

# Aplicação de Técnicas de Aprendizado de Máquina para Previsão do IBOVESPA

Matheus Scalco Hernandez  
Engenharia de Produção, Escola  
Politécnica - EP, Pontifícia  
Universidade Católica do Paraná -  
PUCPR  
Curitiba, Brasil  
matheus.hernandes@pucpr.br

Leandro dos Santos Coelho  
Programa de Pós-Graduação em  
Engenharia de Produção e Sistemas -  
PPGEPS, Escola Politécnica - EP,  
Pontifícia Universidade Católica do  
Paraná - PUCPR  
Departamento de Engenharia  
Elétrica - DEE, Universidade Federal  
do Paraná - UFPR  
Curitiba, Brasil  
leandro.coelho@pucpr.br

Roberto Zanetti Freire  
Programa de Pós-Graduação em  
Engenharia de Produção e Sistemas -  
PPGEPS, Escola Politécnica - EP,  
Pontifícia Universidade Católica do  
Paraná - PUCPR  
Curitiba, Brasil  
roberto.freire@pucpr.br

**Resumo** – A previsão de mercados financeiros é uma tarefa desafiadora devido à complexidade dos dados que, quando analisados, geralmente são caóticos, dinâmicos e não-lineares. Estas características representam altos riscos para os investidores, portanto, a elaboração de modelos mais robustos para previsão de mercados financeiros é um tópico de interesse do setor. Além disso, a grande maioria dos estudos realizados foram feitos em economias já desenvolvidas. Neste contexto, este trabalho se propõe a aplicar métodos de Aprendizado de Máquina na previsão da bolsa de valores de São Paulo. Foram utilizadas quatro abordagens: Redes Neurais Artificiais (RNAs) com arquitetura *feedforward*; RNA associada a uma estrutura de modelo paramétrico não linear, auto-regressivo com entrada exógena (NARX, do inglês *Nonlinear AutoRegressive with exogenous inputs*); Máquina de Vetores de Suporte com Mínimos Quadrados (LS-SVM, do inglês *Least-Squares Support Vector Machines*); LS-SVM também associada a estrutura de modelo NARX. Os modelos propostos foram validados utilizando três critérios de desempenho. Os modelos com estrutura NARX apresentaram qualidade de previsão superior aos métodos tradicionais, sendo que a LS-SVM NARX foi o método com melhor resultado nas etapas de treinamento, validação e teste. Para previsão de um passo à frente, a RNA NARX teve o melhor resultado. Já para a previsão livre considerando 5 e 22 dias, a LS-SVM NARX foi o método com melhor desempenho.

**Palavras-chave**—*IBOVESPA, redes neurais artificiais, máquina de vetores de suporte com mínimos quadrados, aprendizado de máquina, modelo auto-regressivo.*

## I. INTRODUÇÃO

A previsão do comportamento de bolsas de valores e mercados financeiros é, há muitos anos, um desafio para pesquisadores e investidores [1]. Neste contexto, diversos modelos de previsão foram desenvolvidos por meio de modelos estatísticos lineares. No entanto, estes modelos não apresentavam desempenho satisfatório quando aplicados à problemas reais [2]. Segundo [3], os mercados financeiros e as bolsas de valores são complicados, caóticos por natureza, dinâmicos e não-lineares. Além disso, eventos políticos,

condições econômicas, taxas de juros e outros fatores macroeconômicos influenciam os movimentos e comportamentos das bolsas de valores [4]. Estas características representam um alto risco para acionistas e investidores. Assim, a elaboração de modelos mais robustos para a previsão de índices associados aos mercados financeiros é considerada importante quando se ponderam sobre estas propriedades nas estratégias de tomadas de decisões [5].

Nos últimos vinte anos, modelos baseados em Aprendizado de Máquina (*Machine Learning* - ML) apresentaram importantes avanços no contexto de previsão de séries temporais [6], ganhando atenção da comunidade científica ao se destacarem em relação a modelos estatísticos clássicos [7]. Uma série temporal retrata uma sequência discreta de um evento ordenado em função do tempo  $t$ , e sua previsão tem o objetivo determinar um valor de  $y(t + 1)$ , onde  $y(t)$  representa o valor da série de interesse no instante atual. Essa previsão é feita pela compreensão da série em seu tempo atual e de pontos temporais anteriores  $t - 1, t - 2, \dots, (t - n - 1)$  [8].

Tsay [9] afirma que as Redes Neurais Artificiais (RNAs) se tornaram um tópico popular na análise de séries temporais. As RNAs possuem propriedades que faltam a alguns modelos lineares e não-lineares tradicionais de previsão, como tolerância ao ruído [10] e não-estacionariedade dos dados [11]. Além disso, o teorema de aproximação universal motivou a aplicação de RNAs nos dados financeiros [12]. Este teorema declara que as RNAs conseguem aproximar qualquer função contínua de maneira uniforme ao aumentar o número de unidades nas camadas ocultas das redes e pressupostos mínimos nas funções de ativação [13].

Uma RNA com três camadas foi utilizada para a previsão do índice da bolsa de valores de Istambul, Turquia [14]. Os resultados experimentais apresentados mostraram um desempenho de 75,74% da RNA na previsão do índice. Já em [15] foram obtidos resultados satisfatórios na previsão de preços da PETR4, com 93,62% de previsões corretas da direção do índice e erro absoluto médio percentual de 5,45% referente a ação preferencial da PETROBRAS. Assim, os autores

## II. MATERIAIS E MÉTODOS

Esta seção descreve os métodos utilizados para previsão de séries temporais adotados neste trabalho.

### A. Redes Neurais Artificiais

concluíram que o uso de RNAs para previsão do comportamento e tendências de ações são uma opção viável para serem consideradas como técnicas de previsão. Além de séries financeiras, as RNAs foram aplicadas na previsão da velocidade do vento [16], chuva [17], irradiância solar [18] e preço da energia elétrica [19]. Tkáč e Verner [20] compilaram 412 artigos, entre 1994 e 2015, que utilizaram RNAs para atividades de previsão em diversas áreas.

