

PREVISIBILIDADE DE MERCADOS: UM ESTUDO COMPARATIVO ENTRE BOVESPA E S&P500

PREDICTABILITY OF MARKETS: A COMPARATIVE STUDY BETWEEN BOVESPA AND S&P500

EVERTON ANGEL CAVALHEIRO,¹ PAULO SÉRGIO CERETTA,² CARLOS EDUARDO MOREIRA TAVARES³ E LARISSA DE LIMA TRINDADE⁴

Recebido em: 15/04/2010

Aprovado em: 30/06/2010

RESUMO

Este estudo tem por objetivo comparar a previsibilidade do mercado de capitais brasileiro e americano, mediante o uso de redes neurais *Group Method of Data Handling* (GMDH) em uma aproximação indutiva do retorno logarítmico mensal do índice Ibovespa e do S&P500. Inicialmente, foi realizada uma revisão teórica sobre o tema de redes neurais e, após, procurou-se a desmistificação do conceito de redes neurais. Os resultados das redes demonstram uma maior, e satisfatória, previsibilidade para o mercado acionário brasileiro e menor, assim como insatisfatória, previsibilidade do retorno mensal do americano S&P500. **Palavras-chave:** Previsão; Redes neurais; Séries temporais.

ABSTRACT

This study aims to compare the predictability of the Brazilian capital market and American, through the use of neural networks Group Method of Data Handling (GMDH) in an inductive approach of monthly log returns of the index Bovespa Index and the S&P500. Initially, we performed a literature review on the subject of neural networks and, later, tried to demystify the concept of neural networks. The results show a larger network, and satisfactory predictability for the Brazilian stock market and lower, as unsatisfactory, the predictability of the monthly return of the S&P500.

Keywords: Forecast; Neural networks; Time series.

1 Introdução

O *Group Method of Data Handling* (GMDH) é uma derivação do campo das Redes Neurais. Esse modelo é caracterizado pela sua habilidade de selecionar a combinação de variáveis endógenas mais significativas para prever uma variável exógena. A base desse modelo é a manipulação dos dados, a fim de encontrar a melhor combinação de tais variáveis. As redes, auto-organizáveis são baseadas no desenvolvimento de modelos com incremento gradual em sua complexidade, sendo sua evolução avaliada por um critério externo.

Ferson (2007) cita que o interesse pela previsão dos preços dos retornos das ações seja, provavelmente, tão antigo quanto os próprios mercados, bem como a literatura sobre o assunto é vasta e significativa. Nesse sentido, o uso de redes neurais em modelos de previsão tem se mostrado cada vez mais presente na literatura e nas modelagens empíricas. Contudo, raros estudos têm sido desenvolvidos através de modelagens polinomiais. Dessa forma, a pergunta de pesquisa deste artigo é: “O uso de redes polinomiais GMDH é suficientemente capaz demonstrar não aleatoriedade no retorno mensal do IBOVESPA e do S&P500?”.

O objetivo geral deste trabalho é comparar a previsibilidade dos mercados de capitais brasileiros e americanos, através do

¹ Professor auxiliar da Universidade de Cruz Alta (UNICRUZ). Especialista (MBA) pela Fundação Getúlio Vargas (FGV) e mestre em Administração pela Universidade Federal de Santa Maria (UFSM).

² Professor adjunto da UFSM, doutor em Engenharia de Produção pela Universidade Federal de Santa Catarina.

³ Professor assistente da UNICRUZ, mestre em Engenharia de Produção pela UFSM.

⁴ Professora do Centro Universitário Franciscano. É mestre em Administração pela UFSM.

uso de redes neurais GMDH para duas séries de dados do retorno mensal do índice Ibovespa e do S&P500. Secundariamente, procura-se desvelar o conceito de redes neurais polinomiais, pouco difundidas no Brasil, e, por fim, demonstrar passo a passo como são tomadas as decisões durante o cálculo de uma rede neural polinomial *Group Method of Data Handling* (GMDH). Na sequência deste artigo, é apresentada a revisão bibliográfica acerca de redes neurais artificiais.

2 Revisão bibliográfica

Inicialmente esse artigo apresentará uma revisão do conceito de redes neurais, bem como as formas de treinamento existentes para o conjunto de dados. Após, é apresentado o conceito de redes polinomiais, em especial detalhe as do tipo GMDH.

2.1 Redes neurais artificiais

As Redes Neurais Artificiais (RNAs) são um conjunto de técnicas que tentam simular, de uma maneira simplificada, em meio computacional, o funcionamento do cérebro humano. Elas são capazes de reconhecer padrões, extrair regularidades e detectar relações subjacentes em um conjunto de dados aparentemente desconexos. Além disso, elas apresentam habilidade de lidar com dados ruidosos, incompletos ou imprecisos. Sua capacidade de prever sistemas não-lineares torna a sua aplicação no mercado financeiro muito objetiva. Para Haykin (2001, p. 28), rede neural pode ser definida como um processador maciço e paralelamente distribuído, constituído de unidades de processamento simples, que têm a propensão natural para armazenar conhecimento experimental e torná-lo disponível para uso. A rede neural assemelha-se ao cérebro, em dois aspectos: (1) o conhecimento é adquirido pela rede a partir de seu ambiente através de um processo de aprendizagem; (2) forças de conexão entre

neurônios (pesos sinápticos) são utilizadas para armazenar o conhecimento adquirido.

As RNAs apresentam duas fases distintas em sua metodologia, que são denominadas de aprendizagem e teste. Na fase de aprendizagem, são apresentados estímulos à entrada, denominados padrões de treinamento que farão com que o método os encontre nos dados que serão testados na fase de teste. Na fase de teste, é demonstrada a capacidade de generalização da rede, pois seus resultados deverão ser significativos após seus pesos terem sido ajustados na fase anterior. Fausett (1995, p. 280) cita que existem três formas de treinamento: a supervisionada, a não-supervisionada e a híbrida.

