

Aplicação de Redes Neurais Polinomiais GMDH na Previsão do Índice Ibovespa

Everton Anger Cavalheiro

Mestre em Administração pela Universidade Federal de Santa Maria
Professor da Universidade de Cruz Alta
eacavalheiro@hotmail.com

Kelmara Mendes Vieira

Doutora em Administração pela universidade federal do Rio Grande do Sul,
Professora Adjunta da Universidade Federal de Santa Maria
kelmara@smail.ufsm.br

Paulo Sergio Ceretta

Doutor em Engenharia de Produção pela Universidade Federal de Santa Catarina
Professor Adjunto da Universidade Federal de Santa Maria
paulo.ceretta@pq.cnpq.br

Resumo: Este trabalho tem como objetivo analisar a eficiência das redes neurais polinomiais Group Method of Data Handling (GMDH) na previsão do índice Ibovespa como forma de contribuir para o aprofundamento das técnicas utilizadas na previsão de séries temporais financeiras. Secundariamente, procurou-se a disseminação desse tipo de rede neural através de um passo a passo na aproximação indutiva do retorno logaritmo mensal desse índice. Os resultados das redes demonstram uma satisfatória previsibilidade para o mercado acionário brasileiro em que sua fortaleza mostrou-se especialmente na previsão dos sinais (acréscimo ou decréscimo dos retornos). A principal força da modelagem GMDH, neste estudo, pode ser expressa pela excelência na previsão do Ibovespa em $t+1$, sendo ineficaz em modelagens para períodos posteriores a $t+1$.

Palavras-chave: previsão; redes neurais; séries temporais.

APPLICATION OF NEURAL NETWORKS IN POLINOMIAIS GMDH BOVESPA INDEX FORECAST

Abstract

This study aimed to analyze the efficiency of neural networks polynomial Group Method of Data Handling (GMDH) in predicting the Ibovespa index as a contribution to developing the techniques used in forecasting financial time series. Secondly we tried to spread this type of neural network through a step-by-step approach in inductive return index. Such logarithm of monthly results of the networks show a satisfactory predictability for the Brazilian stock market in which their strength was especially shown in the forecast signal (increase or decrease in returns). The main strength of the GMDH model, in this study, can be expressed for excellence in forecasting the Bovespa index in $t+1$, being ineffective in modeling for subsequent periods at $t+1$.

Keywords: forecast; neural networks; time series.

1 Introdução

Ferson (2007) cita que o interesse pela previsão dos preços dos retornos das ações é provavelmente tão antigo quanto os próprios mercados, e passam uma vasta e significativa literatura sobre o assunto. Nesse sentido, o uso de redes neurais em modelos de previsão de séries temporais financeiras tem se mostrado cada vez mais presente na literatura e nas modelagens empíricas internacionais.

De acordo com Braga, Carvalho e Ludemir (2000, p. 2) a capacidade de redes neurais artificiais (RNA) em aprender através de exemplos e de generalizar a informação aprendida é principal o atrativo para a aplicação dessas modelagens na solução de problemas. A generalização que está associada à capacidade da rede

aprender através de um conjunto reduzido de exemplos e posteriormente dar respostas coerentes para dados não-conhecidos é uma demonstração de que a capacidade das RNAs vai muito além do que simplesmente mapear relações de entrada e saída.

Braga, Carvalho e Ludemir (2000, p. 2) comentam que as RNAs são capazes de extrair informações não-apresentadas de forma explícita através dos exemplos. Não obstante, as RNAs são capazes de atuar como mapeadores universais de funções multivariáveis, com custo computacional que cresce apenas linearmente com o número de variáveis. Outra característica importante é a capacidade de auto-organização e de processamento temporal, que, aliada às demais, faz das RNAs uma ferramenta computacional extremamente poderosa e atrativa para a solução de problemas complexos.

