

# Uma estratégia automatizada de investimento por meio de redes neurais artificiais e preditores econométricos

Alternative Title: An automated investment strategy using artificial neural networks and econometric predictors

Ciniro A. L. Nametala  
Programa de Pós-Graduação  
em Engenharia Elétrica  
Universidade Federal de  
Minas Gerais  
ciniro@ifmg.edu.br

Alexandre Pimenta  
Programa de Pós-Graduação  
em Engenharia Elétrica  
Universidade Federal de  
Minas Gerais  
alexandre@ifmg.edu.br

Adriano César M. Pereira  
Departamento de Ciência da  
Computação  
Universidade Federal de  
Minas Gerais  
adrianoc@dcc.ufmg.br

Eduardo G. Carrano  
Departamento de Engenharia Elétrica  
Universidade Federal de Minas Gerais  
carrano@cpdee.ufmg.br

## RESUMO

Sistemas de informação que tem por objetivo realizar previsões em séries temporais financeiras e negociar a partir destas estão sujeitos a diversos riscos, pois o mercado acionário sofre influências de diferentes origens continuamente. O estudo das finanças quantitativas aborda métodos para tratamento de problemas desta natureza, fato que se dá, principalmente, por meio do uso de inteligência computacional. Neste trabalho é apresentada uma estratégia automatizada (robô investidor) que combina previsões feitas por redes neurais artificiais e preditores econométricos em uma segunda rede neural, esta age como um comitê de decisão. As previsões são utilizadas para gerar sinais de compra ou venda por meio de um modelo de negociação incorporado ao algoritmo. Os experimentos foram conduzidos com séries reais referentes a três ações de alta liquidez, uma *commodity* e um índice de mercado. Os resultados obtidos são comparados de forma financeira frente à aplicação individual de cada preditor e, também, a técnicas clássicas de mercado.

## Palavras-Chave

Sistemas de informação inteligentes, finanças quantitativas, econometria de séries temporais, redes neurais artificiais.

## ABSTRACT

Information systems with the objective to make forecasts for financial time series and negotiate from these are subject to various risks, because the stock market is influenced by diffe-

rent sources continuously. The study of quantitative finance addresses methods for treating problems such as these, a fact which occurs mainly through the use of computational intelligence. This paper presents an automated strategy (investor robot) that combines predictions made by artificial neural networks and econometric predictors in a second neural network, this acts like an ensemble. The predictions are used to generate purchase or sell signals through a negotiation model built into the algorithm. The experiments were conducted with real series of three assets with high liquidity, a commodity and a market index. The financial results are compared against the individual application of each predictor and also the classical market techniques.

## Categories and Subject Descriptors

I.2.6 [Artificial Intelligence]: Learning - *connectionism and neural nets*; J.1 [Computer Applications]: Administrative Data Processing - *financial*

## General Terms

Algorithms, Economics, Experimentation

## Keywords

Intelligent information systems, quantitative finance, econometrics time series, artificial neural networks.

## 1. INTRODUÇÃO

Métodos para predição de séries temporais lidam com incertezas. Isso ocorre, pois seja no domínio temporal ou de frequências, modelos probabilísticos e estocásticos baseiam-se na aleatoriedade de uma variável num certo período de tempo, logo, seu valor num próximo instante não pode ser precisamente determinado, somente estimado [7].

Morettin [15], no âmbito das séries especificadamente financeiras, cita que estas apresentam algumas características de difícil modelagem como tendências, sazonalidade, pontos influentes (atípicos), heterocedasticidade condicional e não

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SBSI 2016, May 17th-20th, 2016, Florianópolis, Santa Catarina, Brazil  
Copyright SBC 2016.

linearidades. Este último ponto ele destaca citando que séries financeiras podem responder de maneiras diferentes a choques pequenos ou grandes.

O interesse inicial no estudo de uma série temporal é a determinação de sua classificação como estacionária ou não estacionária [10]. A estacionariedade evidencia que uma série apresenta valores aleatórios situados ao redor de uma média constante no decorrer de um dado período de tempo [15]. Essa característica se faz importante pois séries estacionárias são análogas as séries convergentes da matemática, ao passo que, séries não estacionárias podem ser divergentes, o que dificulta seu estudo [10]. No caso de não estacionariedade, muitas vezes em econometria são aplicadas análises aos resíduos da série e não à série propriamente dita. Para isso são realizadas, por exemplo, diferenciações ou transformações de dados. Isto é necessário, pois somente quando a estacionariedade é constatada, os testes estatísticos envolvidos podem gerar confiança na estimação de valores. Fato que conduz a uma melhor qualidade nas previsões.

Séries de preços, como as utilizadas neste trabalho, são consideradas na maioria das vezes séries não estacionárias [10]. Em razão disso, à modelagem e análise destas cabe à aplicação de técnicas que foquem no tratamento de não linearidades, uma das propostas fundamentais das Redes Neurais Artificiais (RNA). As RNA's são adequadas a este tipo de problema, pois também possuem capacidade de aproximação universal e habilidade de generalização ao estimar soluções para um conjunto de padrões, a princípio, desconhecido [5].

Neste contexto, o trabalho presente propõe a utilização de uma RNA com arquitetura *Perceptron* Múltiplas Camadas (PMC), conjuntamente com preditores econométricos clássicos da análise de séries temporais. Buscou-se a partir destes estimar um conjunto de valores de preço de fechamento que, numa segunda fase, serviram como entrada em outra RNA PMC. Esta foi responsável por combinar ou ativar somente as previsões que tivessem qualidade, ou seja, esta segunda rede atuou como um comitê de decisão. Para avaliação da estratégia foram realizadas comparações de desempenho dos preditores quando aplicados individualmente contra o desempenho da combinação destes feita por meio da segunda RNA. Foram também realizadas comparações dos resultados obtidos frente à técnica de mercado *Buy and Hold* [16] e à operações aleatórias. As métricas utilizadas foram a taxa de acerto e diferentes formas de análise dos rendimentos financeiros percentuais. Utilizou-se, por fim, a linguagem de programação R [20] na implementação dos artefatos de código e a plataforma *MetaTrader 5* [9] para extração de dados.

