

Previsão do Mercado Financeiro

1st Ana Livia Franco
DACOM
UTFPR - CP
Cornélio Procópio, PR
analivia@live.com

2nd Bruna Almeida Osti
DACOM
UTFPR - CP
Cornélio Procópio, PR
brunaosti@alunos.utfpr.edu.br

3rd Katharina Akemi Ikeda Rosa
DACOM
UTFPR - CP
Cornélio Procópio, PR
kath.akemi@gmail.com

Resumo—Este artigo tem como objetivo realizar previsões do mercado financeiro a partir de técnicas de Inteligência Artificial. Para tanto, as previsões foram realizadas a partir de diferentes cenários abordando modelos baseados em Lógica Fuzzy e Redes Neurais Recorrentes. Além disso, serão tratados dados de um trabalho relacionado, sendo este, envolvendo a técnica Bayes-Fuzzy, para um melhor comparativo ao final do artigo.

Index Terms—Aprendizado de Máquina; Lógica Fuzzy; Redes Neurais Recorrentes; Teorema de Bayes.

I. INTRODUÇÃO

Vários aspectos podem gerar instabilidade no mercado financeiro, como mudanças governamentais, catástrofes naturais, crises econômicas, aquisições e fusões de empresas, alterações no mercado exterior e outros fatores econômicos. Juntamente a isso, o avanço da comunicação e da tecnologia nos dias atuais vem aperfeiçoando o mercado financeiro, o tornando mais eficaz e acessível. Como consequência, o volume de informações geradas aumenta vertiginosamente todos os dias [1].

Nesse cenário, a atividade dos investidores pode se tornar difícil, visto a constante necessidade de análise às oscilações do mercado acionário. Desse modo, há uma busca contínua pelo aprimoramento de ferramentas capazes de prever com maior precisão o mercado de ações. Tais ferramentas têm como objetivo principal auxiliar pessoas e empresas na tomada de decisão durante a compra e venda de ações a partir da análise e manipulação das informações existentes.

Neste trabalho analisaremos métodos distintos utilizados para modelagem e previsão das séries temporais de ações do mercado através de métricas levando em conta os resultados encontrados por meio de duas aplicações próprias e por meio do estudo de uma técnica previamente estabelecida.

II. OBJETIVOS

A. Objetivo geral

Comparar os resultados de dois algoritmos de classificação e previsão de dados em relação ao mercado financeiro. E ser capaz de analisar e detectar por meio de métricas as tendências presentes na bolsa de valores.

B. Objetivos específicos

- Identificar dois algoritmos que realizam a previsão do mercado financeiro;

- Coletar e analisar dados da bolsa de valores por meio de métricas;
- Analisar as previsões do aumento ou queda de ações da bolsa de valores por meio dos dois algoritmos;
- Detectar as tendências em relação à bolsa de valores usufruindo de ambos os códigos;
- Comparar os resultados encontrados por meio das duas técnicas distintas.

III. FUNDAMENTAÇÃO TEÓRICA

Nesta seção apresentaremos conceitos de extrema importância para o entendimento deste trabalho, serão abordados conceitos dos métodos: Fuzzy, Redes Neurais Recorrentes e Teorema de Bayes.

A. Lógica Fuzzy

A lógica difusa (Fuzzy) é bem diferente da lógica a qual estamos acostumadas pois a lógica clássica define o elemento como está dentro ou fora de um conjunto, até mesmo associado a lógica booleana (0,1). Essa realidade é bem diferente da maioria dos problemas reais pois em algumas situações, como por exemplo: para classificar as pessoas com limites definidos, no caso, peso = {magro, esbelto, gordo}, idade = {criança, adolescente, jovem, adulto, idoso}, com altura = {baixo, médio, alto}. Portanto, ao descrever alguém você se deparará com a situação de que talvez esta pessoa possa estar numa categoria intermediária a dois valores.

Proposta por Zadeh (1965) [2], a lógica fuzzy propõe uma dualidade onde um elemento pode pertencer e simultaneamente não pertencer ao mesmo conjunto em níveis de forma que a adesão é um valor no intervalo entre [0,1]. No caso deste trabalho, a lógica fuzzy será utilizada no contexto de séries temporais difusas, no qual o modelo fuzzy é utilizado para modelagem e previsão de séries temporais.

O seu funcionamento está descrito abaixo:

Valores crisp (números reais) \rightarrow fuzzyficação \rightarrow valores fuzzy \rightarrow defuzzyficação

A fuzzyficação é feita através de uma função de pertinência como descrita na Figura 1 que mapeia os elementos do conjunto para o intervalo de [0,1], que representa o grau de possibilidade do elemento pertencer ao conjunto.

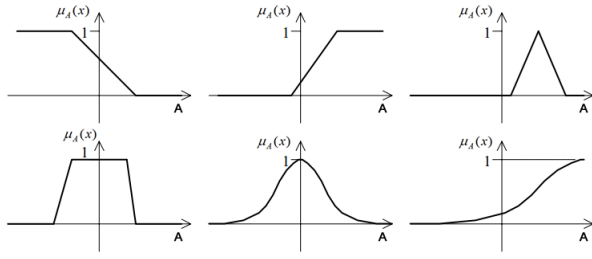


Figura 1. Representação das funções de pertinência mais utilizadas

Portanto, a diferença entre a lógica booleana e a lógica fuzzy pode ser representada pela Figura 2, é importante notar que se x e y forem valores crisp (0 ou 1), então a lógica fuzzy se reduz à lógica booleana.

		Lógica Booleana			Lógica Fuzzy		
x	y	$x \text{ e } y$	$x \text{ ou } y$	não x	Interseção	União	Complemento
					$\min(x,y)$	$\max(x,y)$	$1 - x$
0	0	0	0	1	0	0	1
0	1	0	1	1	0	1	1
1	0	0	1	0	0	1	0
1	1	1	1	0	1	1	0

Figura 2. Lógica Booleana X Lógica Fuzzy

Por fim, há um processo de defuzzyficação do conjunto onde é feita através de algumas estratégias, como por exemplo: método do máximo critério (MAX), método da média dos máximos (MOM) e o método do centro de massa (COA).

