

# **Previsão de retornos de ações de empresas dos setores financeiro e de alimentos por meio de redes neurais e modelos ARIMA-GARCH**

## **AUTORES**

**MAURI APARECIDO DE OLIVEIRA**

Universidade de São Paulo

mauriao@usp.br

**DANIEL REED BERGMANN**

Universidade Presbiteriana Mackenzie

drberg@uol.com.br

**LUIZ PAULO LOPES FÁVERO**

Universidade de São Paulo

lpfaver@usp.br

## **Resumo**

O objetivo deste trabalho é realizar previsões de séries de retornos de ações de empresas dos setores financeiro e de alimentos utilizando redes neurais artificiais (RNA) do tipo feedforward treinadas com algoritmo de Levenberg-Marquardt e modelos ARIMA-GARCH. Foram selecionadas duas séries de cada setor. Para o setor financeiro são analisadas as séries do Bradesco e do Itaú, no setor de alimentos as séries da Perdigão e da Sadia. Todas estas séries possuem 1542 valores que iniciam em 3/01/2000. Estas 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. 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 escolher o melhor modelo ARIMA-GARCH foram utilizados os critérios de informação de Akaike (AIC) e bayesiano (BIC). Para comparar as previsões realizadas são utilizadas duas medidas de desempenho: o coeficiente de desigualdade de Theil (TIC) e a raiz do erro quadrado médio (RMSE). Verificou-se que as previsões realizadas pelas redes neurais artificiais foram superiores às realizadas pelos modelos ARIMA-GARCH para três das quatro séries de retornos consideradas.

## **Abstract**

The main purpose of this work is realize stock returns forecasting for financial and food companies using feedforward neural networks trained with Levenberg-Marquardt algorithm and ARIMA-GARCH models. In each area two time series was selected, to the financial area, Bradesco and Itaú was analyzed, Perdigão and Sadia in the food sector. All the series presents 1542 values starting in 3/01/2000. The series were divided in two parts, the first is used to estimate ARIMA-GARCH models and to neural networks training, the second part is used to forecasting and to performance measures building. For each series were adjusted neural networks topologies with ten neurons in the first layer, one neuron in the exit layer and in the hidden layer the number of neurons between one and nine. To choose the better ARIMA-GARCH model we use the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC). To compare the forecasting two performance measures was used: Theil's inequality coefficient (TIC) and root mean square error (RMSE). The forecasting performed by artificial neural networks had better results in three of the considered returns series.

**Palavras-chave:** Redes Neurais Artificiais, Modelos ARIMA-GARCH, Retornos de Ações

## 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 primeiramente proposto por Bollerslev em 1986. Este 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 e 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 (LAPEDDES, 1987), incorporação do tempo na rede neural (ELMAN, 1988) e o modelamento não-linear para previsão de séries temporais caóticas (CASDAGLI,1989) tem levado a aplicação de redes neurais como ferramenta na tomada de decisão em finanças (HAWLEY,1990; REFENES,1997), análise de mercado (FISHMAN et al., 1991), modelamento não-linear e previsão (CASDAGLI e EUBANK.,1992; AZZOF, 1993; CLEMENTS e HENDRY, 1999). Mais recentemente tem havido a preocupação de comparar e relacionar a tecnologia de redes neurais com a abordagem estatística tradicional (CHENG e TITTERINGTON, 1994; RIPLEY, 1993, 1994, 1996; MEDEIROS et al., 2006), sob a perspectiva econométrica (KUAN e WHITE, 1994), de engenharia financeira (ABU-MOSTAFA et al., 2001) e macroeconômica (TERÄSVIRTA et al., 2005).

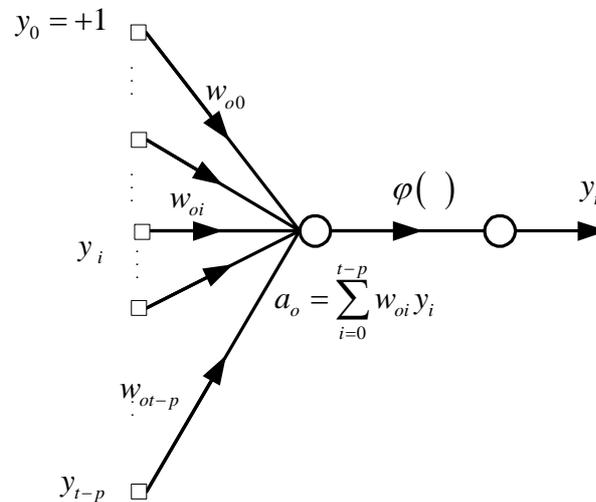
## 2. Redes Neurais Artificiais (RNA) Alimentadas Adiante de Múltiplas Camadas

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 como 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, estes pesos são parâmetros ajustados conforme são apresentados 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 de totalmente conectada. O fluxo de sinal através da rede progride para frente, da esquerda para a direita e de camada em camada.

