UNIVERSIDADE DE SÃO PAULO
FACULDADE DE ECONOMIA, ADMINISTRAÇÃO E CONTABILIDADE
DEPARTAMENTO DE ADMINISTRAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM ADMINISTRAÇÃO

DEEP LEARNING NO MERCADO ACIONÁRIO BRASILEIRO: FATORES QUE POSSIBILITAM PREVISÕES CONSISTENTES PARA A TOMADA DE DECISÃO EM CONDIÇÕES DE RISCO

Felipe Henrique Rubim
Orientador: Prof. Dr. José Roberto Securato

SÃO PAULO
2019

Prof. Dr. Vahan Agopyan
Reitor da Universidade de São Paulo

Prof. Dr. Fabio Frezatti
Diretor da Faculdade de Economia, Administração e Contabilidade

Prof. Dr. Moacir de Miranda Oliveira Júnior
Chefe do Departamento de Administração

Prof. Dr. Eduardo Kazuo Kayo
Coordenador do Programa de Pós-Graduação em Administração

FELIPE HENRIQUE RUBIM

DEEP LEARNING NO MERCADO ACIONÁRIO BRASILEIRO: FATORES QUE POSSIBILITAM PREVISÕES CONSISTENTES PARA A TOMADA DE DECISÃO EM CONDIÇÕES DE RISCO

Dissertação apresentada ao Programa de Pós-Graduação do Departamento de Administração da Faculdade de Economia, Administração e Contabilidade da Universidade de São Paulo, como requisito parcial para a obtenção do título de Mestre em Ciências.

Orientador: Prof. Dr. José Roberto Securato

Versão original

SÃO PAULO

2019

FICHA CATALOGRÁFICA

Elaborada pela Seção de Processamento Técnico do SBD/FEA/USP

Rubim, Felipe Henrique

Deep learning no mercado acionário brasileiro: fatores que possibilitam previsões consistentes para a tomada de decisão em condições de risco / Felipe Henrique Rubim – São Paulo, 2019.

102 p.

Dissertação (Mestrado) - Universidade de São Paulo, 2019.

Orientador: José Roberto Securato.

1. LSTM. 2. Deep Learning. 3. Previsão de Retorno. 4. Processo Decisório. 5. Mercado Acionário Brasileiro. Universidade de São Paulo. Faculdade de Economia, Administração e Contabilidade. II. Título

A minha esposa Jordana, com amor, admiração e gratidão por sua compreensão, carinho, presença e incansável apoio ao longo do período de elaboração desse trabalho.

Ao meu filho Lucas, minha fonte de felicidade.

Agradecimentos

A Deus, pela sabedoria e discernimento para trabalhar na idealização e execução dessa obra.

À Universidade de São Paulo, e à Faculdade de Economia, Administração e Contabilidade, pela excelência de seu corpo docente e recursos essenciais para a minha formação e desenvolvimento dessa dissertação, disponibilizados em um ambiente público, gratuito e de qualidade.

Ao Prof. Dr. José Roberto Securato, pela orientação e valiosos ensinamentos, por me dar instruções claras e objetivas desde o início dessa jornada, sempre generoso ao propor ideias e paciente com os desafios do momento de vida deste recém-papai. Assim, ele se tornou uma referência para mim em conhecimento, didática, paciência e gentileza.

Aos professores da FEA-USP com quem tive a oportunidade de estudar, aprender, e que de uma forma ou outra enriqueceram essa pesquisa com suas ideias, ora transmitidas em sala de aula ora em conversas particulares.

Aos funcionários e demais colegas do PPGA pela disponibilidade e ajudas prestadas em inúmeras solicitações de apoio e por dividirem comigo este processo de aprendizado.

Ao C6 Bank e ao Banco Itaú Unibanco, pelo incentivo à formação de qualidade e apoio em termos de tempo e recursos, além de proporcionarem um ambiente rico em talentos, com quem pude discutir extensivamente muitos dos conceitos aqui tratados.

Finalmente, agradeço especialmente à minha amada esposa Jordana, por me apoiar, me compreender e, além de tudo, por ter me presenteado com a maior inspiração da minha vida: nosso lindo filho Lucas.

“Experiments are fact-finding missions that, over time,

inch scientists toward greater understanding.”