Outra técnica popular de ML em tarefas de regressão é a Máquina de Vetores de Suporte (*Support Vector Machine* – SVM), introduzida por Vapnik [21]. As SVMs são formuladas e representadas como um problema convexo quadrático, onde seu regressor realiza um mapeamento entre o espaço das entradas do modelo até o espaço de características de maiores dimensões. Neste contexto, um regressor linear é construído ao minimizar a função de custo [12]. As SVMs possuem boas propriedades de generalização devido a implementação do princípio de minimização do risco estrutural, onde a capacidade do aprendizado de uma máquina e o risco empírico são minimizados simultaneamente [22]. O método ainda possui a propriedade de otimização global [23]. As SVMs apresentaram resultados interessantes em aplicações de desenvolvimento de sistemas de controle [24], previsão de carga elétrica [25], previsão de consumo de energia [26] e finanças [27].

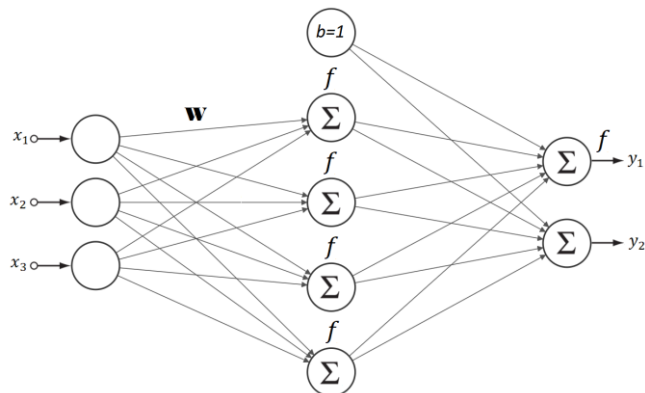
Embora existam trabalhos que lidam com a análise de séries temporais financeiras, a grande maioria dos estudos realizados foram feitos em mercados financeiros já desenvolvidos, como Estados Unidos, Japão e países da Europa Ocidental [14]. No contexto deste trabalho, onde um mercado em expansão foi adotado como estudo de caso, uma minoria de trabalhos é encontrada.

Assim, este estudo tem como objetivo a análise comparativa de diferentes métodos de previsão de séries temporais quando aplicados a um problema associado ao mercado de capitais. Foram utilizados quatro métodos na comparação, são eles: *i*) uma RNA clássica em arquitetura *feedforward*; *ii*) a mesma RNA associada a uma estrutura de modelo paramétrico não linear, auto-regressivo com entrada exógena (NARX) [28]; *iii*) a Máquina de Vetores de Suporte com Mínimos Quadrados (LS-SVM, do inglês *Least-Squares Support Vector Machines*) [29]; *iv*) e a LS-SVM também associada a estrutura de modelo NARX. Todos os modelos citados neste parágrafo foram utilizados na previsão do índice da Bolsa de Valores de São Paulo (IBOVESPA).

A próxima seção deste artigo apresenta os materiais e métodos utilizados com foco nos algoritmos, estruturas de modelos utilizados e ambiente de desenvolvimento escolhido. A seção III introduz os dados utilizados no trabalho, local de obtenção dos mesmos, métodos de seleção, e critérios de desempenho utilizados para quantificar os resultados. Em IV, é retratado a configuração de parâmetros dos métodos utilizados, os diferentes cenários de previsão e quantidade de simulações realizadas. Além disso, apresenta a análise dos resultados deste trabalho. Por último, o capítulo V faz a conclusão deste estudo e sugestões para possíveis trabalhos futuros.

O componente básico de uma RNA é um neurônio artificial. A ideia do primeiro neurônio artificial foi apresentada por McCulloch e Pitts [30], que imitava as funcionalidades de um neurônio biológico. Um modelo mais flexível foi então elaborado por Frank Rosenblatt [31]. Uma rede neural é um processador paralelo, formado por unidades de processamento – os neurônios artificiais – que armazena conhecimento experimental e o disponibiliza para uso. A forma na qual os neurônios são organizados é chamada de arquitetura de rede. Neste trabalho, utilizou-se uma Rede Neural Artificial de Múltiplas Camadas *feedforward*, que pode ser definida pela presença de uma ou mais camadas ocultas. O termo “oculta” é usado pois esta parte da rede não é vista diretamente pelos sinais de entrada ou de saída da rede. Os sinais de entrada da primeira camada são alimentados à segunda camada, e assim por diante, até o final da rede. Uma representação gráfica de uma RNA *feedforward* com 3 camadas, 3 entradas, 4 neurônios na camada oculta, *bias* e duas saídas, pode ser encontrada na Fig. 1.

Fig. 1. Diagrama de uma RNA *feedforward*.



O sinal  $x_j$  da conexão  $j$  interligada ao neurônio  $k$  é multiplicado pelo peso  $w_{kj}$ . Em termos matemáticos, pode-se descrever a saída  $y$  do neurônio  $k$  a partir da equação:

$$y_k = f \left( \sum_{j=1}^m x_j w_{kj} + b_k \right), \quad (1)$$

onde  $x_1, x_2, \dots, x_m$  são os sinais de entrada,  $w_{k1}, w_{k2}, \dots, w_{km}$  são os respectivos pesos da conexão do neurônio  $k$ ,  $b_k$  é o *bias*, e  $f(\cdot)$  é a função de ativação. A função de ativação utilizada nesse trabalho é a função sigmoide apresentada em (2), onde  $g$  é o parâmetro de inclinação da função, conforme equação abaixo:

$$f(\cdot) = \frac{1}{1 + e^{-gx}}. \quad (2)$$

As RNAs aprendem informações do meio a partir de processos de aprendizagem. Na aprendizagem supervisionada, a rede é alimentada com um vetor de treinamento. A saída

computada pela rede é observada e comparada com os resultados desejados. Os parâmetros da RNA são ajustados de acordo com o vetor de treinamento e o erro, este último equivalente à diferença entre o resultado desejado e o resultado real da rede. O algoritmo de retropropagação do erro (*Error Backpropagation*) é considerado um dos maiores avanços para o treinamento de redes neurais [32]. Embora ainda seja muito utilizado, é reconhecido como um algoritmo ineficiente, devido sua lenta convergência e por encontrar mínimos locais, em vez de mínimos globais [33]. O algoritmo Levenberg-Marquardt, outro algoritmo de aprendizagem, utilizado neste estudo, trabalha como um processo de treinamento combinado. Quando ao redor de uma área com curvatura complexa, utiliza o método do gradiente descendente. Ao atingir uma curvatura adequada, o algoritmo utiliza o método de Gauss-Newton, o que acelera a convergência [32].

