

P

PREVISÃO DE RETORNOS DE AÇÕES DOS SETORES FINANCEIRO, DE ALIMENTOS, INDUSTRIAL E DE SERVIÇOS, POR MEIO DE RNA E MODELOS ARIMA-GARCH

STOCK RETURNS FORECASTING FOR FINANCIAL, FOOD, INDUSTRIAL
AND SERVICES COMPANIES USING NEURAL NETWORKS AND ARIMA-
GARCH MODELS

MAURI APARECIDO DE OLIVEIRA

*Doutor em Administração de Empresas pela Faculdade de Economia e Administração
de Empresas da Universidade de São Paulo (FEA-USP-SP).*

Rua Oliveira Melo, 232, Ipiranga – São Paulo – SP – CEP 04271-000

E-mail: mauriao@usp.br

ALESSANDRA DE ÁVILA MONTINI

*Doutora em Administração de Empresas pela Faculdade de Economia e Administração
de Empresas da Universidade de São Paulo (FEA-USP-SP).*

*Professora da Faculdade de Economia, Administração e Contabilidade
da Universidade de São Paulo (FEA-USP-SP).*

*Avenida Professor Luciano Gualberto, 908, sala G-107,
Cidade Universitária – São Paulo – SP – CEP 05508-900*

E-mail: amontini@usp.br

DANIEL REED BERGMANN

*Mestre em Contabilidade pela Faculdade de Economia
e Administração de Empresas da Universidade de São Paulo (FEA-USP-SP).*

*Professor da Universidade Presbiteriana Mackenzie (UPM).
Rua da Consolação, 930, Consolação – São Paulo – SP – CEP 01302-907*

E-mail: danielrb@mackenzie.br

RESUMO

O objetivo deste trabalho é realizar previsões de séries de retornos de ações de empresas dos setores financeiro, de alimentos, industrial e de serviços, utilizando redes neurais artificiais (RNA) do tipo *feedforward* treinadas com algoritmo de Levenberg-Marquardt e modelos Arima-Garch. Selecionaram-se duas séries de cada setor, e os dados foram obtidos da economática. Para o setor financeiro, são analisadas as séries dos bancos Bradesco e Itaú, no setor de alimentos a Perdigão e a Sadia, no setor industrial a Marcopolo e a Gerdau, e no setor de serviços o Pão de Açúcar e Lojas Americanas. Verificou-se que as previsões realizadas pelas duas técnicas têm desempenhos parecidos, não revelando superioridade de nenhuma técnica.

PALAVRAS-CHAVE

Séries temporais; Previsão; Algoritmo de Levenberg-Marquardt; Redes neurais; Arima-Garch.

ABSTRACT

The main purpose of this work is realize stock returns forecasting for financial, food, industrial and services companies using feedforward neural networks trained with Levenberg-Marquardt algorithm and Arima-Garch models. In each area two time series was selected from Economática. To the financial area, Bradesco and Itaú was analyzed, Perdigão and Sadia in the food sector, Marcopolo and Gerdau in the industrial area, finally Pão de Açúcar and Lojas Americanas in the services. The forecasting generated by the two techniques had similar performance implying no significant differences between them.

KEYWORDS

Time series; Forecasting; Levenberg-Marquardt algorithm; Neural networks; Arima-Garch.

1 INTRODUÇÃO

Neste trabalho, vamos analisar duas técnicas utilizadas para previsão de séries temporais: os modelos Arima-Garch e as redes neurais alimentadas adiante (*feedforward*). O modelo Arch generalizado, conhecido como Garch-Generalized Arch, foi inicialmente proposto por Bollerslev (1986). Esse modelo é um dos mais utilizados para o modelamento de volatilidade. As redes neurais constituem uma ferramenta flexível amplamente utilizada para a análise de séries temporais, sua aplicação tem sido feita nas mais variadas situações que envolvem problemas do mundo real de diversas áreas, notadamente finanças e economia. As redes neurais artificiais fornecem uma grande variedade de modelos matemáticos não-lineares, úteis para resolver diferentes problemas em que são empregadas convencionalmente técnicas estatísticas. A metodologia empregada neste trabalho, no caso dos processos Garch, é amplamente conhecida (HAMILTON, 1994; MORETTIN; TOLOI, 2004; ENDERS, 2004). Para as redes neurais, serão utilizadas especificamente redes neurais artificiais do tipo alimentada adiante ou *feedforward* (FINE, 1999). Do ponto de vista teórico, o processamento de sinais não-lineares (LAPEDÉS; FABER 1987), a incorporação do tempo na rede neural (ELMAN, 1988) e o modelamento não-linear para previsão de séries temporais caóticas (CASDAGLI, 1989) têm sido aplicados como ferramenta na tomada de decisão em finanças (HAWLEY et al., 1990; REFENES, 1993), análise de mercado (FISHMAN et al., 1991), modelamento não-linear e previsão (CASDAGLI; EUBANK., 1992; AZZOF, 1993; CLEMENTS; HENDRY, 1999). Mais recentemente tem havido a preocupação de comparar e relacionar a tecnologia de redes neurais com a abordagem estatística tradicional (CHENG; TITTERINGTON, 1994; RIPLEY, 1993, 1994, 1996; MEDEIROS et al., 2006), sob a perspectiva econométrica (KUAN; WHITE, 1994), de engenharia financeira (ABU-MOSTAFA et al., 2001) e macroeconômica (TERÄSVIRTA et al., 2005).

2 REDES NEURAIS ARTIFICIAIS (RNA) ALIMENTADAS ADIANTE DE MÚLTIPLAS CAMADAS

Uma rede neural artificial (RNA) alimentada adiante tem todos os sinais indo em uma direção, dos neurônios de entrada para os neurônios de saída. As redes neurais alimentadas adiante de múltiplas camadas também são chamadas de *multilayer feedforward neural networks*. A RNA *feedforward* de três camadas pode ser esquematizada como mostrado na Figura 1. Basicamente, a RNA é constituída de linhas de ligações para as quais são atribuídos pesos e vértices que representam as conexões.

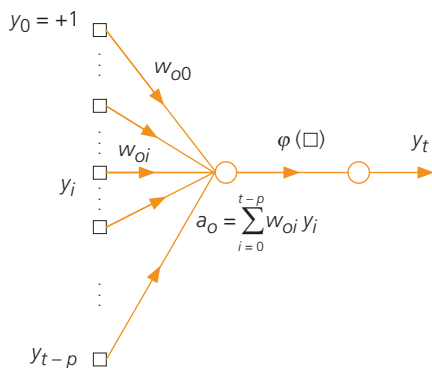
Na primeira camada, representada por quadrados, temos os valores de entrada para a rede, também chamados de nós de entrada. O modelo mostrado na Figura 1 inclui também um bias, y_0 , aplicado externamente, representado por y_0 . Esse bias tem o efeito de aumentar ou diminuir a entrada líquida da função de ativação, dependendo se ele é positivo ou negativo, respectivamente. As linhas de conexão indicam para qual neurônio o valor precedente está se direcionando. Cada uma dessas linhas contém um peso. Os pesos dessas linhas são parâmetros ajustados conforme apresentam-se novos vetores de dados para a RNA. Uma RNA em que um neurônio em qualquer camada está conectado a todos os nós/neurônios da camada anterior é denominada totalmente conectada. O fluxo de sinal através da rede progride para a frente, da esquerda para a direita e de camada em camada.

A segunda camada e todas as subseqüentes contêm neurônios de processamento na forma de círculo. Quaisquer camadas entre as camadas de entrada e de saída são chamadas de intermediárias. Uma apresentação completa de um vetor de dados que será processado constitui uma época. No final da RNA, a camada de saída fornece o estado final da rede. O processamento realizado em um neurônio artificial pode ser dividido em três etapas. Na primeira etapa, os dados passam ao longo das linhas de entrada de um neurônio e são multiplicados pelos pesos correspondentes. A seguir, todos os dados, após a multiplicação pelos respectivos pesos, são somados dentro do neurônio.