O trabalho está organizado em 5 seções. A seção 2 aborda de forma sucinta as características dos preditores utilizados e como estes foram parametrizados. A seção 3 descreve a metodologia utilizada para o tratamento de dados, as características das séries envolvidas no estudo, os modelos de previsão e negociação, além da descrição, passo a passo, da operação do algoritmo. A seção 4 expõe os resultados obtidos e, por fim, a seção 5 trata das conclusões.

## 2. PREDITORES UTILIZADOS

Os preditores utilizados foram escolhidos, pois, antes de tudo, permitem a geração de previsões para o preço de fechamento sem receber como entradas obrigatoriamente valores e métricas que não sejam as já existentes no chamado

gráfico de velas [11], em outras palavras, pode-se trabalhar com os dados diretamente sem ser necessário usar indicadores técnicos ou outros tipos de informação que não sejam os valores diários de preços máximo e mínimo, de abertura e fechamento da bolsa e o volume negociado.

Além das RNA's, foram escolhidos, no domínio da econometria de séries temporais, os preditores: SMA - *Simple Moving Average*, ARMA - *Autoregressive Moving Average Model*, ARIMA - *Autoregressive Integrated Moving Average Model* e GARCH - *Generalized Autoregressive Conditional Heteroskedasticity* (como abordados em Moretin [15] e Bueno [10]). A seleção destes se deu pelo fato de que, a RNA capta padrões de movimentação não lineares com base no aprendizado supervisionado, a SMA lida com tendências acumuladas ao longo do período estudado, o ARMA pode gerar bons resultados se for constada estacionariedade em alguma parcela da janela de operação, o ARIMA integra a tendência com a autoregressão para possibilitar estudos caso a série não seja estacionária e, por fim, o GARCH incorpora à estratégia a capacidade de lidar com as variâncias passadas de forma condicional, buscando prever a volatilidade futura.

Nas subseções a seguir é feita uma breve conceituação de cada preditor. É exposta também a forma como cada um destes foi configurado.

### 2.1 Redes Neurais Artificiais

RNA's podem ser conceituadas como um conjunto de unidades de processamento, caracterizadas pelos chamados "neurônios artificiais", que são conectadas para permitir as "sinapses artificiais" [18]. Braga [5] e Haykin [19] elencam suas diversas aplicações, com destaque para aproximação de funções, categorização de padrões, classificação de dados, estimação de parâmetros e, no interesse deste estudo, também a previsão em séries temporais.

Foram utilizadas duas RNA's. Uma para previsão e outra para atuar como um comitê que decide qual saída ativar. Trabalhos como o de Hansen and Salaman [14] propõe o uso de comitês de decisão para operarem quando se têm respostas oriundas de vários preditores, no caso, estes os membros de um comitê. Em muitos casos, como quando se entendem as respostas dos preditores como votos de um comitê, o uso de uma rede neural pode apresentar vantagens, pois esta não somente busca selecionar as melhores respostas com base em aprendizado prévio, como também combina saídas. RNA's já foram abordadas na literatura com esta finalidade como no trabalho de Chang [8].

Uma das arquiteturas de RNA mais conhecidas é a *Perceptron* com Múltiplas Camadas (PMC). Esta tem por principal característica a utilização de pelo menos uma camada escondida. Comumente é observada também a utilização de funções sigmóides nos neurônios da camada escondida e, lineares ou sigmóides, nos neurônios da camada de saída. Existem diversas estratégias para correção de erros numa RNA PMC, neste estudo optou-se pelo algoritmo de *back-propagation* como abordado em Bergmeirl [1] por sua simplicidade e desempenho.

A configuração de ambas as redes foi a mesma. Estas atuaram com uma camada escondida e neurônios com funções logísticas. Na camada de saída, foi disposto apenas um neurônio com função linear referente ao preço de fechamento. O critério de parada foi determinado como máximo de 3000 épocas e a taxa de aprendizagem foi valorada em 0.07. Não foi realizada mistura dos parâmetros de entrada antes da

etapa de treinamento, pois se tratando de uma série temporal preferiu-se levar em conta a sequência dos valores como dispostos. Os pesos iniciais das ligações foram escolhidos aleatoriamente no intervalo entre -0.5 e 0.5. As quantidades de neurônios e o tamanho dos conjuntos de treinamento em ambas as redes é otimizada ao longo da execução, conforme método abordado na seção 3.

## 2.2 SMA

Considerando um conjunto de observações anteriores a um dado ponto onde se pretende realizar uma previsão (neste trabalho as contidas em uma janela de treinamento), o uso da SMA consiste de realizar uma média aritmética entre este conjunto de valores, logo pode ser descrita conforme Equação 1:

$$M_t = \frac{z_t + z_{t-1} + \dots + z_{t-r+1}}{r} \quad (1)$$

Onde  $M_t$  representa a estimativa do nível seguinte,  $z$  os elementos da série e  $r$  a quantidade de elementos. Este tipo de média é denominada móvel, pois a cada período uma nova observação sobrepõe a mais antiga [15].

Uma SMA é considerada como um modelo  $MA(1)$  quando assume-se nesta um passo de previsão igual a 1, ou seja, avalia-se a série tomando-se médias valor a valor. Essa generalização é também referenciada na literatura como  $MA(q)$ , onde  $q$  representa o passo.

A média móvel utilizada neste trabalho é simples, de passo  $q = 1$  e não usa qualquer ponderação. A quantidade de períodos é determinada dinamicamente de forma a maximizar os rendimentos gerados em cada ativo pelo modelo de negociação. Foram testadas janelas de tamanhos diferentes na etapa de calibração deste preditor.