A segunda camada, e todas as camadas subseqüentes, contêm neurônios de processamento, mostrados na forma de círculo. Quaisquer camadas entre as camadas de entrada e de saída são chamadas de camadas intermediárias. Uma apresentação completa de um vetor de dados que será processado constitui-se em 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 uma 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 onde 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, devido ao seu 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, isto significa a introdução de uma capacidade de processar não linearidade na rede. Na prática estas funções são escolhidas de tal forma a serem 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 necessitará de normalização, este 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 (Azoff, 1994, pág. 25) existem quatro abordagens para normalização. Neste caso 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:

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

$$NDLSERIE_t = \frac{DLSERIE_t - \min(DLSERIE_t)}{\max(DLSERIE_t) - \min(DLSERIE_t)} \quad (2)$$

em que:

$\Delta$  representa a primeira diferença,  $\log$  indica o logaritmo,  $\min$  é o menor valor da série e  $\max$  o maior valor. Primeiro é construído o logaritmo da série, LSERIE, depois disso é construído a série da primeira diferença, DLSERIE 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, 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 função disso foram propostos outros algoritmos como alternativa ao backpropagation (ANDERSEN e WILAMOWSKI, 1995; BATTITI, 1992; CHARALAMBOUS, 1992; HAGAN e MENHAJ, 1994; SHAH e PALMIERI, 1990), sendo que 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 nesta observação, cuja regra de atualização é uma mistura dos algoritmos mencionados acima e é dada por

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

sendo que  $H$  é a matriz Hessiana calculada em  $w_t$ . Se o erro decresce após uma atualização, isto implica que a suposição de forma quadrática para  $f(w)$  está valendo e reduzimos  $\lambda$  para reduzir a influência da descida do gradiente. Por outro lado, se o erro aumenta, poderemos seguir o gradiente e então  $\lambda$  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 deve-se retornar o passo (ou seja, retornar os pesos para os seus valores anteriores) e aumentar  $\lambda$  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 aceitar o passo (ou seja, manter os pesos em seus novos valores) e diminuir  $\lambda$  por um determinado fator e assim por diante. Esse algoritmo tem a desvantagem de que se o valor de  $\lambda$  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. Isto 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, isto torna o problema de minimização difícil de resolver e para tais problemas o algoritmo de Levenberg- Marquardt (LM) é normalmente a melhor escolha. Por esta razão, o método de otimização utilizando o algoritmo LM será utilizado no treinamento das RNA utilizadas 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 e KAWANO, 2001). Este 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 e BURGERS, 1997). Estes modelos são capazes de capturar heterocedasticidade, aglomerados de volatilidade e excesso de curtose.

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

$$y_t = \phi_0 + \phi_1 y_{t-1} + v_t, \quad (5)$$

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

sendo que  $\sigma_\eta^2 = 1$  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, Granger e Engle (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 e ZHANG, 2001). Sendo que 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 escritos sobre o desempenho de previsão dos modelos GARCH (CHAKRABORTY, 1992; BARUCCI e 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 desigualdade de Theil).

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

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

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

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

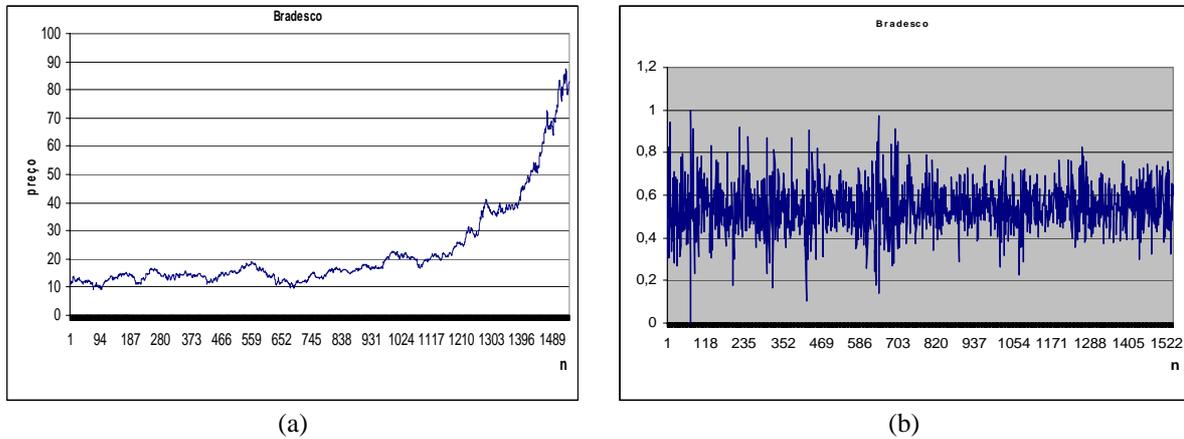
Quanto menor o RMSE significa menor o erro obtido pelo modelo utilizado para realizar a previsão.