Na etapa final o valor somado é passado através de uma função de ativação, cuja saída representa o valor de saída do neurônio.

FIGURA 1

MODELO DE UMA REDE NEURAL



Uma RNA é um ferramenta poderosa e flexível para realizar previsões de séries temporais, quando uma seleção adequada de amostras de entradas e saídas, número de neurônios na camada intermediária, função de ativação e recursos computacionais estão disponíveis.

A RNA também possui a vantagem de aproximar funções não-lineares e ser capaz de resolver problemas em que o relacionamento de entrada e saída não é bem definido. Uma RNA do tipo *feedforward* é especialmente adequada para realizar previsão de séries que apresentam volatilidade, por causa do tratamento da não-linearidade realizada pela utilização de funções sigmóides na camada de entradas.

A função de ativação, $\varphi(\cdot)$, define a saída de um neurônio. A função de ativação é a regra para mapeamento das entradas somadas, a , do neurônio até sua saída e , por uma escolha adequada, isso significa a introdução de uma capacidade de processar a não-linearidade na rede.

Na prática, essas funções são escolhidas de tal forma para que possam ser monotônicas e saturar nos extremos $[0,1]$ ou $[-1,1]$. A função sigmóide é a forma mais comum de função de ativação utilizada na construção de uma RNA:

$$\varphi(y) = \frac{1}{1 + e^{-gy}} \quad (1)$$

onde: g é o ganho, ou parâmetro de inclinação da função sigmóide.

2.1 NORMALIZAÇÃO DOS VALORES DE ENTRADA DA RNA

Os dados de entrada da RNA, na maioria dos casos, necessitarão de normalização. Esse processo é aconselhável, uma vez que coloca o intervalo de dados

dentro de limites que facilitam a tarefa da rede para realizar a minimização do erro de saída. De acordo com Azzof (1994, p. 25), existem quatro abordagens para normalização. No caso em questão, vamos utilizar o processo que considera um vetor de entrada de dimensão n , ou seja, os elementos no vetor variam de $i = 1, \dots, n$. A amostra total da série será normalizada ao longo da extensão $[0,1]$.

Neste trabalho, serão consideradas as normalizações da primeira diferença do logaritmo das séries dadas por:

$$NDLSERIE_t = \frac{\Delta(\log(SERIE_t)) - \min(\Delta(\log(SERIE_t)))}{\max(\Delta(\log(SERIE_t))) - \min(\Delta(\log(SERIE_t)))} \quad (2)$$

$$NDLSERIE_t = \frac{DLSERIE_t - \min(DLSERIE_t)}{\max(DLSERIE_t) - \min(DLSERIE_t)}$$

em que:

Δ representa a primeira diferença, \log indica o logaritmo, \min é o menor valor da série e \max o maior valor. Inicialmente, é construído o logaritmo da série, L SERIE, depois disso se constrói a série da primeira diferença, D LSERIE, e, por fim, a normalização.

Tanto as previsões realizadas pelos modelos Arima-Garch quanto pelas RNA serão realizadas para séries normalizadas de acordo com (2).

2.2 ALGORITMO DE LEVENBERG-MARQUARDT

O algoritmo de *backpropagation* (RUMELHART et. al, 1986a e 1986b; RUMELHART e McCLELLAND, 1986; WERBOS, 1988) é um dos mais importantes no desenvolvimento das redes neurais, no entanto sua taxa de convergência é muito pobre. Em razão disso, foram propostos outros algoritmos como alternativa ao *backpropagation* (ANDERSEN; WILAMOWSKI, 1995; BATTITI, 1992; CHARALAMBOUS, 1992; HAGAN; MENHAJ, 1994; SHAH; PALMIERI, 1990), e o algoritmo de Levenberg-Marquardt tem sido amplamente utilizado e aceito como algoritmo muito eficiente para solucionar o problema de minimização de quadrados para funções não-lineares.

Pode ser mostrado que as iterações do método de Gauss-Newton e de descida do gradiente são complementares nas vantagens que eles fornecem. Levenberg (1944) propôs um algoritmo baseado nessa observação, cuja regra de atualização é uma mistura dos algoritmos anteriormente mencionados e dada por

$$w_{t+1} = w_t - [H + \lambda I]^{-1} \nabla f(w_t) \quad (3)$$

onde H é a matriz hessiana calculada em w_t .

Se o erro decresce após uma atualização, isso implica que a suposição de forma quadrática para $f(w)$ está valendo, e reduzimos λ para reduzir a influência da descida do gradiente. No entanto, se o erro aumenta, poderemos seguir o gradiente, e então λ é aumentado pelo mesmo fator. O algoritmo proposto por Levenberg é então composto basicamente por quatro passos. No passo um, é realizada uma atualização como descrita pela Equação 3. A seguir, avalia-se o erro no novo vetor de parâmetros. No terceiro passo, é verificado se o erro diminuiu como resultado da atualização, e então se deve retornar ao passo (ou seja, retornar os pesos para os seus valores anteriores) e aumentar λ por um determinado fator, usualmente por um fator de 10, ou por algum fator significativo. Então, voltar para o passo 1 e tentar atualizar novamente. No último passo, é avaliado se o erro diminuiu como resultado da atualização, então se deve aceitar o passo (ou seja, manter os pesos em seus novos valores) e diminuir λ por um determinado fator e assim por diante. Esse algoritmo tem a desvantagem de que, se o valor de λ é grande, a matriz hessiana calculada não é utilizada no todo. Pode-se obter alguma vantagem da segunda derivada em tais casos pelo escalonamento de cada componente do gradiente de acordo com a curvatura. Isso pode resultar em grandes movimentos ao longo das direções, onde o gradiente é menor de tal forma que o problema dos vales não ocorra mais. A principal sugestão foi fornecida por Marquardt (1963). Ele substituiu a matriz identidade em (3) pela diagonal da matriz Hessiana, resultando na regra de atualização de Levenberg- Marquardt.

$$w_{t+1} = w_t - (H + \lambda \text{diag}[H])^{-1} \nabla f(w_t) \quad (4)$$

Problemas de minimização utilizando RNA freqüentemente são mal condicionados, o que torna o problema de minimização difícil de resolver. Para tais problemas, o algoritmo de Levenberg- Marquardt (LM) é normalmente a melhor escolha. Por essa razão, o método de otimização que utiliza o algoritmo LM será usado no treinamento das RNA para previsão das séries temporais deste trabalho.

Modificações do algoritmo LM têm sido propostas para melhorar o desempenho de aprendizagem de redes neurais de múltiplas camadas e reduzir a quantidade de oscilação no processo de aprendizagem (SURATGAR et al., 2005; WILAMOWSKI et al., 2001).

3 PROCESSO GARCH

O modelo de volatilidade de séries temporais mais importante para estimar variância condicional é o modelo de processos de heterocedasticidade condicional auto-regressiva generalizada (*generalized autoregressive conditional heteroskedasticity* – Garch) (BOLLERSLEV, 1986; BOLLERSLEV et al., 1992; ENGLE, 1982; MEISSNER; KAWANO, 2001). Esse modelo é um dos mais utilizados para a modelagem da volatilidade de séries temporais, ou seja, ele é utilizado quando a volatilidade da série não é constante (GONZALES; BURGERS, 1997). Essa técnica é capaz de capturar a influência de muitos fatos sobre a série, notadamente heterocedasticidade, aglomerados de volatilidade e excesso de curtose.

Uma série temporal segue um modelo AR-Garch quando puder ser escrita na forma:

$$y_t = \phi_0 + \phi_1 y_{t-1} + v_t,$$

$$v_t = \sqrt{\sigma_t^2} \eta_t, \quad (6)$$

onde: $\sigma_\eta^2 = 1$ e $\sigma_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + \beta_1 \sigma_{t-1}^2$.

Temos ainda que $\{\eta_t\}$ é um processo ruído-branco, dessa forma as médias condicional e incondicional de v_t são iguais a zero (POSEDEL, 2005).