### B. Máquinas de Vetores de Suporte com Mínimos Quadrados (LS-SVM)

Em 1995, Vapnik introduziu as SVMs para solucionar problemas de reconhecimento de padrões. O método realiza um mapeamento dos dados de entrada para um espaço dimensional mais alto e, então, constrói um hiperplano ótimo de separação. Esse procedimento consiste na resolução de um problema de programação quadrática. As SVMs também foram utilizadas para tarefas de aproximação de funções e estimação de regressores com a introdução da função de perda [34]. Uma alteração no método para regressão foi proposta por Suykens [29], a qual foi adotada neste trabalho. Denominada de Máquinas de Vetores de Suporte com Mínimos Quadrados (LS-SVM, do inglês *Least-Squares Support Vector Machines*), a técnica utiliza funções de perda de mínimos quadrados para construir o problema de otimização baseado em restrições de igualdades em vez de desigualdades.

Dado o conjunto de treinamento  $\{x_i, y_i\}$  de tamanho  $n$ , onde  $i = 1, \dots, n$ , a função de regressão que mapeia  $y$  em função de  $x$  pode ser formulada como:

$$y = f(x) = \mathbf{w}^T \varphi(x_i) + b, \quad (3)$$

onde  $\mathbf{w}$  é o vetor peso e  $\varphi$  uma função não-linear que realiza o mapeamento da característica de entrada  $x_i$  a uma dimensão mais alta. Neste caso,  $b$  e  $\mathbf{w}$  são desconhecidos e podem ser obtidos na otimização do problema (durante o treinamento da SVM).

O problema de otimização da LS-SVM, associado à minimização da função de risco  $J$ , é dado por:

$$\min J(\mathbf{w}, e) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + \frac{1}{2} \gamma \sum_{i=1}^n e_i^2, \quad (4)$$

sujeito a  $y_i = \mathbf{w}^T \varphi(x_i) + b + e_i$ , onde  $\gamma$  é um termo de regularização, que controla o equilíbrio entre o erro de treinamento e habilidade de generalização da LS-SVM, e  $e_i$  representa um ruído branco,  $i = 1, \dots, n$ . A primeira parte da função acima tem como objetivo regular os pesos e penalizar aqueles com valores altos, já que cargas altas provocam variações excessivas no modelo. Já a segunda parte da equação

representa o vetor regressor do conjunto de treinamento. Ao introduzir o operador Lagrangiano definido por:

$$L_1(\mathbf{w}, b, e, \alpha) = J_1(\mathbf{w}, b, e) + \sum_{i=1}^n \alpha_i (y_i - \mathbf{w}^T \varphi(x_i) b - e_i), \quad (5)$$

onde  $\alpha$  é um vetor multiplicador Lagrangiano. As condições para o ótimo mapeamento entre entrada e saída podem ser definidas como:

$$\left\{ \begin{array}{l} \frac{\partial L_1}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^n \alpha_i \varphi(x_i) \\ \frac{\partial L_1}{\partial b} = 0 \Rightarrow \sum_{i=1}^n \alpha_i = 0 \\ \frac{\partial L_1}{\partial e_i} = 0 \Rightarrow e_i = \frac{1}{\gamma} \alpha_i \\ \frac{\partial L_1}{\partial \alpha_i} = 0 \Rightarrow y_i = \mathbf{w}^T \varphi(x_i) + b + e_i \end{array} \right. \quad (6)$$

Na equação (6),  $k(x_i, x_j)$  indica uma função kernel. O vetor  $k(x_i, x_j)$  é igual ao produto interno dos vetores  $x_i$  e  $x_j$  na dimensão de características, o que significa  $k(x_i, x_j) = \varphi(x_i)^T \varphi(x_j)$ . O procedimento de adotar núcleos para substituir o cálculo de  $\varphi(x_i)$  e  $\varphi(x_j)$  pode ser feito através de uma função aproximada. Estes núcleos geram um mapeamento entre o espaço de entrada e um espaço de alta dimensão, chamado de espaço de características. O hiperplano construído por este espaço de características, ao ser mapeado de volta ao espaço de entrada, torna-se uma superfície não-linear. O hiperplano de separação não é mais uma função linear do vetor de entrada, e sim uma função linear do vetor do espaço de características.

Neste trabalho uma função kernel de base radial foi adotada:

$$k(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right), \quad (7)$$

onde  $\sigma$  representa o espalhamento da função Gaussiana. Devido à condição de Mercer,  $K = (k_{ij})_{i,j}$ ,  $k_{ij} = k(x_i, x_j)$  e  $V = \text{diag}(1/C, 1/C, \dots, 1/C)$ , o modelo LS-SVM para regressão pode ser definido como:

$$\begin{bmatrix} A & 1 \\ \mathbf{1}^T & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ b \end{bmatrix} = \begin{bmatrix} \mathbf{y} \\ 0 \end{bmatrix}, \quad (8)$$

onde  $A = K + V$  [35]. Por fim, o modelo LS-SVM de regressão resultante é [29]:

$$f(x) = \sum_{i=1}^n \alpha_i k(x_i, x_j) + b \quad (9)$$

### C. Modelo Paramétrico Não Linear, Auto-regressivo com Entrada Exógena (NARX)

A previsão de múltiplos passos adiante é um procedimento complexo [36]. Neste contexto, acredita-se que RNAs recorrentes desempenham um importante papel para este tipo de previsão.

RNAs de múltiplas camadas com arquitetura *feedforward* podem ser adaptadas para Redes Neurais com Atraso de Tempo e também para RNAs Recorrentes ao retroalimentar as saídas dos neurônios da camada oculta ou da camada de saída. Estas topologias de rede são geralmente treinadas com variantes temporais do algoritmo de retropropagação do erro baseado no método do gradiente descendente. No entanto, a aprendizagem de dependências temporais de longo prazo, com métodos baseados no gradiente descendente é mais eficiente em modelos não-lineares, auto-regressivos com entradas exógenas (NARX, do inglês *Nonlinear AutoRegressive with eXogenous input*) [28]. Modelos NARX são considerados poderosos no que diz respeito à identificação de sistemas não-lineares e na previsão de séries temporais. Neste caso, a aprendizagem da rede torna-se mais eficiente, convergem mais rapidamente e generalizam melhor do que outras RNAs [37]. O modelo NARX pode ser matematicamente representado:

$$y(t) = F \left( \begin{matrix} x(t-1), \dots, x(t-D), \\ y(t-1), \dots, y(t-D) \end{matrix} \right), \quad (10)$$

onde  $F$  é o mapeamento entre a entrada e a saída,  $t$  representa o tempo atual e  $D$  os atrasos. Assim,  $y(t)$  representa a saída do modelo no tempo  $t$  e  $y(t-D)$  é um número finito de saídas passadas, sendo  $x(t-D)$  as entradas atrasadas.