2.1.1 Redes neurais supervisionadas

Segundo a autora De Ré (2000, p. 30), redes neurais supervisionadas caracterizam-se pela utilização de um conjunto de treinamento composto por dados de entrada previamente classificados. O ajuste dos pesos ocorre através da apresentação dos estímulos de entrada, disponíveis no conjunto de treinamento. Para o cálculo da resposta, utilizam-se, como parâmetro, os valores dos pesos atuais. Faz-se, então, uma comparação entre a resposta oferecida pela rede atual e a desejada àqueles estímulos e, com base na similaridade entre as duas respostas, os pesos são ajustados. Esse procedimento perdura até que ocorra a classificação mais adequada possível ou caso não haja possibilidade de ganho representativo nas modelagens.

Esse tipo de rede dá uma resposta mais razoável, baseada em uma variedade de padrões de aprendizado, porque em uma rede supervisionada é possível o usuário mostrar à rede como fazer previsões, classificações ou decisões, fornecendo-lhe um grande número de classificações corretas ou previsões com as quais ela pode aprender. Redes de Retropropagação, *General Regression Neural Networks* (GRNN), *Probabilistic Neural Networks* (PNN) e *Group Method Data Handling* (GMDH) são exemplos de redes supervisionadas.

2.1.2 Redes neurais não supervisionadas

De acordo com Fausett (1995, p. 165), as redes neurais não-supervisionadas aprendem sem que exista um conjunto de “respostas esperadas” como referência para as saídas. Os pesos são ajustados à medida que a rede vai sendo renovada de padrões de entrada selecionados como representativos de cada classe, isto é, de acordo com sua similaridade aos padrões de entrada apresentados à rede. Normalmente, a distância euclidiana entre ambos é calculada, e o neurônio, representando o peso mais próximo de vetor de entrada, é considerado como o vencedor, tendo, então, seus pesos ajustados. A taxa de aprendizado deve diminuir com o tempo de treinamento.

2.1.3 Redes neurais híbridas

De acordo com Towell e Shavlik (1994, p. 120), os métodos híbridos usam conhecimento teórico de um domínio e um conjunto de exemplos para desenvolver um método de exemplos classificados corretamente, não vistos durante o treinamento. O desafio do sistema de aprendizado híbrido está em usar a informação proporcionada por uma fonte de dados para compensar a perda de informação de outra. Dessa forma, um sistema híbrido deve aprender mais eficientemente do que um sistema que usa somente uma fonte de informação. *Knowledge-Based Artificial Neural Networks* (KBANN) é um sistema híbrido, construído sobre técnicas de aprendizado conexionistas, que mapeia problemas específicos representados em lógica proposicional em rede e, então, refina os conhecimentos reformulados utilizando *backpropagation*, e, conforme cita Kovács (2006, p. 83), esse método procura retroceder o erro, a fim de melhorar o seu desempenho.

2.2 Redes polinomiais GMDH

No contexto estocástico da análise de dados, encontra-se a rede GMDH. As primeiras investigações foram realizadas por Alexey Ivakhnenko, em 1968, e seguidas

por R. Shankar, em 1972, o qual apresentou o algoritmo do mesmo nome como um método que permitia descobrir de forma sucessiva um sistema complexo de relações a partir de simples operações matemáticas. Trata-se de um método para solucionar problemas de estilo, identificação, predição de curto prazo de processos aleatórios e reconhecimento de padrões. O conteúdo desse algoritmo se desenvolveu como veículo para identificar relações lineares e não-lineares entre *inputs* e *outputs*, gerando uma estrutura que tende a ótima, a partir de um processo sucessivo de várias manipulações dos dados, mediante a incorporação de novas camadas.

De acordo com Schneider e Steiner (2006, p. 20), os algoritmos da rede GMDH apresentam-se mais poderosos que os modelos estatísticos tradicionais devido aos seus procedimentos auto-organizáveis. Esses algoritmos extraem informação dos dados existentes, usando-se da complexidade obtida através de modelos matemáticos flexíveis, ademais quebram o conceito de “caixa preta” característico das redes neurais.

O modelo GMDH pode ser analisado como uma combinação de redes neurais e conceitos estocásticos, conforme cita Valença (2005, p. 24), e são implementadas com funções de ativação nos neurônios das camadas escondidas e um critério de seleção para decidir quantas camadas serão construídas. Na sua formulação original, cada neurônio da camada escondida, a ser construída, recebe duas entradas e tem como função de ativação um polinômio de segundo grau. Portanto, através da combinação de cada par desses neurônios de entrada, é gerada uma função de saída polinomial, cuja complexidade é função do número de camadas, isto é, caso se tenha duas camadas, tem-se uma função polinomial de quarto grau; para três camadas, uma função polinomial de oitavo grau. Dessa forma, tais redes são chamadas de redes polinomiais, pois o modelo que resulta é uma função polinomial.

2.2.1 Algoritmo GMDH

Algoritmo é um processo de cálculo em que se estipulam, com generalidade e sem restrições, regras formais para a obtenção do resultado ou da solução do problema. Nas redes GMDH, são desenvolvidos, entre outros, o algoritmo do mesmo nome, inicialmente desenvolvido por Ivakhnenko (1971, p. 364), em que os neurônios gerados nas camadas intermediárias são funções polinomiais de segundo grau compostos de duas variáveis.

No caso de três entradas x_1 , x_2 e x_3 , há produção de saídas y que são uma combinação quadrática dessas entradas, podendo desdobrarem-se da seguinte forma:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_1^2 + \beta_3 x_1 x_2 + \beta_4 x_1 x_3 + \beta_5 x_2 x_3 + \beta_6 x_2 + \beta_7 x_2^2 + \beta_8 x_3 + \beta_9 x_3^2 + \xi \quad (1)$$

em que y é a saída estimada; x_1 , x_2 e x_3 as variáveis de entrada; β_i seus pesos e $\hat{\epsilon}$ o erro.