Embora muitas variantes dos processos Garch tenham sido sugeridas, Ding et al. (1993) demonstraram com simulações de Monte de Carlo que o modelo original Garch proposto por Bollerslev (1986) é capaz de produzir o padrão de autocorrelação que aparece em dados financeiros. Normalmente, os processos Garch servem como *benchmark* para as avaliações de séries temporais, com presença de volatilidade condicional (QI; ZHANG, 2001). O AR(1)-Garch(1,1) é o modelo mais comumente utilizado.

$$y_t = \phi_0 + \phi_1 y_{t-1} + v_t \sqrt{\alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + \beta_1 \sigma_{t-1}^2} \quad (7)$$

Existem muitos trabalhos publicados sobre o desempenho de previsão dos modelos Garch (CHAKRABORTY, 1992; BARUCCI; RENO, 2002; ANDERS, 2006), e tradicionalmente algumas medidas são mais utilizadas para comparar o ajuste e a precisão da previsão de modelos e técnicas alternativas: o RMSE (raiz do erro quadrado médio), MAE (erro absoluto médio), Mape (erro percentual absoluto médio), Corr (coeficiente de correlação de Pearson), Sign (proporção de vezes em que o sinal é corretamente previsto) e o TIC (coeficiente de desigual-

dade de Theil).

Neste trabalho utilizaremos, como medidas de desempenho, o RMSE e o TIC dados pelas expressões (8) e (9), respectivamente.

$$TIC = \frac{\sqrt{\sum_{t=T+1}^{T+h} \frac{(\hat{y}_t - y_t)^2}{h}}}{\sqrt{\sum_{t=T+1}^{T+h} \frac{\hat{y}_t^2}{h}} + \sqrt{\sum_{t=T+1}^{T+h} \frac{y_t^2}{h}}} \quad (8)$$

O TIC varia entre zero e um, e zero indica ajuste perfeito.

$$RMSE = \sqrt{\sum_{t=T+1}^{T+h} \frac{(\hat{y}_t - y_t)^2}{h}} \quad (9)$$

Quanto menor for o RMSE, menor será o erro obtido pelo modelo utilizado para realizar a previsão.

4 RESULTADOS OBTIDOS

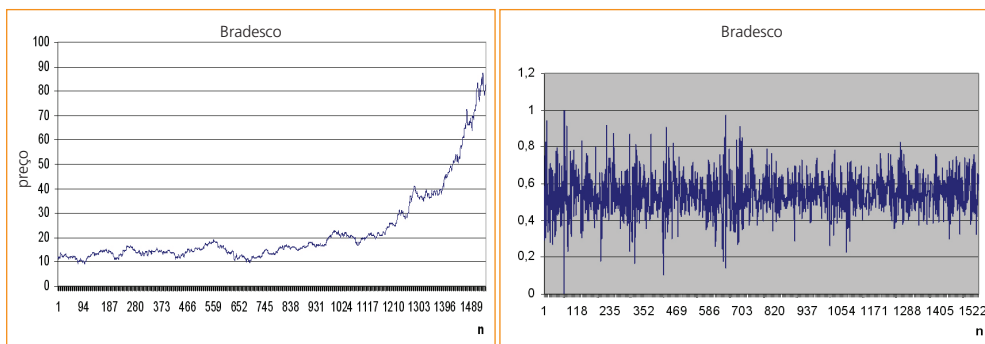
Neste trabalho, foram analisadas duas séries de cada setor. Para o setor financeiro analisaram-se as séries do Bradesco PN e do Itaú PN, no setor de alimentos a Perdigão PN e a Sadia PN, no setor industrial a Marcopolo PN e a Gerdau PN, e no setor de serviços o Pão de Açúcar PN e as Lojas Americanas PN. As séries foram obtidas da economática. Todas essas séries possuem 1.542 valores que iniciam em 3.1.2000. Essas séries foram divididas em duas partes, uma para estimação dos modelos Arima-Garch e treinamento da RNA, e a segunda parte para realizar as previsões e construir as medidas de desempenho. A primeira parte das séries contém 1.300 valores e a segunda 242. Para cada uma das séries, foram ajustadas topologias com dez neurônios na camada de entrada, um neurônio na camada de saída e neurônios na camada intermediária, variando de um a nove. Para cada uma das séries, foi estimado o melhor modelo, e depois disso realizaram-se as previsões. Para escolher o melhor modelo Arima-Garch, foram utilizados os critérios de informação de Akaike (AIC) e bayesiano (BIC). Para escolher a melhor topologia de RNA, utilizaram-se o RMSE e o TIC.

4.1 SETOR FINANCEIRO – BRADESCO E ITAÚ

A Figura 2a apresenta a série original do preço de fechamento do Bradesco, e a Figura 2b mostra a série normalizada da primeira diferença do logaritmo do preço NDLBRAD.

FIGURA 2

**(A) SÉRIE NO NÍVEL DO PREÇO DO BRADESCO,
 (B) SÉRIE TRANSFORMADA NDLBRAD**



(a)

(b)

No caso do Bradesco, o melhor modelo ajustado para a série é um AR(1)-Garch(1,1), dado pela expressão (7). Os parâmetros do modelo são apresentados na Tabela 1.

TABELA 1

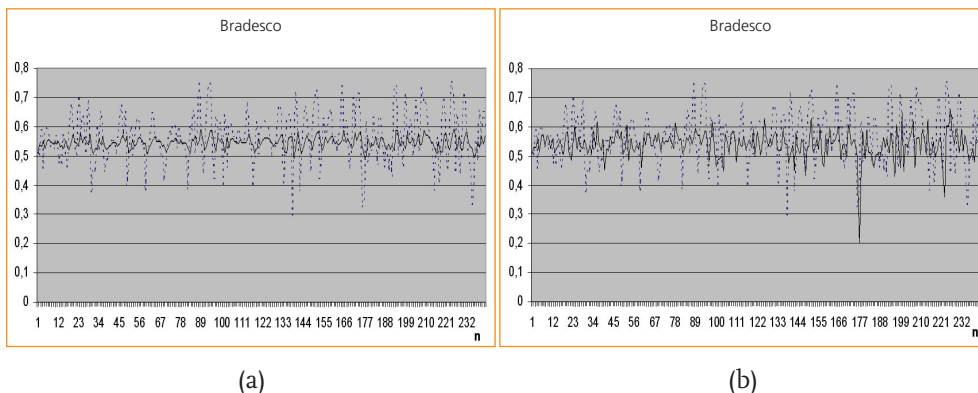
PARÂMETROS ESTIMADOS PARA A SÉRIE NDLBRAD

PARÂMETROS	COEFICIENTE	ERRO PADRÃO	ESTATÍSTICA Z	SIGNIFICÂNCIA
ϕ_0	0,543800	0,003395	160,1913	0,0000
ϕ_1	0,220962	0,028225	7,828568	0,0000
α_0	0,000490	0,000134	3,672624	0,0002
α_1	0,091887	0,015491	5,931627	0,0000
β_0	0,863512	0,023188	37,23988	0,0000

As previsões realizadas utilizando o modelo Arima-Garch e a RNA são mostradas na Figura 3. As linhas tracejadas indicam o valor da série NDLBRAD; e as linhas contínuas, as previsões.

FIGURA 3

**(A) PREVISÃO OBTIDA PELO MODELO ARIMA-GARCH,
 (B) PREVISÃO DA RNA PARA NDLBRAD**



As medidas de desempenho obtidas por TIC e RMSE são mostradas na Tabela 2. Nota-se que a rede neural apresentou o melhor desempenho.