#### 4. Resultados Obtidos

Neste trabalho foram analisadas duas séries de cada setor. Para o setor financeiro foram analisadas as séries do Bradesco PN e do Itaú PN, no setor de alimentos a Perdigão PN e a Sadia PN. As séries foram obtidas da Economática. Todas estas séries possuem 1542 valores que iniciam em 3/01/2000. Estas 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 contem 1300 valores e a segunda 242. Para cada uma das séries foram ajustadas topologias com 10 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 foram realizadas as previsões. Para escolher o melhor modelo ARIMA-GARCH foram utilizados os critérios de informação de Akaike (AIC) e bayesiano (BIC) e para escolher a melhor topologia de RNA foram utilizados 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, a Figura-2b apresenta a série normalizada da primeira diferença do logaritmo do preço NDLBRAD.



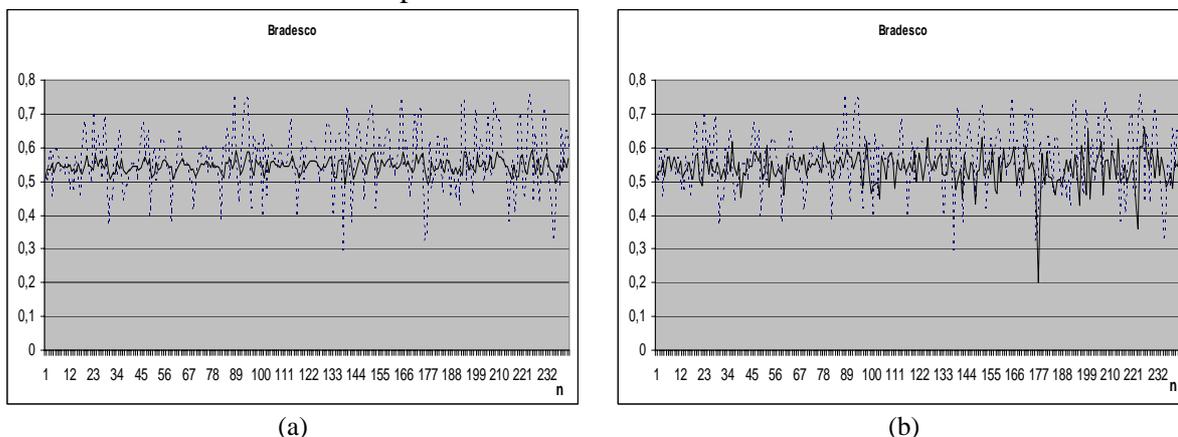
**Figura-2** (a) Série no nível do preço do Bradesco, (b) Série transformada NDLBRAD

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.

Parâmetros	Coefficiente	Erro-padrão	Estatística z	Significância
$\phi_0$	0,543800	0,003395	160,1913	0,0000
$\phi_1$	0,220962	0,028225	7,828568	0,0000
$\alpha_0$	0,000490	0,000134	3,672624	0,0002
$\alpha_1$	0,091887	0,015491	5,931627	0,0000
$\beta_1$	0,863512	0,023188	37,23988	0,0000

**Tabela-1** Parâmetros estimados para a série NDLBRAD

As previsões realizadas utilizando o modelo ARIMA-GARCH e a RNA são mostradas na Figura-3a e 3b, respectivamente. As linhas tracejadas indicam o valor da série NDLBRAD e as linhas contínuas indicam as previsões.



