COMPARAÇÃO DE VARIANTES DE REDES NEURONAIS ARTIFICIAIS E DOS MODELOS MIXED LOGIT E LOGIT MULTINOMIAL NA AQUISIÇÃO DE PRODUTOS EM SUPERMERCADOS

Paulo Alexandre Botelho Rodrigues Pires, Universidade Portucalense Infante D. Henrique

RESUMO

São comparadas variantes das redes neuronais artificiais e os modelos de escolha discreta Mixed Logit e Logit Multinomial na previsão da aquisição de produtos com envolvimento fraco. Os resultados obtidos mostraram que, no critério performance, não existe uma supremacia de um tipo de modelo sobre o outro, mas que os modelos de escolha discreta são mais robustos do que as redes neuronais artificiais. O modelo Mixed Logit teve sempre uma capacidade previsional superior ao modelo Logit Multinomial, mas essa superioridade na medida de desempenho é marginal e, por essa razão, nos problemas analisados não se justifica o uso de um modelo com especificação mais complexa e mais exigente em recursos computacionais, em detrimento de um modelo mais simples e computacionalmente eficiente. Não foi possível estabelecer uma hierarquia das variantes do algoritmo BP, mas os algoritmos BPRPROP, BPGCS e BPLM devem constituir sempre uma primeira escolha.

PALAVRAS CHAVE: Redes Neuronais Artificiais, Logit Multinomial, Mixed Logit.

ABSTRACT

We compared the performance of the artificial neuronal networks with the performance of Mixed Logit and Multinomial Logit models in forecasting the acquisition of products with low involvement. The results obtained from the simulations executions revealed that it does not exist an evident supremacy of one type of models over the others. However, discrete choice models were always more robust and less demanding in computational resources. Furthermore, the practical implementation of these models allowed the attainment of an important set of guidelines on which the literature is omissive. It was not possible to establish an hierarchy in neuronal network variants, but the BPRPROP, BPGCS and BPLM should always be a first choice. Those guidelines, the large number of paradigms of artificial neuronal networks evaluated and the study of neglected concepts in previous studies comprise an important support to future investigations in the area.

KEY WORDS: Artificial Neural Networks, Multinomial Logit, Mixed Logit.

1. INTRODUÇÃO

A relevância e a pertinência da aplicação das redes neuronais artificiais (RNA) e dos modelos Mixed Logit (ML) e Logit Multinomial (LM) para aquisições de produtos em que existe um envolvimento fraco são bem reconhecidas, pois ao contrário das situações com envolvimento forte, em que o analista consegue obter informação relativa aos vários estágios, entre os quais a identificação das necessidades, das motivações, das

atitudes (nas suas componentes cognitiva e afectiva) e comportamento, raramente na primeira situação se consegue informação útil que permita identificar e relacionar a informação com os comportamentos do consumidor.

2. REDES NEURONAIS ARTIFICIAIS

Uma RNA é uma função matemática que tem uma correspondência a um grafo, correspondente à arquitectura da rede e que necessita de ser treinada por um algoritmo de aprendizagem. O algoritmo retropropagação BP é de todos os algoritmos de aprendizagem para RNA o mais conhecido e utilizado. O algoritmo BP tem uma aprendizagem com supervisão, pois requer dois conjuntos P e T que são, respectivamente, o conjunto com os valores das variáveis de entrada e de saída: $P=\{p\{1\},..,p\{N\}\}$ e $T=\{t\{1\},..,t\{N\}\}$. Este algoritmo pode ser aplicado seguindo um método de aprendizagem incremental, *batch*, ou intermédio (Möller, 1990; Heskes e Wiegerinck, 1996; Torresen, 1997). A aprendizagem incremental caracteriza-se pelo processamento de um par de vectores $(p\{i\},t\{i\})$, em que $(p\{i\}\in P)\wedge(t\{i\}\in T)\wedge(i=1..N)$, e a actualização imediata da matriz W, enquanto na aprendizagem *batch* o conjunto de pares de vectores é processado e só posteriormente são actualizadas as matrizes W dos valores das conexões (adaptado de LeCun, 1996 e Sarle, 1997). O algoritmo BP é uma aplicação do método do gradiente ou de outro método de optimização numérico a uma RNA com arquitectura unidireccional para minimizar a função erro. A equação que ajusta os valores das conexões no algoritmo BP é:

$$W\{k+1\}=W\{k\}-\alpha\frac{\partial E}{\partial W}\{k\}$$

Em que: α é o coeficiente de aprendizagem, $W\{k+1\}$ são os novos valores das conexões e $W\{k\}$ são os valores actuais das conexões e $\partial E/\partial W\{k\}$ é a derivada da função erro em relação às conexões.

A função erro média do erro quadrado é a mais utilizada na aprendizagem *batch* e pode ser adaptada para que o algoritmo de aprendizagem produza uma RNA com melhores capacidades de generalização, designando-se por regularização (Demuth e Beale, 1997):

$$E=MEQ=\frac{1}{N}\sum_{i=1}^{N}\left(t\{i\}-a^{M}\{i\}\right)^{2}$$

A nova função é: $MEQREG=\gamma MEQ+(1-\gamma)MEQW$, em que γ é o rácio da performance e $MEQW$: $MEQW=\frac{1}{N_w}\sum_{i=1}^{N_w}w_i^2$

O valor $MEQW$ representa a média da soma de todos os parâmetros da RNA. A função erro $MEQREG$ provoca um ajuste mais pequeno dos valores das conexões, o que apresenta como vantagem uma probabilidade menor de ocorrer sobre-aprendizagem. No entanto, esta função inclui um parâmetro adicional γ a ajustar. O valor óptimo deste parâmetro é difícil de calcular: um valor demasiado pequeno faz com que a RNA não reproduza convenientemente os pares de vectores (p,t); para um valor demasiado grande, a RNA resultante tem um comportamento de sobre-aprendizagem (Demuth e Beale, 1997).

Nos problemas de classificação utiliza-se outro género de funções erro: as funções erro entrópicas cruzadas ou logarítmicas, (Bishop, 1995; Reed e Marks, 1999). As funções erro entrópicas cruzadas têm a particularidade de retropropagar o valor do erro proporcional entre o valor desejado e o valor produzido. Desta forma, evita-se o

problema quando ocorre a saturação das funções sigmóides (Oh e Lee, 1995; Joost e Schiffmann, 1997). A função erro entrópica cruzada mais comum é (Bishop, 1995):

$$E = -\sum_{i=1}^{N}\sum_{j=1}^{S^M}\left(t_j\{i\}\ln\left(1+a_j^M\{i\}\right)+\left(1-t_j\{i\}\right)\ln\left(1-a_j^M\{i\}\right)\right)$$

Após a descoberta do algoritmo BP surgiram inúmeras variantes, cujo propósito foi atenuar as limitações e complexidades identificadas. As primeiras iniciativas incidiram sobretudo na pesquisa de alternativas que proporcionassem convergências mais rápidas, podendo ser divididas em duas categorias (Hagan et al., 1996): a primeira categoria inclui as técnicas heurísticas e na segunda categoria enquadram-se as técnicas de optimização numérica. Existe ainda um conjunto de técnicas que têm actualmente uma repercussão importante e não se enquadram nas categorias definidas atrás, tais como a optimização por algoritmos genéticos, os algoritmos construtivos e outras.