## 2.3 ARMA

Modelos de autoregressão baseiam-se em condições de estacionariedade ditadas, principalmente, por valores de autocorrelação observados em uma série. A partir destes é analisada a autocovariância que é utilizada para derivar-se a esperança estatística da série. A generalização é referenciada como modelo  $AR(p)$  onde  $p$  refere-se à ordem. A representação de um modelo autoregressivo pode ser expressa como na Equação 2:

$$X_t = c + \varepsilon_t + \sum_{i=1}^p \varphi_i X_{t-1} \quad (2)$$

Onde  $\varphi_i$  são os parâmetros da série,  $c$  é uma constante e  $\varepsilon_t$  diz respeito ao ruído branco [15]. O modelo ARMA, por sua vez, é a combinação dos processos  $AR(p)$  e  $MA(q)$ , gerando-se assim o  $ARMA(p, q)$ . A expressão combinada é descrita pela Equação 3:

$$X_t = c + \varepsilon_t + \sum_{i=1}^p \varphi_i X_{t-1} + \sum_{i=1}^q \theta_i \varepsilon_{t-1} \quad (3)$$

Este modelo foi preconizado pela primeira vez por Whittle [21] em 1951 e pode ser utilizado tanto para a previsão móvel de séries como para analisar um período em específico. A parametrização do ARMA diz respeito a quantidade de termos autoregressivo  $p$  e a quantidade de termos de médias móveis  $q$ .

Para este trabalho foi utilizada a biblioteca fArma [23] do R que executa as previsões. Os parâmetros  $p$  e  $q$  foram

definidos com base numa função construída para minimizar a métrica *Akaike Information Criterion* (AIC) [2]. Para isso o algoritmo testa combinações possíveis dos parâmetros e a melhor combinação é retornada.

## 2.4 ARIMA

O modelo ARIMA foi proposto por Box & Jenkins [4] em 1973. A sua construção é baseada no modelo ARMA e serve para descrever tanto comportamentos estacionários como não estacionários. O ARIMA possui parametrização de três valores, sendo  $p$ , oriundo do  $AR(p)$ ;  $q$ , oriundo do  $MA(q)$ ; e mais um terceiro,  $d$ , referente à quantidade de diferenciações necessárias aplicadas à série. A representação final é definida como  $ARIMA(p, d, q)$ .

Como no modelo ARMA, a implementação do ARIMA foi feita por meio da linguagem R com recursos da biblioteca fArma [23]. Já a minimização da métrica AIC para definição dos parâmetros  $p$ ,  $d$  e  $q$  foi realizada utilizando-se o algoritmo de otimização BFGS. O BFGS é um algoritmo da família Quase Newton que tem por premissa possibilitar a otimização em problemas irrestritos não lineares por meio da aproximação iterativa da matriz hessiana com correção simétrica de posto dois. Sua formulação como aplicada neste trabalho é a abordada por Broyden [6] e a implementação na linguagem R é a disponível no método nativo *optim*.

## 2.5 GARCH

As primeiras contribuições em modelos que levam em conta também a variância condicional aleatória das séries foram feitas por Engle [12] e Bollerslev [3] na década de 80.

A volatilidade das séries financeiras é de complexa detecção, logo a avaliação do desempenho dos modelos também é difícil. Apesar disso Furriel [13] cita que a volatilidade apresenta características observáveis, como a existência de *clusters* onde a mesma pode ser elevada ou reduzida em um dado período específico na série. Observa-se também que é raro constatar-se divergências repentinas na volatilidade. O que indica que a mesma pode ser estacionária e possuir capacidade de reagir, previsivelmente, quando ocorrem nas séries, altas ou quedas elevadas. O modelo GARCH de Bollerslev é uma extensão do modelo ARCH de Engle. O GARCH é aplicado levando-se em conta que a variância do erro é condicionada. Este relaciona a mesma não apenas com os quadrados dos valores passados de uma série (como no ARCH), mas também, com a sua condicionalidade [15].

Neste trabalho a parametrização do modelo GARCH foi feita com  $p = 1$  e  $q = 1$  de forma associada a um  $ARMA(0, 2)$ . Para tanto utilizou-se o pacote fGarch [22], biblioteca compatível com a linguagem R.

## 3. METODOLOGIA

Para entendimento da operação do algoritmo e dos resultados gerados são comentadas a seguir as séries financeiras analisadas (3.1) e os métodos para previsão e negociação (3.2).

### 3.1 Séries financeiras analisadas

Foi dada preferência às séries que tivessem grande volume de negociação, característica que é conhecida como liquidez. Em mercados de maior de liquidez as ordens enviadas possuem maiores chances de serem executadas em curtos períodos de tempo [17]. Este fato torna a simulação mais próxima das condições de realidade. Levou-se em conta

**Tabela 1: Parte da série analisada para o ativo ITUB4**

Data	Aber.	Max.	Min.	Fech.	Volume
3-1-2011	19.097	19.130	18.888	18.955	73609160
4-1-2011	19.362	20.059	19.272	19.903	152468992
5-1-2011	19.903	20.741	19.737	20.613	135673056
6-1-2011	20.850	21.277	20.722	20.988	134166088
7-1-2011	20.651	20.831	20.353	20.448	69249720

também a diversificação de setores e o tipo de ativo, sendo que, representando empresas foram utilizadas: Companhia Energética de Minas Gerais S.A. (CMIG4), Embraer S.A. (EMBR3) e Itaú Unibanco Holding S.A. (ITUB4); representando um índice foi utilizado o índice BOVESPA (BOVA11) e para um contrato futuro foi utilizada a *commoditie* de Boi Gordo (BGI\$) negociada na Bolsa de Mercadorias & Futuros (BM&F) como derivativo. Nos períodos de operação todos os ativos apresentaram, em pontos distintos de suas séries, sentidos de alta, baixa e lateralização.

Foram utilizados dados reais obtidos por meio da plataforma *MetaTrader 5* [9] com acesso autorizado via conta de pessoa física registrada em uma corretora de valores brasileira. Foram considerados somente os dias úteis, logo um ano comercial na amostra corresponde a 248 *candles*<sup>1</sup> do gráfico de velas. Para todas a séries foi tomada uma data de início aleatoriamente escolhida, sendo esta, 3 de janeiro de 2011. Deste ponto em diante um conjunto adicional de 744 dias foi selecionado em sequencia perfazendo uma janela de estudos possível para até três anos, esta com fim na data de 30 de dezembro de 2013.