Como salienta Valença (2005, p. 26), cada camada construída terá uma quantidade de neurônios que depende do número de variáveis de entrada. Supondo que o número de variáveis de entrada é n , o número de neurônios na primeira camada é de C_n^2 , isto é, $n(n-1)/2$. Tendo-se três entradas de neurônios, a primeira camada escondida será de $3(3-1)/2=3$, em que os pesos β_i desses neurônios da camada escondida serão determinados pelo conjunto de dados da fase de treinamento \hat{a} . Esses neurônios da camada escondida têm, então, seu desempenho avaliado através de uma função objetivo com limiar, que é utilizada para avaliar o desempenho de cada neurônio e para decidir quais neurônios deverão “continuar” e quais neurônios deverão “parar”. Esse processo é, então, repetido até que reste apenas um neurônio escondido ou, então, quando as camadas adicionais não provocarem nenhuma melhoria.

2.2.2 Algoritmo combinatorial

A diferença básica desse algoritmo é a sua estrutura em uma única camada e, de acordo com Valença (2005, p. 29), funções de soma são geradas para todas as combinações de variáveis de entrada de forma similar a uma busca polinomial. Esse algoritmo se torna eficiente pela utilização de uma técnica recursiva dos mínimos quadrados, uma vez que os modelos gerados são lineares em relação aos parâmetros.

Para um melhor entendimento, supondo-se uma função com três variáveis de entrada (x_1 , x_2 e x_3) e uma variável de saída, haverá uma busca polinomial do melhor modelo até o segundo grau, devendo-se começar pelo modelo mais simples ($y = \beta_0 + \xi$) até o mais complexo, semelhante ao demonstrado na equação [1]. A seguir, demonstra-se o incremento gradual da complexidade para esse algoritmo contendo uma variável:

$$y = \beta_0 + \xi \quad [2]$$

$$y = \beta_1 x_1 + \xi \quad [3]$$

$$y = \beta_2 x_1^2 + \xi \quad [4]$$

...

$$y = \beta_9 x_3^2 + \xi \quad [5]$$

Como demonstrado nas equações [2, 3, 4 e 5], existirão $C_{10}^1 = 10$ neurônios. Os pesos são determinados pelo método dos mínimos quadrados, com um conjunto de treino. Na sequência, os erros obtidos por cada equação são comparados a um limiar estabelecido para cada função objetivo (complemento externo), utilizando-se de um conjunto de dados teste. Finalmente, as unidades selecionadas, ditas sobreviventes, serão consideradas aptas para terem sua complexidade evoluída.

Num segundo passo, todos os neurônios com duas variáveis são determinados:

$$y = \beta_0 + \beta_1 x_1 + \xi \quad [6]$$

$$y = \beta_0 + \beta_2 x_1^2 + \xi \quad [7]$$

...

$$y = \beta_8 x_3 + \beta_9 x_3^2 + \xi \quad [8]$$

Logo, haverá $C_{10}^2 = 45$ equações parciais nesse segundo passo.

Os neurônios serão construídos até que, no último passo, obtenha $C_{10}^{10} = 1$ combinação, que é o polinômio completo demonstrado na equação [1]. É importante notar que, se o número de entradas for grande, irá ocorrer uma explosão de combinações, pois o número de modelos que devem ser testados aumenta rapidamente com o número de variáveis de entrada, como demonstra: $M=2^n-1$. No exemplo de três variáveis de entrada (x_1, x_2 e x_3), acrescida de uma constante, que se desdobram em 10 variáveis, demonstrada na figura [1], haverá $M=2^{10}-1 = 1.023$ neurônios totalmente diferentes.

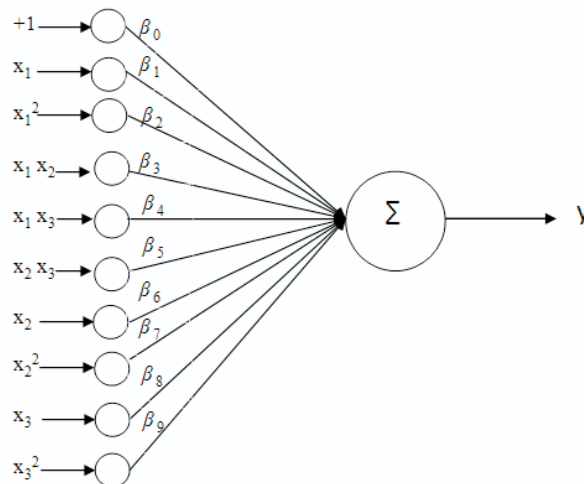


Figura 1 - Detalhe de um neurônio GMDH.

2.2.3 Rede multilayer

De acordo com Valença (2005, p. 32), a rede multilayer é uma estrutura paralela, constituída por um polinômio de segundo grau como função de ativação. O número de neurônios gerados, em cada camada escondida, depende do número de variáveis de entrada.

Nesse modelo, cada duas variáveis são combinadas para gerar um neurônio escondido, por exemplo: x_i e x_j são passados para a unidade k , que pode ser construída por uma função de ativação do tipo:

$$y = \alpha_0 + \alpha_1 x_i + \alpha_2 x_j + \alpha_3 x_i x_j + \alpha_4 x_i^2 + \xi \quad [9]$$

Os pesos são estimados usando-se um conjunto de treinamento e um dado valor de limiar. Um critério de erro é usado para avaliar o ajuste desses neurônios gerados. Quando se tem n variáveis de entrada, a primeira camada gerará C_n^2 , isto é, $n(n-1)/2$ neurônios parciais. Desses, escolhem-se os que são superiores a um dado valor de limiar estabelecido, que passarão a compor a próxima camada. Esse processo é repetido para sucessivas camadas até que o mínimo global para um dado critério de erro seja encontrado.

Os critérios utilizados para finalizar o treinamento, citados por Valença (2005, p. 27), são: a camada de saída possui apenas um neurônio ou a camada adicionada não fornece ganhos significativos em relação à anterior.

3 Método

Para o desenvolvimento deste trabalho, optou-se pela utilização de uma rede neural supervisionada, com algoritmo de seleção para frente. Dentre os modelos disponíveis, o método escolhido foi método GMDH, devido a sua característica híbrida de redes neurais e métodos estatísticos tradicionais (VALENÇA, 2005, p. 24). A rede escolhida foi a rede *multilayer* pela sua possibilidade de testar uma abrangente combinação de variáveis, bem como sua inferência estatística e o cálculo de cada neurônio foi realizado através do método dos mínimos quadrados ordinários.

