perda com Condenação, obteve-se o Valor da Perda Esperada com as ações cíveis massificadas.

Comparado o desempenho dos modelos selecionados, observa-se que a RNA apresentou a média das diferenças, em valores absolutos, 82,4% menor que a AD e

98,8% menor do que o MLG. Observa-se também que a RNA apresentou a RMEQ 9,4% menor que a AD e 2,7% menor que o MLG.

Tabela 16 - Desempenho das técnicas avaliadas na estimação da Perda Esperada com ações cíveis massificadas.

Modelo	Média das Diferenças	Raiz do Erro Quadrático Médio (REQM)
Árvore de Decisão	4,09	285,05
Modelos Lineares Generalizados	-57,83	306,16
Redes Neurais Artificiais	0,72	277,31

6. Conclusões

Verificou-se que na estimação da Perda Esperada o modelo de Redes Neurais Artificiais apresentou resultados bastante satisfatórios, com desempenho superior às Árvores de Decisão e Modelos Lineares Generalizados.

A diferença de 3,37 na média das diferenças entre a RNA e a AD, que a princípio pode parecer não muito significativa, pode ser relevante caso o número de ações cíveis massificadas seja elevado. Supondo que uma instituição tenha cerca de vinte milhões de clientes e que 1% deles ingressem com uma ação cível contrária a ela, a redução no valor da provisão devido à utilização das RNA pode chegar a 1,8% do valor provisionado, que neste estudo corresponde 674.000 unidades monetárias, valor suficiente para provisionar outras 3.608 ações.

Ainda que as Redes Neurais Artificiais sejam uma técnica computacionalmente intensiva, muitas vezes ela tende a ser menos custosa do que os Modelos Lineares Generalizados, pois este último exige que sejam realizadas diversas análises para verificação das suposições do modelo, distribuições de probabilidade e funções de ligação a serem utilizadas. Além disso, as Redes Neurais Artificiais possuem a capacidade de identificar complexas relações entre as variáveis independentes e as variáveis dependentes sem a necessidade de transformações nas variáveis.

Dessa forma, pode-se concluir que as Redes Neurais Artificiais se apresentam como uma ótima alternativa na gestão do risco legal.

REFERÊNCIAS

- Banco Central do Brasil. **Resolução N° 003380**. 2006. Disponível em: <<https://www3.bcb.gov.br/normativo/detalharNormativo.do?method=detalharNormativo&N=106196825>>. Acesso em: 20 jan. 2013.
- Basel Committee on Banking Supervision. **International Convergence of Capital Measurement and Capital Standards – A Revised Framework**. 2004. Disponível em: <<http://www.bis.org/publ/bcbs107.pdf>>. Acesso em: 20 jan. 2013.
- Bolfarine, H., Bussab, W. O. **Elementos de Amostragem**. São Paulo: Edgard Blucher, 2006.
- Breiman, L., Friedman J. H., Olshen, R. A., and Stone, C. J. **Classification and Regression Trees**. California: Wadsworth, 1984.
- Comissão de Valores Mobiliários. **Deliberação N. 489 – Normas e Procedimento de Contabilidade N. 22**. Disponível em: <<http://www.cvm.gov.br/asp/cvmwww/atos/exiato.asp?File=%5Cdeli%5Cdeli489.htm>>. Acesso em: 20 jan. 2013.
- Cruz, M. G. **Modeling, measuring and hedging operational risk**. New York: Wiley, 2002.
- Fox, J., Weisberg J. **An {R} Companion to Applied Regression**, 2nd ed. California: Sage, 2011. Disponível em: <<http://socserv.socsci.mcmaster.ca/jfox/Books/Companion>> Acesso em: 20 jan. 2013.
- Gross, J. Ligges, W. **nortest: Tests for Normality**. 2012. R package version 1.0-2. Disponível em: <<http://CRAN.R-project.org/package=nortest>> Acesso em: 20 jan. 2013.
- Haykin, S. **Neural Networks A Comprehensive Foundation**. 2nd ed. Ontario: Pearson Prentice Hall, 2001.
- McCullagh P. and Nelder, J. A. **Generalized Linear Models**. London: Chapman and Hall, 1989.
- Morettin, P. A., Bussab, W. O. **Estatística Básica**. 7^a Ed. Saraiva, São Paulo: Saraiva, 2011.
- Paula, G. A. **Modelos de Regressão com apoio computacional**. 2012. Disponível em: <http://www.ime.usp.br/~giapaula/texto_2012.pdf>. Acesso em: 20 jan. 2013.

R Core Team. **R: A language and environment for statistical computing**. 2012. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0. Disponível em: <<http://www.R-project.org/>>
Acesso em: 20 jan. 2013.