TABELA 2

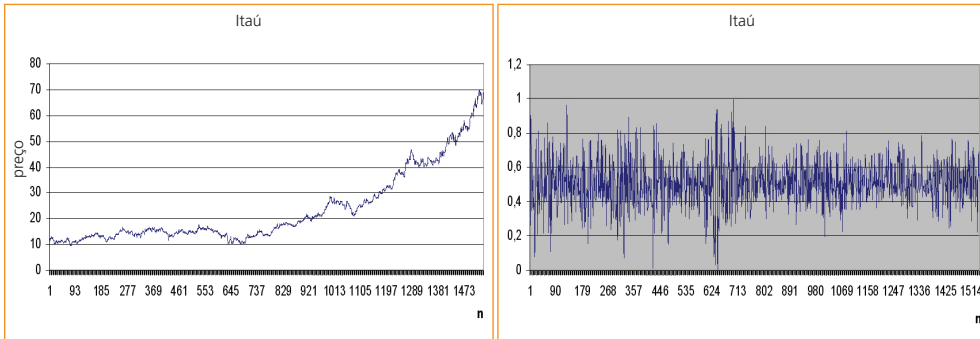
ESTATÍSTICAS DE AJUSTE OBTIDAS PARA A SÉRIE NDLBRAD

	RMSE	TIC
AR(1)-Garch(1,1)	0,090183	0,081444
RNA 10-1-1	0,088427	0,080153

A Figura 4 mostra a série de preço do Itaú e a série normalizada da primeira diferença do logaritmo do preço (NDLITAU). No caso do Itaú, também foi ajustado um modelo AR(1)-Garch(1,1).

FIGURA 4

**(A) SÉRIE NO NÍVEL DO PREÇO DO ITAÚ,
 (B) SÉRIE TRANSFORMADA NDLITAU**



(a)

(b)

Na Tabela 3, apresentamos a estimativa dos parâmetros do modelo AR(1)-Garch(1,1).

TABELA 3

PARÂMETROS ESTIMADOS PARA A SÉRIE NDLITAU

PARÂMETROS	COEFICIENTE	ERRO PADRÃO	ESTATÍSTICA Z	SIGNIFICÂNCIA
ϕ_0	0,515000	0,003781	136,1979	0,0000
ϕ_1	0,158516	0,027529	5,758036	0,0000
α_0	0,000449	0,000144	3,107148	0,0019
α_1	0,075684	0,013720	5,516254	0,0000
β_1	0,894804	0,017243	51,89240	0,0000

As previsões realizadas utilizando o modelo AR-Garch e a RNA 10-2-1 são mostradas na Figura 5. As linhas tracejadas indicam o valor da série NDLITAU; e as linhas contínuas, as previsões. Na Tabela 4, temos as estatísticas TIC e RMSE.

Nota-se que a rede neural apresentou o melhor desempenho.

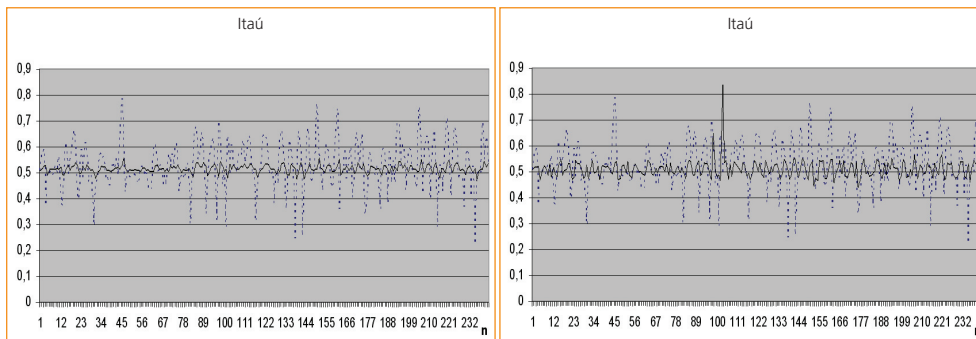
TABELA 4

ESTATÍSTICAS DE AJUSTE OBTIDAS PARA A SÉRIE NDLITAU

	RMSE	TIC
AR(1)-GARCH(1,1)	0,094509	0,090604
RNA 10-2-1	0,092871	0,089322

FIGURA 5

(A) PREVISÃO OBTIDA PELO MODELO ARIMA-GARCH,
(B) PREVISÃO DA RNA PARA NDLITAU



(a)

(b)

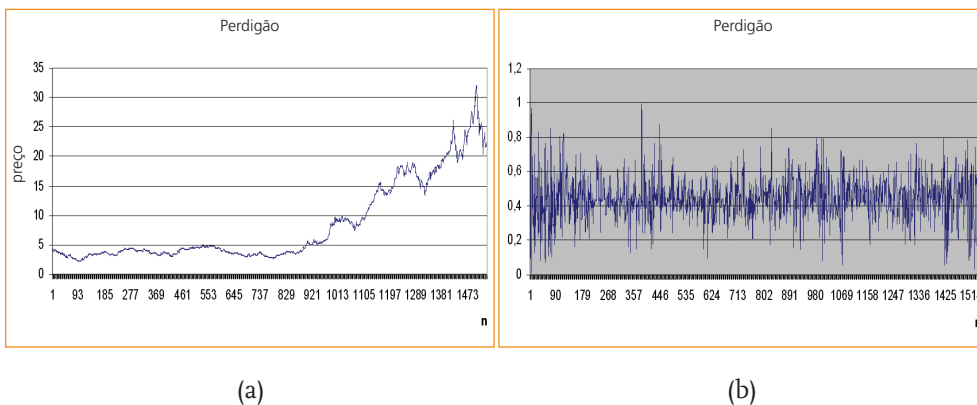
Para as duas séries escolhidas do setor financeiro, o desempenho de previsão das redes neurais foi melhor quando comparado com os modelos autorregressivos de heterocedasticidade condicional.

4.2 SETOR DE ALIMENTOS – PERDIGÃO E SADIA

Na Figura 6, apresentam-se a série de preços da Perdigão e a série normalizada da primeira diferença do logaritmo do preço (NDLPERD).

FIGURA 6

**(A) SÉRIE NO NÍVEL DO PREÇO DA PERDIGÃO,
 (B) SÉRIE TRANSFORMADA NDLPERD**



A Tabela 5 mostra os parâmetros estimados para o modelo AR(1)-Garch(1,1).

TABELA 5

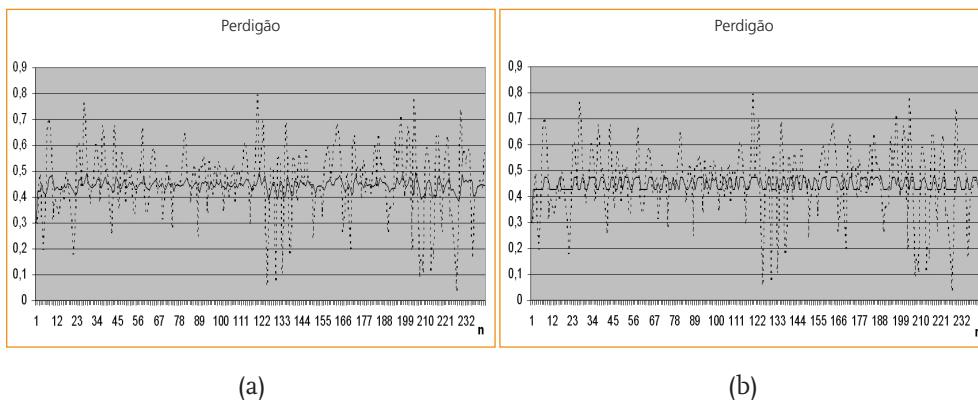
PARÂMETROS ESTIMADOS PARA A SÉRIE NDLPERD

PARÂMETROS	COEFICIENTE	ERRO PADRÃO	ESTATÍSTICA Z	SIGNIFICÂNCIA
ϕ_0	0,445162	0,003311	134,4327	0,0000
ϕ_1	0,147928	0,026878	5,503584	0,0000
α_0	0,000728	0,000163	4,460329	0,0000
α_1	0,072308	0,010961	6,597043	0,0000
β_1	0,865022	0,020639	41,91157	0,0000

As previsões realizadas utilizando o modelo Arima-Garch e a RNA 10-1-1 são mostradas na Figura 7. As linhas tracejadas indicam o valor da série NDLPERD; e as linhas contínuas, as previsões.