(Catmull, 2014, p. 113)

Resumo

Rubim, F. H. (2019). *Deep learning no mercado acionário brasileiro: fatores que possibilitam previsões consistentes para a tomada de decisão em condições de risco.* Dissertação de Mestrado. Faculdade de Economia, Administração e Contabilidade, Universidade de São Paulo, São Paulo.

A concepção e operação de estratégias de compra e venda de ações por mecanismos de inteligência artificial ganharam escala na última década visando resultados acima da média do mercado para seus usuários, em geral, *hedge funds*, investidores institucionais e grandes bancos de investimentos. O aprendizado de máquina se tornou o arcabouço central no desenvolvimento desses processos, em função de sua extraordinária capacidade de revelação de padrões escondidos em um universo obscuro de dados, que até então apenas dificultava o processo de tomada de decisão tradicional, realizado pelos agentes do mercado de capitais. Em particular, este trabalho visa compreender se no mercado acionário brasileiro é possível tomar decisões consistentes, em um processo diário de seleção de ações, com auxílio de redes neurais recorrentes do tipo *Long Short-Term Memory* (LSTM), que em determinados níveis de profundidade e conectividade são conhecidas como *deep learning*. Os modelos baseados na arquitetura LSTM são propícios para a extração de informações relevantes de longas sequências de dados, portanto, são naturalmente aplicáveis ao problema de previsão do retorno futuro de ações a partir de séries temporais financeiras históricas, como os preços diários das próprias ações, os índices gerais de mercado, a cotação do dólar, entre outras. Assim, foi proposta uma metodologia que utiliza redes neurais LSTM, em conjunto com tais séries temporais, como modelos preditivos para apoiar a tomada de decisão no mercado acionário brasileiro. Com isso, diversos experimentos foram realizados para encontrar a melhor arquitetura e verificar seu potencial econômico, compreendendo as ações constituintes do Índice Bovespa (Ibovespa) no período entre 2007 e 2018. Em linha com as evidências apontadas em estudos prévios, os modelos apresentaram ótimos resultados, mesmo em configurações relativamente simples. Grandes estruturas LSTM tiveram performance ligeiramente inferior às arquiteturas menores. Além disso, observou-se que o conjunto de informações complementares aos preços das ações melhoraram o aprendizado e a acurácia dos modelos, sobretudo os índices de mercado e a taxa de câmbio. Em termos econômicos, o retorno médio anual bruto de uma carteira que simula compras diárias de 10 ações atingiu a marca de 22,3%. Esse resultado supera em mais de 100% o retorno médio das ações constituintes do Ibovespa, no mesmo período. Sobretudo, a curva de rentabilidade dessa carteira foi consistentemente maior que a curva referente ao Ibovespa. As evidências encontradas sugerem, portanto, que a metodologia proposta é capaz de revelar fortes variáveis preditivas e proporcionar ganhos de capital relevantes, através da aplicação das redes neurais LSTM no mercado brasileiro.

Palavras-chave: LSTM. Deep Learning. Previsão de Retorno. Processo Decisório. Mercado Acionário Brasileiro.

Abstract

Rubim, F. H. (2019). Deep learning in the Brazilian stock market: factors that enable consistent predictions for risky decision making. Masters dissertation. School of Economics, Business and Accounting, University of São Paulo, São Paulo.