O algoritmo BP Momemtum (BPMO) é técnica heurística, requerendo um coeficiente adicional γ, designado por momentum, e as alterações aos valores das conexões, $\{k+1\}$, dependem de várias iterações ou épocas anteriores (consoante o método de aprendizagem) (Fausett, 1994). Na sua forma mais simples (primeira ordem) o algoritmo BPMO faz depender as actualizações dos valores das conexões $\{k+1\}$ de $\{k\}$ e $\{k-1\}$ (Bishop, 1992; Hagan et al., 1996), traduzida na equação:

$$W\{k+1\} = W\{k\} - \alpha\frac{\partial E}{\partial W}\{k\} + \gamma\Delta W\{k-1\}$$

O coeficiente momentum satisfaz sempre a condição $0 \le \gamma \le 1$. Os valores típicos para os coeficientes são (Reed e Marks, 1999): $0.05 \le \alpha \le 0.75$ e $0 \le \gamma \le 0.9$. O coeficiente de aprendizagem tem como valor mais sugerido $\alpha = 0.1$.

O algoritmo BP com coeficiente de aprendizagem variável assenta em ideias muito simples. Se, após a actualização dos valores das conexões, a função erro aumentou, então provavelmente o algoritmo ultrapassou o mínimo. O coeficiente de aprendizagem deve ser reduzido e as alterações dos valores das conexões da última iteração devem ser ignoradas. Se a função erro decresceu, então as alterações dos valores das conexões são aceites, mas o valor do coeficiente de aprendizagem é aumentado (Bishop, 1995). Os valores aconselhados para os factores multiplicativos são: $\rho = 1.1$ e $\eta = 0.5$.

$$\alpha\{k+1\} = \begin{cases} \alpha\{k\}\times\rho & \text{se } \Delta E < 0 \\ \alpha\{k\}\times\eta & \text{se } \Delta E > 0 \end{cases}$$

O algoritmo Resilient Propagation (BPRPROP) foi proposto por (Riedmiller e Braun, 1993; Riedmiller, 1994) e ajusta os valores das conexões avaliando apenas a variação do sinal do gradiente, ao contrário dos algoritmos descritos atrás, em que os ajustes dependem do valor do gradiente. O algoritmo BPRPROP segue o método de aprendizagem *batch* e os ajustes dos valores das conexões são efectuados com as equações:

$$W_{ij}\{k+1\} = W_{ij}\{k\} - \delta_{ij}\{k\} \quad \text{se } \frac{\partial E}{\partial W_{ij}} > 0$$

$$W_{ij}\{k+1\} = W_{ij}\{k\} + \delta_{ij}\{k\} \quad \text{se } \frac{\partial E}{\partial W_{ij}} < 0$$

$$W_{ij}\{k+1\} = W_{ij}\{k\} \quad \text{se } \frac{\partial E}{\partial W_{ij}} = 0$$

Embora restringidos às condições anteriores, os ajustes só são efectuados se não houver alteração do sinal das derivadas parciais em épocas sucessivas, isto é: $\partial E/\partial W_{ij}\{k-1\}\partial E/\partial W_{ij}\{k\} \geq 0$. Caso contrário, pressupõe-se que o algoritmo ultrapassou a zona onde se encontra o mínimo e, por isso, a última actualização das conexões deve ser ignorada: $\Delta W_{ij}\{k\} = -\Delta W_{ij}\{K-1\}$, se $\partial E/\partial W_{ij}\{k-1\}\partial E/\partial W_{ij}\{k\} < 0$.

Os algoritmos dos gradientes conjugados (BPGC) utilizam a técnica de optimização numérica dos gradientes conjugados para localizar o óptimo. Ao contrário do algoritmo BP que ajusta os valores das conexões na direcção mais inclinada da função erro, no algoritmo BPGC a procura é executada sucessivamente em direcções conjugadas (duas direcções são conjugadas se $cA\{k\}v = 0$). Existem algumas variantes do algoritmo BPGC, que resultam de cálculos da direcção de procura diferentes, mas a sequência de passos dessas variantes é semelhante e está descrita nas linhas seguintes (Jervis e Fitzgerald, 1993; Hagan et al., 1996; Demuth e Beale, 1997; LeCun et al., 1998). O primeiro passo no algoritmo BPGC selecciona como primeira direcção de procura o simétrico do gradiente: $d\{0\} = -g\{0\}$, $g\{k\} \equiv \frac{\partial E}{\partial W}\{k\}$. No segundo passo procede-se ao ajustamento dos valores das conexões, em que o valor de $\alpha\{k\}$ minimiza $E(W)$ na direcção $d\{k\}$: $W\{k+1\} = W\{k\} + \alpha\{k\}d\{k\}$. Os valores sucessivos de $d\{k\}$ são obtidos iterativamente pela equação: $d\{k\} = -g\{k\} + \beta\{k\}d\{k-1\}$. As diferentes versões do algoritmo BPGC são consequência directa de cálculos diferentes do parâmetro $\beta\{k\}$. Obtêm-se, assim, os algoritmos (Moreira e Fiesler, 1995; Hagan et al., 1996; Demuth e Beale, 1997; Reed e Marks, 1999):

$$\text{Fletcher-Reeves: } \beta\{\text{k}\} = \frac{g^T\{\text{k}\}g\{\text{k}\}}{g^T\{\text{k-1}\}g\{\text{k-1}\}},\ \text{Polak-Ribiére: } \beta\{\text{k}\} = \frac{\Delta g^T\{\text{k-1}\}g\{\text{k}\}}{g^T\{\text{k-1}\}g\{\text{k-1}\}},\ \text{Hestenes-Steifel: } \beta\{\text{k}\} = \frac{\Delta g^T\{\text{k}\}g\{\text{k}\}}{\Delta g^T\{\text{k-1}\}d\{\text{k-1}\}}$$

Os algoritmos Quasi-Newton utilizam o método de Newton, mas não requerem o cálculo das derivadas de segunda ordem. Neste algoritmos constrói-se uma matriz aproximadamente igual à inversa da matriz $A\{k\}$, utilizando informação do gradiente de primeira ordem em vez de se calcular $A\{k\}$ e, posteriormente, inverte-la (Bishop, 1995). O método Quasi-Newton mais referenciado e que demonstra uma performance melhor é da autoria de Broyden, Fletcher, Goldfarb e Shanno (o algoritmo é referenciado por BFGS) (Demuth e Beale, 1997; Reed e Marks, 1999). O ajuste dos valores das conexões segue as equações (Bishop, 1995):

$$W\{k+1\} = W\{k\} + \alpha\{k\}G\{k\}g\{k\}$$

Em que: $G\{k+1\} = G\{k\} + \frac{dd^t}{d^tv} - \frac{\left(G\{k\}v^t\right)v^tG\{k\}}{v^tG\{k\}v} + \left(v^tG\{k\}v\right)uu^t$ e $d = W\{k+1\} - W\{k\}$, $v = g\{k+1\} - g\{k\}$, $u = \frac{d}{d^tv} - \frac{G\{k\}v}{v^tG\{k\}v}$

Os algoritmos BP Quasi-Newton, tal como os algoritmos BPGC e BPGCS, requerem a minimização da função erro na direcção de procura. No entanto, ao contrário dos algoritmos BP baseados no método dos gradientes conjugados, a precisão da localização do mínimo não necessita de ser tão eficaz, o que se traduz numa menor exigência computacional (Bishop, 1995; Reed e Marks, 1999).