Ripley, B. D. **Pattern Recognition and Neural Networks**. New York: Cambridge, 1996.

Therneau, T., Atkinson, B, Ripley, B. D. **rpart: Recursive Partitioning**. 2012. R package version 3.1-55. Disponível em: <<http://CRAN.R-project.org/package=rpart>>
Acesso em: 20 jan. 2013.

Venables, W. N. and Ripley, B. D. **Modern Applied Statistics with S**. 4th ed. New York: Springer, 2002.

APENDICE A – Programas desenvolvidos em R

Programa para a importação de dados

```
### IMPORTAÇÃO DOS DADOS
dados = read.table("/Users/FernandoItano/Documents/DADOS.CSV", sep=";", dec=".", header=TRUE)
```

Programa para estimação da probabilidade de perda em ações cíveis

```
### ESTIMAÇÃO DA PROBABILIDADE DE PERDA FINANCEIRA
source("/Users/FernandoItano/Documents/Importacao.r")
dados.originais = dados
(n = length(dados[,1]))
(nT = ceiling(n*0.7))

# DEFINIÇÃO DAS AMOSTRAS DE TREINO E VALIDAÇÃO
set.seed(123456)
T = sample(1:n,nT) ; V = (1:n)[-T]
dados.treino = rbind(dados[T,]) ; str(dados.treino)
dados.valida = rbind(dados[V,]) ; str(dados.valida)

# MODELAGEM POR ÁRVORE DE DECISÃO
require(rpart)
pperda.arv = rpart(TARGET_A ~ INPUT_A + INPUT_B + INPUT_C + INPUT_D, data=dados.treino,
method="class")
summary(pperda.arv)
plot(pperda.arv) ; text(pperda.arv1, cex=0.6)
plotcp(pperda.arv)
printcp(pperda.arv)
par(mfrow=c(2,1)) ; xsq.rpart(pperda.arv)

# MODELAGEM POR MODELOS LINEARES GENERALIZADOS
pperda.mlg1 = glm(TARGET_A ~ INPUT_A + INPUT_B + INPUT_C + INPUT_D, data=dados.treino,
family="binomial")
summary(pperda.mlg1) ; anova(pperda.mlg1, test="Chisq")
pperda.mlg2 = glm(TARGET_A ~ INPUT_A + INPUT_B + INPUT_C, data=dados.treino,
family="binomial")
summary(pperda.mlg2) ; anova(pperda.mlg2, test="Chisq")
pperda.mlg3 = glm(TARGET_A ~ INPUT_B + INPUT_C, data=dados.treino, family="binomial")
summary(pperda.mlg3) ; anova(pperda.mlg3, test="Chisq")
pperda.mlg4 = glm(TARGET_A ~ INPUT_B * INPUT_C, data=dados.treino, family="binomial")
summary(pperda.mlg4) ; anova(pperda.mlg4, test="Chisq")
pperda.mlg = pperda.mlg3

# MODELAGEM POR REDES NEURAIS ARTIFICIAIS
require(nnet)
rna.treinamento = function(neuronios=3, tec.limite=0.10, tentativas.limite=100){
  tentativas = 0
  rna.valida.tec = 1
  while(rna.valida.tec > tec.limite && tentativas < tentativas.limite){
    rna.modelo = nnet(TARGET_A ~ INPUT_A + INPUT_B + INPUT_C + INPUT_D,
data=dados.treino, size=neuronios, maxit=2000)
    tentativas = tentativas + 1
    rna.valida = predict(rna.modelo,dados.valida)
    rna.valida.tec = 1 - mean(round(rna.valida,0) == dados.valida$TARGET_A)
    print(rna.valida.tec)
    print(tentativas)
  }
  return(rna.modelo)
}
(pperda.rnax = rna.treinamento(neuronios=5, tec.limite=0.22, tentativas.limite=500))
pperda.rna = pperda.rnax ; pperda.rna

# CÁLCULO DAS MEDIDAS PARA COMPARAÇÃO DOS MODELOS
# VALORES AJUSTADOS
(pperda.arv.fitted = as.numeric(predict(pperda.arv, dados.treino)[,2]))
(pperda.arv.valida = as.numeric(predict(pperda.arv, dados.valida)[,2]))
(pperda.mlg.fitted = pperda.mlg$fitted.values)
(pperda.mlg.valida = predict.glm(pperda.mlg, dados.valida, type="response"))
(pperda.rna.fitted = pperda.rna$fitted.values)
```

```
(pperda.rna.valida = predict(pperda.rna, dados.valida))

# TAXA DE ERROS DE CLASSIFICAÇÃO
(pperda.arv.tec.treino = 1 - mean(round(pperda.arv.fitted,0) == dados.treino$TARGET_A))
(pperda.arv.tec.valida = 1 - mean(round(pperda.arv.valida,0) == dados.valida$TARGET_A))
(pperda.mlg.tec.treino = 1 - mean(round(pperda.mlg.fitted,0) == dados.treino$TARGET_A))
(pperda.mlg.tec.valida = 1 - mean(round(pperda.mlg.valida,0) == dados.valida$TARGET_A))
(pperda.rna.tec.treino = 1 - mean(round(pperda.rna.fitted,0) == dados.treino$TARGET_A))
(pperda.rna.tec.valida = 1 - mean(round(pperda.rna.valida,0) == dados.valida$TARGET_A))

# Curva ROC
require(pROC)
(pperda.arv.roc.treino = roc(dados.treino$TARGET_A, pperda.arv.fitted, plot=TRUE, smooth=F,
ci=T))
(pperda.arv.roc.valida = roc(dados.valida$TARGET_A, pperda.arv.valida, plot=TRUE, smooth=F,
ci=T))
(pperda.mlg.roc.treino = roc(dados.treino$TARGET_A, pperda.mlg.fitted, plot=TRUE, smooth=F,
ci=T))
(pperda.mlg.roc.valida = roc(dados.valida$TARGET_A, pperda.mlg.valida, plot=TRUE, smooth=F,
ci=T))
(pperda.rna.roc.treino = roc(dados.treino$TARGET_A, pperda.rna.fitted, plot=TRUE, smooth=F,
ci=T))
(pperda.rna.roc.valida = roc(dados.valida$TARGET_A, pperda.rna.valida, plot=TRUE, smooth=F,
ci=T))
```