Um ótimo exemplo de complexidade é o mercado de capitais, especialmente a bolsa de valores. A alta complexidade destes mercados e a eficiência das redes neurais em mapeá-los têm levado diversos pesquisadores internacionais a divulgarem suas pesquisas, contudo raros estudos demonstram a eficiência das redes neurais artificiais Group Method of Data Handling (GMDH) na previsão de séries temporais financeiras, entre os quais se destacam Sarycheva (2003) e Zaychenko e Zayets (2003). Nesse sentido, o problema de pesquisa abordado neste trabalho é: "as redes polinomiais GMDH são suficientemente capazes de demonstrar não aleatoriedade no retorno mensal do índice Ibovespa?"

Dessa forma, este trabalho analisa a eficiência das redes neurais polinomiais GMDH na previsão do índice Ibovespa como forma de contribuir para o aprofundamento das técnicas utilizadas na previsão de séries temporais financeiras. Secundariamente, procurou-se a divulgação desse tipo de rede neural pouco difundida em pesquisas empíricas nacionais.

2 Revisão bibliográfica

Inicialmente esse artigo apresentará uma revisão do conceito de redes neurais, bem como as formas de estrutura da rede e de aprendizado existentes para o conjunto de dados. Após, será 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, 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. Nesse sentido, 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 utilizados para armazenar o conhecimento adquirido.

De acordo com Braga, Carvalho e Ludemir (200, p. 1) redes neurais artificiais (RNA) são sistemas paralelos distribuídos compostos por unidades de processamento simples (nodos) que calculam determinadas funções matemáticas (normalmente não-lineares). Tais unidades são dispostas em uma ou mais camadas e interligadas por um grande número de conexões, geralmente unidirecionais. Na maioria dos modelos, essas conexões estão associadas a pesos os quais armazenam o conhecimento representado no modelo e servem para ponderar a entrada recebida por cada neurônio da rede. O funcionamento destas redes é inspirado em uma estrutura física concebida pela natureza: o cérebro humano.

O primeiro modelo artificial de um neurônio biológico foi fruto do trabalho pioneiro de Warren McCulloch e Walter Pitts em 1943. McCulloch, psiquiatra e neuroanatomista, dedicou vinte anos na tentativa de representar um evento no sistema nervoso. Pitts, um matemático recém-graduado, juntou-se a ele em 1942. O primeiro trabalho publicado nesta área foi chamado "A Logical Calculus of the Ideas Immanent in Nervous Activity".

Contudo, o aprendizado de redes biológicas e artificiais veio a ser objeto de estudo somente alguns anos depois do trabalho de McCulloch e Pitts. O primeiro trabalho que se tem notícia que tem ligação direta com o aprendizado foi apresentado por Donald Hebb, em 1949. Hebb mostrou como a plasticidade da aprendizagem de redes neurais é conseguida através da variação dos pesos de entrada dos nodos. Ele propôs uma teoria para explicar o aprendizado em nodos biológicos baseada no reforço das ligações sinápticas entre nodos excitados. A regra de Hebb foi interpretada do ponto de vista matemático, e é hoje utilizada em vários algoritmos de aprendizado. Mais tarde, Widrow e Hoff sugeriram uma regra de aprendizado, conhecida como regra de Widrow-Hoff, ou regra delta, que é ainda hoje bastante utilizada. Esta, por sua vez, é baseada no método do gradiente para minimização do erro na saída de um neurônio com resposta linear (BRAGA, CARVALHO e LUDEMIR, 2000).

As RNA 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. Existem duas formas principais de aprendizado: a supervisionada e a não supervisionada.

2.2 Aprendizado

Uma definição geral do que vem a ser aprendizagem pode ser expressa da seguinte forma: aprendizagem é o processo pelo qual os parâmetros de uma rede neural são ajustados através de uma forma continuada de estímulo pelo ambiente no qual a rede está operando, sendo o tipo específico de aprendizagem

realizada definido pela maneira particular como ocorrem os ajustes realizados nos parâmetros. (MENDEL e McLAREN, 1970).