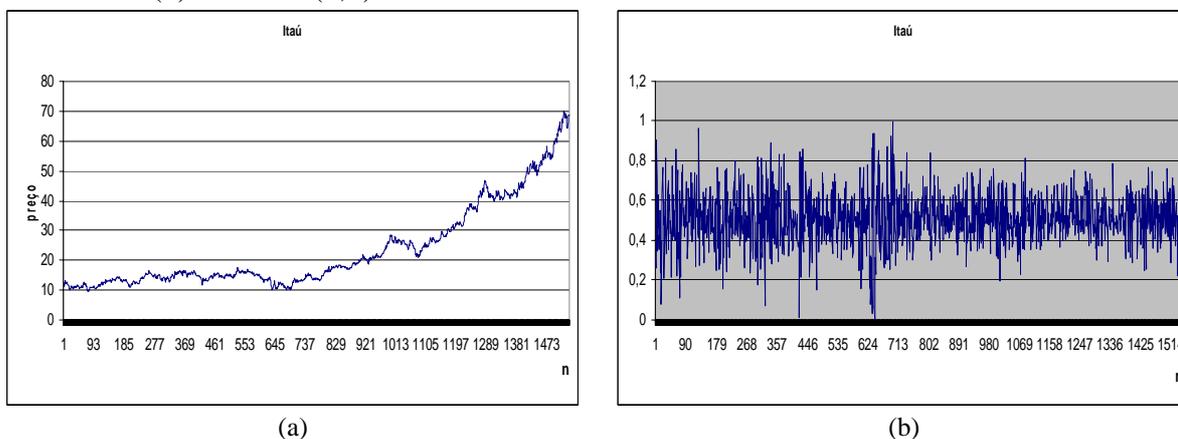
**Figura-3** (a) Previsão obtida pelo modelo ARIMA-GARCH, (b) previsão da RNA para NDLBRAD

As medidas de desempenho, obtidos por cada uma das técnicas, o TIC e o RMSE são mostrados na Tabela-2. Notamos que a rede neural apresentou o melhor desempenho.

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

**Tabela-2** Estatísticas de ajuste obtidas para a série NDLBRAD

A Figura-4a mostra a série de preço do Itaú e a Figura-4b 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

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

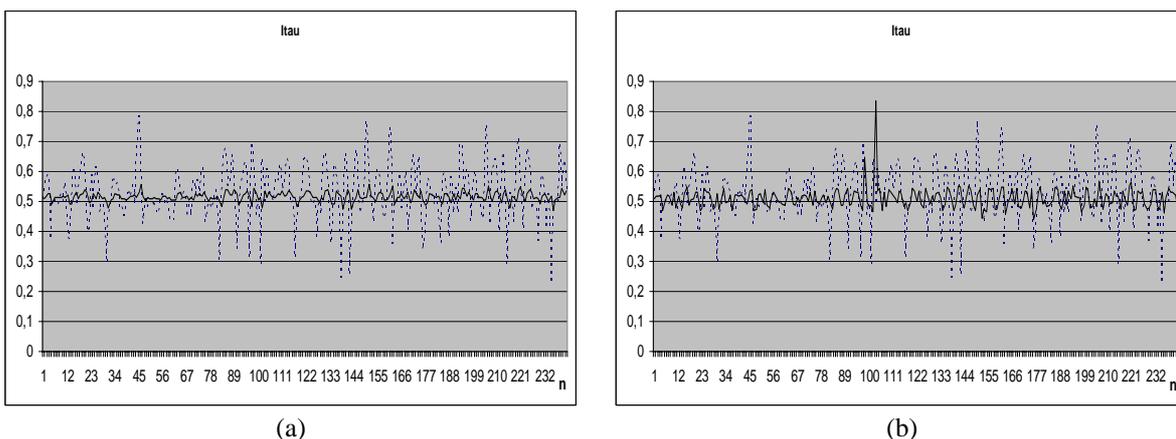
Parâmetros	Coefficiente	Erro-padrão	Estatística z	Significância
$\phi_0$	0,515000	0,003781	136,1979	0,0000
$\phi_1$	0,158516	0,027529	5,758036	0,0000
$\alpha_0$	0,000449	0,000144	3,107148	0,0019
$\alpha_1$	0,075684	0,013720	5,516254	0,0000
$\beta_1$	0,894804	0,017243	51,89240	0,0000

**Tabela-3** Parâmetros estimados para a série NDLITAU

As previsões realizadas utilizando o modelo AR-GARCH e a RNA 10-2-1 são mostradas na Figura-3a e 3b, respectivamente. As linhas tracejadas indicam o valor da série NDLITAU e as linhas contínuas indicam as previsões. Na Tabela-4 temos as estatísticas TIC e RMSE. Notamos que a rede neural apresentou o melhor desempenho.