Com o algoritmo BPGC evita-se o cálculo da matriz hessiana e, simultaneamente, obtém-se o valor para o coeficiente de aprendizagem $\alpha\{k\}$ através da minimização da função erro $E(W)$ na direcção $d\{k\}$. Contudo, este processo é muito ineficiente, pois o método de minimização é iterativo e cada iteração requer o cálculo de $E(W)$. Tanto o processo iterativo, como o cálculo de $E(W)$ são computacionalmente intensivos e têm como

consequência uma degradação da performance final do algoritmo (Bishop, 1995). O algoritmo BPGCS (BP gradientes conjugados escalado), introduzido por (Möller, 1990), evita o processo iterativo de encontrar o mínimo de $E(W)$ através do procedimento de procura por recta, substituindo-o por uma aproximação Levenberg-Marquardt para ajustar o coeficiente de aprendizagem.

O algoritmo BP Levenberg-Marquardt (BPLM), descrito em (Hagan e Menhaj, 1994), é uma variante do método de Newton. Embora o método de Newton seja de segunda ordem e, por isso, requer o cálculo da matriz hessiana, o algoritmo BPLM baseia-se no método Gauss-Newton, evitando esse cálculo. Para o efeito, o algoritmo utiliza uma aproximação à matriz hessiana através da matriz jacobiana (Hagan et al., 1996). É de salientar que este algoritmo só pode ser aplicado com a função erro soma do erro quadrado. Os valores das conexões são ajustados seguindo a equação (Hagan e Menhaj, 1994):

$$W\{k+1\} = W\{k\} - \left(J^t\{k\}J\{k\} + \mu\{k\}I\right)^{-1} J^t\{k\}e\{k\}$$

Em que J é a matriz jacobiana, μ é multiplicado por um factor β quando uma época resulta num aumento do erro e é dividido por β quando o erro decresce após uma época (os valores de referência são $\beta = 10$ e $\mu = 0.01$).

3. MODELOS DE ESCOLHA DISCRETA

Num modelo de escolha discreta (MED) um decisor n selecciona uma alternativa entre J alternativas e $C_{nJ} \subseteq C$, em que C_{nJ} é o conjunto de alternativas do decisor n e C é o conjunto universal. Ao escolher a alternativa i o decisor recebe uma utilidade U_{ni}, em que $i \in C_{nJ}$ e, por conseguinte, a utilidade de uma alternativa é $U_{ni}, \forall i \in C_{nJ}$. Um decisor escolhe sempre a alternativa que lhe dá a utilidade maior: $U_{ni} > U_{nj}, \forall i, j \in C_{nJ} \wedge i \neq j$. O decisor conhece a sua função utilidade e a sua escolha resulta sempre da maximização da utilidade. Para o modelo Logit obtém-se a seguinte equação, podendo-se encontrar a sua derivação em várias referências com destaque para (Ben-Akiva e Lerman, 1985; Train, 1986; Louviere et al., 2000; Train, 2003):

$$P_{ni} = \frac{e^{V_{ni}}}{\sum_{j \in C_{nJ}} e^{V_{nj}}}$$

O modelo de escolha discreta Mixed Logit (ML), também designado por RPL (Random Parameter Logit), Mixed Multinomial Logit (MMNL), Kernel Logit ou Logit Híbrido, é consistente com a maximização da utilidade com elementos aleatórios, mas também permite outros paradigmas de escolha. É um modelo muito flexível e que não tem as limitações do modelo Logit, permitindo variações do gosto aleatórias, padrões de substituição sem restrições e correlações entre os termos da utilidade não observada para observações diferentes (Train, 2003). Tal como o modelo Logit também o modelo ML permaneceu durante algum tempo limitado ao conhecimento de um número reduzido de pessoas, ressurgindo e conhecendo uma difusão assinalável com o trabalho de (McFadden e Train, 1995). Nele, os autores estabelecem que o modelo ML consegue aproximar com a precisão desejada as probabilidades de escolha de qualquer modelo de escolha discreta derivado do modelo RUM. Em face do potencial e da flexibilidade exibidos pelo modelo ML as imposições amostrais crescem, bem como as exigências de especificação e de identificação (Hensher e Greene, 2001).

O modelo ML pode ser definido como sendo um modelo LM com parâmetros aleatórios retirados de uma função f, sendo formulado com (Train, 2003):

$$P_{ni} = \int L_{ni}(\beta) f(\beta) d\beta$$

Em que L_{ni} é a probabilidade obtida com o modelo Logit com os parâmetros β:

$$L_{ni}(\beta) = \frac{e^{V_{ni}(\beta)}}{\sum_{j=1}^{J} e^{V_{nj}(\beta)}}$$

Quando a utilidade V_{ni} é linear nos seus parâmetros β então reduz-se a:

$$P_{ni} = \int \frac{e^{\beta' x_{ni}}}{\sum_{j=1}^{J} e^{\beta' x_{nj}}} f(\beta) d\beta$$

Assim, seguindo indicações de (Train, 1999), as probabilidades do modelo ML são uma média ponderada do modelo Logit e são avaliadas com valores diferentes de β, em que as ponderações são proporcionadas pela função densidade $f(\beta|\theta)$. O objectivo é estimar os parâmetros θ da função f. A função densidade $f(\beta)$ pode ser contínua ou discreta, mas usualmente $f(\beta)$ é especificada contínua, podendo assumir as funções mais usuais de que são exemplo a normal, a uniforme, a triangular, a log-normal (Train, 2003) e a gama (Hensher e Greene, 2001). A função f é também usualmente designada por distribuição de mistura. A derivação do modelo ML segue genericamente o paradigma da maximização da utilidade, sobressaindo duas perspectivas (Hensher e Greene, 2001; Train, 2003): (1) coeficientes aleatórios, em que é permitida a variação do coeficiente de cada variável; (2) componentes dos erros, que introduzem correlações entre alternativas. As duas perspectivas, ainda que formalmente equivalentes, afectam a especificação do modelo, pois têm finalidades diferentes. Especificamente, quando se opta por uma especificação com coeficientes aleatórios pretende-se admitir a variação do gosto não observado entre os decisores em função das variáveis exógenas, enquanto a opção por uma especificação com componentes dos erros é adequada para a obtenção de padrões de substituição entre alternativas mais flexível (Bhat, 2003).

Decorrente da flexibilidade e potencial do modelo ML emergem requisitos adicionais na sua especificação e identificação, que se não forem devidamente ponderados resultam em estimativas enviesadas ou num modelo inadequado para representar a amostra (Walker, 2002), a qual afirma que não existem soluções tipificadas que permitam responder aos problemas da especificação e da identificação, podendo o analista recorrer a regras empíricas e enquadrá-las no seu caso. Os conceitos e as ideias subjacentes à especificação e identificação podem ser revistos detalhadamente em (Ben-Akiva et al., 2001; Hensher e Greene, 2001; Walker, 2002).