No qual,

- Método do máximo critério (MAX): escolhe o ponto onde a função inferida tem seu máximo;
- Método da média dos máximos (MOM): representa o valor médio dentre todos pontos de máximo quando existe mais de um máximo;
- Método do centro de massa (COA): retorna o centro de área da função inferida.

B. Teorema de Bayes

O Teorema de Bayes, relacionado à teorias de probabilidade e estatística, trabalha com conhecimentos prévios como base para a descrição da probabilidade de um evento. Portanto, este mostra como modificar as probabilidades iniciais, considerando novas informações, para obter resultados diferentes.

Sendo assim, a definição formal para o teorema, seria um colário da da lei da probabilidade total, expresso matematicamente na forma da equação (1).

$$P(B|A) = \frac{P(A|B) P(B)}{P(A)} \quad (1)$$

Considerando, A e B eventos e $P(B) \neq 0$.

C. Rede Neural Recorrente

No campo de Inteligência Artificial as Redes Neurais Artificiais, do inglês *Artificial Neural Network* (ANN), estão entre as ferramentas mais utilizadas, uma vez que se busca prover máquinas capacitadas a realizar determinadas atividades mentais do ser humano. A contribuição das ANNs se dão principalmente em problemas de reconhecimento de padrões, classificação e problemas de predição, tanto para dados transversos, quanto para dados de séries temporais ou espaciais.

Uma ANN é composta por uma estrutura de processamento de informações paralela e distribuída que consiste em processar elementos interconectados a partir de sinais unidirecionais [3]. Sua construção é dada pela conexão de unidades de processamento simples, conhecidas como neurônios artificiais que estão dispostos na rede a partir de camadas.

Inspirados em neurônios biológicos, os neurônios artificiais são compostos, basicamente, por sinais de entrada, pesos, função de ativação e sinal de saída, de modo que cada neurônio produz uma sequência de ativação de valores reais [4]. Desse modo, são instituídas relações entre os estímulos de entrada e de saída de um sistema, de forma que essas relações são aprendidas com base em seus próprios dados, a partir de um processo de treinamento, de forma correlata ao aprendizado do cérebro humano [5].

Uma das estruturas básicas de uma ANN é a rede multicamadas, ou redes *feedforward*. Essa estrutura é caracterizada por sua capacidade em modelar tanto estruturas lineares quanto não-lineares. Em uma rede *feedforward* cada camada da rede se conecta à próxima camada, alimentando-se das informações de forma direta, nunca passando por um mesmo nó duas vezes, como demonstrado pela Figura 1. Desse modo, os exemplos de entrada são transformados em uma saída, a partir do Aprendizado Supervisionado, que faz com que os dados brutos das categorias existentes em um *dataset* rotulado sejam mapeados, reconhecendo os padrões existentes para a criação das saídas apropriadas.

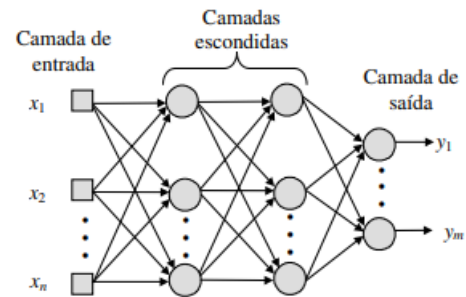


Figura 3. Topologia de rede *feedforward* multicamada

A estrutura de uma Rede Neural Recorrente, do inglês *Recurrent Neural Network* (RNN), é baseada nas redes *feedforward*, porém possuem algumas modificações. Diferentemente das redes *feedforward*, uma RNN utiliza como entrada não somente o exemplo da entrada atual, mas também os anteriores, gerando um sistema de realimentação. As realimentações são dadas pelas saídas de neurônios de determinada camada sendo reintroduzidas como entradas para neurônios de camadas anteriores ou da própria camada. Dessa forma um neurônio pode ser realimentado por sua própria saída, gerando ciclos entre suas unidades, como exemplificado na Figura 2.

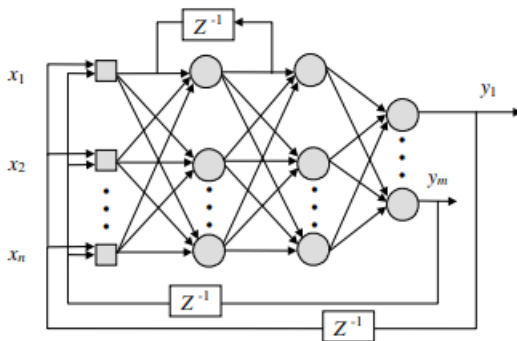


Figura 4. Topologia de Rede Neural Recorrente com realimentação

É dito que esse tipo de rede possui capacidade de memória, reproduzindo seu comportamento com relação aos padrões apresentados anteriormente, possibilitando o processamento de longas sequências de informações. Por conta de tal característica, as RNNs são comumente aplicadas em sistemas combinatórios, sistemas dinâmicos, séries temporais, sistemas de predição e reconhecimento de padrões.

Contudo, ainda que na teoria as RNNs sejam capazes de lidar com dependências de longo termo, as longas sequências, pesquisadores apontam através de experimentos que na prática isso muitas vezes não é possível, uma vez que em métodos de treinamento baseados em gradientes a informação de erro pode ser perdida. [6]

Uma Rede *Long-Short Term Memory* (LSTM) se trata de um modelo recorrente e profundo de redes neurais introduzido por Hochreiter e Schmidhuber[1997] com o intuito de oferecer solução para o problema de perda da informação de erro ocorrente em RNNs quando lidam com grandes sequências de dados [7].