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

**Tabela-4** Estatísticas de ajuste obtidas para a série NDLITAU

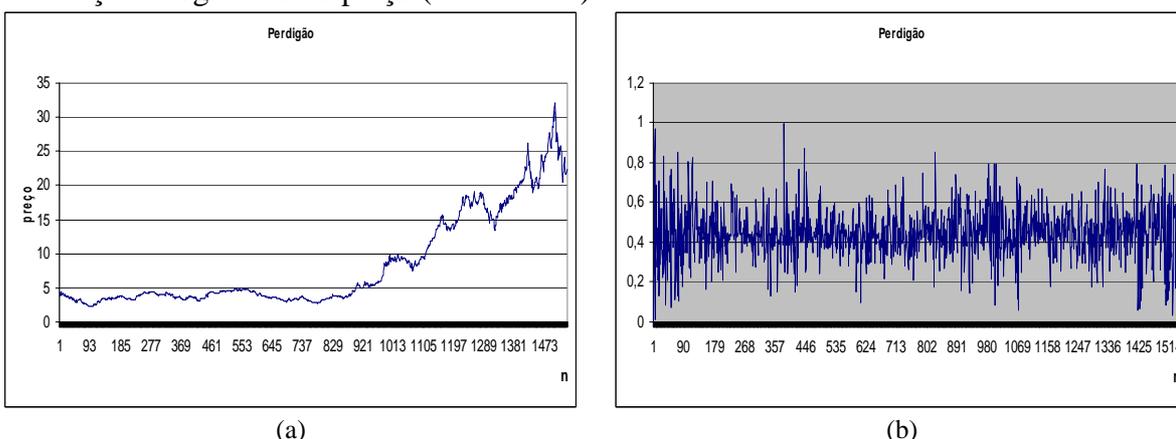


**Figura-5** (a) Previsão obtida pelo modelo ARIMA-GARCH, (b) previsão da RNA para NDLITAU

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

#### 4.2 Setor de Alimentos – Perdigão e Sadia

Na Figura-6a é mostrada a série de preços da Perdigão e na 6b 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

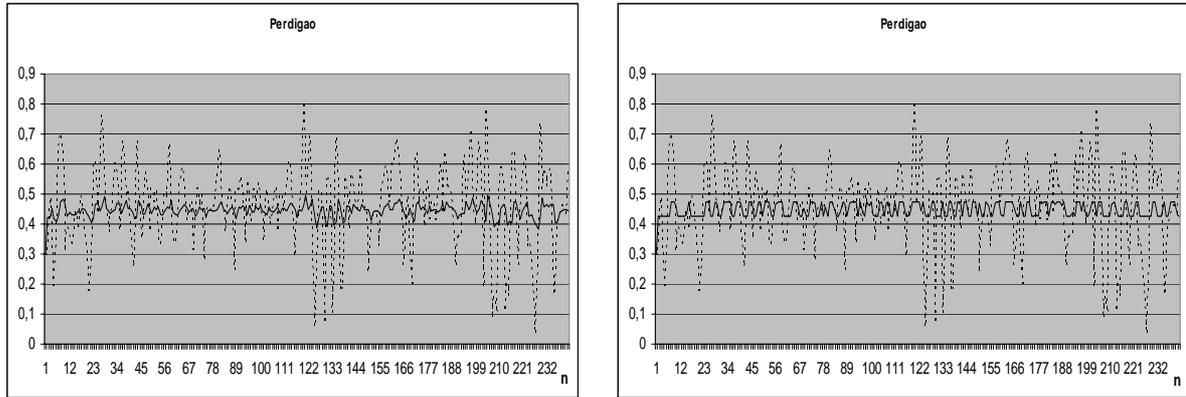
A Tabela-5 mostra os parâmetros estimados para o modelo AR(1)-GARCH(1,1).

Parâmetros	Coefficiente	Erro-padrão	Estatística z	Significância
$\phi_0$	0,445162	0,003311	134,4327	0,0000
$\phi_1$	0,147928	0,026878	5,503584	0,0000
$\alpha_0$	0,000728	0,000163	4,460329	0,0000

$\alpha_1$	0,072308	0,010961	6,597043	0,0000
$\beta_1$	0,865022	0,020639	41,91157	0,0000

**Tabela-5** Parâmetros estimados para a série NDLPED

As previsões realizadas utilizando o modelo ARIMA-GARCH e a RNA 10-1-1 são mostradas na Figura-7a e 7b, respectivamente. As linhas tracejadas indicam o valor da série NDLPED e as linhas contínuas indicam as previsões.