The design and operation of stock buying and selling strategies by artificial intelligence mechanisms has gained scale over the past decade aiming at above-average market outcomes for its users, generally hedge funds, institutional investors and large investment banks. Machine learning has become the central framework in the development of these processes, due to its extraordinary ability to reveal patterns hidden in a dark universe of data, which until then only hindered the traditional decision-making process undertaken by market players. In particular, this work aims to understand if in the Brazilian stock market it is possible to make consistent decisions, in a daily stock selection process, with the aid of some type of recurrent neural networks called Long Short-Term Memory (LSTM), which at certain depth levels and connectivity are known as deep learning. Models based on the LSTM architecture are lucky to extracting relevant information from long data strings, so they are naturally applicable to the problem of predicting future stock returns from historical financial time series, such as daily stock prices, the general market indexes, the dollar quotation, among others. Thus, a methodology was proposed that uses LSTM neural networks, together with such time series, as predictive models to support decision making in the Brazilian stock market. Several experiments were performed to find the best architecture and verify its economic potential, including the actions that composed the Bovespa Index (Ibovespa) in years between 2007 and 2018. In line with the evidences pointed out in previous studies, the models presented excellent results, even in relatively simple configurations. Large LSTM structures performed slightly lower than smaller architectures. In addition, it was observed that the complementary set of stock price information improved the learning and accuracy of the models, especially the market indexes and the exchanges rates. In economic terms, the average annual return on a portfolio that simulates daily purchases of 10 shares reached 22.3% annually, excluding fees and taxes. This result exceeds by more than 100% the average return of Ibovespa's constituent shares in the same period. Withal, the portfolio yield curve was consistently higher than the Ibovespa curve. Therefore, there is positive evidence on the temporal consistency of the proposed methodology, on its ability to reveal strong predictive variables, and thereafter potential application of LSTM neural networks in the Brazilian market.

Keywords: LSTM. Deep learning. Forecast return. Decision making process. Brazilian Stock Market.

Lista de Siglas

AM	Aprendizado de Máquina
B3	Brasil, Bolsa, Balcão
CAPM	Capital Asset Pricing Model
DJI	Dow Jones Industrial
DNN	Deep Neural Network
EuroStoxx50	Índice EURO STOXX 50
HEM	Hipótese da Eficiência dos Mercados
HSI	Hang Seng Index
IA	Inteligência Artificial
Ibovespa	Índice Bovespa
IPEA	Instituto de Pesquisa Econômica Aplicada
iRMF	Ressonância Magnética Funcional
KOSPI	Korea Composite Stock Price Index
LSTM	Long Short-Term Memory
MTF	Moderna Teoria Financeira
NASDAQ	National Association of Securities Dealers Automated Quotations
RNA	Rede Neural Artificial
RNR	Rede Neural Recorrente
S&P500	Standard & Poor's 500
VSt	Ventral Striatum

Lista de Figuras

Figura 1 – Organização dos elementos de um processo decisório 34
Figura 2 – Função de utilidade assimétrica proposta pela Teoria dos Prospectos..... 39
Figura 3 – Ilustração do conceito de agrupamento de um conjunto de pontos 45
Figura 4 – Ilustração da análise de classificação dos retornos de ITUB4..... 47
Figura 5 – Concepção de um neurônio cerebral 55
Figura 6 – Diagrama de blocos de um neurônio artificial 56
Figura 7 – Grafo do fluxo de sinal em um perceptron 57
Figura 8 – Perceptron de múltiplas camadas..... 59
Figura 9 – Neurônio recorrente 62
Figura 10 – Representação de um neurônio recorrente desdobrado 63
Figura 11 – Sequência de blocos de processamento de uma RNR comum..... 65
Figura 12 – Estrutura conceitual de uma unidade LSTM 65
Figura 13 – Conjuntos de treino, validação e teste 76

Lista de Gráficos

Gráfico 1 – Volume de publicações sobre aplicações de RNAs no mercado de capitais 24
Gráfico 2 – Índice Bovespa 42
Gráfico 3 – Curva do erro de predição *versus* a complexidade de um modelo 51
Gráfico 4 – Taxa média de eventos da variável 73
Gráfico 5 – Retorno acumulado por portfólios *long position*: LSTM simples 1 x 25 80
Gráfico 6 – Retorno acumulado por portfólios *short position*: LSTM simples 1 x 25 81
Gráfico 7 – Retorno acumulado por portfólios *long-short position*: LSTM simples 1 x 25 83
Gráfico 8 – Acurácia do modelo LSTM simples 86
Gráfico 9 – Retorno acumulado por portfólios *long position*: LSTM *deep learning* 3 x 32 88
Gráfico 10 – Retorno acumulado pelos portfólios *long position* via *deep learning* 2 x 100 89

Lista de Tabelas

Tabela 1 – Matriz de confusão 52
Tabela 2 – Estatísticas dos portfólios de compra e venda de ações: LSTM simples 1 x 25 82
Tabela 3 – Resultados do portfólio *long-short*: LSTM simples 1 x 25 83
Tabela 4 – Performance e rentabilidade referente à variável *target* 2 87