Braga, Carvalho e Ludemir (2000) comentam que as redes neurais artificiais possuem a capacidade de aprender por exemplos e fazer interpolações e extrapolações do que aprenderam. Um conjunto de procedimentos bem-definidos para adaptar os parâmetros de uma RNA para que a mesma possa aprender uma determinada função é chamado de algoritmo de aprendizado.

Diversos métodos para treinamento de redes foram desenvolvidos, podendo ser agrupados em dois paradigmas principais: aprendizado supervisionado e aprendizado não-supervisionado. Outros dois grupos bastante conhecidos são os de aprendizado por reforço (que é um caso particular de aprendizado supervisionado) e aprendizado por competição (que é um caso particular de aprendizado não-supervisionado).

2.2.1 Aprendizado supervisionado

Este método de aprendizado é o mais comum no treinamento das RNA. É chamado de aprendizado supervisionado porque a entrada e saída desejadas para a rede são fornecidas por um supervisor (professor) externo. O objetivo é ajustar os parâmetros da rede, de forma a encontrar uma ligação entre os pares de entrada e saída fornecidos. 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.

Braga, Carvalho e Ludemir (2000) citam que o professor indica explicitamente um comportamento bom ou ruim para a rede visando a direcionar o processo de treinamento. A rede tem sua saída corrente (calculada) comparada com a saída desejada, recebendo informações do supervisor sobre o erro da resposta atual. A cada padrão de entrada submetido à rede, compara-se a resposta desejada (que representa uma ação ótima para ser realizada pela rede) com a resposta calculada, ajustando-se os pesos das conexões para minimizar o erro. A minimização da diferença é incremental, já que pequenos ajustes são feitos nos pesos a cada etapa de treinamento, de tal forma que estes caminhem - se houver solução possível - para uma solução. A soma dos erros quadráticos de todas as saídas é normalmente utilizada como medida de desempenho da rede e também como função de custo a ser minimizada pelo algoritmo de treinamento.

A desvantagem do aprendizado supervisionado é que, na ausência do professor a rede não conseguirá aprender novas estratégias para situações não cobertas pelos exemplos do treinamento da rede. Os exemplos mais conhecidos de algoritmos para aprendizado supervisionado são a regra delta e a sua generalização para redes de múltiplas camadas, o algoritmo backpropagation (RUMELHART, HINTON e WILLIAMS, 1986).

2.3 Estruturas de redes

Uma rede neural consiste num conjunto de neurônios interligados. O tipo de ligações entre as unidades de processamento que constituem uma rede define a sua estrutura. Assim sendo, as redes neuronais podem ser divididas em duas classes principais: as redes com ligações para frente e as redes recorrentes.

2.3.1 Redes com ligações para frente

Nesta classe de redes neurais incluem-se estruturas como a Perceptron Multicamada (RUMELLHART e McCLELLAND, 1986) e as redes com funções de base radial (BROOMHEAD e LOWE, 1988; MOODY e DARKEN, 1989), as quais são largamente utilizadas em problemas de identificação de padrões e controle. Dessa classe constam ainda estruturas como as redes Group Method of Data Handling (GMDH).

2.3.2 Redes recorrentes

As redes recorrentes contêm ligações para trás e/ou ligações entre neurônios da mesma camada. Assim, o fluxo dos sinais é bidirecional. Ao contrário das redes com ligações para frente, nessa classe as propriedades dinâmicas da rede revestem-se de grande importância. De fato, as redes recorrentes contêm memória dinâmica, pelo que as suas saídas num dado instante refletem o efeito da entrada corrente, assim como de entradas e saídas passadas. Em virtude dessa propriedade, estruturas como a rede de Elman (ELMAN, 1990) ou a rede de Jordan (JORDAN, 1986) são utilizadas na modelagem, de sistemas dinâmicos (HENRIQUES e DOURADO, 1998). Outra rede incluída nessa classe é a rede de Hopfield (HOPFIELD, 1982), a qual funciona como memória associativa, sendo também utilizada em problemas de otimização conduzidos por uma função objetivo.