4. COMPARAÇÃO ENTRE OS MED E AS RNA – APLICAÇÃO PRÁTICA

As fontes de informação incidentes na comparação entre a performance dos MED e as RNA é vasta, salientando-se as seguintes: (Agrawal e Schorling, 1996; West et al., 1997; Bentz e Merunka, 2000; Hruschka, 2000; Hruschka et al., 2001; Yip et al., 2001; Fish e Blodgett, 2002; Fish et al., 2002; Papatla et al., 2002; Vroomen et al., 2004). Este problema já foi objecto de estudo anterior em (Pires, 2005), abrindo novas perspectivas nesta comparação.

Para avaliar a performance dos MED e das RNA procedeu-se à aplicação dos diversos modelos em três amostras em que são adquiridos produtos com envolvimento fraco, nomeadamente detergentes líquidos, bacon e manteigas. Os dados foram recolhidos através das compras efectuadas em supermercados pelos elementos do agregado familiar. Ao agregado familiar foi atribuído um código único e as compras reflectem o seu comportamento durante dois anos consecutivos. Os produtos foram igualmente codificados com um código UPC, vulgarmente designado por código de barras (esta base de dados é conhecida por ERIM[1]).

Na vertente das RNA são utilizados: o algoritmo BP e as suas variantes. São igualmente analisados factores relevantes referidos nas fontes de informação como a inicialização dos valores das conexões, a aprendizagem incremental e *batch*, a regularização e a paragem antecipada da aprendizagem. As funções erro também são comparadas e avaliadas no desempenho dos algoritmos, nomeadamente a função erro média da soma do erro quadrado e a função erro entrópica. Nos MED foram seleccionados o modelo LM e o modelo ML.

O objectivo primordial desta comparação residiu, essencialmente, na identificação do modelo com melhores capacidades previsionais para produtos com envolvimento fraco, pretendendo-se ainda obter outras ilações como a facilidade de implementação e a robustez. Para o efeito, os dois conjuntos de modelos, RNA e MED, foram calibrados independentemente e reconheceu-se a imperatividade de iniciar o processo pelas RNA porque, sendo um método não paramétrico, proporcionam poucas indicações da adequação das variáveis explicativas. Para o efeito, foram também definidas duas medidas de avaliação de desempenho: (1) média da soma do erro quadrado; (2) percentagem dos vectores classificados correctamente.

A função activação para as camadas interiores foi sempre a função activação tangente hiperbólica, porque proporciona uma convergência mais rápida do que a função activação logística. Para a camada de saída utilizou-se a função softmax. Procederam-se a diferentes simulações em RNA com arquitecturas com várias camadas interiores, variando o número de neurónios de cada camada, mas as RNA com uma camada interior obtiveram sempre resultados superiores. As RNA com múltiplas camadas interiores evidenciaram quase sempre uma tendência para convergir prematuramente para um mínimo local, obtendo resultados inferiores. Em consequência desta experimentação foram sempre optimizadas RNA com uma camada interior. Nas RNA a amostra foi dividida em três conjuntos: conjunto de aprendizagem (50% dos pares de vectores); conjunto de validação (25% dos pares de vectores); conjunto de teste (25% dos pares de vectores). Esta divisão é necessária para utilizar o método da paragem antecipada da aprendizagem, o qual assegura uma identificação mais rápida da arquitectura óptima. É ainda de referir dois aspectos importantes. O conjunto de teste é sempre disjunto do conjunto de aprendizagem e do conjunto de validação. Os conjuntos são formados aleatoriamente em cada simulação e são utilizados para os vários modelos. As medidas de desempenho foram sempre aplicadas aos conjuntos de aprendizagem e de teste. Mas a avaliação das medidas de desempenho finais foi feita sobre o conjunto de teste (quer para RNA quer para os MED) pois é esta que indica a capacidade previsional dos modelos.

Para a aquisição de produtos com envolvimento fraco as fontes de informação são determinantes, sobressaindo a importância da informação de aquisições anteriores. A variável a incorporar foi designada por Fid, sugerida por Guadagni e Little. Como a amostra em estudo não avalia a fidelização quanto à dimensão da alternativa, a

[1] A base de dados ERIM está disponível em ftp://gsbper.uchicago.edu/acnpanel/.

variável Fid contempla apenas a fidelização à marca. Assim, a equação que define Fid é: $Fid_{j,t} = \gamma Fid_{j,t-1} + (1-\gamma) y_{j,t-1}$. Em que γ é um parâmetro de atenuação, $y_{j,t-1}$ assume o valor unitário se a alternativa j foi a escolhida na aquisição $t-1$. A condição inicial para esta variável impõe que $\sum(Fid_{j,1}) = 1$. O valor atribuído a γ foi de 0.9.

A primeira amostra tem 3055 observações pertencentes a 400 agregados familiares. Das 3055 observações 398 são observações incompletas e, por essa razão, foram removidas da amostra. Assim, o total de observações disponíveis com informação completa é de 2657. O número de alternativas do conjunto de escolha é seis: (1) Tide; (2) Wisk; (3) Eraplus; (4) Surf; (5) Solo; (6) All. As variáveis incluídas na amostra, caracterizadoras da alternativa e do agregado familiar, são as seguintes: Gasto Noutros Produtos na mesma aquisição; Dimensão do Agregado Familiar; Volume Adquirido na Aquisição Precedente; Tempo Decorrido entre Aquisições; Preço de cada Alternativa; Promoção para cada Alternativa; Destaque para cada Alternativa. As variáveis de saída/dependentes são em número de seis e correspondem a cada alternativa. Para cada registo a variável escolhida assume o valor unitário, enquanto as outras variáveis assumem o valor nulo. Realizou-se uma avaliação à amostra para identificar outliers e para obter as percentagens de cada alternativa. Concluiu-se que a amostra não é balanceada e analisando os histogramas das variáveis concluiu-se, igualmente, a probabilidade de existirem outliers na amostra.

A primeira fase da calibração das RNA começa com a identificação do número de neurónios óptimo, o qual é calculado através de um conjunto de simulações sequenciais em que se vai acrescentando neurónios interiores. O número de neurónios óptimo calculado com o algoritmo BPLM foi de 6. A fase seguinte consta na identificação das variáveis relevantes, consistindo igualmente num conjunto de simulações sequenciais, nas quais, em cada simulação, se elimina uma variável e se avalia o aumento do erro do conjunto de testes. O número de variáveis de entrada da RNA passou a ser 17, enquanto o número de variáveis de saída permaneceu em 6, tendo-se uma arquitectura 17-6-6. Na Tabela 1 são exibidos os resultados obtidos com as variantes do algoritmo BP para a arquitectura óptima, identificada com as simulações anteriores.

Tabela 1 – Melhor variante do algoritmo BP para a amostra de detergentes líquidos.