Lista de Quadros

Quadro 1 – Oito passos para o treinamento de redes neurais 72
Quadro 2 – Catálogo de fontes de informações 74
Quadro 3 – Previsões com LSTM simples em tempos de desvalorização 84
Quadro 4 – Previsões com LSTM simples em tempos de valorização 85

Sumário

1. Introdução 19
1.1. Situação Problema 19
1.2. Objetivos da Pesquisa 25
1.3. Metodologia 26
1.4. Estrutura do Trabalho 27
2. Fundamentação Teórica 29
2.1. Tomada de Decisão no Mercado Acionário 29
2.1.1. Processo da Tomada de Decisão 29
2.1.2. Escolha de Ativos em Condições de Risco 37
2.1.4.1. Atitude Racional em Relação à Incerteza 37
2.1.4.2. Atitude Irracional em Relação à Incerteza 38
2.1.3. Contexto Decisório no Mercado Acionário Brasileiro 41
2.1.4.3. Índice Bovespa 42
2.2. Aprendizado de Máquina 43
2.2.1. Aprendizado Não-Supervisionado 44
2.2.2. Aprendizado Supervisionado 46
2.3. *Deep Learning* 53
2.3.1. Redes Neurais 54
2.3.2. Perceptrons de Camada Única 57
2.3.3. Perceptrons de Múltiplas Camadas e o Método *Back-Propagation* 58
2.3.4. Redes Neurais Recorrentes 62
2.3.4.1. Redes Neurais Recorrentes do Tipo *Long Short-Term Memory* 64
2.3.5. *Deep Learning* no Mercado Acionário 67
3. Metodologia 71
3.1. Introdução 71
3.2. Hipóteses 71
3.3. Método de Pesquisa 71
3.3.1 Definição de Variáveis 72
3.3.2 Coleta dos Dados 74
3.3.3 Pré-processamento dos Dados 75
3.3.4 Conjuntos de Treino, Validação e Teste 75
3.3.5 Paradigma da Rede Neural 76
3.3.6 Critérios de Avaliação 76
3.3.7 Treinamento da Rede Neural 77
3.3.8 Implementação 78

3.4. Ambiente Computacional 78
4. Análise dos Resultados 79
4.1. Modelos LSTM Simples para Ações do Ibovespa 79
4.2. Modelos LSTM *Deep Learning* para Ações do Ibovespa 88
5. Considerações Finais 91
Bibliografia 95
Apêndice 101

1. Introdução

1.1. Situação Problema

O processo de seleção de ações proposto por (Markowitz, 1952) marcou o início da moderna teoria de carteiras. No trabalho original, o processo foi dividido em dois estágios. O primeiro, que se inicia pela análise de informações e uso de experiências passadas, busca determinar as expectativas sobre a performance futura de um conjunto de ativos disponíveis. O segundo estágio, por sua vez, parte das expectativas futuras mais relevantes e culmina na escolha de um portfólio com potencial de satisfazer certos objetivos.

Com sua abordagem analítica e matemática, Herry Markowitz desenvolveu uma solução elegante para o segundo estágio, buscando equilibrar a relação entre o retorno esperado e o risco aceitável pelo investidor. Assim, do modelo média-variância proposto, obtém-se a fronteira eficiente, sobre a qual, estão as carteiras com o maior retorno médio possível para cada nível de risco. Um imenso esforço intelectual foi empreendido a partir desta base teórica (Fabozzi, Gupta, & Markowitz, 2002).

Os conceitos sobre a racionalidade dos investidores, a natureza da distribuição de retornos dos ativos e diversificação de carteiras, implícitos no modelo média-variância, foram questionados e analisados por inúmeros pesquisadores ao longo do tempo. Com isso, novos conceitos e alternativas surgiram para tratar os desafios encontrados na prática do processo de otimização de portfólios. Em particular, um dos principais assuntos refere-se a sensibilidade dos resultados às estimativas de retorno e às medidas de covariância dos ativos, que são os principais dados de entrada do processo decisório (Elton & Gruber, 1997; Kolm, Tütüncü, & Fabozzi, 2014).