Programa para estimação da probabilidade de condenação em ações cíveis

```
### ESTIMAÇÃO DA PROBABILIDADE DE CONDENAÇÃO
source("/Users/FernandoItano/Documents/Importacao.r")
dados.originais = dados
dados = dados[dados$TARGET_A == 1,]
str(dados)
(n = length(dados[,1]))
(nT = ceiling(n*0.7))

# DEFINIÇÃO DAS AMOSTRAS DE TREINO E VALIDAÇÃO
set.seed(123456)
T = sample(1:n,nT) ; V = (1:n)[-T]
dados.treino = rbind(dados[T,]) ; str(dados.treino)
dados.valida = rbind(dados[V,]) ; str(dados.valida)

# MODELAGEM POR ÁRVORE DE DECISÃO
require(rpart)
pcond.arv = rpart(TARGET_B ~ INPUT_A + INPUT_B + INPUT_C + INPUT_D, data=dados.treino,
method="class")
summary(pcond.arv)
plot(pcond.arv) ; text(pcond.arv, cex=0.6)
plotcp(pcond.arv)
printcp(pcond.arv)
par(mfrow=c(2,1)) ; rsq.rpart(pcond.arv) ; par(mfrow=c(1,1))

# MODELAGEM POR MODELOS LINEARES GENERALIZADOS
pcond.mlg1 = glm(TARGET_B ~ INPUT_A + INPUT_B + INPUT_C + INPUT_D, data=dados.treino,
family="binomial")
summary(pcond.mlg1) ; anova(pcond.mlg1, test="Chisq")
pcond.mlg2 = glm(TARGET_B ~ INPUT_B + INPUT_C + INPUT_D, data=dados.treino, family="binomial")
summary(pcond.mlg2) ; anova(pcond.mlg2, test="Chisq")
pcond.mlg = pcond.mlg1
summary(pcond.mlg)

# MODELAGEM POR REDES NEURAIS ARTIFICIAIS
require(nnet)
rna.treinamento = function(neuronios=3, tec.limite=0.10, tentativas.limite=100){
  tentativas = 0
  rna.valida.tec = 1
  while(rna.valida.tec > tec.limite && tentativas < tentativas.limite){
    rna.modelo = nnet(TARGET_B ~ INPUT_A + INPUT_B + INPUT_C + INPUT_D,
data=dados.treino, size=neuronios, maxit=2000)
    tentativas = tentativas + 1
    rna.valida = predict(rna.modelo,dados.valida)
    rna.valida.tec = 1 - mean(round(rna.valida,0) == dados.valida$TARGET_B)
    print(rna.valida.tec)
    print(tentativas)
  }
}
```

```

    }
    return(rna.modelo)
}
(pcond.rnax = rna.treinamento(neuronios=4, tec.limite=0.17, tentativas.limite=500))
pcond.rna = pcond.rnax ; pcond.rna
summary(pcond.rna)

# CÁLCULO DAS MEDIDAS PARA COMPARAÇÃO DOS MODELOS
# VALORES AJUSTADOS
(pcond.arv.fitted = as.numeric(predict(pcond.arv, dados.treino)[,2]))
(pcond.arv.valida = as.numeric(predict(pcond.arv, dados.valida)[,2]))
(pcond.mlg.fitted = pcond.mlg$fitted.values)
(pcond.mlg.valida = predict(glm(pcond.mlg, dados.valida, type="response"))
(pcond.rna.fitted = pcond.rna$fitted.values)
(pcond.rna.valida = predict(pcond.rna, dados.valida))

# TAXA DE ERROS DE CLASSIFICAÇÃO
(pcond.arv.tec.treino = 1 - mean(round(pcond.arv.fitted,0) == dados.treino$TARGET_B))
(pcond.arv.tec.valida = 1 - mean(round(pcond.arv.valida,0) == dados.valida$TARGET_B))
(pcond.mlg.tec.treino = 1 - mean(round(pcond.mlg.fitted,0) == dados.treino$TARGET_B))
(pcond.mlg.tec.valida = 1 - mean(round(pcond.mlg.valida,0) == dados.valida$TARGET_B))
(pcond.rna.tec.treino = 1 - mean(round(pcond.rna.fitted,0) == dados.treino$TARGET_B))
(pcond.rna.tec.valida = 1 - mean(round(pcond.rna.valida,0) == dados.valida$TARGET_B))

# Curva ROC
require(pROC)
(pcond.arv.roc.treino = roc(dados.treino$TARGET_B, pcond.arv.fitted, plot=TRUE, smooth=F,
ci=T))
(pcond.arv.roc.valida = roc(dados.valida$TARGET_B, pcond.arv.valida, plot=TRUE, smooth=F,
ci=T))
(pcond.mlg.roc.treino = roc(dados.treino$TARGET_B, pcond.mlg.fitted, plot=TRUE, smooth=F,
ci=T))
(pcond.mlg.roc.valida = roc(dados.valida$TARGET_B, pcond.mlg.valida, plot=TRUE, smooth=F,
ci=T))
(pcond.rna.roc.treino = roc(dados.treino$TARGET_B, pcond.rna.fitted, plot=TRUE, smooth=F,
ci=T))
(pcond.rna.roc.valida = roc(dados.valida$TARGET_B, pcond.rna.valida, plot=TRUE, smooth=F,
ci=T))