3.1 Conjunto de variáveis utilizadas no estudo

Neste estudo, foram utilizadas, como variáveis, os índices IBOVESPA e S&P500. A escolha dos índices deu-se pelo fato de serem considerados os mais importantes indicadores do desempenho médio das cotações do mercado de ações brasileiro e americano. Foram utilizados os retornos mensais de janeiro de 2000 até fevereiro de 2007, perfazendo 86 observações na fase de treinamento. Na fase de teste foram simuladas 24 observações, ou 25% do tempo de treinamento, entre março de 2007 até junho de 2009.

3.1.1 Padronização dos dados

Um dos recursos mais utilizados em pesquisa de séries temporais financeiras é a transformação dos dados. As transformações diretas dos dados mais comumente utilizadas são: a logarítmica, cúbica, a quadrática, a inversa e a raiz quadrada. Para definir o tipo de transformação dos dados que poderia ser utilizada em cada uma das séries de dados, foi realizado o teste do qui quadrado, conforme demonstrado no quadro 1.

Analisando-se a tabela 1, é possível evidenciar que a padronização dos dados com melhor p valor foi a logarítmica. Dessa forma, foi utilizado o retorno mensal, na forma logarítmica, do índice Ibovespa e do S&P500. Para formar as variáveis x_1 , x_2 e x_3 após a normalização logarítmica (conforme demonstrado na equação [11]), foram realizadas defasagens (Lag) $t-1$, $t-2$ e $t-3$, respectivamente.

Tabela 1 – Transformação dos dados com os seus respectivos resultados do teste qui quadrado

| Tipo de transformação | Qui quadrado | p valor | Qui quadrado | p valor |
|-----------------------|--------------|---------|--------------|---------|
| Índice | S&P500 | | IBOVESPA | |
| Cúbica | 9.26 | 0.010 | 37.91 | 0.000 |
| Quadrática | 9.84 | 0.007 | 21.86 | 0.000 |
| Identidade | 7.65 | 0.022 | 10.91 | 0.004 |
| Raiz quadrada | 6.55 | 0.038 | 19.81 | 0.000 |
| Logarítmica | 6.50 | 0.039 | 41.91 | 0.020 |
| 1/Raiz Quadrada | 7.92 | 0.019 | 32.5 | 0.000 |
| Inversa | 10.75 | 0.005 | 11.77 | 0.003 |
| 1/Quadrática | 19.53 | 0.000 | 18.62 | 0.000 |
| 1/Cúbica | 30.81 | 0.000 | 41.93 | 0.000 |

Tsay (2002, p. 2) cita que a maioria dos estudos de séries temporais financeiras utiliza os retornos, em vez dos preços dos ativos em questão. Neste sentido, o autor comenta que existem duas razões principais para o uso dos retornos em estudos financeiros: em primeiro lugar, para a média dos investidores, o retorno dos ativos é uma medida adequada para a comparação entre oportunidade de investimentos, e, em segundo lugar, as séries de retorno são mais fáceis de lidar do que uma série de preços, porque os primeiros apresentam propriedades estatísticas mais atraentes. Dentre tais propriedades, cita-se a ausência de tendenciosidade comum em séries de dados não estacionados. Tsay (2002, p. 11) comenta que a utilização de log-retornos em estudos financeiros é indicada pela hipótese de que os retornos dos ativos são independente e identicamente distribuídos (i.i.d.) com média μ e variância σ^2 .

3.1.2 A forma logarítmica do retorno

Com capitalização contínua, o preço de um título é dado por $P_t = P_{t-1}e^r$. Onde r é a taxa de retorno, P_t é o preço da ação na data t , P_{t-1} é o preço da ação na data $t-1$;

A expressão pode ser reescrita da seguinte forma:

$$\frac{P_t}{P_{t-1}} = e^r \quad [10]$$

Extraindo-se o logaritmo natural dos dois lados da igualdade, obtém-se:

$$r = \ln\left(\frac{P_t}{P_{t-1}}\right) \quad [11]$$

De acordo com Soares, Rostagno e Soares (2002, p. 8), ao extrair o logaritmo natural da razão P_t / P_{t-1} , a curva representativa da distribuição de frequência torna-se menos assimétrica. Isso ocorre porque o logaritmo natural de números situados entre zero e um é negativo, e o logaritmo natural de números maiores que um é positivo. Dessa forma, quando P_t for menor que P_{t-1} , o logaritmo natural de P_t / P_{t-1} tende a $-\infty$ e se P_t for maior que P_{t-1} ,

o logaritmo tende a ∞ . Como existem infinitos resultados possíveis para a razão P_t / P_{t-1} , assim como infinitos números maiores que um, a curva relativa à distribuição de frequência dos retornos seria uma distribuição mais simétrica, tendendo a centrar em zero.

3.1.3 Desdobramento das variáveis

Inicialmente, as variáveis puras, x_1 , x_2 e x_3 serão acrescidas de uma constante. Na sequência, essas variáveis formarão a uma nova combinação, multiplicando-se duas a duas, formando as novas variáveis: $1, x_1, x_1^2, x_1x_2, x_1x_3, x_2x_3, x_2, x_2^2, x_3, x_3^2$

3.2 Cálculo da primeira camada da rede

Essa subseção propõe o método de cálculo da primeira camada da rede.

3.2.1 Algoritmo combinatorial

Para iniciar o cálculo da primeira camada, foi utilizado o conceito do algoritmo combinatorial (COMBI) para a definição do número de neurônios a serem testados nessa camada. Haverá o desenvolvimento de combinações entre as dez variáveis, citadas no subitem 3.1.3. Na equação [12], é apresentado o modelo de cálculo do total de combinações que serão testadas nessa camada:

$$N = c_{10}^1 + c_{10}^2 + c_{10}^3 + c_{10}^4 + c_{10}^5 + c_{10}^6 + c_{10}^7 + c_{10}^8 + c_{10}^9 + c_{10}^{10}, \quad [12]$$

em que N é o número de neurônios que formará a primeira camada da rede.