Para tanto, a LSTM possui uma estrutura em cadeia, como demonstrado na Figura 3, que contém diferentes blocos de memória chamados células. A informação é retida pelas células e as manipulações de memória são feitas unidades especiais, chamadas portões (*gates*).

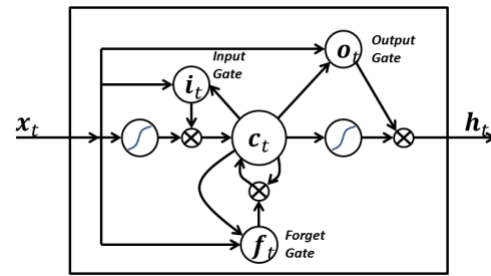


Figura 5. Rede Long-Short Term Memory

Os portões são responsáveis por manter o fluxo do erro constante. Além disso, eles também permitem o ajuste de pesos e o truncamento da sequência quando a informação não for necessária, simbolizando um esquecimento. Essa estrutura permite que as informações dos intervalos anteriores sejam armazenados em células da LSTM, fazendo com que não se percam [8].

D. Rede TDNN - Time delay neural network

As redes de retardo no tempo (TDNN) são um grupo de redes neurais que possuem uma topologia especial, é uma arquitetura de rede neural artificial de multicamadas cujo objetivo é 1) classificar padrões com invariância à mudança e 2) contexto do modelo em cada camada da rede [9].

Classificação invariável por turno significa que o classificador não requer segmentação explícita antes da classificação. Para a classificação de um padrão temporal (como a fala) evita-se determinar os pontos inicial e final dos sons antes de classificá-los [10].

Para modelagem contextual em um TDNN, cada unidade neural em cada camada recebe entrada não apenas de ativações / recursos na camada abaixo, mas de um padrão de saída da unidade e seu contexto. Para sinais de tempo, cada unidade recebe como entrada os padrões de ativação ao longo do tempo das unidades abaixo. Aplicado à classificação bidimensional (imagens, padrões de frequência de tempo), o TDNN pode ser treinado com invariância a deslocamento no espaço de coordenadas e evita segmentação precisa no espaço de coordenadas.

A Rede Neural com atraso de tempo, como outras redes neurais, opera com várias camadas interconectadas de perceptrons e é implementada como uma rede neural avançada. Todos os neurônios (em cada camada) de um TDNN recebem entradas das saídas de neurônios na camada abaixo, mas com duas diferenças [9]:

- 1) Diferentemente dos perceptrons de várias camadas comuns, todas as camadas obtêm entradas de uma janela contextual de saídas da camada abaixo. Para sinais que variam no tempo cada unidade possui conexões com a saída das unidades abaixo, mas também com as saídas com atraso de tempo (passadas) dessas mesmas unidades. Isso modela o padrão temporal das unidades.
- 2) invariância à mudança é alcançada removendo explicitamente a dependência da posição durante a

retropropagação. Isso é feito com cópias deslocadas no tempo de uma rede na dimensão da invariância. O gradiente de erro é então calculado pela retropropagação através de todas essas redes a partir de um vetor de destino geral, mas antes de executar a atualização de peso, os gradientes de erro associados às cópias deslocadas são calculados em média e, assim, compartilhados e as restrições são iguais.

E. Trabalhos Relacionados

Realizando uma revisão sistemática da literatura encontrou-se o estudo do Departamento de Engenharia da Computação e Sistemas Digitais (PCS) da Escola Politécnica da Universidade de São Paulo "Um Estudo da Aplicação de Modelo "Bayes-Fuzzy" para a Previsão de um Mercado Financeiro".

Sendo assim, este estudo constatou o crescimento da bolsa de valores como uma importante fonte de financiamento das empresas e uma importante opção para pessoas físicas começarem a diversificar seus investimentos. Portanto, o mesmo propõe um modelo e estudo da aplicação da técnica Bayes-Fuzzy para a previsão do mercado financeiro baseado em um fundo quantitativo de ações - fundos de investimento que utilizam uma "receita" de aplicação de certos ativos, usufruindo de modelos matemáticos para prever o comportamento do preço dos ativos e assim investir. [11]

Em outras palavras, o trabalho busca mostrar como o teorema de Bayes e a lógica de fuzzy foram utilizados para esse domínio, seguido dos resultados experimentais obtidos com a implementação prática. A base utilizada para a medição do impacto da utilização deste modelo nesse domínio é a comparação com a estratégia de investimento *buy and hold* (comprar e manter), estratégia passiva de investimento, na qual o investidor compra certo ativo - ação negociada em bolsa de valores - e o mantém no portfólio indefinidamente. Por fim, esta técnica foi implementada e testada com ações específicas da bolsa de valores de São Paulo (Bovespa) e sua utilização se mostrou mais rentável que o uso da estratégia passiva anteriormente citada. [11]

Ademais, este modelo é inicial, sendo assim, trata-se de uma versão base para uma aplicação à qual seria colocada em utilização no mercado. Isto pois, a técnica atual realiza a previsão de um determinado fundo quantitativo, o qual deseja-se aplicar, tomando como base o retorno após um investimento direito, sendo assim, considerando o lucro. Contudo, alguns outros aspectos devem também ser considerados, como, os custos das operações de compra e venda junto às corretoras de valores e os riscos do investimento.

Enfim, devido as limitações anteriormente citadas e pelo fato do contexto de aplicação de fundos quantitativos ser bastante incipiente e pouco estudado, em destaque o brasileiro, este modelo apenas propõe uma nova estratégia válida de investimento automatizado. Logo, o diferencial do mesmo reside na abordagem voltada a fundos quantitativos e a consideração de ativos específicos e voláteis, ao invés de portfólio de ativos ou índice Bovespa.