FIGURA 7

**(A) PREVISÃO OBTIDA PELO MODELO ARIMA-GARCH,
 (B) PREVISÃO DA RNA PARA NDLPERD**



Na Tabela 6, temos as estatísticas TIC e RMSE obtidas pelo modelo $AR(1)$ Garch(1,1) e pela RNA 10-1-1. Notamos que o modelo $AR(1)$ -Garch(1,1) apresentou o melhor desempenho.

TABELA 6

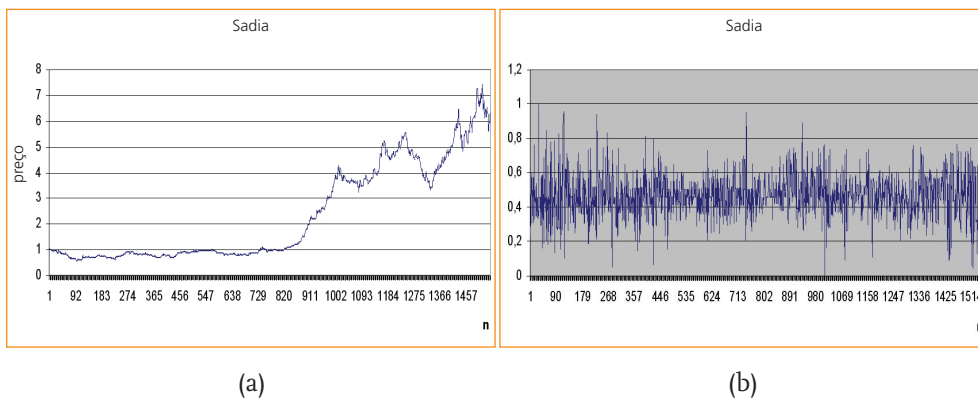
ESTATÍSTICAS DE AJUSTE OBTIDAS PARA A SÉRIE NDLPERD

	RMSE	TIC
AR(1)-Garch(1,1)	0,137140	0,150141
RNA 10-1-1	0,140295	0,152999

Na Figura 8, apresentam-se a série no nível da Sadia e a série normalizada da primeira diferença do logaritmo do preço (NDLSADI).

FIGURA 8

**(A) SÉRIE NO NÍVEL DO PREÇO DA SADIA,
 (B) SÉRIE TRANSFORMADA NDLSADI**



No caso da Sadia, o modelo selecionado também foi o AR(1)-Garch(1,1). Os parâmetros estimados são mostrados na Tabela 7.

TABELA 7

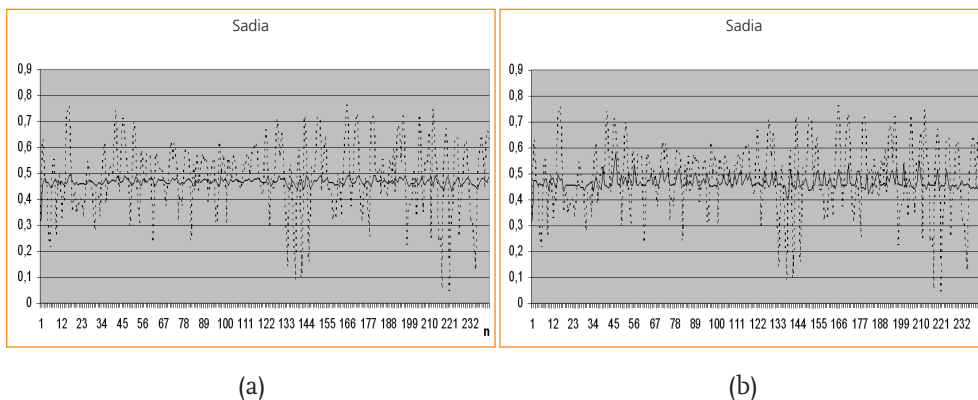
PARÂMETROS ESTIMADOS PARA A SÉRIE NDLSADI

PARÂMETROS	COEFICIENTE	ERRO PADRÃO	ESTATÍSTICA Z	SIGNIFICÂNCIA
ϕ_0	0,468054	0,002993	156,3786	0,0000
ϕ_1	0,087591	0,026559	3,297963	0,0010
α_0	0,000546	0,000150	3,649578	0,0003
α_1	0,102279	0,015240	6,711293	0,0000
β_1	0,860133	0,021157	40,65519	0,0000

As previsões realizadas utilizando o modelo Arima-Garch e a RNA 10-2-1 são mostradas na Figura 9. As linhas tracejadas indicam o valor da série NDLSADI; e as linhas contínuas, as previsões.

FIGURA 9

**(A) PREVISÃO OBTIDA PELO MODELO ARIMA-GARCH,
 (B) PREVISÃO DA RNA PARA NDLSADI**



As estatísticas TIC e RMSE obtidas pelo modelo AR(1)-Garch(1,1) e pela RNA 10-2-1 são mostradas na Tabela 8. Nota-se que as medidas de desempenho dos dois modelos são próximas, mas com valores menores para a RNA.

TABELA 8

ESTATÍSTICAS DE AJUSTE OBTIDAS PARA A SÉRIE NDLSADI

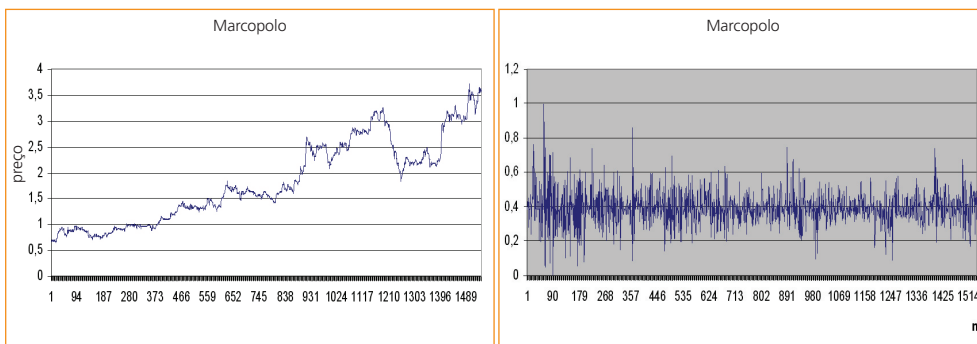
	RMSE	TIC
AR(1)-Garch(1,1)	0,137650	0,143399
RNA 10-2-1	0,136794	0,142475

4.3 SETOR INDUSTRIAL – MARCOPOLO E GERDAU

Na Figura 10, apresentam-se a série de preços da Marcopolo e a série normalizada da primeira diferença do logaritmo do preço (NDLMARC).

FIGURA 10

**(A) SÉRIE NO NÍVEL DO PREÇO DA MARCOPOLO,
 (B) SÉRIE TRANSFORMADA NDLMARC**



(a)

(b)

O melhor modelo ajustado para a Marcopolo foi o AR(1)-Garch(2,2). Os parâmetros estimados são mostrados na Tabela 9.

TABELA 9

PARÂMETROS ESTIMADOS PARA A SÉRIE NDLMARC

PARÂMETROS	COEFICIENTE	ERRO PADRÃO	ESTATÍSTICA Z	SIGNIFICÂNCIA
ϕ_0	0,391193	0,002024	193,2875	0,0000
ϕ_1	-0,064709	0,032222	-2,008227	0,0446
α_0	8,04E-05	2,60E-05	3,096130	0,0020
α_1	0,265585	0,031714	8,374344	0,0000
α_2	-0,235462	0,031861	-7,390223	0,0000
β_1	1,246308	0,076712	16,24667	0,0000
β_2	-0,284268	0,069959	-4,063321	0,0000

As previsões realizadas utilizando o modelo Arima-Garch e a RNA 10-5-1 são mostradas na Figura 11. As linhas tracejadas indicam o valor da série NDLMARC; e as linhas contínuas, as previsões. Na Tabela 10, são apresentadas as estatísticas TIC e RMSE. Notamos que a rede neural apresentou melhor desempenho.