3.2.2 Treinamento conforme algoritmo iterativo *multilayer*

O cálculo de cada neurônio e, como consequência, a seleção de cada parâmetro foram realizados através do método dos mínimos quadrados ordinários. A determinação das variáveis que formarão cada neurônio deu-se de forma semelhante ao método *stepwise*. Segundo Hair et al. (2005), o método *stepwise* envolve a inclusão das variáveis independentes (endógenas) na função, uma por vez, com base no seu poder discriminatório. A abordagem começa, assim, como no algoritmo *multilayer*, escolhendo a melhor variável discriminatória. A variável inicial faz par, então, com cada uma das outras variáveis independentes, uma de cada vez, e a variável mais adequada para melhorar o poder discriminatório da função em combinação com a primeira variável é escolhida. A terceira e as demais variáveis são selecionadas de maneira semelhante. Conforme variáveis adicionais são incluídas, algumas variáveis previamente escolhidas podem ser removidas se a informação que elas contêm sobre diferenças de grupos estiver disponível em algumas combinações das outras variáveis incluídas em estágios posteriores. Eventualmente todas as variáveis serão incluídas na função ou as variáveis excluídas não contribuirão significativamente para a discriminação futura.

Este algoritmo é útil quando o pesquisador quer considerar um número relativamente grande de variáveis independentes para a inclusão na função, selecionando-se sequencialmente, a próxima melhor variável discriminante em cada passo, as variáveis que não são úteis na discriminação entre os grupos são eliminadas e um conjunto reduzido de variáveis é identificado. O conjunto reduzido geralmente é quase tão bom quanto e, às vezes, melhor que o conjunto completo de variáveis.

A diferença básica do algoritmo combinatorial, quando comparado ao método *stepwise*, está no critério de decisão adotado para a determinação dos modelos ótimos. O critério de decisão adotado nesta pesquisa foi o *Predict Squared Error (PSE)*. Esse critério além de levar em consideração o erro médio quadrático (EMQ), também leva em consideração a

variância do erro e uma penalidade para os modelos mais complexos, o que pode evitar o problema de *overfitting*, conforme cita Valença (2005). Após, as equações foram ranqueadas, conforme os resultados do PSE, em que os menores resultados foram considerados os melhores. Na equação [13] é demonstrado o método de cálculo do PSE:

$$PSE = \left[\frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{n-2} \right] + 2\sigma_\varepsilon^2 (p/n) \rightarrow \min, \quad [13]$$

em que σ_ε^2 é a variância do erro do modelo; p é o número de parâmetros no modelo; n o número de observações no arquivo de treinamento da rede; y_i são os valores tabelados da variável explicada e \hat{y}_i são os valores calculados de acordo com o modelo.

3.2.2.1 Seleção dos melhores neurônios

Foram selecionados os três melhores neurônios nessa camada, segundo o critério *PSE*. Como o *PSE* apresenta uma relação direta entre o erro médio quadrático e a variância do erro, quanto menor o valor calculado em cada neurônio, maior será o ajustamento da variável prevista (\hat{y}_i) com a variável observada (y_i).

3.3 Cálculo nas camadas subsequentes

Os três neurônios selecionados, conforme citado no subitem 3.2.2.1., formaram as novas variáveis de entrada da próxima camada, isto é, w_1 , w_2 e w_3 . Houve um novo desdobramento das variáveis, conforme citado no subitem 3.1.2 e, após, houve uma busca polinomial, entre as próximas combinações de variáveis, selecionando os três melhores neurônios, a fim de diminuir o critério de decisão *Predict Squared Error (PSE)*. Esse processo de treinamento e seleção foi, então, repetido para cada nova camada.

3.4 Critério para a finalização do treinamento

Conforme destaca Valença (2005, p. 27), os critérios utilizados para finalizar o treinamento são: a camada adicionada não oferece ganhos significativos em relação à anterior ou o valor do erro médio quadrático (EMQ) para o conjunto de treinamento é superior ao EMQ da camada anterior. Nesse caso, o modelo ótimo encontra-se na camada anterior.

3.5 Análise dos resultados das previsões

O critério utilizado para avaliar o sucesso das previsões, é demonstrado na equação [14], conforme propõem Ivakhnenko, Ivakhnenko e Müller (1993, p. 420). Com $\delta^2 < 0,5$, o resultado do modelo é considerado bom, com $0,5 < \delta^2 < 0,8$ é considerado satisfatório e com $\delta^2 > 1,0$, a modelagem é considerada como falha e os resultados do modelo devem ser considerados como desinformação, conforme citam Ivakhnenko e Ivakhnenko (1995, p. 532).

$$\delta_i^2 = \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{\sum_{i=1}^N (y_i - \bar{y})^2} \rightarrow \min, \quad [14]$$

Conforme citam Ivakhnenko e Ivakhnenko (1995, p. 532), com essa abordagem, cada modelagem a ser realizada em redes GMDH usa dois critérios. Em um primeiro momento, há uma exaustiva pesquisa entre todos os neurônios candidatos, conforme as combinações citadas no subitem 3.2.1, e um pequeno número de modelos cuja estrutura

está perto de ótima selecionado, nesse caso, são adotados o *PSE*, conforme é demonstrado na equação [13]. Então, apenas o neurônio selecionado como ótimo na última camada é testado em sua conformidade com um especial critério discriminatório, sendo, neste artigo, demonstrado na equação [14].