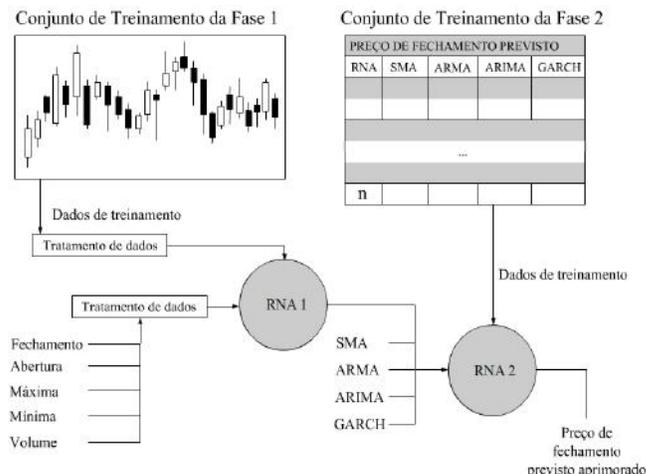
É apresentado na Tabela 1, como exemplo, um período de cinco dias da série de dados referente ao ativo ITUB4.

### 3.2 Método de previsão

O método de previsão foi dividido em duas fases. Na primeira fase é realizado o tratamento dos dados das séries, o treinamento da primeira rede neural (chamada deste ponto em diante de RNA1), a calibração da RNA1 e dos preditores econométricos, além da geração do conjunto de treinamento que serve à próxima rede neural. Na segunda fase ocorre o treinamento da segunda rede neural (chamada deste ponto em diante de RNA2), a calibração da RNA2 e a geração de previsões na janela de operação. As previsões são, por fim, inseridas no modelo de negociação que emite sinais de compra ou venda. A operação do algoritmo pode ser observada de forma geral no diagrama da Figura 1 até a geração da previsão de preço. O modelo de negociação, que usa este preço previsto para dar sinais de compra ou venda é abordado na seção 3.2.1.

Primeiramente tomam-se as dimensões dos *candles* em uma janela de tamanho específico como conjunto de treinamento na Fase 1 para treinar a RNA1, dá-se então como entrada um padrão tratado como no conjunto de treinamento. A saída da RNA1 juntamente com as previsões dos métodos econométricos são usadas para gerar um novo conjunto de treinamento que serve para treinar a segunda rede, onde por fim, fornece-se a previsão para o dia seguinte de cada

<sup>1</sup>O *candle* é uma representação gráfica onde podem ser observadas as dimensões de preço de fechamento, preço de abertura, preço mínimo, preço máximo e volume negociado em períodos de tempo previamente determinados.

**Figura 1: Arquitetura do sistema.**

preditor como segundo padrão de entrada. A saída gerada pela segunda rede é o preço de fechamento aprimorado.

A simulação foi realizada conforme demonstrado na Figura 2. Utilizou-se o ano de 2011 (amostra com 248 observações) como janela de calibração dos preditores. Para essa calibração foram geradas 100 previsões em configurações diferentes em cada preditor. As configurações que retornaram maior rentabilidade no período foram utilizadas para gerar o segundo conjunto de treinamento e também o padrão de entrada na ocasião da previsão. Para cada ativo testou-se com cada preditor as seguintes configurações:

- RNA1: Combinações de 3, 5, 15, 30 e 50 neurônios na camada escondida com janelas de tamanho 5 (uma semana), 10 (duas semanas), 20 (um mês), 35 (um mês e três semanas) e 60 (três meses) dias.
- SMA: Janelas de 9, 21, 30, 45, 60 e 120 dias.
- ARMA: Janelas de 5, 20, 40, 62, 124 e 248 dias.
- ARIMA: Janelas de 62, 124, 186 e 248 dias.
- GARCH: Janelas de 62, 124, 186 e 248 dias.

Com os preditores calibrados com as configurações mais rentáveis foram gerados então conjuntos de treinamento de tamanhos diferentes. Estes foram inseridos na RNA2 que também foi submetida a testes com diferentes configurações. As opções testadas contabilizam 27 combinações distintas envolvendo:

- Conjuntos de treinamento de 5, 15 e 35 dias.
- Tempos de operação sem realização de novo treinamento de 5, 10 e 15 dias.
- Quantidades de neurônios na camada escondida em 5, 10 e 15 unidades.

Por meio da análise de 100 predições geradas pela RNA2 definiu-se a configuração mais rentável em cada ativo. A partir deste ponto foi então feita a previsão de fato na janela de operação para os 248 dias seguintes, ou seja, de novembro de 2012 a outubro de 2013. Neste período a configuração de

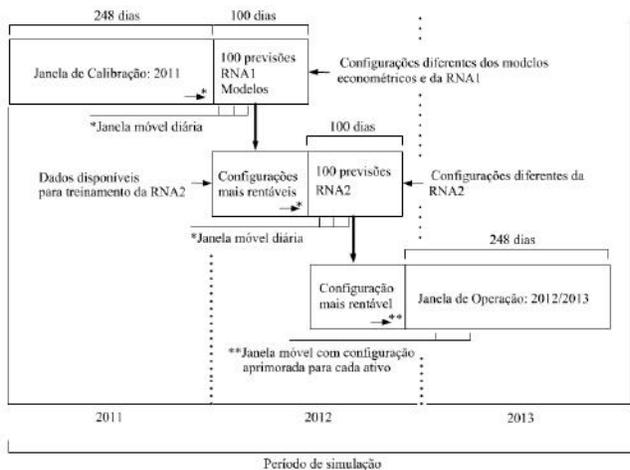


Figura 2: Período de simulação com janelas de calibração e operação

nenhum dos preditores foi alterada, somente as RNA's foram novamente treinadas conforme melhor tempo de operação selecionado pelo algoritmo.