### III. BASE DE DADOS E CRITÉRIO DE DESEMPENHO

Esta seção descreve a base de dados e os critérios de desempenho que foram utilizados para avaliar a qualidade dos modelos de previsão.

#### A. Base de Dados

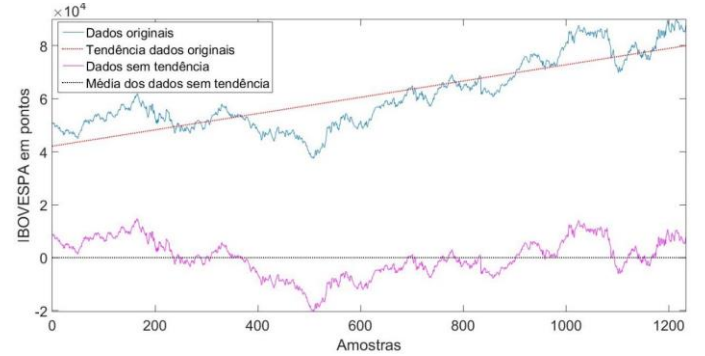
Os dados utilizados neste trabalho foram obtidos por relatórios no site da B3 [38], e datam entre 01 de janeiro de 2014 até 31 dezembro de 2018, totalizando 1234 amostras temporais. Os relatórios possuíam informações sobre a data dos dias de negociação, códigos de ativos, preço de abertura, preço máximo diário, preço mínimo diário, preço de fechamento, preço médio diário, quantidade total de títulos negociados e o volume financeiro movimentado pelos ativos diariamente (preço médio multiplicado pela quantidade de títulos negociados no dia).

O índice da Bolsa de Valores de São Paulo (IBOVESPA) é uma carteira teórica composta pelos ativos negociados de maior expressividade. No período selecionado para realização deste trabalho, o número de ativos na carteira era de 67. Empresas como Petróleo Brasileiro S.A., Vale S.A., grandes bancos e Cervejaria AmBev são muito negociadas e movimentam grandes quantidades de dinheiro, e, por tanto, possuem uma representação – peso – maior no índice do que ações menos negociadas. A partir dos relatórios, foram selecionados 27 ativos, com base na representatividade e correlação com o índice, para compor este estudo. Os ativos selecionados são

responsáveis por aproximadamente 82% do IBOVESPA. Neste trabalho, utilizou-se o volume financeiro diário movimentado por cada um dos ativos para serem alimentados como conjunto de entrada. O conjunto do volume financeiro diário dos 27 ativos é o vetor de entrada dos modelos, e a pontuação do IBOVESPA no respectivo dia representa o vetor de valores desejados. Ambos possuem 1234 amostras em função do tempo.

Depois de compilados, retirou-se a tendência dos conjuntos de dado, conforme Fig. 2. Ao observar as séries sem tendência, é difícil saber se houveram padrões cíclicos.

Fig. 2. Série original dos dados e a série resultante após remoção da tendência.



As análises e testes de sazonalidade produziram resultados insatisfatórios quanto às conclusões dos modelos e foram deixados de lado em relação ao que está sendo apresentado neste trabalho. A normalização ocorreu considerando valores máximos e mínimos de cada série, de forma que:

$$x_{k(i)} = \frac{x_i - x_{min}}{x_{max} - x_{min}}, \quad (11)$$

onde  $x_{k(i)}$  é o valor normalizado de  $x_i$  para um conjunto de dados de tamanho  $n$ ,  $x_{min}$  é o menor valor encontrado no conjunto, e  $x_{max}$  o maior valor do conjunto.  $x_{k(i)}$  possui um valor entre  $[0,1]$ . Esse procedimento garante que os grandes valores não oprimam os valores menores, reduzindo erros de previsão [39]. Os dados foram divididos entre conjunto de treinamento, teste e validação, com as proporções de 70%, 15% e 15%, respectivamente.

#### B. Critérios de Desempenho

Para avaliação dos métodos, três métricas de desempenho foram utilizadas: o erro absoluto médio (*Mean Absolute Error – MAE*) (12); o erro quadrático médio (*Mean Squared Error – MSE*) (13); e o coeficiente de correlação de Pearson,  $\rho$ , (14). O primeiro é calculado da seguinte maneira:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - d_i|, \quad (12)$$

onde  $y_i$  é o valor de saída e  $d_i$  é o valor desejado. MSE e  $\rho$  são apresentados a seguir, respectivamente.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - d_i)^2, \quad (13)$$

$$\rho = \frac{\sum_{i=1}^n (y_i - \bar{y})(d_i - \bar{d})}{(n-1)S_y S_d}. \quad (14)$$

Nas duas últimas equações, onde  $\bar{y}$  é a média do conjunto de saída da rede,  $\bar{d}$  é a média do conjunto de valores desejados,  $S_y$  é o desvio padrão para a primeira variável,  $S_d$  é o desvio padrão para a segunda variável, e  $n$  é o tamanho do conjunto. O coeficiente de correlação de Pearson mede o grau de relação linear entre duas variáveis. Este assume um valor no intervalo  $[-1, 1]$ . Se uma variável tende a aumentar à medida que as outras diminuem, o coeficiente é negativo. Por outro lado, se as duas variáveis tendem a aumentar em conjunto, o coeficiente de correlação é positivo.

#### IV. RESULTADOS

Esta seção descreve os parâmetros dos modelos e também os resultados obtidos para previsão do IBOVESPA. Para todos os algoritmos implementados foi utilizada o ambiente computacional MATLAB R2015a (*The MathWorks, Inc, 2015*).