1) *Modelo "Bayes-Fuzzy"*: O modelo proposto se trata de um fundo quantitativo de ações *black box*, portanto, totalmente automatizado, o qual investe em apenas um ativo por vez. Portanto, inicialmente considera-se que o investimento foi realizado com uma quantidade x de capital. Sendo assim, todo este valor sempre estará inteiramente aplicado ou não ao ativo. Por conseguinte, será necessário uma tomada de decisão quanto ao ativo, a venda ou não do mesmo. Desta forma, neste passo é aonde será aplicado o modelo "Bayes-Fuzzy". Consequentemente, a técnica considera que no momento inicial da análise o capital já foi aplicado, o que torna a única tomada de decisão sendo em relação à venda, devido ao padrão adotado e considerando que a decisão "Devo comprar?" é uma condição logicamente complementar a "Devo vender?". Em suma, toda esta explicação pode ser esclarecida com a observação da figura 6. [11]

Além disso, o modelo "Bayes-Fuzzy" realiza a tomada de decisão baseando-se em um histórico de valorizações de um ativo para então prever o que deve ocorrer com o mesmo em um determinado período de tempo, tornando o fundo o qual utilizar desta técnica *trend follower* - fundos que usam como base a análise técnica para fundamentar o modelo.

Por fim, especificamente neste estudo o uso da lógica fuzzy junto com o uso do teorema de Bayes tem por objetivo lidar melhor com a classificação das valorizações do ativo como positivas, negativas ou estáveis antes de mapear as probabilidades.

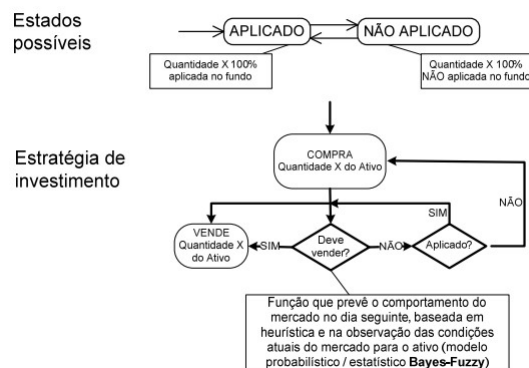


Figura 6. Modelo ou estratégia de investimento.

2) *Resultados*: Neste trabalho, o modelo Bayes-Fuzzy foi implementado em programa de computador para o sistema operacional Windows.

As figuras 7 e 8 mostram os resultados utilizando a aplicação em dois testes com dados reais de valorização de dois ativos: ações da Gerdau e Petrobrás. Considerando que, ambos os ativos possuem alto volume de negociação diário na Bovespa e compõe o índice desta bolsa, sendo assim muito representativos. Enfim, foram coletados dados de preços diários de quatro anos consecutivos para ambos os ativos: 2004, 2005, 2006 e 2007.

Finalmente, para as ações da Gerdau, o fundo deste estudo teve valorização de 40% contra 26% da outra estratégia. A valorização adicional final foi de 14% e a média de 3%. Enquanto, para as ações da Petrobrás, a valorização do fundo foi de 131% contra 124% da estratégia passiva. A valorização adicional final foi de 7% e a média de 15%.

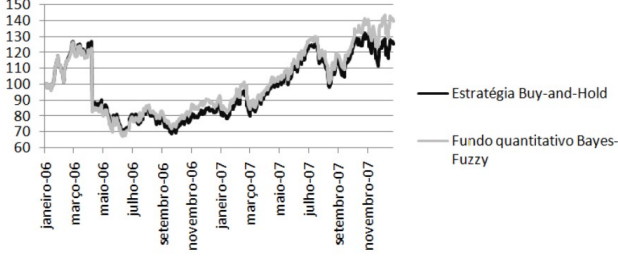


Figura 7. Simulação para as ações da Gerdau.

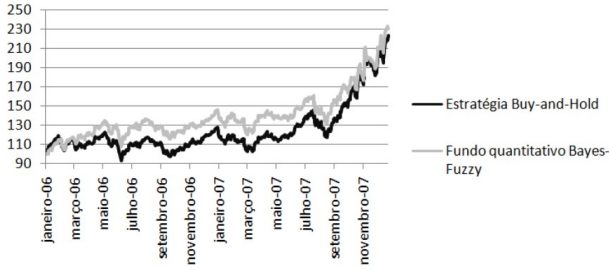


Figura 8. Simulação para as ações da Petrobrás.

IV. METODOLOGIA PROPOSTA

Nesta seção será descrito como será cada etapa do trabalho, desde os datasets utilizados, a utilização dos descritores e como será feita as métricas para comparação dos métodos, além disso, quais métricas de avaliação seriam relevantes.

A. Datasets

Em primeira instância, foi realizado a escolha de quais os datasets de séries temporais do mercado de ações seriam utilizados, os quais foram adquiridos através da plataforma do Yahoo Finanças.

1) **Empresa Magazine Luíza - MGLU3**: Este dataset contém 2123 dados entre 2011 e 2019 a respeito dos dados de flutuações das ações, destes utilizou-se 80% dos dados para treinamento e 20% dos dados para teste. A Figura 9 demonstra a sua linha de tendência desde a sua entrada no mercado de ações.

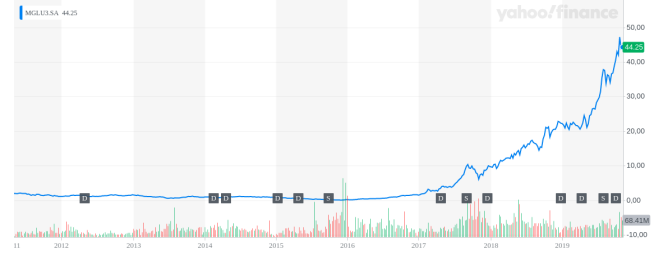


Figura 9. Flutuação do valor das ações do Magazine Luíza

2) **Empresa Oi - OIBR3**: Este dataset contém 4371 dados entre 2002 e 2019 a respeito dos dados de flutuações das ações, destes utilizou-se 80% dos dados para treinamento e 20% dos dados para teste. A Figura 10 demonstra a sua linha de tendência desde a sua entrada no mercado de ações.