(a) (b)

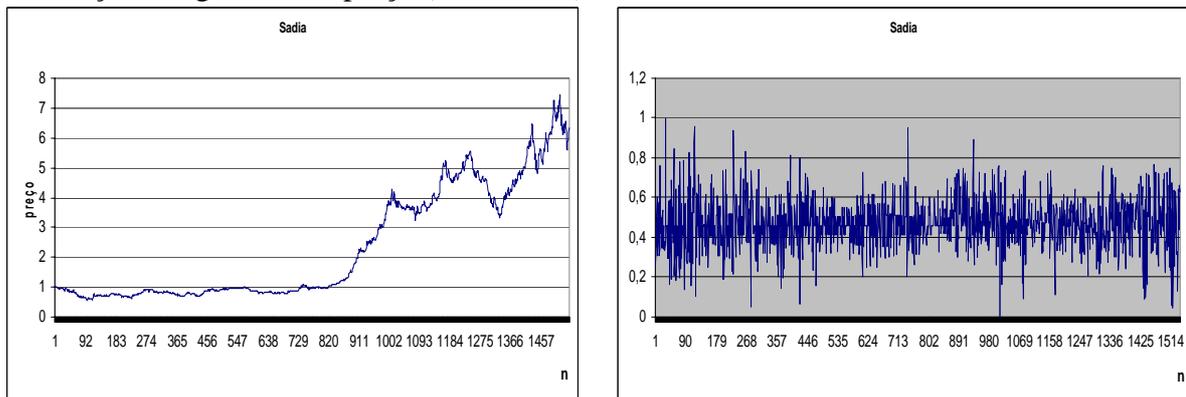
**Figura-7** (a) Previsão obtida pelo modelo ARIMA-GARCH, (b) previsão da RNA para NDLPED

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.

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

**Tabela-6** Estatísticas de ajuste obtidas para a série NDLPED

Na Figura-8a é apresentada a série no nível da Sadia e na 8b a série normalizada da primeira diferença do logaritmo do preço (NDLSADI).



(a) (b)

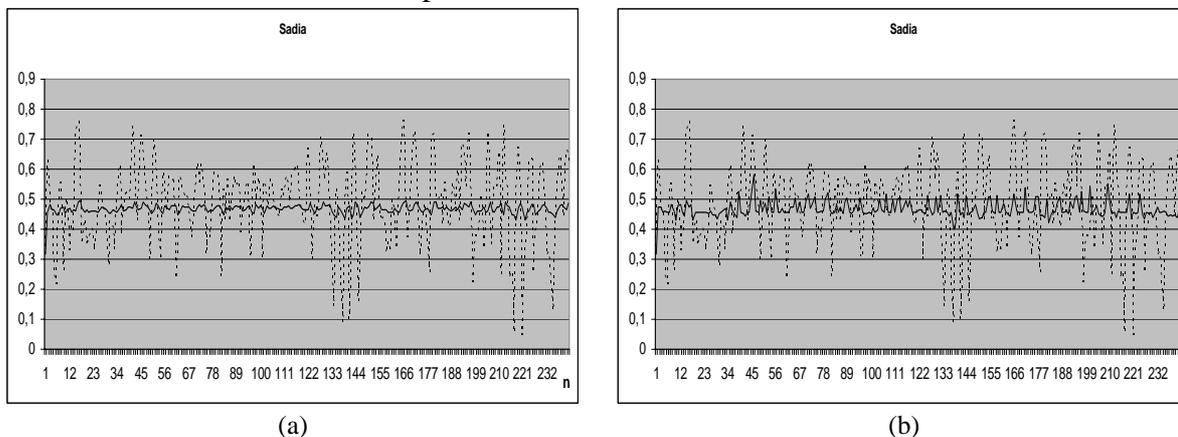
**Figura-8** (a) Série no nível do preço da Sadia, (b) Série transformada NDLSADI

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

Parâmetros	Coefficiente	Erro-padrão	Estatística z	Significância
$\phi_0$	0,468054	0,002993	156,3786	0,0000
$\phi_1$	0,087591	0,026559	3,297963	0,0010
$\alpha_0$	0,000546	0,000150	3,649578	0,0003
$\alpha_1$	0,102279	0,015240	6,711293	0,0000
$\beta_1$	0,860133	0,021157	40,65519	0,0000

**Tabela-7** Parâmetros estimados para a série NDLSADI

As previsões realizadas utilizando o modelo ARIMA-GARCH e a RNA 10-2-1 são mostradas na Figura-9a e 9b, respectivamente. As linhas tracejadas indicam o valor da série NDLSADI e as linhas contínuas indicam 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. Notamos que as medidas de desempenho dos dois modelos são próximas, mas com valores menores para a RNA.