Variantes do algoritmo BP	Erro do conjunto de teste			Percentagem de vectores classificados correctamente					
				Teste			Aprendizagem		
	Mínimo	Máximo	Média	Mín.	Max.	Méd.	Mín.	Max.	Méd.
Quasi-Newton BFGS	0.7883	1.8069	0.8629	2.31	34.57	24.76	2.92	39.24	26.59
Powell-Beale GC	0.6283	1.7612	0.9260	2.48	51.77	29.30	2.75	53.81	30.47
Fletcher-Powell GC	0.7835	2.2345	1.0672	3.55	39.72	22.94	2.57	40.21	22.67
Polak-Ribiere GC	0.6448	1.6995	0.9623	2.31	48.76	26.28	3.28	48.45	27.00
Gradiente (incremental)	0.7981	2.0114	1.1801	2.48	40.43	22.07	2.92	39.68	22.75
Gradiente com coeficiente aprendizagem variável	0.7994	2.3794	1.3022	1.60	36.52	21.44	3.10	38.39	22.29
Gradiente com momentum	0.8123	2.6619	1.2326	2.31	28.55	20.41	2.48	30.56	20.22
Gradiente com momentum e coeficiente aprendizagem variável	0.7979	2.4538	1.3102	2.84	36.70	21.70	2.92	34.01	21.59
BPLM	0.2950	0.5667	0.3958	55.32	81.56	73.66	58.72	87.95	80.27
BPRPROP	0.3299	0.8019	0.4164	29.08	78.72	71.55	32.24	81.40	74.64
BPGCS	0.7789	1.8111	0.9307	3.19	38.12	26.67	2.39	39.06	27.31

É notória, neste caso, a supremacia do algoritmo BPLM obtendo valores melhores para todas as medidas de desempenho, aproximando-se de perto o algoritmo BPRPROP. Na Tabela 2 são apresentados os resultados das simulações com regularização e paragem antecipada da aprendizagem, regularização e função erro entrópica.

Tabela 2 – Medidas de desempenho com variantes para a amostra de detergentes líquidos.

Valor de regularização	Erro do conjunto de teste			Percentagem de vectores classificados correctamente					
				Teste			Aprendizagem		
	Mínimo	Máximo	Média	Mín.	Max.	Méd.	Mín.	Max.	Méd.
Medidas de desempenho com regularização e paragem antecipada da aprendizagem									
0.6	0.2905	0.6432	0.3761	48.76	80.67	75.29	57.22	87.95	82.26
Medidas de desempenho com regularização									
0.5	0.3307	0.3940	0.3617	75.00	78.90	77.38	82.40	86.36	84.45
Identificação da melhor variante do algoritmo BP com função erro entrópica									
BPGCS	0.3400	0.3785	0.3559	74.82	79.61	76.67	83.70	86.06	84.87
Quasi-Newton BFGS	0.3296	0.4111	0.3720	71.99	78.72	75.51	82.28	85.29	83.60
Gradiente incremental	0.3591	0.4326	0.3961	71.81	77.48	75.09	79.62	83.93	81.97

Para estas simulações, a aplicação simultânea da paragem antecipada da aprendizagem e da regularização com o valor 0.6 obteve resultados superiores às variantes do algoritmo BP. A utilização de regularização obteve os melhores resultados, suplantando todos os outros métodos com o valor 0.5. O algoritmo BP com gradientes conjugados escalado e função erro entrópica foi superior às outras variantes do algoritmo BP com a mesma função erro e às outras variantes com função erro média da soma do erro quadrado.

O modelo LM foi estimado com 75% dos valores da amostra, sendo os 25% restantes guardados para avaliar a capacidade previsional do modelo. Foram sempre executadas 30 simulações com conjuntos disjuntos e formados aleatoriamente. A construção da função utilidade de cada alternativa segue um processo iterativo de eliminação de variáveis tendo como finalidade obter o $\bar{\rho}^2$ maior e, simultaneamente, obter um modelo com o menor número de parâmetros. A eliminação das variáveis requer algum cuidado, pois, por vezes, verificou-se que a eliminação de uma variável baseada apenas no rácio t diminui o $\bar{\rho}^2$ e afecta as outras variáveis, significando uma interacção entre as mesmas. Na Tabela 3 estão as medidas de desempenho obtidas para as 30 simulações. O modelo estimado exibe medidas de desempenho boas e demonstra uma boa capacidade previsional. A característica mais saliente é, no entanto, a sua robustez, pois foi obtido um intervalo de valores reduzido.

Tabela 3 – Medidas de desempenho para o LM com a variável Fid para a amostra de detergentes líquidos.

Erro do conjunto de teste			Percentagem de vectores classificados correctamente					
			Teste			Aprendizagem		
Mínimo	Máximo	Média	Mínimo	Máximo	Média	Mínimo	Máximo	Média
0.2881	0.3099	0.2998	68.45	70.92	69.62	69.44	71.07	69.95

No modelo ML a opção por coeficientes aleatórios ou por componentes do erro e a escolha das funções distribuição teve sempre como premissa a maximização da capacidade previsional do modelo, considerando as medidas de desempenho definidas. Como tal, a especificação seleccionada foi a de coeficientes aleatórios, escolhendo-se a distribuição triangular para o coeficiente do preço. As funções utilidade das alternativas são as mesmas que foram identificadas para o modelo LM. Os resultados obtidos para as 30 simulações estão representados na Tabela 4.

Tabela 4 – Medidas de desempenho para o Mixed Logit com a variável Fid.

Erro do conjunto de teste			Percentagem de vectores classificados correctamente					
			Teste			Aprendizagem		
Mínimo	Máximo	Média	Mínimo	Máximo	Média	Mínimo	Máximo	Média
0.2788	0.3124	0.2954	68.90	71.28	70.21	69.19	70.11	69.64

O modelo ML apresenta resultados superiores ao modelo LM, ainda que essa superioridade seja reduzida ou

mesmo insignificante. O acréscimo computacional e a complexidade inerentes ao modelo ML não justificam de todo a sua utilização para este problema.

As amostras bacon e manteigas referem-se também a aquisições de produtos com envolvimento fraco. A estrutura das duas amostras é semelhante, contendo as variáveis: QtAdUA - Quantidade Adquirida na Última Aquisição; Gasto - Gasto total na aquisição; Inv – Inventário; Preço - Preço (de cada alternativa); DE - Destaque (de cada alternativa); Feat - Característica (de cada alternativa); Educ - Educação do agregado familiar; DAgFa - Dimensão do Agregado Familiar; Mret - Número de homens na reforma; Fret - Número de mulheres na reforma; Mwork - Número de homens a trabalhar; Fwork - Número de mulheres a trabalhar; Venc - Vencimento do agregado familiar; InvA - Inventário de cada alternativa; Fidelização - Alternativa escolhida na aquisição anterior; Fid - Variável Fid (de cada alternativa).

A amostra Bacon tem 7 alternativas, enquanto a amostra Manteiga tem 5 alternativas. Estas amostras têm uma dimensão pequena, sendo compostas por 1354 aquisições e 298 agregados familiares e por 1303 e 245 agregados familiares, respectivamente para as amostras Bacon e Manteiga. Os procedimentos de calibração das RNA e de estimação dos MED seguiram a mesma sequência de passos que foram aplicados na amostra de detergentes líquidos, residindo a excepção na inclusão da variável Fid. É de notar que as amostras já incluem a variável Fidelização, não necessitando de reduzir a dimensão para construir essa mesma variável. As duas amostras têm as mesmas características da amostra detergentes líquidos, contendo valores dispersos e não são balanceadas.

Começando por analisar a amostra Bacon com as RNA, foram identificadas as variáveis independentes relevantes, reduzindo-se ao conjunto: QtAdUA; Inv, Preço (de cada alternativa); Fidelização; Fid (de cada alternativa). O número de variáveis de entrada ficou estabelecido em 17 e o número de variáveis de saída em 7.