O processo de análise das melhores configurações, tanto nos preditores quanto na RNA2 (que seleciona qual preditor ou combinação de preditores ativar) garante que, na etapa de previsão, sejam utilizadas as configurações mais rentáveis quando aplicadas aos dados históricos.

As subseções a seguir detalham o modelo de negociação e o fluxo dos dados nas fases 1 e 2 do algoritmo.

### 3.2.1 Modelo de negociação

Apesar de gerarem previsões, os preditores econométricos e a rede neural por si só não atuam diretamente no mercado, ou seja, não dão sinais de compra ou venda ao investidor. Para isso é necessário um modelo de negociação.

Existem diversas estratégias para se operar no mercado financeiro e uma delas é a chamada arbitragem. Segundo Siegel [17], a arbitragem consiste em avaliar as discrepâncias temporárias nos preços de séries de bens que sejam idênticos (como é o caso do dólar cheio com o mini dólar) ou bens que estejam altamente correlacionados, como papéis da mesma empresa (PETR3 e PETR4, por exemplo). Caso o valor de um bem caia ou aumente suficientemente em relação ao seu par entende-se que uma oportunidade de operação no mesmo sentido exista no ativo que se está operando.

Foi utilizada a estratégia de arbitragem entre as previsões realizadas por cada um dos preditores e a série de preços real para contabilizarem-se rendimentos no algoritmo. O módulo de negociação pode ser visto no diagrama na Figura 3 e funciona da seguinte forma: Caso a valor previsto para o dia seguinte seja maior que o preço previsto no dia corrente, entende-se que, no dia seguinte haverá uma alta na série real. Ao contrário, caso o valor previsto para o dia seguinte seja menor que o preço previsto no dia corrente, entende-se então que, no dia seguinte haverá uma baixa no preço real. Com os sinais de alta e baixa definidos parte-se para a etapa de operação com lançamento de ordens.

No caso de um sinal de alta no dia seguinte, executa-se uma ordem de compra à mercado na abertura do pregão. Ao

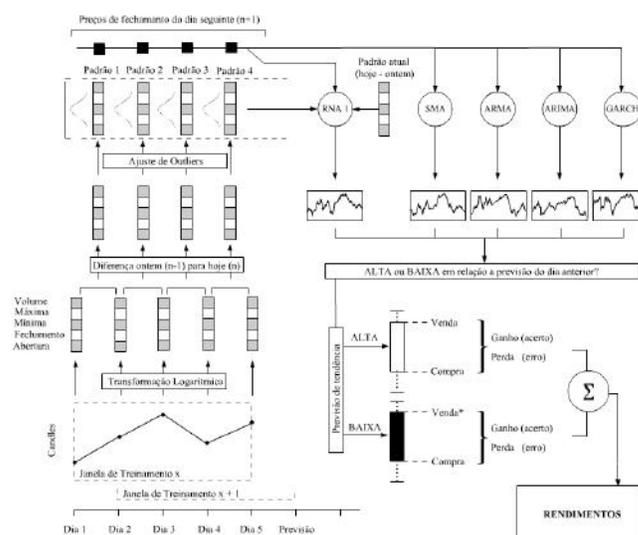


Figura 3: Tratamento de dados, calibração dos preditores e modelo de negociação

fim do dia, independentemente do valor no fechamento do pregão, fecha-se posição realizando-se uma ordem de venda. A situação é semelhante quando o sinal é de baixa: Na abertura executa-se uma venda descoberta (para isso é necessária geralmente uma garantia financeira na corretora) e no fim do dia fecha-se a posição realizando uma compra para devolução do ativo emprestado a bolsa.

Como se pode perceber, os rendimentos correspondem ao somatório dos lucros e prejuízos tidos em todas as operações, que neste caso contabilizam um total de 40 em um mês com 20 dias úteis. É um sistema de operações *daytrade*, pois se opera todos os dias, duas vezes. Quando o algoritmo tem um acerto no tipo de operação, o tamanho completo do *candle* é integralizado como lucro, caso contrário, como prejuízo. Isso ocorre pois opera-se na diferença existente entre o preço de abertura e o preço de fechamento.

### 3.2.2 Fase 1: Tratamento de dados e calibração de preditores

A Fase 1 trata das rotinas responsáveis pela calibração dos preditores. Para isso, realiza-se nesta etapa um tratamento de dados prévio antes de testarem-se as diferentes configurações da RNA1. Nos preditores econométricos SMA, ARMA, ARIMA e GARCH o tratamento é dispensado, pois se têm como entradas, diretamente, as séries de preço de fechamento históricas.

Quanto a RNA1, a cada uma das séries que compõe as cinco dimensões do *candle* é aplicada a sequência de operações listadas a seguir antes da calibração:

1. Transformação logarítmica;
2. Faz-se a diferença entre o dia corrente e o dia anterior, ou seja, obtém-se a variação na dimensão em questão. Essa variação é transformada em variação percentual.
3. À série de variação log-transformada percentual é aplicada uma função que ajusta os possíveis *outliers* existentes. São considerados *outliers* os valores que estão

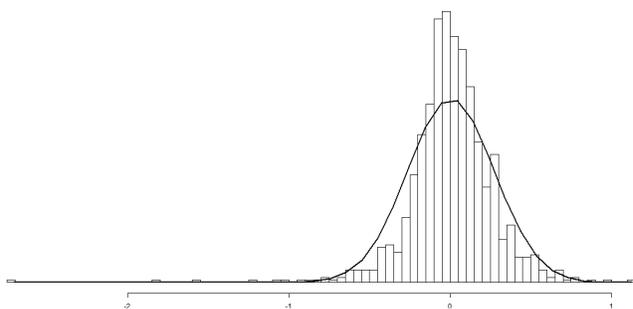


Figura 4: Distribuição das variações absolutas de preço de fechamento em CMIG4 em 2011