2.4 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.

De acordo com Liu, Lee e Tarn (2000) as redes neurais polinomiais propostos por Ivakhnenko (1971) são um método de

manipulação de dados em que sistemas complexos são decompostos em unidades menores e mais simples, sendo agrupados em várias camadas. Os insumos (inputs) da rede são manipulados (realizadas combinações) para em seguida serem enviadas aos nós da rede. Esses nós avaliam um número limitado de entradas por uma função polinomial e geram uma saída para servir de entrada para os nós subsequentes da camada seguinte. A metodologia geral, segundo os autores, trata de lidar com um número limitado de entradas em um momento, então resumindo esta informação, e depois a transmitindo a um nível mais elevado de raciocínio. Tal procedimento está diretamente relacionado com comportamento humano, conforme observado por Miller (1956). Por isso, as redes polinomiais podem ser reconhecidas como uma classe especial das redes neurais biologicamente inspiradas e pode ser utilizada efetivamente como uma forma de prever estimas para sistemas complexos (LIU, LEE, TARN, 2000).

2.4.1 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 desdobrar-se da seguinte forma:

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

onde y é a saída estimada, x_1, x_2 e x_3 as variáveis de entrada, β_i seus pesos e ξ 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 3 entradas de neurônios, a primeira camada escondida será de $3(3-1)/2=3$, onde os pesos β_i desses neurônios da camada escondida serão determinados pelo conjunto de dados da fase de treinamento α . 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.

2.4.2 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 $C_0 = 10$ 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 $C_0^2 = 45$ equações parciais nesse segundo passo.

Os C_0 neurônios serão construídos até que no último passo obtenha $C_0 = 1$ 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 desdobraram em 10 variáveis, haverá $M = 2^{10} - 1 = 1.023$ neurônios (nós) totalmente diferentes.

2.4.3 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_1 e x_2 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 , 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 desse trabalho optou-se pela utilização de uma rede neural supervisionada, com algoritmo de seleção para frente. Apesar das redes recorrentes serem reconhecidas como uma estratégia mais robusta, a plataforma de cálculo utilizada (Excel) não permite este tipo de manipulação. 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

Nesse estudo foi utilizada como variável exógena o retorno logarítmico do índice Ibovespa. A escolha deste índice deu-se pelo fato de ser considerado o mais importante indicador do desempenho médio das cotações do mercado de ações brasileiro. 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.

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. Nesse 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. Entre tais propriedades, pode-se citar 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 .

3.1.1 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:

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

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

3.1.2 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 3.1.2. Na equação [12], é apresentado o modelo de cálculo do total de combinações que serão testadas nessa camada:

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

onde 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 foi realizado 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 contribuindo 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. De acordo com Hair et al (2005), 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), 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, e os menores resultados foram considerados os melhores. Na equação [13], é demonstrado o método de cálculo do PSE:

$$PSE = \left[\frac{\sum_1^N (y_i - \hat{y}_i)^2}{n-2} \right] + 2\sigma_\varepsilon^2 (p/n) \rightarrow \min, \quad [13]$$

onde 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 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).

$$\delta_i^2 = \frac{\sum_1^N (y_i - \hat{y}_i)^2}{\sum_1^N (y_i - \bar{y})^2} \rightarrow \min, \quad [14]$$

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

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]:

$$R^2 = 1 - \frac{\sum_1^N (\hat{y}_i)^2}{\sum_1^N (y_i - \bar{y})^2}, \quad [15]$$