O número de neurónios óptimo foi identificado como sendo 7, impondo uma arquitectura 17-7-7. Após a identificação do número de neurónios óptimo seguiram-se as simulações para identificar a melhor variante do algoritmo BP, que são apresentadas na Tabela 5.

Tabela 5 – Melhor variante do algoritmo BP para a amostra Bacon.

Variantes do algoritmo BP	Erro do conjunto de teste			Percentagem de vectores classificados correctamente					
				Teste			Aprendizagem		
	Mínimo	Máximo	Média	Mín.	Max.	Méd.	Mín.	Max.	Méd.
Quasi-Newton BFGS	0.5976	1.3915	0.8359	7.37	59.29	43.49	9.76	67.16	44.25
Powell-Beale GC	0.5099	1.2570	0.7637	22.71	69.03	49.13	29.29	70.12	50.79
Fletcher-Powell GC	0.5429	1.7724	0.8649	1.47	66.96	41.28	2.66	66.72	42.65
Polak-Ribiere GC	0.5574	1.0676	0.6977	23.01	65.19	52.97	22.63	69.08	55.74
Gradiente (incremental)	0.6014	2.3529	1.2998	1.77	62.83	32.13	1.63	64.05	31.92
Gradiente com coeficiente aprendizagem variável	0.5406	1.4949	0.7348	2.36	66.67	46.80	2.07	64.79	48.88
Gradiente com momentum	0.6111	2.1221	1.1990	2.33	63.22	39.21	2.53	66.98	36.42
Gradiente com momentum e coeficiente aprendizagem variável	0.4972	1.6499	0.9109	2.65	66.67	38.48	1.78	66.72	38.86
BPLM	0.4768	2.1441	0.7059	5.29	70.50	52.54	12.78	75.89	57.65
BPRPROP	0.4748	0.6200	0.5031	55.46	68.14	63.11	59.62	71.60	66.20
BPGCS	0.4924	1.7216	0.7035	2.95	66.96	51.40	3.40	71.30	53.34

Da análise da Tabela 5 retira-se que o algoritmo BPRPROP obteve as medidas de desempenho melhores, mostrando também uma robustez superior a todos os outros. As simulações posteriores utilizaram sempre este algoritmo. A Tabela 6 contém os resultados das simulações com regularização e paragem antecipada da aprendizagem, regularização e função erro entrópica.

Tabela 6 – Medidas de desempenho com variantes para a amostra Bacon.

Valor de regularização	Erro do conjunto de teste			Percentagem de vectores classificados correctamente					
				Teste			Aprendizagem		
	Mínimo	Máximo	Média	Mín.	Max.	Méd.	Mín.	Max.	Méd.
Medidas de desempenho com regularização e paragem antecipada da aprendizagem									
0.8	0.4413	0.5607	0.4996	60.18	71.09	65.56	68.20	76.48	71.99
Medidas de desempenho com regularização									
0.7	0.4540	0.5938	0.5279	60.77	68.73	64.53	73.00	78.33	75.30
Identificação da melhor variante do algoritmo BP com função erro entrópica									
BPGCS	0.5176	0.6062	0.5465	57.52	65.78	62.83	75.67	79.41	76.74
Quasi-Newton BFGS	0.5067	0.6795	0.5661	52.04	64.01	60.67	62.96	74.58	71.28
Gradiente (incremental)	0.4944	0.5901	0.5393	58.41	67.85	62.98	70.34	74.48	71.80

A regularização com paragem antecipada da aprendizagem melhorou as medidas de desempenho e tornou o algoritmo BPRPROP mais robusto. O valor 0.8 para o termo de regularização obteve o erro médio menor. Nesta amostra, a aprendizagem com regularização não melhorou a performance do algoritmo. A função erro entrópica melhorou os resultados da aprendizagem dos algoritmos analisados quando comparada com a função erro média da soma do erro quadrado (não melhorou quando se utiliza regularização). Note-se que os algoritmos de segunda ordem obtiveram resultados piores do que o algoritmo BP.

Os procedimentos para a estimação dos modelos de escolha discreta seguiram a sequência de etapas descritas na amostra anterior. Em primeiro lugar foi estimado o modelo LM que identificou as variáveis independentes relevantes: Preço (de cada alternativa); DE (de cada alternativa); Feat (de cada alternativa); QtAdUA; Venc; InvA; Gasto; Fid (de cada alternativa); Fidelização. As simulações proporcionaram os resultados apresentados na Tabela 7.

Tabela 7 – Medidas de desempenho para o LM para a amostra Bacon.

Erro do conjunto de teste			Percentagem de vectores classificados correctamente					
			Teste			Aprendizagem		
Mínimo	Máximo	Média	Mínimo	Máximo	Média	Mínimo	Máximo	Média
0.4180	0.4785	0.4473	55.26%	58.43%	57.14%	55.71%	57.58%	56.66%

O modelo ML foi estimado utilizando uma distribuição triangular para o coeficiente da variável Feat e uma distribuição uniforme para o coeficiente da variável DE.

Tabela 8 – Medidas de desempenho para o Mixed Logit para a amostra Bacon.

Erro do conjunto de teste			Percentagem de vectores classificados correctamente					
			Teste			Aprendizagem		
Mínimo	Máximo	Média	Mínimo	Máximo	Média	Mínimo	Máximo	Média
0.4112	0.4864	0.4445	55.76%	58.23%	56.74%	57.73%	59.89%	58.70%

O modelo Mixed Logit não reflectiu um aumento significativo da medida de desempenho média da soma do erro quadrado para o conjunto de teste, verificando-se um decréscimo, ainda que ligeiro, dos vectores classificados correctamente para o mesmo conjunto.

A amostra com os dados referentes a aquisições de marcas de manteiga manteve a sucessão de passos efectuados para a amostra com os dados das aquisições dos produtos de bacon. As primeiras simulações tiveram como finalidade identificar as variáveis independentes relevantes. O número de variáveis de entrada ficou estipulado em 27 e o número de variáveis de saída em 5 e o erro médio menor para 30 simulações foi obtido com 5 neurónios, constituindo esse número de neurónios o mais adequado e resultando numa arquitectura 27-5-5.

Tabela 9 – Melhor variante do algoritmo BP para a amostra Manteiga.