Figura 10. Flutuação do valor das ações da Oi

3) **Empresa Petrobrás - PETR4**: Este dataset contém 1267 dados entre 2013 e 2018 a respeito dos dados de flutuações das ações, destes utilizou-se 80% dos dados para treinamento e 20% dos dados para teste. A Figura 11 demonstra a sua linha de tendência desde a sua entrada no mercado de ações.



Figura 11. Flutuação do valor das ações da Petrobrás

B. Métrica

Para este trabalho visto que é um problema que trata de séries temporais, optamos pela utilização do MSE que é o erro quadrado médio que é muito útil na comparação de estimadores. No caso, o MSE de um estimador $\hat{\theta}$ de um parâmetro escalar θ é definido pela Equação 2, é importante notar que quanto mais próximo de zero, mais as soluções são iguais.

$$MSE(\hat{\theta}) = E(\hat{\theta} - \theta)^2 \quad (2)$$

No qual, "E" denota o valor esperado ou variância. t

V. AVALIAÇÃO EXPERIMENTAL

Nesta seção avaliaremos a metodologia de cada abordagem sendo elas: Lógica Fuzzy, Redes Neurais Recorrentes.

Todos os métodos propostos serão validados em experimentos utilizando dados de diferentes ativos da bolsa de valores de forma a averiguar se o mesmo pode ser generalizado ou se possui determinado padrão. Visto isso, foi definido para trabalho o intervalo de 90 dias anteriores ao alvo de previsão.

Para melhor desempenho da rede, os *datasets* utilizados necessitaram de manipulação, descartando dados nulos ou incorretos, bem como colunas irrelevantes para a aplicação do método abordado.

Posteriormente, os dados resultantes da manipulação foram normalizados para que a convergência no treinamento e generalização na previsão fosse facilitada. Para tanto, foi escolhida a normalização Min-Max, onde os valores do atributo são normalizados linearmente, entre $[0,1]$, com base nos valores máximo (max) e mínimo (min) e original (v) de cada atributo. O cálculo realizado para a normalização é descrito por (3).

$$v' = \frac{v - min1}{max1 - min1} \quad (3)$$

A. Lógica Fuzzy

O método proposto será validado em experimentos utilizando dados de diferentes ativos da bolsa de valores, além disso, a quantidade máxima de partições do universo será mantida com valor de 500 para todos os experimentos.

Para melhor desempenho do modelo, utilizou-se uma biblioteca pyFTS que é pronta para oferecer suporte na utilização da Lógica Fuzzy visto que não é uma abordagem trivial.

Primeiramente o modelo é treinado através de 80% do dataset, posteriormente o método é testado e a métrica é aplicada.

B. Rede Neural Recorrente

O modelo da rede neural foi construído, sendo composto, inicialmente, por uma camada LSTM com 100 células de memória, seguidas de outras duas camadas LSTM com 50 células de memória cada. Entre as camadas foi adicionado o *dropout*, uma técnica de regularização caracterizada por descartar neurônios de forma aleatória no processo de treinamento da rede, evitando assim a superadaptação (*overfitting*) do modelo, aumentando assim a capacidade de generalização. Para essa abordagem foi utilizada uma taxa de 0.5 para cada *dropout*.

O modelo também é constituído por uma camada densa que recebe somente um neurônio referente a saída. A função de ativação escolhida para tal camada foi a *Sigmoid*, demonstrada em (3), pois, desse modo a rede retornará valores entre zero e um, que serão arredondados para a classe mais próxima.

$$\sigma(x) = \frac{1}{1 + e^x} \quad (4)$$

A RNN foi treinada utilizando o método de otimização *RMSProp*, um método que utiliza a magnitude das recentes descidas do gradiente para normalizar o próprio gradiente. Além disso, o *RMSProp* atualiza a taxa de aprendizagem de forma automática e escolhe uma taxa de aprendizado diferente para cada parâmetro.

Por fim, a estrutura final do modelo de RNN desenvolvido para o presente trabalho é representado na Tabela I.

Tabela I
REPRESENTAÇÃO DO MODELO DESENVOLVIDO

Layer (type)	Output Shape	Param
LSTM	(None, 90, 100)	40800
Dropout	(None, 90, 100)	0
LSTM	(None, 90, 50)	30200
Dropout	(None, 90, 50)	0
LSTM	(None, 90, 50)	20200
Dropout	(None, 90, 50)	0
LSTM	(None, 50)	20200
Dropout	(None, 50)	0
Dense	(None, 2)	102

C. Rede TDNN - Time delay neural network

O método proposto foi modelado de acordo com a Tabela II, no qual demonstra que o método não tem característica da existência de camadas de Dropout como no método anterior, ela apenas segue retropropagando os atrasos de tempo das outras camadas.

Tabela II
REPRESENTAÇÃO DO MODELO DESENVOLVIDO

Layer (type)	Activation
Dense	Relu
Dense	Relu
Dense	Sigmoid

VI. RESULTADOS E DISCUSSÕES

Nesta seção apresentaremos os resultados encontrados em cada abordagem, além da análise e comparação desses dados de acordo com cada dataset utilizado.

A. Lógica Fuzzy

Os testes utilizando a Lógica Fuzzy foram realizados de modo que a quantidade máxima de partições do universo seja mantida em 500 para todos os experimentos, além disso, por se tratar de dados sequenciais a métrica mais adequada é o MSE que está disposto na Tabela III.

Tabela III
MSE DO TREINO E TESTE PARA OS DIFERENTES DATASETS

Dataset	Treino	Teste
OIBR3	0.004247812879290096	0.0008448517916530
PETR4	0.000132944973311514	0.0002170743944633
MGLU3	0.000006077047386003	1.6320998664804127

1) *Dataset Oi*: Podemos analisar através das Figuras 12 e 13 que para este caso em específico, a abordagem foi extremamente eficiente para prever os casos de teste, pois o seu MSE se aproximou mais do que os outros datasets do zero absoluto como descrito na Tabela III.