```

Programa para estimação do valor da perda com acordo em ações cíveis

```

### ESTIMAÇÃO DO VALOR DA PERDA COM ACORDO
source("/Users/FernandoItano/Documents/Importacao.r")
dados.originais = dados
dados = dados[dados$TARGET_A == 1 & dados$TARGET_B == 0,]
str(dados)
(n = length(dados[,1]))
(nT = ceiling(n*0.7))

# TRANSFORMAÇÃO DE ESCALA DA RESPOSTA
summary(dados$TARGET_C)
maxim.rna = max(dados$TARGET_C)
dados$TARGET_CT = (dados$TARGET_C - min(dados$TARGET_C)) / maxim.rna
summary(dados$TARGET_CT)

# DEFINIÇÃO DAS AMOSTRAS DE TREINO E VALIDAÇÃO
set.seed(123456)
T = sample(1:n,nT) ; V = (1:n)[-T]
dados.treino = rbind(dados[T,]) ; str(dados.treino)
dados.valida = rbind(dados[V,]) ; str(dados.valida)

# MODELAGEM POR ÁRVORE DE DECISÃO
require(rpart)
vacor.arv = rpart(TARGET_C ~ INPUT_A + INPUT_B + INPUT_C + INPUT_D, data=dados.treino,
method="anova")
summary(vacor.arv)
plot(vacor.arv) ; text(vacor.arv, cex=0.6)
plotcp(vacor.arv)
printcp(vacor.arv)
par(mfrow=c(2,1)) ; rsq.rpart(vacor.arv) ; par(mfrow=c(1,1))

### ANÁLISE DA DISTRIBUIÇÃO DA VARIÁVEL RESPOSTA
require(car)