Com a modernização dos sistemas computacionais e a alta disponibilidade de informações nos últimos anos, muitos pesquisadores estão se dedicando ao desenvolvimento de algoritmos no âmbito da Inteligência Artificial (IA) para auxiliar a tomada de decisão em diferentes segmentos do mercado financeiro. Em linha com a necessidade do primeiro estágio fundamentado por (Markowitz, 1952), estes métodos estão sendo aplicados, com evidências de sucesso, na difícil tarefa de predizer a performance futura de empresas e suas respectivas ações negociadas em bolsas de valores, conforme Cavalcante et al. (2016).

No entanto, há pelo menos dois grandes argumentos contrários à possibilidade de se prever com sucesso o movimento do mercado. Um deles refere-se à teoria do mercado eficiente de (Fama, 1970), baseada na hipótese de que os mercados são robustos em relação a informação disponível no ato do investimento. Com isso, seria impossível para um agente econômico superar consistentemente a performance do mercado sem tomar deliberadamente mais risco.

A ideia de que o preço de um ativo financeiro segue um processo estocástico aleatório, quando negociado em um ambiente competitivo, está intimamente ligado ao primeiro argumento, conforme (Fama, 1995). Na teoria do *random walk*, as mudanças nos preços de ações negociadas em bolsas de valores, por exemplo, não podem ser previstas com base nos movimentos passados, tampouco pela tendência dos preços no histórico recente.

Mas o potencial preditivo das técnicas de aprendizado de máquinas aumentou consideravelmente nos últimos anos devido a proposição e implementação de programas e algoritmos mais eficientes, integrados às novas infraestruturas computacionais para *big data analytics*. Redes Neurais Artificiais (RNAs) complexas e com muitas camadas de neurônios, dedicadas ao aprendizado de padrões complexos e não lineares, chamadas *Deep Neural Network* (DNN) ou simplesmente *deep learning*, são consideradas o estado da arte neste campo e estão disponíveis em softwares *open source* (Fischer & Krauss, 2018).

O mercado financeiro já utiliza toda essa tecnologia, encapsulada em sistemas inteligentes de execução de ordens de compra e venda de ativos, visando alcançar resultados acima da média. Em particular, estratégias *high frequency trading* são operadas exclusivamente com IA, em mercados líquidos como a bolsa eletrônica NASDAQ, nos Estados Unidos (Cavalcante et al., 2016).

A tecnologia e a rapidez no fluxo de informações estão influenciando o processo decisório no mercado financeiro, mas é possível, de fato, aprender os padrões intrínsecos do mercado de ações? Quais são as variáveis que permitem um aprendizado consistente? A IA pode auxiliar o investidor no processo de tomada de decisão para vencer o mercado? Como se comportam as estratégias baseadas nas previsões obtidas com *deep learning* em momentos de alta volatilidade? Informações derivadas do mercado de câmbio e de índices futuros influenciam a capacidade preditiva de um sistema inteligente?

As bases da moderna teoria de portfólios foram definidas pelos trabalhos de (Markowitz, 1952), (Sharpe, 1964), (Ross, 1976) e (Merton, 1986), dentre outros. Estes introduziram os conceitos sobre carteiras eficientes, sobre o *Capital Asset Pricing Model* (CAPM), sobre fatores de risco que poderiam impactar a formação de preços, como fatores econômicos e tamanho das firmas e, finalmente, a ideia de que deveria haver um prêmio para o investidor que incorre em maior incerteza devido à assimetria de informação referente a um certo ativo, respectivamente. Estas teses harmonizam com a hipótese do mercado eficiente de (Fama, 1970), dito anteriormente. Uma implicação direta desta afirmação é que seria impossível, para um agente financeiro, superar o retorno médio do mercado consistentemente.

Autores destacados apontam evidências sobre os benefícios da aplicação de modelos de previsão em ambientes de risco. Para Securato (2007), no âmbito das finanças, "é em função de previsões de resultados que são tomadas as decisões" (p. 20). Por sua vez, Johannes et al. (2014) concordam que a previsibilidade é uma característica relevante no momento da tomada de decisão sobre o aporte de capital em um determinado investimento ou em uma carteira de ativos. De acordo com os autores, os benefícios da previsibilidade são significativos e existem "fortes evidências de que os investidores podem usar a previsão sobre o retorno dos ativos para melhorar o desempenho de seus portfólios, desde que incorporem uma série de características sensíveis em seus problemas de otimização".