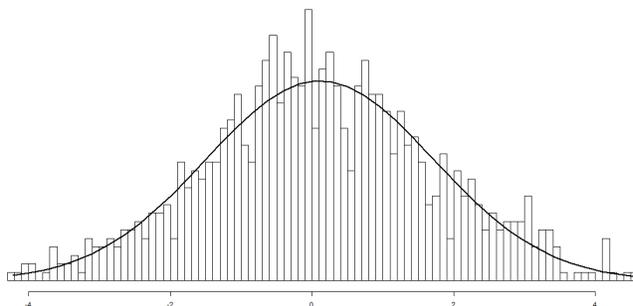


Figura 5: Distribuição das variações percentuais de preço de fechamento log-transformadas com ajuste de outliers em CMIG4 em 2011

no primeiro ou no quarto quartil da distribuição de dados. Estes são substituídos pelo resultado de uma média simples entre valor imediatamente anterior e o valor imediatamente posterior à sua ocorrência.

Este procedimento de tratamento dos dados, como visto na Figura 3, é feito com o objetivo de aproximar a distribuição das variações tratadas de uma distribuição normal, isso independentemente da época em que a série está sendo estudada. Para exemplificar pode-se observar na Figura 4 a distribuição das variações absolutas da dimensão “preço de fechamento” no ano de 2011 para CMIG4. Como se nota é uma distribuição com características leptocúrticas e assimétricas. A Figura 5 mostra o mesmo conjunto de dados agora com a aplicação do tratamento. Foram realizados outros estudos quanto a caracterização de diversas séries levando-se em conta a importância de inserir dados normalizados na RNA1. Depois de diversos testes essa estratégia se mostrou a com maior eficácia para o maior número de casos.

Após o tratamento das séries de preço de fechamento, abertura, máxima, mínima e volume, tomou-se o preço de fechamento do dia seguinte como padrão esperado no treinamento da RNA1. Nesta lógica pode-se, ao fim de um dia de operação no mercado, tomar-se a variação das dimensões do *candle*, aplicar-se o tratamento de dados discutido e, com esse padrão de entrada, gerar-se a previsão de preço de fechamento para o dia seguinte.

A forma de gerar-se previsão para os preditores econométricos trata apenas da realização do cálculo correspondente ao método chamado, passando-se a estes a série histórica de

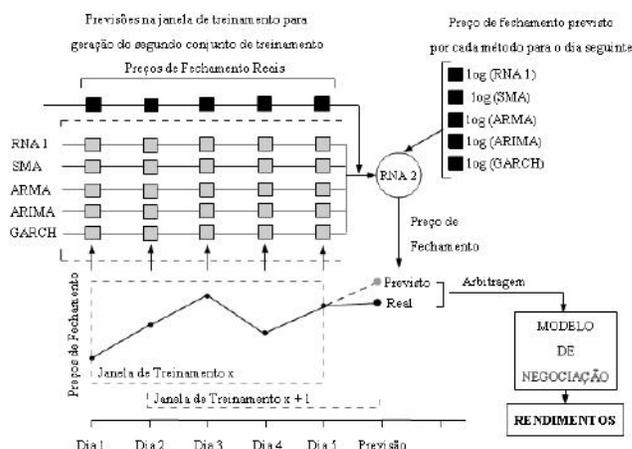


Figura 6: Avaliação das melhores configurações para a RNA2 na Fase 2

preços de fechamento no ativo de interesse.

Todas as previsões feitas nas diversas configurações são então submetidas ao modelo de negociação que retorna os rendimentos de cada uma. As melhores são utilizadas para gerar o conjunto de treinamento que servirá à RNA2 na fase seguinte.

### 3.2.3 Fase 2: Treinamento do comitê e previsão do preço de fechamento aprimorado

Após a geração do segundo conjunto de treinamento com as melhores configurações identificadas para cada preditor na Fase 1, este é utilizado para treinar a RNA2. Na Fase 2 o conjunto de treinamento e o padrão de entrada são log-transformados mais uma vez com o objetivo de aproximar-se da normalização, entretanto não é aplicada nesta fase as variações e o ajuste de outliers como visto na Fase 1. O processo pode ser observado no diagrama da Figura 6. Os valores de preço de fechamento da série real são vinculados a cada predição correspondente feita, dessa forma, a RNA2 passa a estar preparada para possíveis desajustes nas previsões realizadas no conjunto. Após o treinamento e o fechamento do mercado no dia corrente, gera-se a previsão para o dia seguinte com cada um dos preditores calibrados. Essa previsão é log-transformada e inserida como padrão de entrada na RNA2, o valor retornado é o preço de fechamento aprimorado. São testadas configurações levando-se em conta o número de neurônios, o tamanho do conjunto de treinamento e o tempo de operação da rede antes de um novo treinamento em cada ativo. A configuração mais rentável segundo o modelo de negociação é a utilizada para a realização de negociações na janela de operação (ver Figura 2).

Os resultados obtidos na janela de operação tratam do algoritmo já calibrado em todos os seus elementos. Os resultados são discutidos na seção a seguir.

## 4. RESULTADOS

A análise dos resultados gerados pelo robô investidor foi realizada frente aos resultados de cada um dos preditores quando se operou estes individualmente. Foi utilizada como comparativo também a estratégia *Buy and Hold* [16] que

consiste em adquirir determinado ativo e, na espera de sua valorização, manter a custódia do papel até o final de determinado período quando então se realiza a venda do mesmo. Como último *benchmark* foi aplicado o processo de compra e venda aleatória dentro do mesmo modelo de negociação. Foi feita inicialmente uma análise comparativa entre os resultados obtidos em cada um dos papéis, na sequência, levando-se em conta a média de retornos percentuais conseguidos na carteira como um todo e, por fim, avaliou-se os rendimentos percentuais totais. Todos os preditores foram utilizados sob as mesmas circunstâncias de experimentação. Os valores obtidos podem ser vistos na Tabela 2.