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

$$MSE = \sqrt{\frac{1}{N} \sum_1^N (y_i - \hat{y}_i)^2}, \quad [16]$$

$$MAE = \frac{1}{N} \sum_1^N \left| \sqrt{y_i^2} - \sqrt{\hat{y}_i^2} \right|, \quad [17]$$

Adicionalmente foi analisado o coeficiente de desigualdade de Theil, também chamado de U . O numerador de U é o MSE, 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].

$$U = \frac{\sqrt{\frac{1}{N} \sum_1^N (y_i - \hat{y}_i)^2}}{\sqrt{\frac{1}{N} \sum_1^N (y_i)^2 + \frac{1}{N} \sum_1^N (\hat{y}_i)^2}}, \quad [18]$$

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

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

$$\lambda = \sqrt{\frac{1}{(N-1) \times \sum_1^N (X - \bar{X})^2}}. \quad [19]$$

Em [19] é a volatilidade mensal dos mercados, N o número de meses, o retorno mensal de cada ativo e a média dos retornos de cada índice.

4 Resultados

Todos os cálculos foram realizados em planilha eletrônica Excel e nenhum software estatístico foi utilizado e o método dos mínimos quadrados foi calculado através de álgebra matricial. Foram calculadas até 9 camadas na rede para cada uma das 28 previsões (somente em $t+1$) dos índices Ibovespa. Um total de 132.990 neurônios foram testados.

4.1 Resultados da fase de teste

Na Tabela 1 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.

Tabela 1: Resultados das previsões para o índice Ibovespa

Camada	R^2	Correlação	Sinais	MSE	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 R2 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.

A baixa volatilidade dos log-retornos do índice brasileiro (0,1052) e o grau de maturidade deste mercado explica, parcialmente, o aumento da capacidade de previsão do método à medida em que as camadas foram adicionadas na rede, conforme pode ser evidenciado na Tabela 1 em que, apesar do número de acerto de sinais não ter variado, a correlação e o R2 aumentaram, demonstrando que o grau de explicação da variância total dos dados através dessas variáveis exógenas aumentou significativamente durante o incremento da complexidade dos modelos. Na Figura 1, é demonstrado o Plot do gráfico do retorno logarítmico real e do previsto do índice Ibovespa.

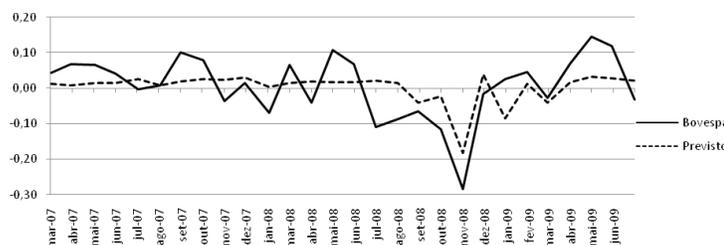


Figura 1 – Comparativo entre o retorno esperado e explicado para o índice BOVESPA.

Na Figura 1, 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.

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 nesse trabalho apresentou resultados significativos na previsão do índice Ibovespa. A previsibilidade do índice brasileiro pode ser explicada por uma menor maturidade do mercado de capitais brasileiro e pela boa capacidade do método em prever o acréscimo e decréscimo do retorno deste índice.

Raros estudos brasileiros utilizam o método GMDH, em que a maioria destes estudos estão relacionados à previsão de vazão de rios ou previsão de falhas em equipamentos. Espera-se que as melhores contribuições deste artigo tenham 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), bem como o passo a passo deste promissor algoritmo. 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 acompanhar toda a volatilidade do retorno deste índice.