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

**Tabela-8** Estatísticas de ajuste obtidas para a série NDLSADI

## 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 e de alimentos. Para cada um dos setores foram escolhidas, sem nenhum critério específico, duas séries. No setor financeiro foram analisadas as séries do Bradesco e do Itaú, no setor de alimentos a Perdigão e a Sadia. Todas as séries continham tamanho 1542, sendo que os 1300 primeiros foram utilizados para estimação dos modelos ARIMA-GARCH e treinamento das RNA, o restante dos valores foi utilizado para medir a capacidade de previsão dos modelos. Para cada uma das séries foi escolhido o melhor modelo ARIMA-GARCH e a melhor RNA, sendo que foram testadas redes com 10 entradas, conteúdo entre 1 e 9 neurônios na camada intermediária e um neurônio na saída. Para medir o desempenho de cada uma dessas duas técnicas foram utilizadas duas medidas: a raiz do erro quadrático médio (RMSE) e o coeficiente de desigualdade de Theil (TIC).

Dos resultados obtidos nos experimentos verifica-se que as previsões dos modelos ARIMA-GARCH foram melhores nos casos da empresa Perdigão. As RNA realizaram melhores previsões nos casos das empresas Bradesco, Itaú e Sadia. O que se observa é que as duas técnicas implementadas podem ser utilizadas como ferramentas complementares e de comparação para poder-se aumentar a capacidade de decisão do analista.

## 6. Referências

- ABU-MOSTAFA, Y. S., ATIYA, A. F., MAGDON-ISMAIL, M, e WHITE, H. Introduction to the special issue on neural networks in financial engineering. *IEEE Transactions on Neural Networks* 12: 653–655, 2001.
- ANDERS, W. Garch forecasting performance under different distribution assumptions, *Journal of Forecasting*, 2006.
- ANDERSEN, T. J. e WILAMOWSKI, B.M. “A. Modified Regression Algorithm for Fast One Layer Neural Network Training”, *World Congress of Neural Networks*, vol. 1, pp. 687-690, Washington DC, USA, 1995.
- AZZOF, E. M. Reducing Error in Neural Network Time Series Forecasting. *Neural Computing and Applications*, vol. 1, pp. 240-247, 1993.
- AZZOF, E. M. *Neural Network Time Series Forecasting of Financial Markets*, John Wiley & Sons, 1994.
- educing Error in Neural Network Time Series Forecasting. *Neural Computing and Applications*, vol. 1, pp. 240-247, 1993.
- BARUCCI, E. e RENO, R. On measuring volatility and the GARCH forecasting performance, Article provided by Elsevier in its journal *Journal of International Financial Markets, Institutions and Money*. Vol.12, 2002.
- BATTITI, R., “First- and second-order methods for learning: between steepest descent and Newton’s method”, *Neural Computation*, vol. 4, no. 2, pp. 141-166, 1992.
- BOLLERSLEV, T., CHOU, R. Y., KRONER, K. F., ARCH Modeling in Finance: A Review of the Theory and Empirical Evidence. *Journal of Econometrics* 52, pp. 5-59, 1992.
- BOLLERSLEV, T., Generalized Autoregressive Conditional Heteroskedasticity, *Journal of Econometrics*, 31, pp. 307-327, 1986.
- CASDAGLI, M. e EUBANK, S. *Nonlinear Modeling and Forecasting*, Santa Fe Institute Proceedings vol. XII, Addison-Wesley, 1992.
- CASDAGLI, M. Nonlinear Prediction of chaotic time series. *Physica D*, vol. 35, pp. 335-356, 1989.
- CHAKRABORTY, K. Forecasting the behaviour of Multivariate Time Series Using Neural Networks. *Neural Networks*, vol. 5, pp. 961-970, 1992.
- CHARALAMBOUS, C., “Conjugate gradient algorithm for efficient training of artificial neural networks”, *IEEE Proceedings*, vol. 139, no. 3, pp. 301-310, 1992.
- CHENG, B. E TITTERINGTON, D. M., *Neural Networks: A Preview from Statistical Perspective*. Statistical Science, 1994.
- CLEMENTS, M. P. e HENDRY, D. F. *Forecasting Non-stationary Economic Time Series*. MIT Press: Cambridge, MA, 1999.
- DING, Z., GRANGER, C.W. J. e ENGLE, R. F., A long memory property of stock market returns and a new model, *Journal of Empirical Finance*, 1, 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*, Wiley, 2004.
- ENGLE, R. F. Autoregressive conditional heteroskedasticity with estimates of the variance of U. K. inflation, *Econometrica* 50, pp. 987-1008, 1982.
- FINE, T. L. *Feedforward Neural Network Methodology*. Springer: New York, 1999.
- FISHMAN, M. B., BARR, D. S. e LOICK, W. J. Using Neural Nets in Market Analysis. *Technical Analysis of Stocks and Commodities*, vol. 9, no. 4, 1991.
- GONZALES, M. F., e BURGESS, N. Modeling Market Volatilities: The Neural Network Perspective. *European Journal of Finance* 3: pp. 137-157, 1997.
- HAGAN, M. T. e MENHAJ, M., “Training feedforward networks with the Marquardt algorithm”, *IEEE Transactions on Neural Networks*, vol. 5, no. 6, pp. 989-993, 1994.
- HAMILTON, J.D., *Time Series Analysis*. Princeton University Press, 1994.