Variantes do algoritmo BP	Erro do conjunto de teste			Percentagem de vectores classificados correctamente					
				Teste			Aprendizagem		
	Mínimo	Máximo	Média	Mín.	Max.	Méd.	Mín.	Max.	Méd.
Quasi-Newton BFGS	0.3100	0.7400	0.4555	52.15	81.60	71.23	53.61	84.02	71.72
Powell-Beale GC	0.2952	0.5556	0.3649	56.44	81.90	77.22	63.13	86.33	79.67
Fletcher-Powell GC	0.2864	0.6640	0.4026	44.48	81.90	74.24	44.85	84.18	76.28
Polak-Ribiere GC	0.3135	0.8232	0.3768	47.24	81.29	76.10	45.97	86.02	79.17
Gradiente (incremental)	0.4067	1.7766	0.8517	3.07	74.54	45.86	4.15	74.96	45.08
Gradiente com coeficiente aprendizagem variável	0.3021	0.8868	0.3768	37.42	81.60	75.51	39.17	83.41	77.24
Gradiente com momentum	0.4170	2.1534	0.8541	3.99	75.15	49.70	4.76	74.35	49.24
Gradiente com momentum e coeficiente aprendizagem variável	0.2966	1.0506	0.3970	23.62	81.60	73.74	25.19	85.87	76.25
BPLM	0.3163	0.6023	0.4090	54.60	80.37	73.71	61.90	94.93	83.09
BPRPROP	0.2948	0.4050	0.3300	70.25	83.13	79.04	76.04	87.71	83.27
BPGCS	0.2919	0.6635	0.3444	47.85	82.82	77.92	52.23	90.48	82.02

A Tabela 9 revela que o algoritmo BPRPROP teve as melhores medidas de desempenho e, simultaneamente, foi o algoritmo mais robusto. As simulações posteriores utilizaram sempre este algoritmo. A aprendizagem com regularização e paragem antecipada da aprendizagem diminui a capacidade previsional do algoritmo e não o torna mais robusto, o mesmo acontecendo com a aprendizagem com regularização, a qual exibe resultados piores, apresentando um erro médio maior para os diversos valores, e não melhora a robustez do algoritmo. Seguem-se os resultados das simulações com as variantes do algoritmo BP treinadas com a função erro entrópica, indicadas na Tabela 10.

Tabela 10 – Melhor variante do algoritmo BP com função erro entrópica para a amostra Manteiga.

Variantes do algoritmo BP	Erro do conjunto de teste			Percentagem de vectores classificados correctamente					
				Teste			Aprendizagem		
	Mínimo	Máximo	Média	Mín.	Max.	Méd.	Mín.	Max.	Méd.
BPGCS	0.3206	0.4816	0.3863	71.47	80.06	76.44	88.23	92.94	90.70
Quasi-Newton BFGS	0.3317	0.4831	0.3813	69.94	80.79	76.53	83.62	88.95	86.18
Gradiente (incremental)	0.3106	0.4656	0.3638	70.86	79.14	76.63	84.95	89.76	86.89

Os resultados obtidos com a função erro entrópica revelam que alguns algoritmos aplicados com esta função têm desempenhos superiores do que quando são aplicados com a função erro média da soma do erro quadrado. Em particular, o algoritmo BP evidencia um aumento razoável das medidas de desempenho. De notar que o mesmo algoritmo mostrou ser superior às variantes de segunda ordem.

As simulações com o modelo LM identificaram as seguintes variáveis independentes relevantes: QtAdUA; Venc; Gasto; Inv; Preço (de cada alternativa); Feat (de cada alternativa); De (de cada alternativa); InvA (de cada alternativa); Fid (alternativas 3,4 e 5). O modelo Mixed Logit foi estimado utilizando uma distribuição triangular para o coeficiente da variável Feat. Os resultados das simulações dos modelos LM e ML estão na Tabela 11.

Tabela 11 – Medidas de desempenho para o LM e ML para a amostra Manteiga.

Erro do conjunto de teste			Percentagem de vectores classificados correctamente					
			Teste			Aprendizagem		
Mínimo	Máximo	Média	Mínimo	Máximo	Média	Mínimo	Máximo	Média
0.3457	0.3737	0.3616	65.34	70.86	67.08	66.70	69.12	67.53
0.3111	0.3720	0.3393	65.95	69.00	67.42	66.32	69.44	68.28

Uma vez mais, o modelo Mixed Logit (segunda linha) obteve uma diminuição pouco significativa para o erro

médio, diminuindo também a percentagem média de vectores classificados correctamente.

4. CONCLUSÕES

Numa perspectiva objectiva, baseada nos resultados e nas medidas de desempenho definidas, não é possível afirmar que as RNA têm uma capacidade previsional superior aos MED ou o oposto. No entanto, se se considerar a medida de desempenho média da percentagem de vectores classificados correctamente, as RNA foram sempre superiores aos MED, sendo de referir que essa diferença é muito significativa. Contudo, esta medida é pouco adequada para avaliar probabilidades. Conclui-se também que os modelos de escolha discreta são mais robustos do que as RNA.

Nas fontes de informação o modelo Logit Multinomial foi sempre citado como algo limitado e com inúmeras restrições, em clara oposição à flexibilidade e às capacidades do modelo Mixed Logit. Mas esta investigação não chegou às mesmas conclusões. O modelo Logit Multinomial é muito mais rápido do que o modelo Mixed Logit, exigindo, por conseguinte, menos recursos computacionais e a sua especificação também é mais simples. O modelo Mixed Logit obteve resultados sempre superiores com a medida de desempenho média da soma do erro quadrado, mostrando uma maior capacidade previsional no comportamento do consumidor. No entanto, essa superioridade na medida de desempenho é marginal e, por essa razão, nos problemas analisados não se justifica o uso de um modelo com especificação mais complexa e mais exigente em recursos computacionais, em detrimento de um modelo mais simples e computacionalmente eficiente.

A aprendizagem incremental mostrou ser inferior à aprendizagem *batch* quando aplicada com a função erro média da soma do erro quadrado. A excepção reside quando se aplica a função erro entrópica, em que a aprendizagem incremental pode ser superior à aprendizagem *batch*. Ressalve-se, todavia, que o melhor resultado das variantes com aprendizagem incremental não supera o melhor resultado das variantes com aprendizagem *batch*.

A aplicação da função erro entrópica, citada frequentemente nas fontes de informação como a mais adequada para problemas de classificação, não demonstrou ser sempre superior à função erro média da soma do erro quadrado. Também neste enquadramento a melhor variante dos algoritmos com função erro média da soma do erro quadrado foi superior à melhor variante dos algoritmos com função erro entrópica. No entanto, a função erro entrópica demonstrou uma vantagem significativa na avaliação da robustez, pois a sua aplicação torna sempre os algoritmos mais robustos.

Na aplicação das várias técnicas, nomeadamente a regularização, a paragem antecipada da aprendizagem ou a aplicação simultânea das duas, nenhuma se mostrou superior às outras em todos os testes realizados.

Não foi possível estabelecer uma hierarquia das variantes do algoritmo BP baseada na medida de desempenho média da soma do erro quadrado. Nas diversas amostras o comportamento das variantes não manteve um padrão constante, mas é possível indicar um conjunto de algoritmos que tiveram sempre boas medidas de desempenho. Assim, em consequência dos resultados obtidos, os algoritmos BPRPROP, BPGCS e BPLM devem constituir sempre uma primeira escolha. O trabalho desenvolvido na parte prática, onde são apresentadas as medidas de desempenho das simulações, mostra, inequivocamente, que estas variantes são superiores aos algoritmos mais usados em estudos anteriores.

BIBLIOGRAFIA

Agrawal, Deepak e Schorling, Christopher - Market share forecasting: An empirical comparison of artificial neural networks and multinomial logit model. Journal of Retailing. 72:4, p. 383-407.

Ben-Akiva, Moshe e Lerman, Steven R. - Discrete choice analysis : theory and application to travel demand. Cambridge, Mass.: MIT Press, 1985. xx, 390 p. ISBN 0262022176.