FIGURA II

**(A) PREVISÃO OBTIDA PELO MODELO ARIMA-GARCH,
 (B) PREVISÃO DA RNA PARA NDLMARC**

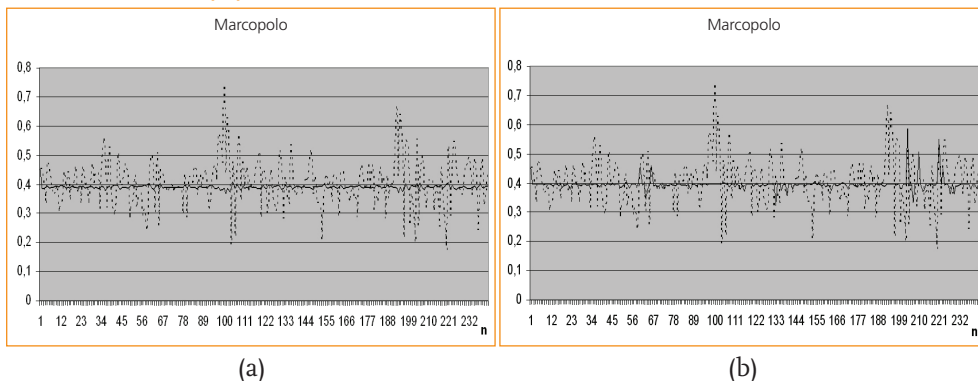


TABELA IO

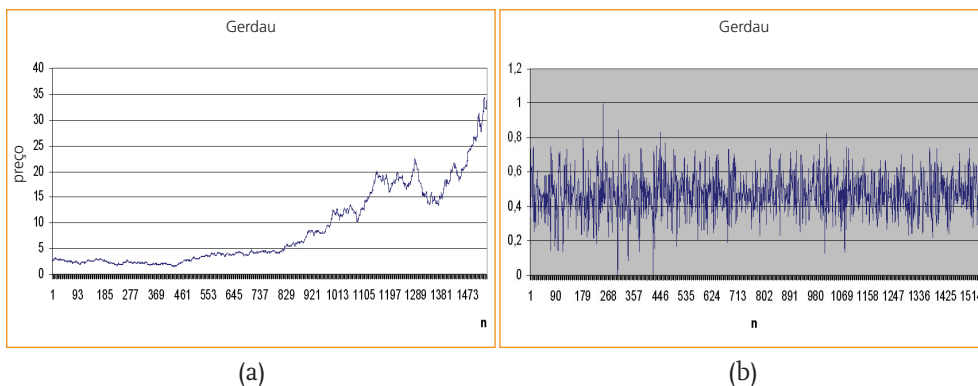
ESTATÍSTICAS DE AJUSTE OBTIDAS PARA A SÉRIE NDLMARC

	RMSE	TIC
AR(1)-Garch(2,2)	0,083067	0,104228
RNA 10-5-1	0,082152	0,102387

Na Figura 12, apresentam-se a série de preços da Gerdau e a série normalizada da primeira diferença do logaritmo do preço (NDLGERD). Para a série transformada, foi ajustado um modelo AR(1)-Garch(2,1).

FIGURA I2

**(A) SÉRIE NO NÍVEL DO PREÇO DA GERDAU,
 (B) SÉRIE TRANSFORMADA NDLGERD**



Na Tabela II, apresentamos a estimativa dos parâmetros do modelo AR(1)-Garch(2,1).

TABELA II

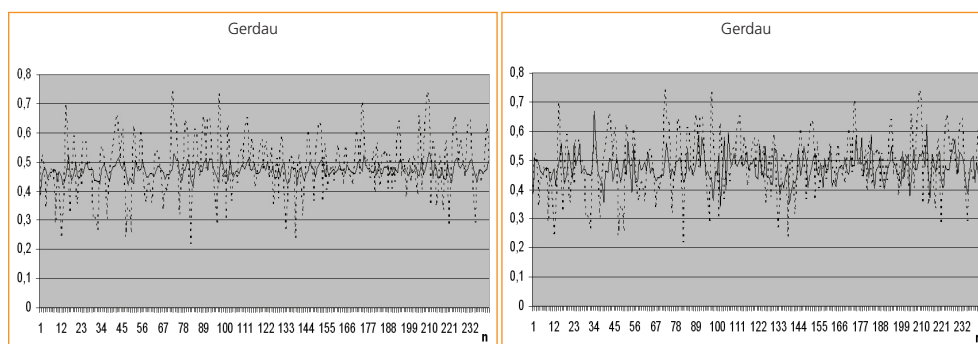
PARÂMETROS ESTIMADOS PARA A SÉRIE NDLGERD

PARÂMETROS	COEFICIENTE	ERRO PADRÃO	ESTATÍSTICA Z	SIGNIFICÂNCIA
ϕ_0	0,471030	0,003927	119,9384	0,0000
ϕ_1	0,222096	0,027457	8,088847	0,0000
α_0	0,021785	0,000829	26,28612	0,0000
α_1	0,076448	0,029315	2,607856	0,0091
α_2	0,074794	0,029117	2,568751	0,0102
β_1	-0,984060	0,009324	-105,5355	0,0000

As previsões obtidas pelo modelo AR(1)-Garch(2,1) e pela RNA 10-4-1 estão na Figura 13.

FIGURA 13

**(A) PREVISÃO OBTIDA PELO MODELO ARIMA-GARCH,
 (B) PREVISÃO DA RNA PARA NDLGERD**



(a)

(b)

Na Tabela 12, são apresentadas as estatísticas RMSE e TIC. Verificamos que as previsões obtidas pelo modelo AR(1)-Garch(2,1) são melhores do que da RNA 10-4-1.

TABELA 12

ESTATÍSTICAS DE AJUSTE OBTIDAS PARA A SÉRIE NDLGERD

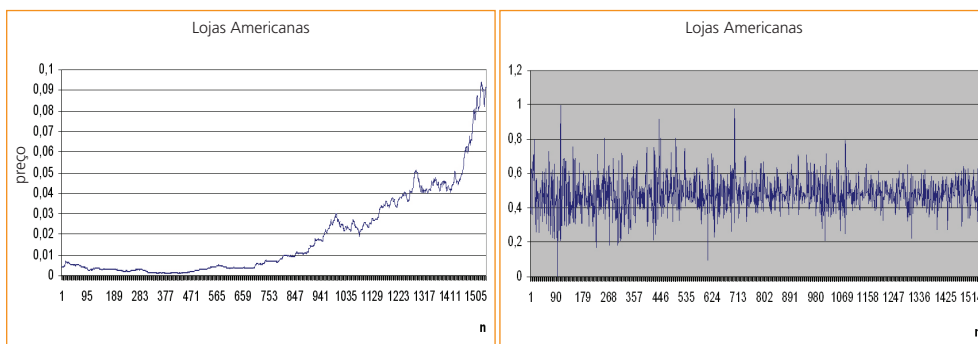
	RMSE	TIC
AR(1)-Garch(2,1)	0,101360	0,105466
RNA 10-4-1	0,102436	0,106732

4.4 SETOR DE SERVIÇOS – LOJAS AMERICANAS E PÃO DE AÇÚCAR

A Figura 14 mostra a série de preços das Lojas Americanas e a série normalizada da primeira diferença do logaritmo do preço (NDLLOJA). Para a série transformada, foi ajustado um modelo AR(1)-Garch(1,1).

FIGURA 14

**(A) SÉRIE NO NÍVEL DO PREÇO DAS LOJAS AMERICANAS,
 (B) SÉRIE TRANSFORMADA NDLLOJA**



(a)

(b)

Os parâmetros do modelo AR(1)-Garch(1,1) para as Lojas Americanas são apresentados na Tabela 13.