3.6 Análise dos erros e comparação entre mercados

A fim de se comparar a previsibilidade dos mercados, foram utilizados o coeficiente de determinação amostral R^2 , pois esse procura medir a proporção ou percentual de variação de Y previsto pelos modelos, conforme demonstrado na equação [15]:

$$R^2 = 1 - \frac{\sum_i^N (\hat{y}_i)^2}{\sum_i^N (y_i - \bar{y})^2}, \quad [15]$$

Outros dois indicadores foram utilizados: o erro quadrático médio (EMQ) e o erro absoluto médio (MAE), que são demonstrados nas equações [16] e [17]:

$$MSE = \sqrt{\frac{1}{N} \sum_i^N (y_i - \hat{y}_i)^2}, \quad [16]$$

$$MAE = \frac{1}{N} \sum_i^N \left| \sqrt{y_i^2} - \sqrt{\hat{y}_i^2} \right|, \quad [17]$$

Adicionalmente foi analisado o coeficiente de desigualdade de Theil, também chamado de U . O numerador de U é o EMQ, mas a escala do denominador é tal que U estará sempre entre 0 e 1. Se $U=0$ há um ajustamento perfeito da previsão com o valor observado. Se $U=1$, o desempenho da previsão do modelo é o pior possível. O coeficiente de desigualdade de Theil é demonstrado na equação [18].

$$U = \frac{\sqrt{\frac{1}{N} \sum_i^N (y_i - \hat{y}_i)^2}}{\sqrt{\frac{1}{N} \sum_i^N (y_i)^2} + \sqrt{\frac{1}{N} \sum_i^N (\hat{y}_i)^2}}, \quad [18]$$

Além do coeficiente de desigualdade de Theil é possível as proporções U^M e U^S que são denominadas respectivamente de proporção da tendenciosidade e proporção da variância que permitem decompor o erro em suas fontes características.

Conforme Pindyck e Rubinfeld (2004), a proporção de tendenciosidade (U^M) é uma indicação do erro sistemático, pois mede quanto os valores médios das séries simulada e efetiva se desviam uns dos outros. Qualquer que seja o valor do coeficiente de desigualdade (U), espera-se que U^M seja próximo de 0. Um valor elevado de U^M (acima de 0,1 ou 0,2) seria preocupante, porque indicaria a presença de tendenciosidade sistemática, de modo que seria necessário rever os modelos.

A proporção de variância U^S , conforme citam Pindyck e Rubinfeld (2004), indica a capacidade de replicar o grau de variabilidade na variável que interessa. Se U^S é alto, significa que a série efetiva flutuou muito enquanto a série simulada mostrou pouca flutuação, ou vice-versa. Isso também seria preocupante e poderia levar à revisão dos modelos.

3.7 Volatilidade dos mercados

A fim de se verificar possíveis diferenças de volatilidade entre os retornos mensais de cada um dos mercados, utilizou-se a equação [19].

$$\lambda = \sqrt{\frac{1}{(N-1) \times \sum_{i=1}^N (X - \bar{X})^2}} \quad [19]$$

Em [19], λ é a volatilidade mensal dos mercados; N é o número de meses; X é o retorno mensal de cada ativo e \bar{X} é a média dos retornos de cada índice.

4 Resultados

Todos os cálculos foram realizados em planilha eletrônica Excel, nenhum *software* estatístico foi utilizado e o método dos mínimos quadrados foi calculado através de álgebra matricial. Foram calculadas até nove camadas na rede para cada uma das 28 previsões (somente em $t+1$) dos índices IBOVESPA e S&P500 e um total de 245.520 neurônios foram testados.

4.1 Resultados da fase de teste

Na tabela 2, são demonstrados os resultados das previsões do índice S&P500.

Tabela 2 – Resultados das previsões para o índice S&P500

| Camada | R^2 | Correlação | Sinais | EMQ | MAE | U | U^M | U^S | Ivakhnenko |
|----------|---------|------------|---------|---------|---------|---------|---------|---------|------------|
| Camada 1 | 0,13199 | -0,36330 | 0,32143 | 0,08320 | 0,06005 | 0,80148 | 0,02816 | 0,00015 | 1,83111 |
| Camada 2 | 0,12509 | -0,35368 | 0,39286 | 0,13200 | 0,07597 | 0,80043 | 0,06933 | 0,00030 | 4,60941 |
| Camada 3 | 0,05258 | -0,22929 | 0,35714 | 0,13328 | 0,08363 | 0,79126 | 0,00380 | 0,00053 | 4,69928 |
| Camada 4 | 0,02646 | -0,16267 | 0,39286 | 0,11504 | 0,07531 | 0,74625 | 0,04862 | 0,00017 | 3,50140 |
| Camada 5 | 0,02611 | -0,16159 | 0,42857 | 0,11503 | 0,07509 | 0,74621 | 0,04751 | 0,00017 | 3,50037 |
| Camada 6 | 0,02582 | -0,16070 | 0,42857 | 0,11498 | 0,07505 | 0,74591 | 0,04762 | 0,00017 | 3,49744 |
| Camada 7 | 0,02580 | -0,16063 | 0,42857 | 0,11498 | 0,07508 | 0,74592 | 0,04751 | 0,00017 | 3,49760 |
| Camada 8 | 0,02580 | -0,16063 | 0,42857 | 0,11498 | 0,07508 | 0,74592 | 0,04750 | 0,00017 | 3,49760 |
| Camada 9 | 0,02580 | -0,16063 | 0,42857 | 0,11498 | 0,07508 | 0,74592 | 0,04750 | 0,00017 | 3,49760 |

Ao se analisar os resultados do indicador de sucesso da previsão para o S&P 500, demonstrado na equação [14], todos os resultados foram considerados insatisfatórios e, de acordo com esse critério, os resultados são considerados como desinformação. Todos os demais indicadores, à exceção do percentual de acerto de sinais e da correlação entre os retornos observados e os retornos previstos, degeneraram ao longo das camadas, denotando uma ineficiência do modelo para a previsão daquele índice.

Na figura 2, evidencia-se que a modelagem foi ineficiente para a previsão do americano S&P 500. Apesar dos 112.530 neurônios testados através de redes polinomiais *Group Method of Data Handling* (GMDH) com até nove camadas, os resultados foram insatisfatórios para as previsões em $t+1$, fato que pode ser explicado pela maturidade do mercado de capitais americano.