```

```

require(nortest)
require(MASS)
parametros.gamma1 = fitdistr(dados$TARGET_C, "gamma")
parametros.gamma2 = fitdistr(sqrt(dados$TARGET_C), "gamma")
parametros.gamma3 = fitdistr(log(dados$TARGET_C), "gamma")

qqPlot(dados$TARGET_C, distribution="gamma",
       shape=parametros.gamma1$estimate[1],
       rate=parametros.gamma1$estimate[2],
       main="Valor da Perda com Acordo",
       xlab="Distribuição Gamma", ylab="Valor da Perda com Acordo")
ks.test(dados$TARGET_C, "pgamma", shape=parametros.gamma1$estimate[1],
       rate=parametros.gamma1$estimate[2])

qqPlot(sqrt(dados$TARGET_C), distribution="gamma",
       shape=parametros.gamma2$estimate[1],
       rate=parametros.gamma2$estimate[2],
       main="Raiz do Valor da Perda com Acordo",
       xlab="Distribuição Gamma", ylab="Raiz do Valor da Perda com Acordo")
ks.test(sqrt(dados$TARGET_C), "pgamma", shape=parametros.gamma2$estimate[1],
       rate=parametros.gamma2$estimate[2])

qqPlot(log(dados$TARGET_C), distribution="gamma",
       shape=parametros.gamma3$estimate[1],
       rate=parametros.gamma3$estimate[2],
       main="Log do Valor da Perda com Acordo",
       xlab="Distribuição Gamma", ylab="Log do Valor da Perda com Acordo")
ks.test(log(dados$TARGET_C), "pgamma", shape=parametros.gamma3$estimate[1],
       rate=parametros.gamma3$estimate[2])

# MODELAGEM POR MODELOS LINEARES GENERALIZADOS
vacor.mlg1 = glm(sqrt(TARGET_C) ~ INPUT_A + INPUT_B + INPUT_C + INPUT_D, data=dados.treino,
family="Gamma")
summary(vacor.mlg1) ; anova(vacor.mlg1, test="Chisq")
vacor.mlg2 = glm(sqrt(TARGET_C) ~ INPUT_B + INPUT_C + INPUT_D, data=dados.treino,
family="Gamma")
summary(vacor.mlg2) ; anova(vacor.mlg2, test="Chisq")
vacor.mlg3 = glm(sqrt(TARGET_C) ~ INPUT_B + INPUT_C, data=dados.treino, family="Gamma")
summary(vacor.mlg3) ; anova(vacor.mlg3, test="Chisq")
vacor.mlg4 = glm(sqrt(TARGET_C) ~ INPUT_B, data=dados.treino, family="Gamma")
summary(vacor.mlg4) ; anova(vacor.mlg4, test="Chisq")
vacor.mlg = vacor.mlg3
summary(vacor.mlg)

lms = summary(vacor.mlg) ; str(lms)
r = lms$deviance.resid
qqPlot(r, distribution="norm",
       main="Resíduos Deviance",
       xlab="Distribuição Normal", ylab="Resíduos Deviance")
hist(r)
ad.test(r)

# MODELAGEM POR REDES NEURAIS ARTIFICIAIS
require(nnet)
rna.treinamento = function(neuronios=3, reqm.limite=0.10, tentativas.limite=100){
  reqm.hist = vector()
  tentativas = 0
  rna.valida.reqm = reqm.limite + 1
  while(rna.valida.reqm > reqm.limite && tentativas < tentativas.limite){
    rna.modelo = nnet(TARGET_CT ~ INPUT_A + INPUT_B + INPUT_C + INPUT_D,
data=dados.treino, size=neuronios, linout=TRUE, maxit=2000)
    tentativas = tentativas + 1
    rna.valida = (predict(rna.modelo,dados.valida) * maxim.rna) +
min(dados$TARGET_C)
    rna.valida.reqm = sqrt(mean((dados.valida$TARGET_C - rna.valida) ^ 2))
    reqm.hist[tentativas] = rna.valida.reqm
    plot(reqm.hist, type="l", ylim=c(reqm.limite*0.9, reqm.limite*1.1))
    print(rna.valida.reqm)
    print(tentativas)
  }
  print(min(reqm.hist))
  return(rna.modelo)
}
(vacor.rnax = rna.treinamento(neuronios=4, reqm.limite=334, tentativas.limite=500))
vacor.rna = vacor.rnax ; vacor.rna

# CÁLCULO DAS MEDIDAS PARA COMPARAÇÃO DOS MODELOS