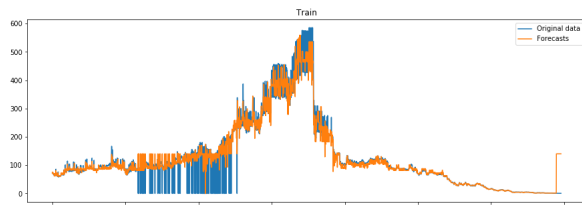


Figura 12. Resultado do treino do modelo utilizando Lógica Fuzzy

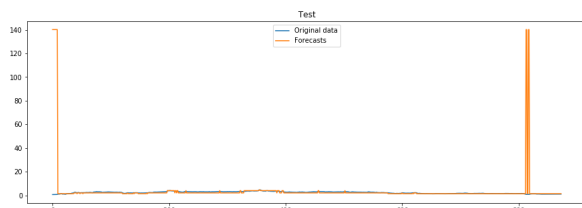


Figura 13. Resultado do teste do modelo utilizando Lógica Fuzzy

2) *Dataset Petrobrás*: Podemos analisar através das Figuras 14 e 15 que para este caso a abordagem também foi muito precisa, ela apresentou resultados um pouco piores do que o "dataset da OI" como descrito na Tabela III, mas houve uma diferença quase imperceptível dos resultados de um para o outro. Portanto, nesse caso a abordagem é adequada para o dataset.

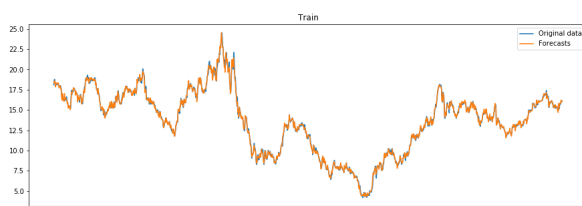


Figura 14. Resultado do treino do modelo utilizando Lógica Fuzzy

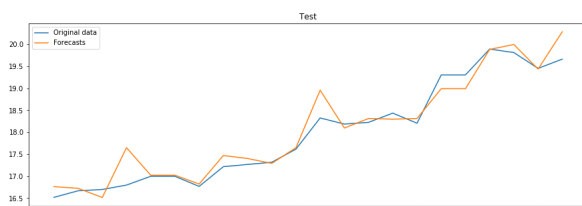


Figura 15. Resultado do teste do modelo utilizando Lógica Fuzzy

3) *Dataset Magazine Luíza*: Neste dataset podemos perceber que o método é inadequado para a análise visto que há uma variação muito alta dos dados que sempre estão

em tendência de crescimento, portanto o classificador não consegue generalizar bem o modelo. E o erro presente no teste do modelo acaba sendo alto como mostra a Tabela III.

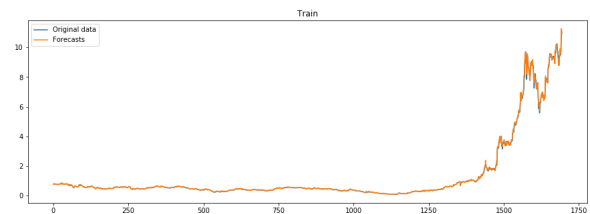


Figura 16. Resultado do treino do modelo utilizando Lógica Fuzzy

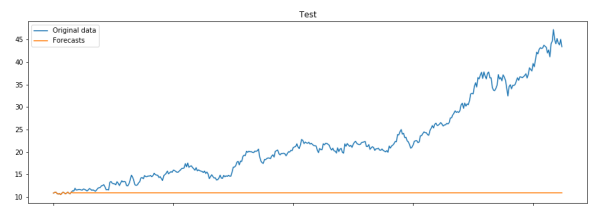


Figura 17. Resultado do teste do modelo utilizando Lógica Fuzzy

B. Rede Neural Recorrente

Os testes utilizando RNN foram realizados de modo que o modelo descrito anteriormente foi treinado ao longo de 100 épocas, os resultados alcançados para cada *dataset* estão descritos ao longo da Tabela III.

Tabela IV
MÉTRICAS ENCONTRADAS PARA O TESTE DOS DIFERENTES DATASETS

dataset	Explained Variance Score	R2 Score	Mean Squared Error	Mean Absolute Error
OIBR3	0.0520955291802321	-1.030995018675297	1.383114390464152	0.9519156442827
PETR4	0.9523065419551643	0.9349951721662217	0.0861093507825221	0.1981040337801
MGLU3	0.62023771124982867	0.5150263854104788	36.4405817664277	3.0609831890156

1) *Dataset Oi*: Neste *dataset* o método proposto não obteve bons resultados. A partir de análises verificou-se que o modelo sofreu *overfitting*, uma vez que não foi capaz de generalizar corretamente. O mesmo previu corretamente o conjunto de treino, como disposto na figura 18, porém, para o conjunto de teste, de acordo com a 19 e com métricas apresentadas na Tabela III, o modelo não prevê as variações dos valores existentes no conjunto, mantendo suas previsões praticamente constantes ao longo do tempo.