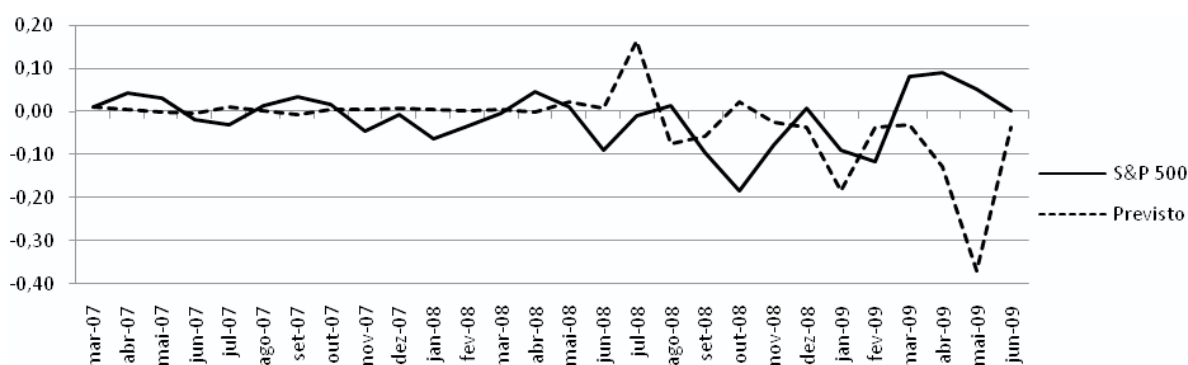


Figura 2 - Comparativo entre o retorno esperado e explicado para o índice S&P500.

Na tabela 3, são demonstrados os resultados das previsões do mercado brasileiro, representado pelo índice IBOVESPA. Nessa modelagem foram testados 132.990 neurônios em redes com até nove camadas, semelhante ao S&P500.

Tabela 3 – Resultados das previsões para o índice Ibovespa

| Camada | R^2 | Correlação | Sinais | EMQ | MAE | U | U^M | U^S | Ivakhnenko |
|----------|---------|------------|---------|---------|---------|---------|---------|---------|------------|
| Camada 1 | 0,23387 | 0,48360 | 0,67857 | 0,08095 | 0,06372 | 0,71192 | 0,02761 | 0,00237 | 0,83473 |
| Camada 2 | 0,25718 | 0,50713 | 0,67857 | 0,07923 | 0,06408 | 0,68947 | 0,01530 | 0,00217 | 0,79962 |
| Camada 3 | 0,27278 | 0,52228 | 0,67857 | 0,07861 | 0,06389 | 0,68035 | 0,01501 | 0,00210 | 0,78708 |
| Camada 4 | 0,17357 | 0,41661 | 0,67857 | 0,08130 | 0,06656 | 0,69851 | 0,01096 | 0,00200 | 0,84191 |
| Camada 5 | 0,21512 | 0,46381 | 0,67857 | 0,07949 | 0,06516 | 0,67049 | 0,00932 | 0,00178 | 0,80477 |
| Camada 6 | 0,40904 | 0,63956 | 0,67857 | 0,06928 | 0,06087 | 0,52015 | 0,00182 | 0,00084 | 0,61138 |
| Camada 7 | 0,41157 | 0,64153 | 0,67857 | 0,06915 | 0,06064 | 0,51891 | 0,00184 | 0,00084 | 0,60915 |
| Camada 8 | 0,41069 | 0,64085 | 0,67857 | 0,06920 | 0,06075 | 0,51943 | 0,00170 | 0,00084 | 0,61001 |
| Camada 9 | 0,41069 | 0,64085 | 0,67857 | 0,06920 | 0,06075 | 0,51943 | 0,00170 | 0,00084 | 0,61001 |

Ao se analisar os resultados das previsões em $t+1$ para o índice IBOVESPA, no teste proposto por Ivakhnenko e Ivakhnenko (1995, p. 532), percebe-se que houve uma evolução significativa nos resultados na medida em que foram sendo incrementadas novas camadas e todos os resultados foram considerados como satisfatórios. Esse fato pode ser evidenciado na evolução de todos os demais indicadores e, em especial, é possível se destacar o R^2 que evoluiu de 0,23 para a primeira camada para 0,41 para a última camada, demonstrando a eficiência da modelagem polinomial, do tipo GMDH, na previsão desse índice. A correlação entre os valores observados e os valores previstos evoluiu durante a criação da rede, corroborando com a eficiência da rede em prever o IBOVESPA.

Na figura 3, evidencia-se que o sucesso da previsão do Ibovespa pode ser parcialmente explicado pelo acerto dos sinais (indicando queda ou aumento do retorno em $t+1$), visto que os retornos previstos variaram próximos de zero.

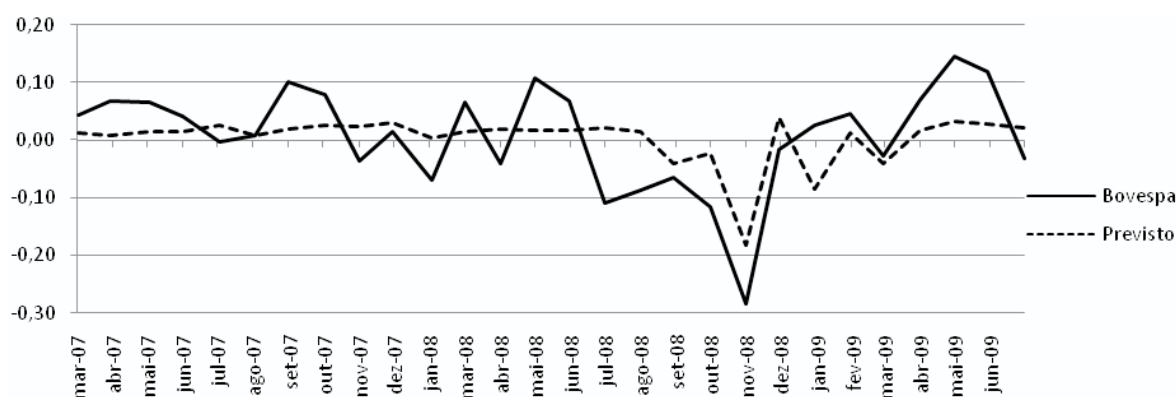


Figura 3 - Comparativo entre o retorno esperado e explicado para o índice BOVESPA.