```

```

# VALORES AJUSTADOS
(vacor.arv.fitted = as.numeric(predict(vacor.arv, dados.treino)))
(vacor.arv.valida = as.numeric(predict(vacor.arv, dados.valida)))
(vacor.mlg.fitted = (vacor.mlg$fitted.values^2))
(vacor.mlg.valida = (predict.glm(vacor.mlg, dados.valida, type="response")^2))
(vacor.rna.fitted = (vacor.rna$fitted.values * maxim.rna) + min(dados$TARGET_C))
(vacor.rna.valida = (predict(vacor.rna, dados.valida) * maxim.rna) + min(dados$TARGET_C))

# RAÍZ DO ERRO MÉDIO QUADRÁTICO
(vacor.arv.reqm.treino = sqrt(mean((dados.treino$TARGET_C - vacor.arv.fitted)^2)))
(vacor.arv.reqm.valida = sqrt(mean((dados.valida$TARGET_C - vacor.arv.valida)^2)))
(vacor.mlg.reqm.treino = sqrt(mean((dados.treino$TARGET_C - vacor.mlg.fitted)^2)))
(vacor.mlg.reqm.valida = sqrt(mean((dados.valida$TARGET_C - vacor.mlg.valida)^2)))
(vacor.rna.reqm.treino = sqrt(mean((dados.treino$TARGET_C - vacor.rna.fitted)^2)))
(vacor.rna.reqm.valida = sqrt(mean((dados.valida$TARGET_C - vacor.rna.valida)^2)))

# ERRO COM INTERVALO DE CONFIANÇA
(vacor.arv.ci.treino = cbind(mean(dados.treino$TARGET_C - vacor.arv.fitted) + c(-qnorm(0.975)
* vacor.arv.reqm.treino, qnorm(0.975) * vacor.arv.reqm.treino)))
(vacor.arv.ci.valida = cbind(mean(dados.valida$TARGET_C - vacor.arv.valida) + c(-qnorm(0.975)
* vacor.arv.reqm.valida, qnorm(0.975) * vacor.arv.reqm.valida)))
(vacor.mlg.ci.treino = cbind(mean(dados.treino$TARGET_C - vacor.mlg.fitted) + c(-qnorm(0.975)
* vacor.mlg.reqm.treino, qnorm(0.975) * vacor.mlg.reqm.treino)))
(vacor.mlg.ci.valida = cbind(mean(dados.valida$TARGET_C - vacor.mlg.valida) + c(-qnorm(0.975)
* vacor.mlg.reqm.valida, qnorm(0.975) * vacor.mlg.reqm.valida)))
(vacor.rna.ci.treino = cbind(mean(dados.treino$TARGET_C - vacor.rna.fitted) + c(-qnorm(0.975)
* vacor.rna.reqm.treino, qnorm(0.975) * vacor.rna.reqm.treino)))
(vacor.rna.ci.valida = cbind(mean(dados.valida$TARGET_C - vacor.rna.valida) + c(-qnorm(0.975)
* vacor.rna.reqm.valida, qnorm(0.975) * vacor.rna.reqm.valida)))

```

Programa para estimação do valor da perda com condenação em ações cíveis

```

### ESTIMAÇÃO DO VALOR DA PERDA COM CONDENAÇÃO
source("/Users/FernandoItano/Documents/Importacao.r")
dados.originais = dados
dados = dados[dados$TARGET_A == 1 & dados$TARGET_B == 1,]
str(dados)
(n = length(dados[,1]))
(nT = ceiling(n*0.7))

# TRANSFORMAÇÃO DE ESCALA DA RESPOSTA
summary(dados$TARGET_C)
maxim.rna = max(dados$TARGET_C)
dados$TARGET_CT = (dados$TARGET_C - min(dados$TARGET_C)) / maxim.rna
summary(dados$TARGET_CT)

# DEFINIÇÃO DAS AMOSTRAS DE TREINO E VALIDAÇÃO
set.seed(123456)
T = sample(1:n, nT) ; V = (1:n)[-T]
dados.treino = rbind(dados[T,]) ; str(dados.treino)
dados.valida = rbind(dados[V,]) ; str(dados.valida)

# MODELAGEM POR ÁRVORE DE DECISÃO
require(rpart)
vcond.arv = rpart(TARGET_C ~ INPUT_A + INPUT_B + INPUT_C + INPUT_D, data=dados.treino,
method="anova")
summary(vcond.arv)
plot(vcond.arv) ; text(vcond.arv, cex=0.6)
plotcp(vcond.arv)
printcp(vcond.arv)
par(mfrow=c(2,1)) ; rsq.rpart(vcond.arv) ; par(mfrow=c(1,1))

# ANÁLISE DA DISTRIBUIÇÃO DA VARIÁVEL RESPOSTA
require(car)
require(nortest)
require(MASS)

hist(dados$TARGET_C)
hist(sqrt(dados$TARGET_C))
hist(log(dados$TARGET_C))

# DISTRIBUIÇÃO GAMMA
parametros.gammal = fitdistr(dados$TARGET_C, "gamma")