##### A. Configuração de Parâmetros dos Modelos

A RNA criada possui três camadas. A camada de entrada é formada por 27 neurônios, um para cada ativo selecionado. A camada de saída possui um único neurônio. Quanto à camada oculta, não há uma teoria consolidada para definição da quantidade de neurônios [40]. No entanto, segundo o mesmo autor, uma regra geral utilizada para a escolha do número de neurônios na camada oculta é escolher um valor entre a quantidade de neurônios na camada de entrada e na camada de saída. Testou-se diferentes cenários com 5, 10, 15, 20 e 25 neurônios na camada oculta. O melhor resultado foi obtido com 20 neurônios.

A função de ativação utilizada na camada oculta foi a função sigmoide, função de ativação mais utilizada nos problemas de redes neurais com múltiplas camadas [32]. Já na camada de saída, foi utilizada uma função de ativação linear. O método de aprendizagem utilizado foi o algoritmo de retropropagação Levenberg-Marquardt.

Para a RNA NARX, utilizou-se a mesma quantidade de camadas e neurônios e o mesmo algoritmo de treinamento. Foi feito um laço de retroalimentação, com dois atrasos no conjunto de entrada e dois atrasos no conjunto de valores desejados.

Já para as LS-SVM, considerando a função kernel de base radial utilizada, seus parâmetros foram encontrados utilizando *leave-one-out crossvalidation*. Os parâmetros  $\sigma^2$  – em (7) – e  $\gamma$  – em (4) – tiveram valores diferentes para a LS-SVM e LS-SVM NARX, já que para a primeira, seus valores foram obtidos a partir do conjunto de dados original. Já no segundo método, os valores foram obtidos a partir de uma matriz de regressores com os atrasos embutidos. Para a LS-SVM NARX, atrasos de duas amostras temporais foram considerados. Em ambas as

utilizações da LS-SVM, foi considerada uma revisão do método Simplex [41], como algoritmo de otimização.

Para a previsão do IBOVESPA foram considerados os horizontes de previsão de 1, 5 e 22 passos à frente considerando amostras diárias. O mercado de capitais funciona apenas em dias úteis, neste contexto, 5 passos à frente equivalem a uma semana, já 22 passos à frente equivalem a um horizonte de um mês de previsão. Para cada um dos modelos e cada horizonte de previsão foram feitas 30 simulações, com o objetivo de se obter uma amostra de dados mais confiável. As métricas de desempenho e saídas dos modelos foram exportadas para um arquivo com extensão .xlsx, e depois de 30 simulações, a média aritmética de cada uma das métricas foi feita, e são apresentadas no próximo capítulo.

##### B. Análise de Desempenho para as Etapas de Treinamento, Validação e Teste

Para a RNA, embora o modelo tenha ajustado a estrutura dos dados de maneira satisfatória, nas etapas de validação e teste o modelo apresentou uma fraca correlação. O número médio de iterações até interrupção do treinamento foi 8,67 e o tempo médio de 0,6798 s.

Em comparação com a RNA, a LS-SVM obteve resultados ligeiramente superiores. A média dos parâmetros  $\sigma^2$  e  $\gamma$  encontrados a partir do conjunto de dados, foi de 9,306 e 25,136, respectivamente. Para um  $\gamma$  baixo, a minimização da complexidade do modelo é enfatizada. Para um  $\gamma$  alto, o ajuste dos dados de treinamento é ressaltado. Em um kernel de base radial, um  $\sigma^2$  maior significa uma maior suavidade. O processo de ajuste dos parâmetros da função ocorre na seguinte sequência: para cada kernel, encontra-se pontos de partida adequados e, em seguida, estes pontos são otimizados com *leave-one-out crossvalidation*. O tempo médio de configuração da LS-SVM foi de 30,12 s.

A Tabela I apresenta o resultado dos modelos nas três etapas.

TABELA I. DESEMPENHO DOS MODELOS NA ETAPA DE TREINAMENTO, VALIDAÇÃO E TESTE.

Critérios de Desempenho	Modelo			
	RNA	LS-SVM	RNA NARX	LS-SVM NARX
<b>Treinamento</b>				
MAE	0,1239	0,0434	0,0207	0,0157
MSE	0,0265	0,0032	0,0007	0,0004
$\rho$	0,790	0,9624	0,9947	0,9951
<b>Validação</b>				
MAE	0,1288	0,1481	0,0283	0,0243
MSE	0,0277	0,0332	0,0014	0,0009
$\rho$	0,460	0,5200	0,9745	0,9853
<b>Teste</b>				
MAE	0,1944	0,1663	0,0401	0,0329
MSE	0,0593	0,0428	0,0029	0,0018
$\rho$	0,1450	0,1231	0,9312	0,9602

No treinamento da LS-SVM, processo que tem como objetivo otimizar os vetores de suporte e o *bias* através do algoritmo de otimização Simplex, a média de tempo do treinamento foi de 0,043 s.

Na RNA NARX, o coeficiente de correlação para as três etapas foi positivo e muito forte (ver Tabela I). O número de atrasos utilizados no conjunto de entrada e conjunto de valores desejados foi dois. O tempo médio de treinamento na RNA NARX foi de 1,9922 s, e a quantidade média de iterações 10,05.

Para o último método, a LS-SVM NARX, a média dos parâmetros  $\sigma^2$  e  $\gamma$  foi de 96114994,57 e 6612924,64, respectivamente. A diferença dos valores dos parâmetros entre a LS-SVM e a LS-SVM NARX se dá pelo fato de que  $\sigma^2$  e  $\gamma$  terem sido encontrados a partir de estruturas de dados diferentes. Enquanto na LS-SVM regular os parâmetros foram obtidos com base no conjunto de dados original, no método NARX os parâmetros foram encontrados a partir de uma reordenação dos dados: os pontos foram arranjados em uma matriz de regressores, fundamentados nos atrasos definidos, para a modelagem das séries temporais no modelo NARX. No método LS-SVM NARX os valores dos parâmetros também foram encontrados utilizando *leave-one-out crossvalidation*.

No caso do modelo LS-SVM NARX, a média de tempo de treinamento foi de 0,0409 s. A comparação do tempo de otimização para os modelos que envolvem a LS-SVM é exibida na Tabela II.

TABELA II. TEMPO DE OTIMIZAÇÃO DOS PARÂMETROS  $\gamma$  E  $\sigma^2$ .