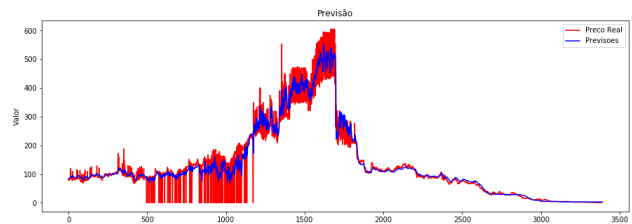


Figura 18. Resultado do treino do modelo utilizando Redes Neurais Recorrentes

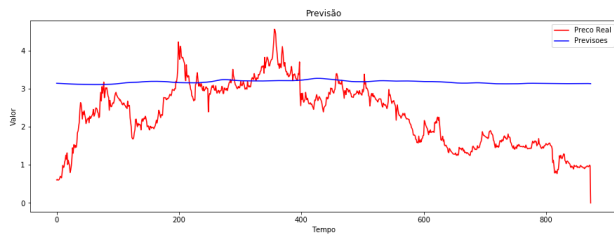


Figura 19. Resultado do teste do modelo utilizando Redes Neurais Recorrentes

2) *Dataset Petrobras*: Para este *dataset* bons resultados foram alcançados com a utilização do modelo, como pode ser constatado pelas métricas dispostas na Tabela III. As previsões realizadas tanto para o conjunto de treinamento, demonstrado pela Figura 20, como para o conjunto de teste, dado pela Figura 21, foram satisfatórias, uma vez que se aproximaram do valor real de cada ação.

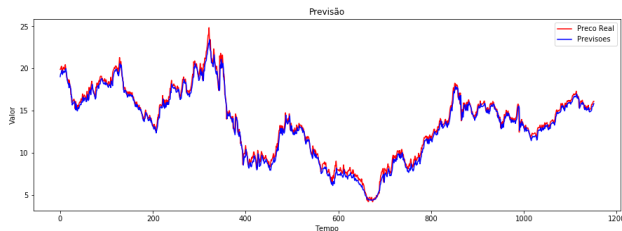


Figura 20. Resultado do treino do modelo utilizando Redes Neurais Recorrentes

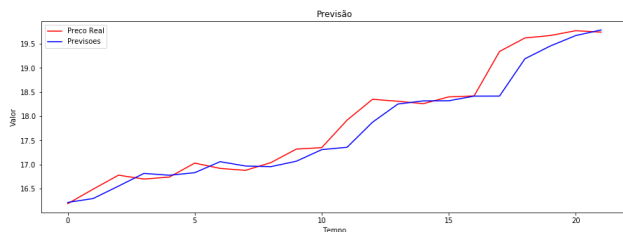


Figura 21. Resultado do teste do modelo utilizando Redes Neurais Recorrentes

3) *Dataset Magazine Luiza*: Analisando os resultados encontrados conclui-se que o método proposto não é ideal para este *dataset*. Isso se dá pois o modelo não foi capaz de generalizar para esse cenário. O modelo proposto se adaptou bem ao conjunto de treino, atingindo bons resultados, como representado pela Figura 22. Contudo, o mesmo não se adaptou corretamente para o conjunto de teste, o que é constatado pelas métricas dispostas na Tabela III e pela Figura 23, visto a taxa de erro nos valores previstos para ações com valores elevados do conjunto.

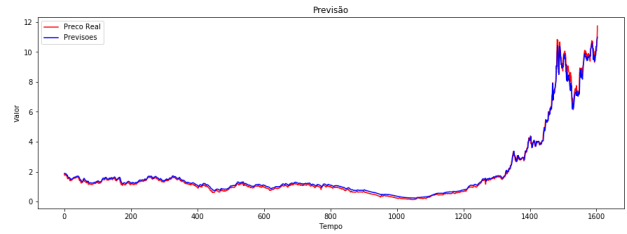


Figura 22. Resultado do treino do modelo utilizando Redes Neurais Recorrentes

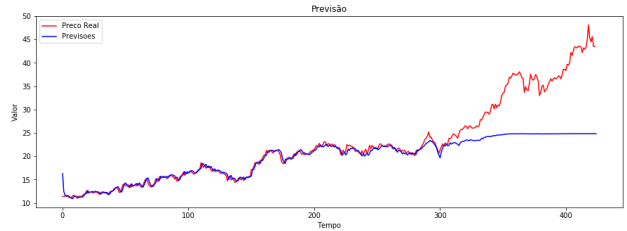


Figura 23. Resultado do teste do modelo utilizando Redes Neurais Recorrentes

C. Rede TDNN - Time delay neural network

Os testes utilizando TDNN foram realizados de modo que o modelo descrito anteriormente foi treinado ao longo de 100 épocas, os resultados alcançados para cada *dataset* estão descritos ao longo da Tabela V.

Tabela V
MÉTRICAS ENCONTRADAS PARA O TESTE DOS DIFERENTES DATASETS

Dataset	Treino	Teste
OIBR3	0.0036657813138932164	0.00004338645746727692
PETR4	0.000880554000183869	0.0006998302934962369
MGLU3	0.00024076262044671703	1.6258615097920004

1) *Dataset Oi*: Neste *dataset* o método obteve um dos melhores SME de todas os *datasets* como na Tabela V, ele se adequou bem ao *dataset* mantendo a linha de tendência com o padrão parecido com o dado real, tendo apenas como diferença o deslocamento no preço como descreve a Figura 24 e a Figura 25.

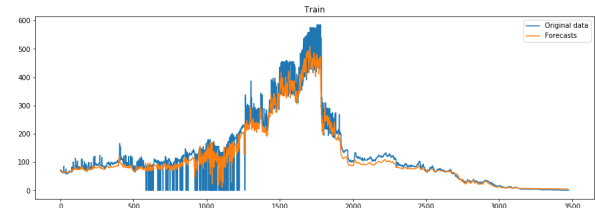


Figura 24. Resultado do treino do modelo utilizando Redes TDNN

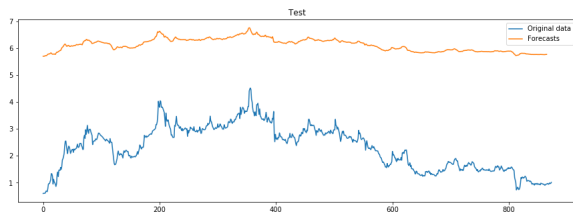


Figura 25. Resultado do teste do modelo utilizando Redes TDNN

2) *Dataset Petrobras*: Para este *dataset* o método obteve bons resultados bem próximo do melhor resultado como descrito na Tabela V, portanto, se adequou corretamente ao tipo de dado, pois ele manteve a linha de tendência similar ao resultado esperado como mostram a Figura 26 e a Figura 27.