```

```

parametros.gamma2 = fitdistr(sqrt(dados$TARGET_C), "gamma")
parametros.gamma3 = fitdistr(log(dados$TARGET_C), "gamma")

qqPlot(dados$TARGET_C, distribution="gamma",
       shape=parametros.gamma1$estimate[1],
       rate=parametros.gamma1$estimate[2],
       main="Valor da Perda com Acordo",
       xlab="Distribuição Gamma", ylab="Valor da Perda com Acordo")
ks.test(dados$TARGET_C, "pgamma", shape=parametros.gamma1$estimate[1],
       rate=parametros.gamma1$estimate[2])

qqPlot(sqrt(dados$TARGET_C), distribution="gamma",
       shape=parametros.gamma2$estimate[1],
       rate=parametros.gamma2$estimate[2],
       main="Raiz do Valor da Perda com Acordo",
       xlab="Distribuição Gamma", ylab="Raiz do Valor da Perda com Acordo")
ks.test(sqrt(dados$TARGET_C), "pgamma", shape=parametros.gamma2$estimate[1],
       rate=parametros.gamma2$estimate[2])

qqPlot(log(dados$TARGET_C), distribution="gamma",
       shape=parametros.gamma3$estimate[1],
       rate=parametros.gamma3$estimate[2],
       main="Log do Valor da Perda com Acordo",
       xlab="Distribuição Gamma", ylab="Log do Valor da Perda com Acordo")
ks.test(log(dados$TARGET_C), "pgamma", shape=parametros.gamma3$estimate[1],
       rate=parametros.gamma3$estimate[2])

# DISTRIBUIÇÃO NORMAL
parametros.norm1 = fitdistr(dados$TARGET_C, "normal")
parametros.norm2 = fitdistr(sqrt(dados$TARGET_C), "normal")
parametros.norm3 = fitdistr(log(dados$TARGET_C), "normal")

qqPlot(dados$TARGET_C, distribution="norm",
       mean=parametros.norm1$estimate[1],
       sd=parametros.norm1$estimate[2],
       main="Valor da Perda com Condenação",
       xlab="Distribuição Normal", ylab="Valor da Perda com Condenação")
ks.test(dados$TARGET_C, "pnorm", mean=parametros.norm1$estimate[1],
       sd=parametros.norm1$estimate[2])
ad.test(dados$TARGET_C)

qqPlot(sqrt(dados$TARGET_C), distribution="norm",
       mean=parametros.norm2$estimate[1],
       sd=parametros.norm2$estimate[2],
       main="Raiz Quadrada do Valor da Perda com Condenação",
       xlab="Distribuição Normal", ylab="Raiz Quadrada do Valor da Perda com Condenação")
ks.test(sqrt(dados$TARGET_C), "pnorm", mean=parametros.norm2$estimate[1],
       sd=parametros.norm2$estimate[2])
ad.test(sqrt(dados$TARGET_C))

qqPlot(log(dados$TARGET_C), distribution="norm",
       mean=parametros.norm3$estimate[1],
       sd=parametros.norm3$estimate[2],
       main="Ln do Valor da Perda com Condenação",
       xlab="Distribuição Normal", ylab="Ln do Valor da Perda com Condenação")
ks.test(log(dados$TARGET_C), "pnorm", mean=parametros.norm3$estimate[1],
       sd=parametros.norm3$estimate[2])
ad.test(log(dados$TARGET_C))

# MODELAGEM POR MODELOS LINEARES GENERALIZADOS
vcond.mlg1 = glm(log(TARGET_C) ~ INPUT_A + INPUT_B + INPUT_C + INPUT_D, data=dados.treino,
family="gaussian")
summary(vcond.mlg1) ; anova(vcond.mlg1, test="Chisq")
vcond.mlg2 = glm(log(TARGET_C) ~ INPUT_A + INPUT_B + INPUT_D, data=dados.treino,
family="gaussian")
summary(vcond.mlg2) ; anova(vcond.mlg2, test="Chisq")
vcond.mlg = vcond.mlg2
summary(vcond.mlg)

lms = summary(vcond.mlg) ; str(lms)
r = lms$deviance.resid
qqPlot(r, distribution="norm",
       main="Resíduos Deviance",
       xlab="Distribuição Normal", ylab="Resíduos Deviance")
hist(r)
ad.test(r)