Modelo	LS-SVM	LS-SVM NARX
Tempo de otimização (s)	30, 1191	29, 2108

Em comparação entre os métodos tradicionais e os que usaram arquitetura NARX, os últimos obtiveram melhores resultados. Já comparando os quatro métodos, a LS-SVM NARX obteve o melhor desempenho nas três etapas, treinamento, validação e teste, seguido pela RNA NARX. Nas Fig. 3, Fig. 4 e Fig. 5 é possível observar a saída da LS-SVM NARX, em comparação com os valores desejados, nas etapas de treinamento, validação e teste, respectivamente.

Fig. 3. Saídas da LS-SVM NARX comparadas com os valores desejados na etapa de treinamento.

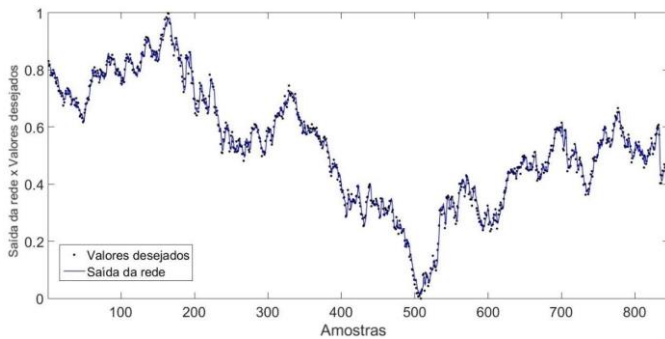


Fig. 4. Saídas da LS-SVM NARX comparadas com os valores desejados na etapa de validação.

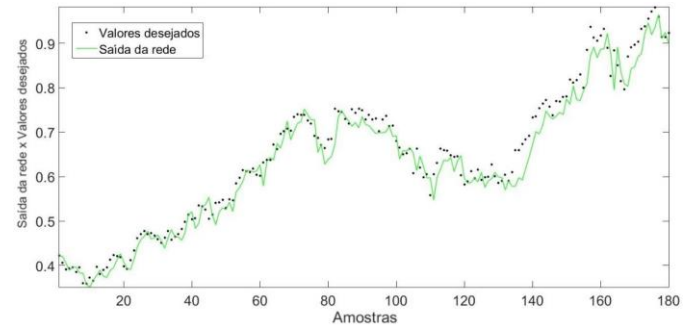
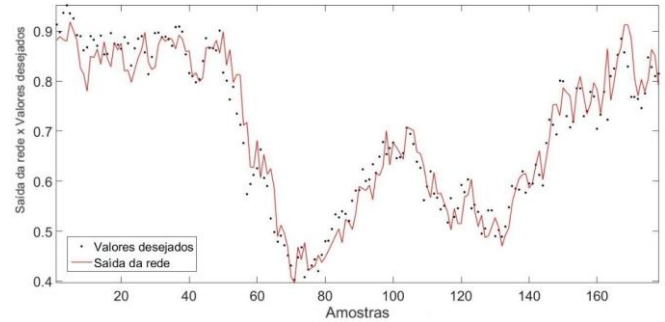


Fig. 5. Saídas da LS-SVM NARX comparadas com os valores desejados na etapa de teste.



### C. Previsões Considerando 1, 5 e 22 Passos à Frente

As previsões foram feitas em três cenários: um passo à frente, cinco passos à frente e 22 passos à frente. Como a bolsa de valores não opera aos fins de semana, o horizonte de previsão,  $h$ , de tamanho cinco é equivalente à uma antecipação de uma semana. Já com um horizonte de 22 passos, a previsão é compatível com um mês. O coeficiente de correlação  $\rho$  não será apresentado nas previsões de um passo à frente, já que a correlação entre dois pontos distintos define um relacionamento linear perfeito.

A Tabela III apresenta os índices de desempenho, para cada um dos métodos, e os respectivos horizontes de previsão.

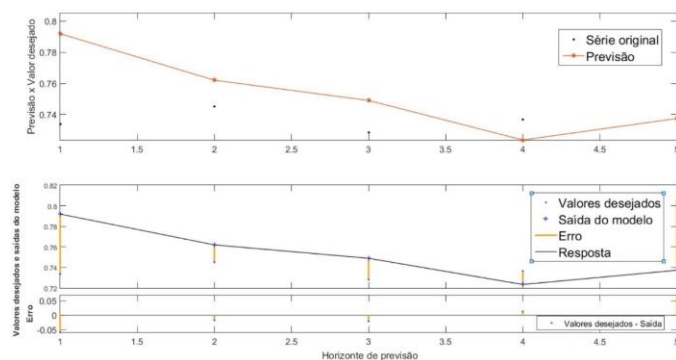
TABELA III. ANÁLISE DE DESEMPENHO DOS MODELOS PARA DIFERENTES HORIZONTES DE PREVISÃO: 1, 5 E 22 PASSOS À FRENTE.

Critérios de desempenho	Modelos			
	RNA	LS-SVM	RNA NARX	LS-SVM NARX
$MAE_{h=1}$	0,2176	0,2219	0,0616	0,0666
$MAE_{h=5}$	0,1653	0,1051	0,0939	0,0362
$MAE_{h=22}$	0,1570	0,1360	0,1250	0,0318
$MSE_{h=1}$	0,0504	0,0492	0,0040	0,0045
$MSE_{h=5}$	0,0497	0,0179	0,0145	0,0018
$MSE_{h=22}$	0,0428	0,0289	0,0263	0,0016
$\rho_{h=5}$	-0,6305	-0,5892	0,1590	-0,3168
$\rho_{h=22}$	0,0977	0,1477	-0,1718	-0,3346

Para o horizonte de previsão de um passo à frente, a RNA NARX obteve o melhor desempenho. O segundo melhor método foi a LS-SVM NARX. Em comparação com os métodos

tradicionais, para  $h = 1$ , os métodos NARX tiveram resultados muito superiores. Já para  $h = 5$ , a LS-SVM NARX (Fig. 6) apresentou os melhores resultados para ambos  $MAE$  e  $MSE$ . No entanto, o coeficiente de correlação apresentou uma correlação fraca e negativa, ou seja, com a subida do IBOVESPA, a saída da LS-SVM NARX diminuía levemente. O método com melhor coeficiente de correlação para um horizonte de 5 passos foi a RNA NARX, embora seja uma correlação de força desprezível, foi o único método com correlação positiva.