Observando-se os rendimentos percentuais percebe-se que o robô desenvolvido obteve os melhores resultados quando operando o índice BOVA11 e as ações CMIG4 e ITUB4. Em ITUB4 os resultados foram expressivos gerando-se um ganho maior que 50% ao longo do período de operação. Em BGI os resultados não apresentaram prejuízos, entretanto o lucro foi ínfimo, muito perto de zero. A série BGI inclusive, para todos os ativos testados, apresentou resultados que não geraram lucros expressivos, mostrando-se ser uma série de difícil previsão. Nesta o melhor caso foi observado com uma média móvel simples de 9 períodos. Em EMBR3, apesar de existirem retornos positivos em outros preditores (como na RNA1), o robô não conseguiu gerar lucros terminando o processo de previsão com um prejuízo de quase 5%.

Levando-se em conta a carteira como um todo, o robô apresentou resultados vencedores frente aos outros preditores e a operação aleatória, tanto na média dos rendimentos quanto no rendimento percentual total. Com um retorno médio muito próximo de 23% e um rendimento percentual total de 114,01%, este obteve o melhor resultado, seguido pela RNA1 que obteve um lucro médio de 17,5% e um retorno percentual total de 87,55%.

Quando se analisa a estratégia de mercado *Buy and Hold* vê-se que esta obteve lucro médio de 10,2% e total de 50,33%, fato devido principalmente à expressiva valorização de EMBR3 no período. Seus retornos são maiores que os gerados por qualquer um dos preditores econométricos, entretanto, quando estes são combinados à RNA1 pela segunda rede neural, na média e nos rendimentos totais, observam-se lucros maiores do que quando cada um foi aplicado individualmente. Assim, pode-se afirmar com base neste experimento que o robô foi mais rentável que *Buy and Hold* em 12,78% na média e 63,68% nos rendimentos percentuais totais.

Outro ponto perceptível, mas que não se pode afirmar sem maiores estudos é o fato de que, a segunda rede neural mesmo teoricamente ativando somente os bons preditores, também é influenciada por previsões ruins quando estas são na sua maioria ruins, ou seja, em ativos como o EMBR3 em que a maioria dos preditores não conseguiu boas previsões, mesmo com o ajuste do preço de fechamento final, ainda assim não foi possível gerar resultados positivos. Esta situação pode ser mais bem percebida quando se observa o bom resultado obtido com a aplicação individual da RNA1.

Foi avaliada também a taxa de acerto na previsão de altas ou baixas, ou seja, a acurácia do sinal de compra e venda gerado pelo método de negociação a partir da previsão do preço de fechamento para o dia seguinte. Levando-se em conta que o robô realizou 496 operações em 248 dias, observaram-se taxas de acerto positivas em todos os ativos com exceção de EMBR3. São expostos na Figura 7 os valores extraídos da matriz de confusão de cada análise. Nota-se que, em algu-

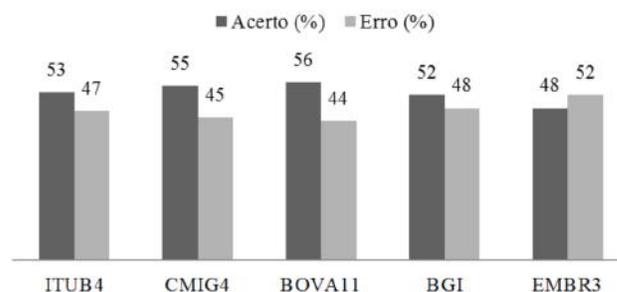


Figura 7: Taxa de acerto percentual em altas e baixas pelo robô investidor de novembro de 2012 a novembro de 2013

mas das vezes, mesmo que os resultados observados sejam positivos, ainda assim os retornos financeiros podem não ser expressivos. Como exemplo pode-se realizar um comparativo entre BGI e ITUB4. ITUB4 teve apenas 1% a mais de verdadeiros positivos que BGI, entretanto os lucros foram 50,9% maiores. Atribui-se esta situação ao fato de que a volatilidade em cada série está diretamente relacionada ao retorno, visto que, diferenças maiores entre preços de abertura e fechamento geram *candles* maiores e, conseqüentemente, maiores integralizações de lucros ou prejuízos.

## 5. CONCLUSÕES

Neste trabalho foi apresentada uma estratégia de negociação automatizada realizada por meio de um robô investidor.

O esquema geral do algoritmo baseou-se na utilização de preditores econométricos que levaram em conta a tendência como o SMA, a existência de estacionariedade como o ARMA, os não estacionários como o ARIMA e, os que tratam as variâncias condicionais, como o GARCH. Junto a estes preditores foi combinada uma RNA treinada para previsão de preço de fechamento do dia seguinte. Com deslizamento diário de janelas, várias configurações foram testadas para todos os preditores a fim de determinarem-se as que gerassem maiores rentabilidades. Os preditores calibrados geraram então previsões para que um novo conjunto de treinamento fosse utilizado em uma segunda rede neural, esta por sua vez, teve por objetivo ajustar o preço de fechamento usando as previsões realizadas pelos métodos econométricos e pela primeira rede neural.

Os resultados mostraram que para a maioria dos ativos, na média da carteira e nos retornos percentuais totais, os lucros obtidos pela combinação que o robô faz dos preditores geram resultados melhores do que quando cada um deles é aplicado individualmente. Os retornos para a ação EMBR3, que não foram de lucro, podem ter sido afetados pelo fato de que muitas previsões ruins podem influenciar negativamente a rede utilizada como comitê. Neste caso, a aplicação individual da RNA1 gerou o melhor resultado dentre todos os preditores.