TABELA 13

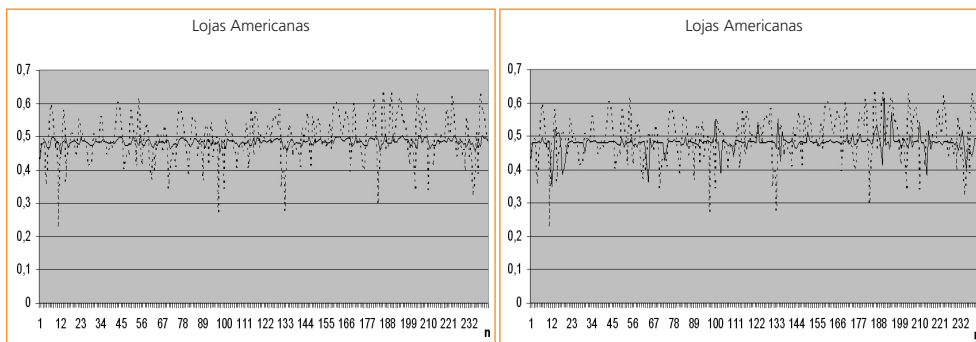
PARÂMETROS ESTIMADOS PARA A SÉRIE NDLLOJA

PARÂMETROS	COEFICIENTE	ERRO PADRÃO	ESTATÍSTICA Z	SIGNIFICÂNCIA
ϕ_0	0,483810	0,002892	167,2855	0,0000
ϕ_1	0,143069	0,029393	4,867435	0,0000
α_0	0,000535	0,000113	4,738079	0,0000
α_1	0,152176	0,017889	8,506704	0,0000
β_1	0,795944	0,023461	33,92677	0,0000

As previsões realizadas utilizando o modelo AR(1)-Garch(1,1) e a RNA 10-3-1 são mostradas na Figuras 15. As linhas tracejadas indicam o valor da série NDLLOJA; e as linhas contínuas, as previsões.

FIGURA 15

**(A) PREVISÃO OBTIDA PELO MODELO ARIMA-GARCH,
 (B) PREVISÃO DA RNA PARA NDLLOJA**



(a)

(b)

As estatísticas de desempenho apresentadas na Tabela 14 indicam que o modelo AR(1)-Garch(1,1) realizou melhores previsões.

TABELA 14

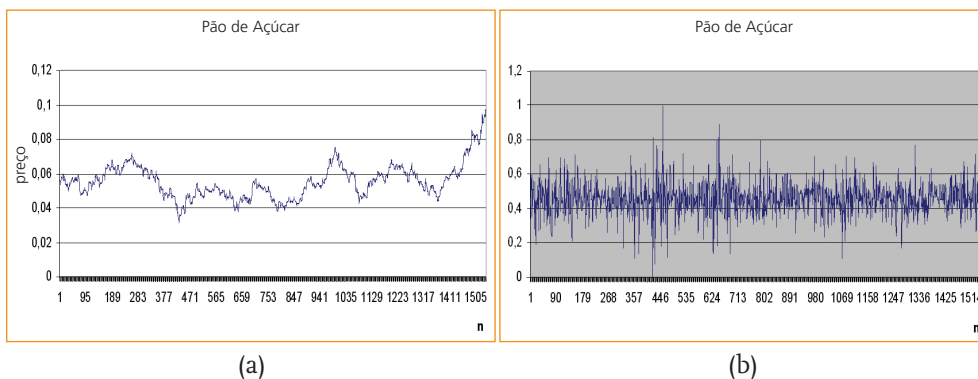
ESTATÍSTICAS DE AJUSTE OBTIDAS PARA A SÉRIE NDLLLOJA

	RMSE	TIC
AR(1)-Garch(1,1)	0,069215	0,070940
RNA 10-3-1	0,070732	0,072219

A última empresa analisada é o Pão de Açúcar. A Figura 16 apresenta a série original e a série normalizada da primeira diferença do logaritmo do preço (NDLPAOD).

FIGURA 16

**(A) SÉRIE NO NÍVEL DO PREÇO DO PÃO DE AÇÚCAR,
 (B) SÉRIE TRANSFORMADA NDLPAOD**



No caso do Pão de Açúcar foi ajustado o modelo AR(1)-Garch(1,1), e os coeficientes são apresentados na Tabela 15.

TABELA 15

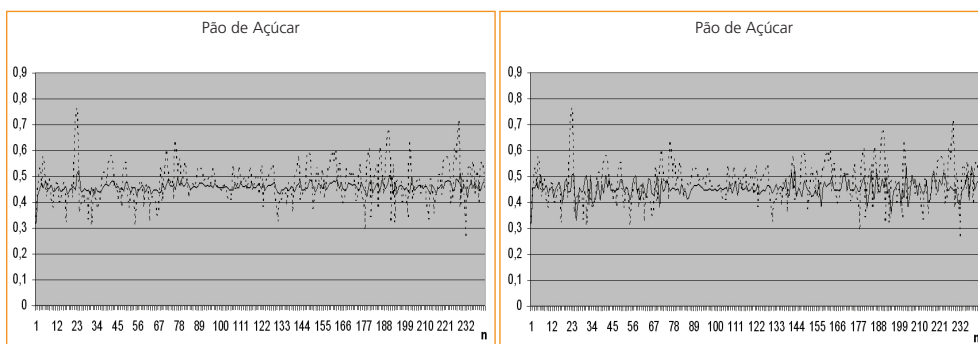
PARÂMETROS ESTIMADOS PARA A SÉRIE NDLPAOD

PARÂMETROS	COEFICIENTE	ERRO PADRÃO	ESTATÍSTICA Z	SIGNIFICÂNCIA
ϕ_0	0,457502	0,002965	154,3246	0,0000
ϕ_1	0,207943	0,026693	7,790197	0,0000
α_0	0,000243	7,63E-05	3,180343	0,0015
α_1	0,046951	0,008607	5,454829	0,0000
β_1	0,923116	0,015717	58,73518	0,0000

As previsões realizadas utilizando o modelo Arima-Garch são mostradas na Figura 17. As linhas tracejadas indicam o valor da série NDLPAD; e as linhas contínuas, as previsões.

FIGURA 17

**(A) PREVISÃO OBTIDA PELO MODELO ARIMA-GARCH,
 (B) PREVISÃO DA RNA PARA NDLPAD**



(a)

(b)

A Tabela 16 mostra o melhor desempenho de previsão do modelo AR(1)-Garch(1,1).

TABELA 16

ESTATÍSTICAS DE AJUSTE OBTIDAS PARA A SÉRIE NDLPAD

	RMSE	TIC
AR(1)-Garch(1,1)	0,077001	0,082483
RNA 10-1-1	0,077475	0,083264

5 CONCLUSÕES

Neste trabalho, nosso interesse concentrou-se em implementar modelos auto-regressivos integrados e de médias móveis com heterocedasticidade condicional auto-regressiva generalizada (Arima-Garch) e redes neurais artificiais alimentadas adiante (RNA *feedforward*), para realizar previsões de um passo à frente de séries univariadas dos setores financeiro, de alimentos, industrial e de serviços. Para cada um dos setores, foram escolhidas, sem nenhum critério

específico, duas séries. No setor financeiro, analisaram-se as séries do Bradesco e do Itaú, no setor de alimentos a Perdigão e a Sadia, no setor industrial a Marcopolo e a Gerdau, e no setor de serviços as Lojas Americanas e o Pão de Açúcar. Todas as séries continham tamanho 1.542, e os 1.300 primeiros foram utilizados para estimativa dos modelos Arima-Garch e treinamento das RNA. Utilizou-se o restante dos valores para medir a capacidade de previsão dos modelos. Para cada uma das séries, escolheu-se o melhor modelo Arima-Garch e a melhor RNA, e foram testadas redes com dez entradas, contento entre um e nove neurônios na camada intermediária, e um neurônio na saída. Para medir o desempenho de cada uma dessas duas técnicas, utilizaram-se duas medidas: a raiz do erro quadrático médio (RMSE) e o coeficiente de desigualdade de Theil (TIC).