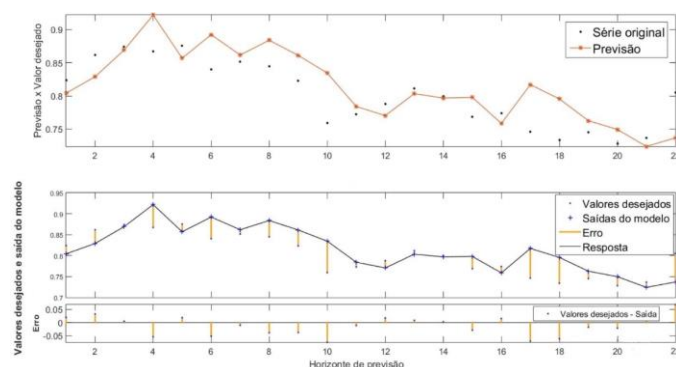
Fig. 6. Previsão livre cinco passos à frente com LS-SVM NARX.



Na parte superior da Fig. 6 é apresentado a comparação entre a previsão do modelo com a série original. Já na divisão inferior, exibe-se os valores da previsão comparados com valores desejados (série original). A diferença entre os dois valores é o erro.

Na última análise em relação a variação do horizonte de previsão, para  $h = 22$  passos à frente, a LS-SVM NARX novamente apresentou os melhores valores para o  $MAE$  e para o  $MSE$ . O coeficiente de correlação da LS-SVM NARX novamente foi fraco e negativo. O único método com correlação positivo para  $h = 22$  foi a LS-SVM tradicional. A RNA NARX obteve o segundo melhor resultado, seguido pela LS-SVM tradicional. A Fig. 7 ilustra a previsão da LS-SVM NARX para  $h = 22$ .

Fig. 7. Previsão livre de 22 passos à diante com LS-SVM NARX.



## V. CONCLUSÃO

Este trabalho teve como objetivo a previsão do índice da Bolsa de Valores de São Paulo utilizando quatro métodos de Aprendizado de Máquina: RNA, LS-SVM, RNA NARX e LS-SVM NARX. A partir da composição da carteira teórica do

IBOVESPA, 27 ações foram selecionadas para integrar o conjunto de entrada dos métodos e, em vez dos preços dos ativos, utilizou-se o volume financeiro movimentado diariamente pelas ações. O período dos dados coletados foi entre 02 de janeiro de 2014 a 28 de dezembro de 2018. Após a seleção das ações, foi retirada a tendência dos dados, que foram em seguida normalizados. Os dados foram então divididos entre conjuntos de treinamento, validação e teste para serem aplicados aos métodos.

Os horizontes de previsão considerados foram de um, cinco e 22 passos à frente. Os resultados dos métodos foram avaliados em relação ao  $MSE$ ,  $MAE$  e coeficiente de correlação de Pearson, estes adotados como critérios de desempenho.

Nas etapas de treinamento, os métodos NARX mostraram desempenho superior aos métodos tradicionais, sendo que a LS-SVM NARX apresentou os melhores resultados para todos os critérios de desempenho, seguida pela RNA NARX. Já para as tarefas de previsão, os métodos NARX também demonstraram melhores resultados. Para previsão de um passo à frente, a RNA NARX foi o método com melhor resultado, seguida pela LS-SVM NARX. Para cinco passos à frente, a LS-SVM NARX teve o melhor desempenho, mas com uma correlação fraca negativa. A RNA NARX, embora com erros maiores, teve uma correlação positiva. No último horizonte de previsão, 22 passos à frente, a LS-SVM NARX também foi o método com melhor desempenho, mas apresentou uma correlação fraca e negativa novamente. Tanto nas etapas de treinamento, validação e teste, os métodos NARX apresentaram resultados superiores aos métodos tradicionais.

Sugere-se para trabalhos futuros utilizar um número maior de ativos como entrada dos modelos, e inserir fatores políticos, internacionais e econômicos nos métodos. Para as redes neurais, outros algoritmos de aprendizagem podem ser testados, assim como distintas funções de ativação nas camadas e demais topologias de rede, considerando diferentes quantidades de camadas e neurônios, além do uso de outras classes de redes neurais. Para as máquinas de vetores de suporte, é possível utilizar outras funções kernel e encontrar os parâmetros via outros métodos de otimização. Além das alterações nos métodos, sugere-se uma análise mais detalhada sobre o desempenho dos modelos, com a utilização de testes estatísticos.

## REFERENCES

- [1] G. Iuhasz, M. Tirea, e V. Negru, "Neural network predictions of stock price fluctuations," 2012 14th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, pp. 505-512, 2012.
- [2] J. G. De Gooijer, e R. J Hyndman, "25 years of time series forecasting," International Journal of Forecasting, vol. 22, n. 3, pp. 443-473, 2006.
- [3] T. Z. Tan, C. Quek, e G. S. Ng, "Brain inspired genetic complementary learning for stock market prediction," IEEE congress on evolutionary computation, vol. 3, pp. 2653-2660, 2005.
- [4] K. Miao, F. Chen, Z. -G. Zhao, "Stock price forecast based on bacterial colony rbf neural network [ j ]," Journal of Qingdao University (Natural Science Edition), vol. 2, 2007.
- [5] B. Krollner, B. Vanstone, e G. Finnie, "Financial time series forecasting with machine learning techniques: A survey," European Symposium on Artificial Neural Networks: Computational and Machine Learning, Bruges, Bélgica, 2010.
- [6] G. Bontempi, S. B. Taieb, e Y. L. Borgne, "Machine learning strategies for time series forecasting," Aufaure MA., Zimányi E. (Eds) Business