A simulação foi realizada levando-se em conta os anos de 2011 e 2012 como períodos de calibração e treinamento. O período de operação foi do final de 2012 ao final de 2013. O robô apresentou taxas de acertos percentuais maiores que as taxas de erros em quatro das cinco séries. Ainda neste mesmo período utilizado para negociação, o robô foi compa-

**Tabela 2: Retornos percentuais em cada papel, média percentual na carteira e rendimentos percentuais totais operando-se com cada preditor em cada um dos ativos no período de novembro de 2012 a novembro de 2013**

	<i>Buy and Hold</i>	Aleatório	RNA1	SMA	ARMA	ARIMA	GARCH	Robô
<b>BOVA11</b>	-4,43	-23,71	27,51	0,34	20,47	9,73	0,6	<b>33,58</b>
<b>CMIG4</b>	-17,23	-47,1	29,73	20,42	2,1	30,66	14,39	<b>34,31</b>
<b>EMBR3</b>	52,56	-13,22	<b>16,98</b>	15,98	-1,08	-20,38	-6,05	-4,78
<b>ITUB4</b>	11,18	-24,32	9,43	-11,42	21,04	15,32	-1,95	<b>51,38</b>
<b>BGI</b>	8,95	-10,63	3,9	<b>10,08</b>	-6,5	-2,73	-5,96	0,48
<b>Média na carteira</b>	10,206	-23,796	17,51	7,08	7,206	6,52	0,206	<b>22,994</b>
<b>Rendimentos totais</b>	50,33	-118,58	87,55	35,4	36,03	32,6	1,03	<b>114,01</b>

rado com um sistema de compra e venda aleatória e com a estratégia de mercado *Buy and Hold*, em ambos os casos ele foi superior.

Notou-se que apesar de apresentar resultados positivos na grande maioria dos ativos testados e nos diferentes cenários, existem ainda séries nas quais o robô não obteve lucro, neste sentido, como trabalho futuro, pretende-se vincular ao sistema uma estratégia que permita selecionar ativos para construção de uma carteira específica. Essa seleção poderia ser realizada pelo próprio algoritmo quando da fase de configurações dos preditores, antes do uso da segunda rede neural.

Outra proposta de continuidade é que, neste formato, a previsão de 248 dias com a mesma calibração pode ser considerada longa. Neste sentido, pretende-se ainda inserir um modelo de ajuste de parâmetros dinâmico e que atue em intervalos de tempo menores. Isso permitiria ao algoritmo ter, a todo o tempo, as configurações mais apropriadas para uma dada janela e não somente o “re-treinamento” das redes.

## 6. AGRADECIMENTOS

O presente trabalho foi realizado com o apoio financeiro da CAPES - Brasil.

## 7. REFERÊNCIAS

- [1] C. N. Bergmeir and J. M. Benítez Sánchez. Neural networks in r using the stuttgart neural network simulator: Rsnns. American Statistical Association, 2012.
- [2] A. Boisbunon, S. Canu, D. Fourdrinier, W. Strawderman, and M. T. Wells. Akaike’s information criterion, cp and estimators of loss for elliptically symmetric distributions. *International Statistical Review*, 82(3):422–439, 2014.
- [3] T. Bollerslev. Generalized autoregressive conditional heteroskedasticity. *Journal of econometrics*, 31(3):307–327, 1986.
- [4] G. Box and G. Jenkins. Some comments on a paper by chatfield and prothero and on a review by kendall. *Journal of the Royal Statistical Society. Series A (General)*, 136(3):337–352, 1973.
- [5] A. d. P. Braga, A. Carvalho, and T. B. Ludermir. *Redes neurais artificiais: teoria e aplicações*. Editora LTC, 2007.
- [6] C. G. Broyden. A class of methods for solving nonlinear simultaneous equations. *Mathematics of computation*, 19(92):577–593, 1965.
- [7] B. Buscariolli and J. Emerick. *Econometria com eviews: Guia essencial de conceitos e aplicações*. Editora Saint Paul, 2011.
- [8] P.-C. Chang, C.-H. Liu, C.-Y. Fan, J.-L. Lin, and C.-M. Lai. An ensemble of neural networks for stock trading decision making. In *Emerging Intelligent Computing Technology and Applications. With Aspects of Artificial Intelligence*, pages 1–10. Springer, 2009.
- [9] M. S. Corporation. Metatrader 5 - trading platform. <http://www.metatrader5.com/>, 2016.
- [10] R. D. L. da Silveira Bueno. *Econometria de séries temporais*. Cengage Learning, 2008.
- [11] C. A. Debastiani. *Candlestick: um método para ampliar lucros na Bolsa de Valores*. Novatec Editora, 2007.
- [12] R. F. Engle. Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. *Econometrica: Journal of the Econometric Society*, pages 987–1007, 1982.
- [13] A. Furriel. *Modelos Heterocedásticos-ARCH e GARCH*. PhD thesis, Faculdade de Economia da Universidade do Porto, 2011.
- [14] L. K. Hansen and P. Salamon. Neural network ensembles. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (10):993–1001, 1990.
- [15] P. A. Morettin and C. Tolo. *Análise de séries temporais*. Blucher, 2006.
- [16] J. J. Murphy. *Technical analysis of the financial markets: A comprehensive guide to trading methods and applications*. Penguin, 1999.
- [17] J. J. Siegel and D. G. Coxe. *Stocks for the long run*, volume 3. McGraw-Hill New York, 2002.
- [18] I. d. Silva, D. H. Spatti, and R. A. Flauzino. *Redes neurais artificiais para engenharia e ciências aplicadas*. São Paulo: Artliber, pages 33–111, 2010.
- [19] H. Simon. *Redes Neurais-Princípios e Prática*. Porto Alegre: Bookman, 2001.
- [20] R. C. Team. R: A language and environment for statistical computing. r foundation for statistical computing, vienna, austria. 2013, 2014.
- [21] P. Whittle. *Hypothesis testing in time series analysis*, volume 4. Almqvist & Wiksells, 1951.
- [22] D. Wuertz, Y. Chalabi, and M. Miklovic. fgarch: Rmetrics-autoregressive conditional heteroskedastic modelling. <http://CRAN.R-project.org/package=fGarch>, 2110, 2009.
- [23] D. Wuertz and M. Taqqu. farma: Arma time series modelling. <http://CRAN.R-project.org/package=fArma>, 2013.