Na tabela 4, evidencia-se que o método foi mais eficaz para encontrar não-aleatoriedade para o índice brasileiro (IBOVESPA), fato que pode ser parcialmente explicado pela menor volatilidade do retorno mensal deste índice, quando comparado com o S&P500. A alta volatilidade dos log-retornos do índice americano e a maturidade deste mercado explica, parcialmente, a degeneração dos resultados das previsões à medida que as camadas da rede foram adicionadas na rede.

Tabela 4 – Volatilidade mensal dos retornos e resultados das previsões dos índices

| País | Volatilidade | R^2 | Correlação | Sinais | EMQ | MAE | U | U^M | U^S | Ivakhnenko |
|--------|--------------|--------|------------|--------|--------|--------|--------|--------|--------|------------|
| Brasil | 0,1052 | 0,4107 | 0,6409 | 0,6786 | 0,0048 | 0,0608 | 0,0360 | 0,0017 | 0,0008 | 0,6100 |
| EUA | 0,3635 | 0,0258 | -0,1606 | 0,4286 | 0,1150 | 0,0751 | 0,7459 | 0,0475 | 0,0002 | 3,4976 |

5 Considerações Finais

As redes neurais, em especial as redes auto-organizáveis, apresentam uma capacidade de encontrar padrões em dados ruidosos e imprecisos e aparentemente desconexos. Essa característica poderá ser utilizada em pesquisas financeiras, em especial de séries temporais. Nesse sentido, a tentativa de utilização do método Group Method of Data Handling (GMDH) neste trabalho apresentou resultados significativos na previsão do índice IBOVESPA e não-significativos para a previsão do S&P500, fato que pode ser explicado pela alta volatilidade dos log-retornos e a maturidade do mercado de capitais americano. A previsibilidade do índice brasileiro pode ser explicada pela menor volatilidade e uma menor maturidade do mercado de capitais brasileiro.

Raros estudos brasileiros utilizam o método GMDH, em que a maioria desses estudos estão relacionados à previsão de vazão de rios ou previsão de falhas em equipamentos. Espera-se que a melhor contribuição deste artigo tenha sido a apresentação do método como alternativa para modelagens de previsão, especialmente no que se refere à previsão dos sinais (acréscimo ou decréscimo dos retornos). A principal força da modelagem GMDH pode ser expressa pela excelência na previsão do IBOVESPA em $t+1$, sendo ineficaz em modelagens para períodos posteriores a $t+1$. Uma das fraquezas pode ser evidenciada na dificuldade do método em encontrar padrões em dados altamente voláteis, como o americano.

Como sugestão, indica-se o uso de modelos com aprendizagem por correção de erros ou aprendizagem baseada em memória ou, ainda, modelos de redes recorrentes,

já que essa modelagem foi feita através de redes alimentadas diretamente com múltiplas camadas. Para a criação de modelagens com aprendizagem por correção, indica-se ainda a regra delta ou regra de Widrow-Hoff. Para a modelagem com aprendizado baseado em memória, orienta-se o uso de conceitos da lógica difusa na determinação de padrões.

Referências Bibliográficas

DE RÉ, Angelita Maria. Um método para identificar características predominantes em empreendedores que obtiveram sucesso utilizando um sistema neurodifuso. 2000. Tese (Doutorado em Engenharia da Produção)-Programa de Pós Graduação em Engenharia da Produção- Universidade Federal de Santa Catarina, Florianópolis, 2000.

FAUSETT, L.V. Fundamentals of neural networks: architectures, algorithms and applications. New Jersey: Prentice Hall International, 1995.

GOMES, Luana Ferreira Costa. Previsão de vazões naturais diárias ao reservatório de três marias usando a técnica de redes neurais. Dissertação - Universidade Federal de Pernambuco. CTG. Departamento de Engenharia Civil, 2006.

HAIR, J. F. et al. Análise multivariada de dados. 5. ed. Porto Alegre: Bookman, 2005.

HAYKIN, Simon. Redes neurais: princípios e prática. 2. ed. Porto Alegre: Bookman, 2001.

IVAKNENKO, A. G. Polynomial theory of complex systems. IEEE Transaction on Systems, Man, Cybernetics, v. 1, p. 364-378, Oct. 1971.

IVAKNENKO, A. G.; IVAKNENKO, G.A.; MÜLLER, J.A. Self-organization of optimum physical clustering of the data sample for weakened description and

forecasting of fuzzy objects. Pattern Recognition and Image Analysis, v. 3, n. 4, p. 415-422, 1993.

IVAKNENKO, A. G.; IVAKNENKO. The review of problems solvable by algorithms of the Group Method of Data Handling (GMDH). Pattern Recognition and Image Analysis, v. 5, n. 4, p. 527-535, 1995.

KOVÁCS, Zsolt L. Redes neurais artificiais: fundamentos e aplicações. Editora Livraria Física, 2006.

PINDYCK, Robert S.; RUBINFELD, Daniel L. Econometria: modelos e previsões. Rio de Janeiro: Elsevier, 2004.

SCHNEIDER, Sebastian; STEINER, Manfred. Conditional asset pricing: predicting time varying beta-factors with group method of data handling methods. SSRN-ID667468. 2006.

SOARES, Rodrigo Oliveira; ROSTAGNO, Luciano Martin; SOARES, Karina Talamini Costa. Estudo de evento: o método e as formas de cálculo do retorno anormal. Enanpad, 2002.

TSAY, R. S. Analysis of financial time series. New York: Wiley, 2002.

TOWELL, G.; SHAVLIK, J. Knowledge-based artificial neural networks. Artificial Intelligence, v.70, n.1-2, p.119-165, 1994.

VALENÇA, Mêuser. Aplicando redes neurais: um guia completo. Olinda: Livro Rápido, 2005.

VALENÇA, M.; LUDERMIR, T. Redes compostas por blocos de regressões sigmóides não-lineares: uma eficiente rede de alta ordem com aplicações na previsão de séries temporais, V Congresso Brasileiro de Redes Neurais, pp. 67-72. Rio de Janeiro, 2001.