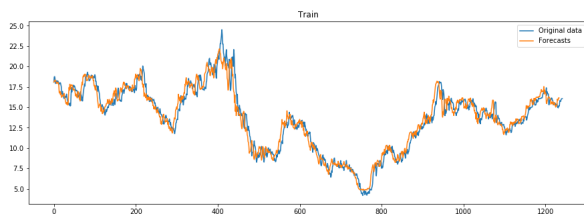


Figura 26. Resultado do treino do modelo utilizando Redes TDNN

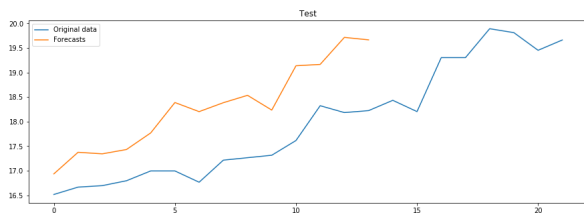


Figura 27. Resultado do teste do modelo utilizando Redes TDNN

3) *Dataset Magazine Luiza*: Para este *dataset* ficou comprovado que o método não é adequado para sua classificação, visto que houve uma alta na tendência dos preços que não seguia o padrão anterior como descrito na Figura 28 e na Figura 29 porque foi determinada por outro fator provavelmente, que pode ser tanto a respeito de política, quanto de uma alta na economia toda do país.

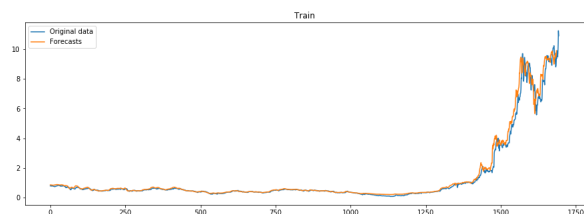


Figura 28. Resultado do treino do modelo utilizando Redes TDNN

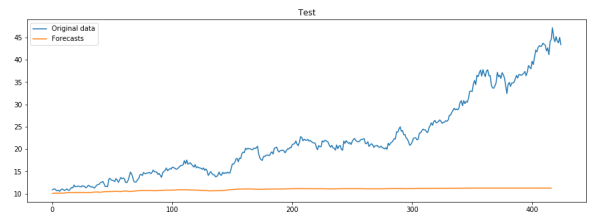


Figura 29. Resultado do treino do modelo utilizando Redes TDNN

VII. CONCLUSÃO E TRABALHOS FUTUROS

Neste trabalho podemos perceber que os diferentes métodos apresentados obtiveram resultados muito bons e muito similares, apesar de um dos *datasets* ter sofrido influência de fatores externos da economia, portanto, a predição de séries temporais é um instrumento importante no entendimento de fenômenos que tem influência do tempo de propagação.

Os resultados obtidos foram bastante satisfatórios o que nos permite em trabalhos futuros analisar fatores externos mais a fundo com a aplicação de linguagem natural e análises probabilísticas para entender o quanto esses fatores influenciam os valores de flutuação do mercado de ações.

REFERÊNCIAS

- [1] Brokers, “O que é a bolsa de valores e mercado de ações?” Maio 2016.
- [2] L. Zadeh, “Fuzzy sets,” *Information and Control*, vol. 8, no. 3, pp. 338–353, Jun. 1965. [Online]. Available: [https://doi.org/10.1016/s0019-9958\(65\)90241-x](https://doi.org/10.1016/s0019-9958(65)90241-x)
- [3] R. Hecht-Nielsen, “Theory of the backpropagation neural network,” *International 1989 Joint Conference on Neural Networks*, pp. 593–605 vol.1, 1988.
- [4] J. Schmidhuber, “Deep learning in neural networks: An overview,” *Neural Networks*, vol. 61, pp. 85–117, Jan. 2015. [Online]. Available: <https://doi.org/10.1016/j.neunet.2014.09.003>
- [5] M. Y. W. Teow, “Understanding convolutional neural networks using a minimal model for handwritten digit recognition,” in *2017 IEEE 2nd International Conference on Automatic Control and Intelligent Systems (I2CACIS)*, Oct 2017, pp. 167–172.
- [6] Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, March 1994.
- [7] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997. [Online]. Available: <http://dx.doi.org/10.1162/neco.1997.9.8.1735>
- [8] A. Graves, M. Liwicki, S. Fernandez, R. Bertolami, H. Bunke, and J. Schmidhuber, “A Novel Connectionist System for Unconstrained Handwriting Recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, pp. 855–868, May 2009. [Online]. Available: <http://dx.doi.org/10.1109/tpami.2008.137>
- [9] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. Lang, “Phoneme recognition using time-delay neural networks,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 3, pp. 328–339, Mar. 1989. [Online]. Available: <https://doi.org/10.1109/29.21701>
- [10] A. Haffner, Patrick e Waibel, “Redes neurais com atraso de tempo em vários estados para reconhecimento contínuo de fala,” in *Anais da 4ª Conferência Internacional sobre Sistemas de Processamento de Informação Neural*, ser. NIPS’91, M. K. P. Inc., Ed., São Francisco, CA, EUA, pp. 135–142. [Online]. Available: <http://dl.acm.org/citation.cfm?id=2986916.2986933>
- [11] J. M. Silva, J. S. Sichman, and P. S. Cugnasca, “Um estudo da aplicação de modelo bayes-fuzzy para a previsão de um mercado financeiro,” *BDBComp*, pp. 2–9, 2009. [Online]. Available: <http://www.lbd.dcc.ufmg.br/colecoes/enia/2009/024.pdf>