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

- BRAGA, A. P.; CARVALHO, A. C. P. L. F.; LUDEMIR, T. B. Redes Neurais Artificiais: teoria e aplicações. 1 ed. Rio de Janeiro: LTC, 2000.
- BROOMHEAD, D. S.; LOWE, D. Multivariable function interpolation and adaptive networks. Complex Systems, Vol. 2, pp. 321-355, 1988.
- de RÉ, A. M. Um método para identificar características predominantes em empreendedores que obtiveram sucesso utilizando um sistema neurodifuso. Tese (Doutorado em Engenharia da Produção) Programa de Pós Graduação em Engenharia da Produção. Universidade Federal de Santa Catarina, 2000.
- ELMAN, J. L. Finding structure in time, Cognitive Science, vol. 14, pp. 179-211, 1990.
- FERSON, W. in Forecasting Expected Returns in the Financial Markets, Ed. Elsevier, San Diego, Califórnia, 2007.
- HAIR, J. F. et al. Análise multivariada de dados. 5ª ed. Porto Alegre: Bookman, 2005.
- HAYKIN, S. Redes Neurais: princípios e prática. Segunda edição, Editora Bookman. Porto Alegre, 2001.
- HENRIQUES, J.; DOURADO, A. Adaptive control using a recurrent neural network observer. Proceedings of the 3rd Portuguese Conference on Automatic Control, Coimbra, pp. 583-589, 1998.
- HOPFIELD, J. J. Neural networks and physical systems with emergent collective computational abilities. Proceedings of the National Academy of Sciences, Vol. 79, pp. 2554-2558, 1982.
- IVAKHNENKO, A. G. Polynomial theory of complex systems. IEEE Transaction on Systems, Man, Cybernetics, Vol. 1, pp 364-378, October, 1971.
- IVAKHNENKO, A. G., IVAKHNENKO, G.A., e MULLER, 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, 1993, vol. 3, no. 4, pp. 415-422.
- IVAKHNENKO, A. G., IVAKHNENKO. The Review of Problems Solvable by Algorithms of the Group Method of Data Handling (GMDH). Pattern Recognition and Image Analysis, 1995 Vol. 5, No. 4, 1995,

JORDAN, M. I. Attractor dynamics and parallelism in connectionist sequential machines. Proceedings of the 8th Annual Conference of the cognitive Science Society, pp. 531-546. 1986.

LIU, H. S.; LEE, B. Y.; TARNG, Y. S. in Process prediction of corner wear in drilling operations. Journal of Materials Processing Technology, V. 101, pp. 152-158, Abril 2000.

MENDEL, M.; McLAREN, R. W. Adaptive, Learning, and Pattern Recognition Systems: Theory and Applications, chapter Reinforcement-learning control and pattern recognition systems, pages 287-318. New York: Academic Press, 1970.

MILLER, G.A. The magic number seven, plus or minus two: some limits on our capacity for processing information. Psychol. Rev. 63 81-97, 1956.

MOODY, J. E.; DARKEN, C. J. Fast learning in networks of locally-tuned processing units. Neural Computation, Vol. 1, pp. 281-294, 1989.

PINDYCK, R. S.; RUBINFELD, D. L. Econometria: Modelos e previsões. Editora Elsevier. Rio de Janeiro, 2004.

RUMELHART, D. E.; HINTON, G.E. e WILLIAMS, R.J. Learning representations by backpropagating errors. Nature, 323:533-536, 1986.

RUMELHART, D. E.; McCLELLAND, J.L. Parallel Distributed Processing: Explorations in the Microstructure of Cognition. MIT Press, Cambridge, MA 1: 45-76, 1986.

SARYCHEVA, L. Using GMDH for modeling economical indices of mine opening. Systems Analysis Modelling Simulation, v.43 n.10, p.1341-1349, October 2003.

SCHNEIDER, S.; STEINER, M. Conditional Asset Pricing: Predicting time varying Beta-Factors with Group Method of Data Handling Methods. SSRN-ID667468. 2006.

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

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

ZAYCHENKO, Y. P.; ZAYETS, I. O. The fuzzy group method of data handling and its application to the tasks of the macroeconomic indexes forecasting. Systems Analysis Modelling Simulation, v.43 n.10, p.1321-1329, October 2003