Ben-Akiva, Moshe, Bolduc, Denis e Walker, Joan - Specification, Identification, & Estimation of the Logit Kernel (or Continuous Mixed Logit) Model. Massachusetts Institute of Technology, 2001. 61 p.

Bentz, Yves e Merunka, Dwight - Neural networks and the multinomial logit for brand choice modellnig: A hybrid approach. Journal of Forecasting. 19:3, p. 117-200.

Bhat, Chandra - Random Utility-Based Discrete Choice Models for Travel Demand Analysis. In Goulias, K. - *Transportation Systems Planning: Methods and Applications*: CRC Press, 2003. p. 1-30.

Bishop, Christopher M. - Exact Calculation of the Hessian Matrix for the Multi-layer Perceptron. Neural Computation. 4:4, p. 494-501.

Bishop, Christopher M. - Neural networks for pattern recognition. Oxford New York: Clarendon Press; Oxford University Press, 1995. xvii, 482 p. ISBN 0198538642.

Demuth, Howard B. e Beale, Mark H. - *Neural Network Toolbox*: The MathWorks, Inc., 1997.

Fausett, Laurene V. - Fundamentals of neural networks: architectures, algorithms, and applications. Englewood Cliffs, NJ: Prentice Hall, 1994. xvi, 461 p. ISBN 0133341860.

Fish, Kelly E. e Blodgett, Jeffrey G. - A visual method for determining variable importance in an artificial neural network: an empirical benchmark study. Journal of Targeting , Measurement and Analysis for Marketing. 11:3, p. 244-254.

Fish, Kelly E., Johnson, John D., Dorsey, Robert E. e Blodgett, Jeffrey G. - Using an Artificial Neural Network Trained with a Genetic Algorithm to Model Brand Share. Journal of Business Research. 57:1, p. 79-85.

Hagan, Martin T. e Menhaj, Mohammad B. - Training feedforward networks with the Marquardt algorithm. IEEE Transactions on Neural Networks. 5:6, p. 989-993.

Hagan, Martin T., Demuth, Howard B. e Beale, Mark H. - Neural network design. Boston: PWS Pub., 1996. 1 v. (various pagings) p. ISBN 0534943322 (hard cover).

Hensher, David A. e Greene, William H. - The Mixed Logit Model: The State of Practice and Warnings for the Unwary. Institute of Transport Studies, Faculty of Economics and Business, University of Sydney, 2001. 39 p.

Heskes, Tom e Wiegerinck, Wim - A theoretical comparison of batch-mode, on-line, cyclic, and almost cyclic learning. IEEE Transactions on Neural Networks. 7, p. 919-925.

Hruschka, Harald - An Artificial Neural Net Attraction Model (ANNAM) to analyze market share effects of marketing instruments. Vienna University of Economics and Business Administration, 2000. 14 p.

Hruschka, Harald, Fettes, Werner e Probst, Markus - An Empirical Comparison of the Validity of aNeural Net Based Multinomial Logit ChoiceModel to Alternative Model Specifications. University of Regensburg, Germany, 2001. 33 p.

Jervis, T. T. e Fitzgerald, W. J. - Optimization Schemes for Neural Networks. 1993. 33 p.

Joost, Merten e Schiffmann, Wolfram - Speeding up backpropagation algorithms by using cross-entropy combined with pattern normalization. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems based Systems (IJUFKS), p. 10.

LeCun, Yann - Efficient BackProp. New York: AT&T Laboratories, Holmdel, NJ, USA, 1996. 75 p.

LeCun, Yann, Bottou, Leon, Orr, Genevieve B. e Müller, Klaus-Robert - Efficient BackProp - *Neural Networks: tricks of the trade*: Springer, 1998. p. 44.

Louviere, Jordan J., Hensher, David A., Swait, Joffre D. e Adamowicz, Wiktor - Stated Choice Methods: Analysis and Application: Cambridge University Press, 2000. 402 p. ISBN 0521788307.

McFadden, Daniel e Train, Kenneth - Mixed MNL Models for Discrete Response. Department of Economics, University of California, 1995. 23 p.

Möller, Martin - A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning. Aarhus: Computer Science Department, University of Aarhus, Denmark, 1990. 21 p.

Moreira, M. e Fiesler, E. - Neural Networks with Adaptive Learning Rate and Momentum Terms. VALAIS - SUISSE: Institut dalle Molle D' Intelligence Artificielle Perceptive, 1995. 30 p.

Oh, Sang-Hoon e Lee, Youngjik - A Modified Error Function to Improve the Error Back-Propagation Algorithm for Multi-Layer Perceptrons. ETRI Journal. 17:1, (April), p. 11-22.

Papatla, Purushottam, Zahedi, Mariam (Fatemeh) e Zekic-Susac, Marijana - Leveraging the strengths of choice models and neural networks: A multiproduct comparative analysis. Decision Sciences. 33:3, p. 433-468.

Pires, Paulo - Avaliação da performance das redes neuronais artificiais e dos modelos de escolha discreta na aquisição de produtos com envolvimento fraco. XV Jornadas Hispano-Lusas de Gestión Científica. Sevilha, Espanha. ISBN ISBN 84-96378-10-1, (2005), p. 135-150.

Reed, Russell D. e Marks, Robert J. - Neural smithing: supervised learning in feedforward artificial neural networks. Cambridge, Mass.: The MIT Press, 1999. viii, 346 p. ISBN 0262181908 (hc alk. paper).

Riedmiller, Martin - Rprop - Description and Implementation Details. Karlsruhe: Institut für Logik, Komplexität und Deduktionssyteme, University of Karlsruhe, 1994. 2 p.

Riedmiller, Martin e Braun, Heinrich - A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm. Proceedings of the IEEE International Conference on Neural Networks. San Francisco, CA, (1993), p. 586--591.

Sarle, Warren S. - *Neural Network FAQ*, 1997. Disponível em <URL:ftp://ftp.sas.com/pub/neural/FAQ.html>.

Torresen, Jim - The convergence of backpropagation trained neural networks for various weight update frequencies. International Journal of Neural Systems. 8:3, p. 263-277.

Train, Kenneth - Qualitative Choice Analysis: Theory, Econometrics, and an Application to Automobile Demand. Cambridge, Mass.: MIT Press, 1986. 252 p.

Train, Kenneth - Halton Sequences for Mixed Logit. Department of Economics, Institute for Business and Economic Research, UC Berkeley, 1999. 18 p.

Train, Kenneth - Discrete Choice Methods with Simulation: Cambridge University Press, 2003. 342 p. ISBN 0521017157.

Vroomen, B., Franses, P. H. e Nierop, E. van - Modeling consideration sets and brand choice using artificial neural networks. European Journal of Operational Research. 154:1, p. 206-217.

Walker, Joan - The Mixed Logit (or Logit Kernel) Model: Dispelling Misconceptions of Identification. Massachusets: Caliper Corporation, 2002. 24 p.

West, Patricia M., Brockett, Patrick L. e Golden, Linda L. - A comparative analysis of neural networks and statistical methods for predicting consumer choice. Marketing Science. 16:4, p. 370-391.

Yip, Yewmun, Kurubarahalli, Gururaj e Su, Yuli - Influence of data structure in choice modeling: An empirical investigation using neural networks. American Business Review. 19:1, p. 67-75.