- HAWLEY, D. D., JOHNSON, J. D. e RAINA, D. Artificial Neural Systems: A New Tool for Financial Decision-Making. *Finan Analy J*, 1990.
- KUAN, C. M., e WHITE, H. Artificial Neural Networks: An Econometric Perspective. *Econometric Reviews* 13, pp. 1-91, 1994.
- LAPEDES, A. e 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 Vol. 2*, pp. 164-168, 1944.
- MARQUARDT, D., An algorithm for least-squares estimation of nonlinear parameters, *Journal of Applied Mathematics Vol. 11*, pp. 431-441, 1963.
- MEDEIROS M., TERÄSVIRTA, T. e RECH, G. Building Neural Network Models for Time Series: A Statistical Approach, *Journal of Forecasting* 25, pp. 49-75, 2006.
- MEISSNER, G., e KAWANO, N. Capturing the volatility smile of options on high-tech stocks-a combined GARCH-Neural Network approach. *Journal of Economics and Finance* 25(3), pp. 276-293, 2001.
- MORETTIN, P. A., TOLOI, C. M. C., *Análise de Séries Temporais*, Edgard Blucher, 2004.
- POSEDEL, P., Properties and Estimation of GARCH(1,1) Model, *Metodoloski zvezki*, Vol. 2, No.2, 243-257, 2005.
- QI, M., e ZHANG, G. P. An investigation of model selection criteria for neural network time series forecasting. *European Journal of Operational Research* 132(3) (August), pp. 666-680, 2001.
- REFENES, A. N. Currency Exchange Rate Prediction and Neural Network Design Strategies. *Neural Computing and Applications*, vol.1, pp. 46-58, 1993.
- RIPLEY, B. D., *Neural Networks and Related Methods of Classification*, *Journal of the Royal Statistical Society, Series B*, 56, 3, pp. 409-456, 1994.
- RIPLEY, B. D., *Pattern Recognition and Neural Networks*. Cambridge Press, 1996.
- RIPLEY, B. D., *Statistical Aspects of Neural Networks*. Chapman e Hall, 1993.
- RUMELHART, D. E. e McCLELLAND, J. L. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, MIT Press, 1986.
- RUMELHART, D. E., HINTON, G. E. e WILLIAMS, R. J, "Learning internal representations by error propagation", In *Parallel Distributed Processing*, vol 1, pp. 318-362. Cambridge, MA: MIT Press, 1986a.
- SHAH, S. e PALMIERI, F.. "MEKA - A fast, local algorithm for training feedforward neural networks", *Proceedings of International Joint Conference on Neural Networks*, San Diego, CA, 3, 41-46, 1990.
- SURATGAR, A. A., TAVAKOLI, M. B. e HOSEINABADI, A. Modified Levenberg-Marquardt Method for Neural Networks Training, *Transactions on Engineering, Computing and Technology*, 2005.
- TERÄSVIRTA, T., VAN DIJK, D. e 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", *Proceeding of International Conference on Neural Networks*, San Diego, CA, 1, pp. 343-354, 1988.
- WILAMOWSKI, B. M., IPLIKCI, S., KAYNAK, O. e EFE, M. O. An Algorithm for Fast Convergence in Training Neural Networks, *IEEE*, 2001.