```

```

cvm.test(r)
lillie.test(r)

# MODELAGEM POR REDES NEURAIS ARTIFICIAIS
require(nnet)
rna.treinamento = function(neuronios=3, reqm.limite=0.10, tentativas.limite=100){
  reqm.hist = vector()
  tentativas = 0
  rna.valida.reqm = reqm.limite + 1
  while(rna.valida.reqm > reqm.limite && tentativas < tentativas.limite){
    rna.modelo = nnet(TARGET_CT ~ INPUT_A + INPUT_B + INPUT_C + INPUT_D,
data=dados.treino, size=neuronios, linout=TRUE, maxit=2000)
    tentativas = tentativas + 1
    rna.valida = (predict(rna.modelo, dados.valida) * maxim.rna) +
min(dados$TARGET_C)
    rna.valida.reqm = sqrt(mean((dados.valida$TARGET_C - rna.valida) ^ 2))
    reqm.hist[tentativas] = rna.valida.reqm
    plot(reqm.hist, type="l", ylim=c(reqm.limite*0.95, reqm.limite*1.05))
    print(rna.valida.reqm)
    print(tentativas)
  }
  print(min(reqm.hist))
  return(rna.modelo)
}
(vcond.rnax = rna.treinamento(neuronios=3, reqm.limite=277, tentativas.limite=500))
vcond.rna = vcond.rnax ; vcond.rna
vcond.rna

# CÁLCULO DAS MEDIDAS PARA COMPARAÇÃO DOS MODELOS
# VALORES AJUSTADOS
(vcond.arv.fitted = as.numeric(predict(vcond.arv, dados.treino)))
(vcond.arv.valida = as.numeric(predict(vcond.arv, dados.valida)))
(vcond.mlg.fitted = (exp(vcond.mlg$fitted.values)))
(vcond.mlg.valida = (exp(predict.glm(vcond.mlg, dados.valida, type="response"))))
(vcond.rna.fitted = (vcond.rna$fitted.values * maxim.rna) + min(dados$TARGET_C))
(vcond.rna.valida = (predict(vcond.rna, dados.valida) * maxim.rna) + min(dados$TARGET_C))

# RAIZ DO ERRO MÉDIO QUADRÁTICO
(vcond.arv.reqm.treino = sqrt(mean((dados.treino$TARGET_C - vcond.arv.fitted)^2)))
(vcond.arv.reqm.valida = sqrt(mean((dados.valida$TARGET_C - vcond.arv.valida)^2)))
(vcond.mlg.reqm.treino = sqrt(mean((dados.treino$TARGET_C - vcond.mlg.fitted)^2)))
(vcond.mlg.reqm.valida = sqrt(mean((dados.valida$TARGET_C - vcond.mlg.valida)^2)))
(vcond.rna.reqm.treino = sqrt(mean((dados.treino$TARGET_C - vcond.rna.fitted)^2)))
(vcond.rna.reqm.valida = sqrt(mean((dados.valida$TARGET_C - vcond.rna.valida)^2)))

# ERRO COM INTERVALO DE CONFIANÇA
(vcond.arv.ci.treino = cbind(mean(dados.treino$TARGET_C - vcond.arv.fitted) + c(-qnorm(0.975)
* vcond.arv.reqm.treino, qnorm(0.975) * vcond.arv.reqm.treino)))
(vcond.arv.ci.valida = cbind(mean(dados.valida$TARGET_C - vcond.arv.valida) + c(-qnorm(0.975)
* vcond.arv.reqm.valida, qnorm(0.975) * vcond.arv.reqm.valida)))
(vcond.mlg.ci.treino = cbind(mean(dados.treino$TARGET_C - vcond.mlg.fitted) + c(-qnorm(0.975)
* vcond.mlg.reqm.treino, qnorm(0.975) * vcond.mlg.reqm.treino)))
(vcond.mlg.ci.valida = cbind(mean(dados.valida$TARGET_C - vcond.mlg.valida) + c(-qnorm(0.975)
* vcond.mlg.reqm.valida, qnorm(0.975) * vcond.mlg.reqm.valida)))
(vcond.rna.ci.treino = cbind(mean(dados.treino$TARGET_C - vcond.rna.fitted) + c(-qnorm(0.975)
* vcond.rna.reqm.treino, qnorm(0.975) * vcond.rna.reqm.treino)))
(vcond.rna.ci.valida = cbind(mean(dados.valida$TARGET_C - vcond.rna.valida) + c(-qnorm(0.975)
* vcond.rna.reqm.valida, qnorm(0.975) * vcond.rna.reqm.valida)))

# AMPLITUDE DO INTERVALO DE CONFIANÇA
(vacor.arv.ampci.treino = vacor.arv.ci.treino[2]*2)
(vacor.arv.ampci.valida = vacor.arv.ci.valida[2]*2)
(vacor.mlg.ampci.treino = vacor.mlg.ci.treino[2]*2)
(vacor.mlg.ampci.valida = vacor.mlg.ci.valida[2]*2)
(vacor.rna.ampci.treino = vacor.rna.ci.treino[2]*2)
(vacor.rna.ampci.valida = vacor.rna.ci.valida[2]*2)

```