Dos resultados obtidos nos experimentos, constatou-se que as previsões dos modelos Arima-Garch foram melhores nos casos das empresas Perdigão, Gerdau, Lojas Americanas e Pão de Açúcar. As RNA realizaram melhores previsões nos casos das empresas Bradesco, Itaú, Sadia e Marcopolo. Quanto aos critérios de desempenho adotados, as maiores diferenças de previsões aconteceram nos casos da Perdigão e das Lojas Americanas, em favor dos modelos Arima-Garch, que foram em torno de 2% melhores que as RNA. As previsões das duas técnicas utilizadas foram mais próximas no caso da Sadia em que a RNA foi melhor 0,6% do que os modelos Arima-Garch. O que se observa é que as duas técnicas implementadas podem ser utilizadas como ferramentas complementares e de comparação para aumentar a capacidade de decisão do analista.

REFERÊNCIAS

- ABU-MOSTAFA, Y. S. et al. Introduction to the special issue on neural networks in financial engineering. *IEEE Transactions on Neural Networks*, v. 12, p. 653-655, 2001.
- ANDERS, W. Garch forecasting performance under different distribution assumptions. *Journal of Forecasting*, 2006.
- ANDERSEN, T. J.; WILAMOWSKI, B. M. A modified regression algorithm for fast one layer neural network training. *World Congress of Neural Networks*, Washington, v. 1, p. 687-690, 1995.
- AZZOF, E. M. Reducing error in neural network time series forecasting. *Neural Computing and Applications*, v. 1, p. 240-247, 1993.
- _____. *Neural network time series forecasting of financial markets*. Chichester: John Wiley & Sons, 1994.
- BARUCCI, E.; RENO, R. On measuring volatility and the Garch forecasting performance. *Journal of International Financial Markets, Institutions and Money*, v. 12, 2002.
- BATTITI, R. First – and second – order methods for learning: between steepest descent and Newton's method. *Neural Computation*, v. 4, n. 2, p. 141-166, 1992.

- BOLLERSLEV, T. Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, v. 31, p. 307-327, 1986.
- BOLLERSLEV, T. et al. Modeling in finance: a review of the theory and empirical evidence. *Journal of Econometrics*, v. 52, p. 5-59, 1992.
- CASDAGLI, M. Nonlinear prediction of chaotic time series. *Physica D*, v. 35, p. 335-356, 1989.
- CASDAGLI, M.; EUBANK, S. *Nonlinear modeling and forecasting*. Santa Fe Institute. *Proceedings...* v. XII, Addison-Wesley, 1992.
- CHAKRABORTY, K. Forecasting the behaviour of multivariate time series using neural networks. *Neural Networks*, v. 5, p. 961-970, 1992.
- CHARALAMBOUS, C., Conjugate gradient algorithm for efficient training of artificial neural networks. *IEEE Proceedings*, v. 139, n. 3, p. 301-310, 1992.
- CHENG, B.; TITTERINGTON, D. M. Neural networks: a preview from statistical perspective. *Statistical Science*, 1994.
- CLEMENTS, M. P.; HENDRY, D. F. Forecasting non-stationary economic time series. Cambridge: MIT Press, 1999.
- DING, Z.; GRANGER, C. W. J.; ENGLE, R. F. A long memory property of stock market returns and a new model. *Journal of Empirical Finance*, v. 1, p. 83-106, 1993.
- ELMAN, J. L. Finding structure in time (CRL Technical Report 8801). La Jolla: University of California, San Diego, Center for Research in Language, 1988.
- ENDERS, W. *Applied econometric time series*. Hoboken: Wiley, 2004.
- ENGLE, R. F. Autoregressive conditional heteroskedasticity with estimates of the variance of U. K. inflation. *Econometrica* v. 50, p. 987-1008, 1982.
- FINE, T. L. *Feedforward neural network methodology*. New York: Springer, 1999.
- FISHMAN, M. B.; BARR, D. S.; LOICK, W. J. Using neural nets in market analysis. *Technical Analysis of Stocks and Commodities*, v. 9, n. 4, 1991.
- GONZALES, M. F.; BURGESS, N. Modeling market volatilities: the neural network perspective. *European Journal of Finance*, v. 3, p. 137-157, 1997.
- HAGAN, M. T.; MENHAJ, M. Training feedforward networks with the Marquardt algorithm. *IEEE Transactions on Neural Networks*, v. 5, n. 6, p. 989-993, 1994.
- HAMILTON, J. D. *Time series analysis*. Princeton: Princeton University Press, 1994.
- HAWLEY, D. D.; JOHNSON, J. D.; RAINA, D. Artificial neural systems: a new tool for financial decision-making. *Finan Analy J.*, 1990.
- KUAN, C. M.; WHITE, H. Artificial neural networks: an econometric perspective. *Econometric Reviews*, v. 13, p. 1-91, 1994.
- LAPEDES, A.; FARBER, R. Nonlinear signal processing using neural networks, prediction and system modelling. Los Alamos Report LA-UR-87-2662, Los Alamos National Laboratory, 1987.
- LEVENBERG, K. A method for the solution of certain problems in least squares. *Applied Mathematics*, v. 2, p. 164-168, 1944.
- MARQUARDT, D. An algorithm for least-squares estimation of nonlinear parameters. *Journal of Applied Mathematics*, v. 11, p. 431-441, 1963.
- MEDEIROS M.; TERÄSVIRTA, T.; RECH, G. Building neural network models for time series: a statistical approach. *Journal of Forecasting*, v. 25, p. 49-75, 2006.

- MEISSNER, G.; KAWANO, N. Capturing the volatility smile of options on high-tech stocks-a combined Garch-Neural Network approach. *Journal of Economics and Finance*, v. 25, n. 3, p. 276-293, 2001.
- MORETTIN, P. A.; TOLOI, C. M. C. *Análise de séries temporais*. São Paulo: Edgard Blucher, 2004.
- POSEDEL, P. Properties and Estimation of Garch(1,1) model. *Metodoloski zvezki*, v. 2, n. 2, p. 243-257, 2005.
- QI, M.; ZHANG, G. P. An investigation of model selection criteria for neural network time series forecasting. *European Journal of Operational Research*, v. 132, n. 3, p. 666-680, 2001.
- REFENES, A. N. Currency exchange rate prediction and neural network design strategies. *Neural Computing and Applications*, v. 1, p. 46-58, 1993.
- RIPLEY, B. D. *Statistical aspects of neural networks*. Londres: Chapman e Hall, 1993.
- _____. Neural networks and related methods of classification. *Journal of the Royal Statistical Society, Series B*, v. 56, n. 3, p. 409-456, 1994.
- _____. *Pattern recognition and neural networks*. Cambridge: Cambridge Press, 1996.
- RUMELHART, D. E.; MCCLELLAND, J. L. *Parallel distributed processing: explorations in the micro-structure of cognition*. Cambridge: MIT Press, 1986.
- RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning internal representations by error propagation. In: _____. *Parallel distributed processing*. Cambridge: MIT Press, 1986a. v. 1, p. 318-362.
- _____. Learning representations of back-propagation errors, *Nature*, v. 323, p. 533-536. 1986b.
- SHAH, S.; PALMIERI, F. MEKA – A fast, local algorithm for training feedforward neural networks. In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS, 1990, San Diego. *Proceedings...* San Diego, 1990. v. 3, p. 41-46.
- SURATGAR, A. A.; TAVAKOLI, M. B.; HOSEINABADI, A. Modified Levenberg-Marquardt method for neural. *Networks Training, Transactions on Engineering, Computing and Technology*, 2005.
- TERÄSVIRTA, T.; VAN DIJK, D.; MEDEIROS M. Linear models, smooth transition autoregressions, and neural networks for forecasting macroeconomic time series: a re-examination. *International Journal of Forecasting*, 2005.
- WERBOS, P. J. Back-propagation: past and future. In: INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, 1, 1988, San Diego. *Proceeding...* San Diego, 1988.
- WILAMOWSKI, B. M. et al. An algorithm for fast convergence in training neural networks. *IEEE*, 2001.

TRAMITAÇÃO

Recebido em 24/4/2007

Aprovado em 12/6/2007