- Intelligence. Lecture Notes in Business Information Processing, vol. 138. Springer, Berlin, Heidelberg, 2012.
- [7] L. dos S. Coelho, A. A. P. Santos, e N. C. A. da Costa JR., "Podemos prever a taxa de cambio brasileira? Evidência empírica utilizando inteligência computacional e modelos econométricos," *Gestão da Produção*, São Carlos, vol. 15, n. 3, pp. 635-647, 2008.
- [8] A. Konar, e D. Bhattacharya, "Time-series prediction and applications: a machine intelligence approach," Cham, Switzerland: Springer International Publishing, 2017.
- [9] R. S. Tsay, "Analysis of financial time series," 3 ed. Hoboken, United States of America: John Wiley & Sons, 2010.
- [10] G. Zhang, B. E. Patuwo, e M. Y. Hu, "Forecasting with artificial neural networks: The state of the art," *International journal of forecasting*, vol. 14, n. 1, pp. 35-62, 1998.
- [11] M. I. Széliga, P. F. Verdes, P. M. Granitto, e H. A. Ceccatto, "Artificial neural network learning of nonstationary behavior in time series," *International Journal of Neural Systems*, vol. 13, n. 02, pp. 103-109, 2003.
- [12] T. Van Gestel, J.A.K. Suykens, D.-E. Baestaens, A. Lambrechts, G. Lanckriet, B. Vandaele, B. de Moor, e J. Vandewalle, "Financial time series prediction using least squares support vector machines within the evidence framework," *IEEE Transactions on neural networks*, vol. 12, n. 4, pp. 809-821, 2001.
- [13] K. Hornik, M. Stinchcombe, e H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, n. 5, pp. 359-366, 1989.
- [14] Y. Kara, M. A. Boyacioglu, e Ö. K. Baykan, "Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange," *Expert systems with Applications*, vol. 38, n. 5, pp. 5311-5319, 2011.
- [15] F. A. de Oliveira, L. E. Zárate, M. de A. Reis, e C. N. Nobre, "The use of artificial neural networks in the analysis and prediction of stock prices," 2011 IEEE International Conference on Systems, Man, and Cybernetics. IEEE, pp. 2151-2155, 2011.
- [16] A. Meng, J. Ge, H. Yin, e S. Chen, "Wind speed forecasting based on wavelet packet decomposition and artificial neural networks trained by crisscross optimization algorithm," *Energy Conversion and Management*, vol. 114, pp. 75-88, 2016.
- [17] C. L. Wu, K. W. Chau, e C. Fan, "Prediction of rainfall time series using modular artificial neural networks coupled with data-preprocessing techniques," *Journal of Hydrology*, vol. 389, n. 1-2, pp. 146-167, 2010.
- [18] A. Mellit, e A. M. Pavan, "A 24-h forecast of solar irradiance using artificial neural network: Application for performance prediction of a grid-connected PV plant at Trieste, Italy," *Solar Energy*, vol. 84, n. 5, pp. 807-821, 2010.
- [19] D. Singhal, e K. S. Swarup, "Electricity price forecasting using artificial neural networks," *International Journal of Electrical Power & Energy Systems*, vol. 33, n. 3, pp. 550-555, 2011.
- [20] M. Tkáč, e R. Verner, "Artificial neural networks in business: Two decades of research," *Applied Soft Computing*, vol. 38, pp. 788-804, 2016.
- [21] V. Vapnik, "The nature of statistical learning theory," New York, United States of America: Springer-Verlag, 1995.
- [22] L. Zhang, W.-D. Zhou, P.-C. Chang, J.-W. Yang, e F.-Z. Li, "Iterated time series prediction with multiple support vector regression models," *Neurocomputing*, vol. 99, pp. 411-422, 2013.
- [23] C.-Y. Yeh, C.-W. Huang, e S.-J. Lee, "A multiple-kernel support vector regression approach for stock market price forecasting," *Expert Systems with Applications*, vol. 38, n. 3, pp. 2177-2186, 2011.
- [24] X.-C. Xi, A.-N. Poo, e S.-K. Chou, "Support vector regression model predictive control on a HVAC plant," *Control engineering practice*, vol. 15, n. 8, pp. 897-908, 2007.
- [25] M. Espinoza, J. A. K. Suykens, R. Belmans, e B. De Moor, "Electric load forecasting," *IEEE Control Systems Magazine*, vol. 27, n. 5, pp. 43-57, 2007.
- [26] K. Kavaklioglu, "Modeling and prediction of Turkey's electricity consumption using support vector regression," *Applied Energy*, vol. 88, n. 1, pp. 368-375, 2011.
- [27] W. Huang, Y. Nakamori, e S.-Y. Wang, "Forecasting stock market movement direction with support vector machine," *Computers & operations research*, vol. 32, n. 10, pp. 2513-2522, 2005.
- [28] P. M. Nørgård, O. Ravn, N. K. Poulsen, e L. K. Hansen, "Neural networks for modelling and control of dynamic systems-a practitioner's handbook," Springer-Verlag London, 2000.
- [29] J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, e J. Vandewalle, "Least squares support vector machines," Singapura: World Scientific, 2002.
- [30] W. Pitts, e W. S. McCulloch, "How we know universals the perception of auditory and visual forms," *The Bulletin of mathematical biophysics*, vol. 9, n. 3, pp. 127-147, 1947.
- [31] F. Rosenblatt, "Principles of neurodynamics. perceptrons and the theory of brain mechanisms," Cornell Aeronautical Lab Inc Buffalo NY, 1961.
- [32] S. Haykin, "Neural networks and learning machines," 3 ed. New Jersey, The United States of America: Prentice Hall, 2008.
- [33] H. Yu, e B. M. Wilamowski "Levenberg-marquardt training," *Industrial electronics handbook*, v. 5, n. 12, p. 1, 2011.
- [34] V. Vapnik, "The nature of statistical learning theory," New York, United States of America: Springer-Verlag, 1995.
- [35] J. Mercer, "Functions of positive and negative type, and their connection the theory of integral equations," *Philosophical transactions of the royal society of London*, vol. 209, n. 441-458, pp. 415-446, 1909.
- [36] J. M. P. Menezes JR, e G. A. Barreto, "Long-term time series prediction with the NARX network: An empirical evaluation," *Neurocomputing*, vol. 71, n. 16-18, pp. 3335-3343, 2008.
- [37] E. Diaconescu, "The use of narx neural networks to predict chaotic time series," *Wseas Transactions on computer research*, vol. 3, n. 3, pp. 182-191, 2008.
- [38] B3. [B]<sup>3</sup> Brasil Bolsa Balcão, 2019. Página inicial. Disponível em: <[http://www.b3.com.br/pt\\_br/](http://www.b3.com.br/pt_br/)>. Acesso em: 24 de ago. de 2019.
- [39] K.-J. Kim, "Financial time series forecasting using support vector machines," *Neurocomputing*, vol. 55, n. 1-2, pp. 307-319, 2003.
- [40] A. Blum, "Neural networks in c++: an object-oriented framework for building connectionist systems," New York: Wiley, 1992.
- [41] C. Sentelle, G. C. Anagnostopoulos, e M. Georgiopoulos, "A fast revised simplex method for svm training," 19th International Conference on Pattern Recognition. Tampa, United States of America, 2008.