Para se obter previsões consistentes, é preciso cuidado com a preparação e engenharia de dados, e uma definição adequada das técnicas consideradas no empreendimento, especialmente no mercado financeiro. Kaastra e Boyd (1996) propuseram uma sequência metodológica para o desenvolvimento de modelos preditivos baseados em RNAs, seguida por vários autores desde então. Outro exemplo de engenharia de dados é dado por Chong et al. (2017), que percebeu que o desempenho das RNAs em desenvolvimento melhorou após diversas informações normalizadas, combinadas com os resíduos de um modelo autorregressivo, serem incluídas como inputs da rede. Neste caso, em que estruturas *deep learning* estavam sendo avaliadas, nota-se que os resíduos introduzidos no modelo causaram impacto positivo na performance *out-of-sample* da previsão; por outro lado, testes incorporando os resíduos das RNAs em um modelo linear não obtiveram melhoria significativa. Segundo Chong et al. (2017), este resultado é possível devido a capacidade das RNAs atuarem com relações não lineares complexas entre as variáveis de entrada e saída.

No braço da bolsa chinesa dedicado às ações de empresas de crescimento rápido, o *Shenzhen Growth Enterprise Market*, Zhang et al. (2018) aplicaram um conjunto de métodos de aprendizagem de máquina para alcançar maior acurácia na previsão do movimento dos preços. Tal combinação resultou em níveis entre 70% e 80% de acurácia em horizontes com 30 a 40 dias, respectivamente. Estes resultados foram obtidos com variáveis explicativas derivadas das próprias cotações das ações, com indicadores técnicos e com o volume de negócios. A inclusão das variáveis referentes aos indicadores técnicos, neste caso, alavancou cerca de 3% a performance do *framework* proposto, na média. A consistência dos resultados em termos do desvio padrão da acurácia em diferentes amostras de treino e teste, em média igual a 3,7%, mostra evidências sobre capacidade deste *framework* prever bem em condições de uso. No entanto, o erro de generalização do modelo não é apresentado de forma clara.

Alguns exemplos de aplicação de RNAs no mercado brasileiro apresentam evidências sobre o seu potencial preditivo. Observam-se aplicações combinadas com o modelo de seleção de carteira de Markowitz, assim como estratégias mais simples de operações de compra e venda de ações sugeridas pela probabilidade de alta e baixa dos preços dos papéis, respectivamente, entre outras abordagens.

Freitas et al. (2001) usaram RNAs para estimar o retorno de 66 ações negociadas na Bolsa de São Paulo. Estas estimativas mais precisas foram, então, utilizadas como *proxy* do retorno esperado no modelo de Markowitz. Nesse caso, o retorno médio da carteira ótima obtida por Freitas et al. (2001) foi cerca de 12% acima do modelo média-variância de referência. É necessário cuidado com conclusões definitivas, pois o período analisado foi de apenas 21 semanas, considerado curto para os padrões de pesquisas internacionais. Além disso, como os próprios autores sugerem, é necessário a verificação de especificações mais robustas das redes neurais desenvolvidas.

Em Oliveira et al. (2013), os resultados obtidos com a metodologia proposta exclusivamente para a negociação das ações da Petrobrás foram excelentes, chegando em uma acurácia de 93% na tarefa de prever a direção do preço das ações da empresa, com até uma semana de antecedência. No entanto, não há indícios suficientes sobre a capacidade de generalização desta proposta para um conjunto amplo de ações, como o conjunto que constitui o Índice Bovespa, no qual um investidor poderia efetivamente selecionar ativos com o objetivo de

maximizar seu retorno esperado, sob a condição de manter o risco dentro de seu limite de tolerância.

Já em sua tese, Paiva (2014) apresentou uma abordagem exitosa no treinamento de RNAs que puderam aprender os “padrões de comportamento das cotações dos preços das ações”. As melhores configurações das redes treinadas foram capazes de prever, com erro relativamente baixo, as cotações futuras de um dado conjunto de ativos. Os retornos obtidos com a estratégia de compra e venda adotada por Paiva foram satisfatórios, chegando em uma rentabilidade anualizada de 70,56% com sua carteira teórica de ações.

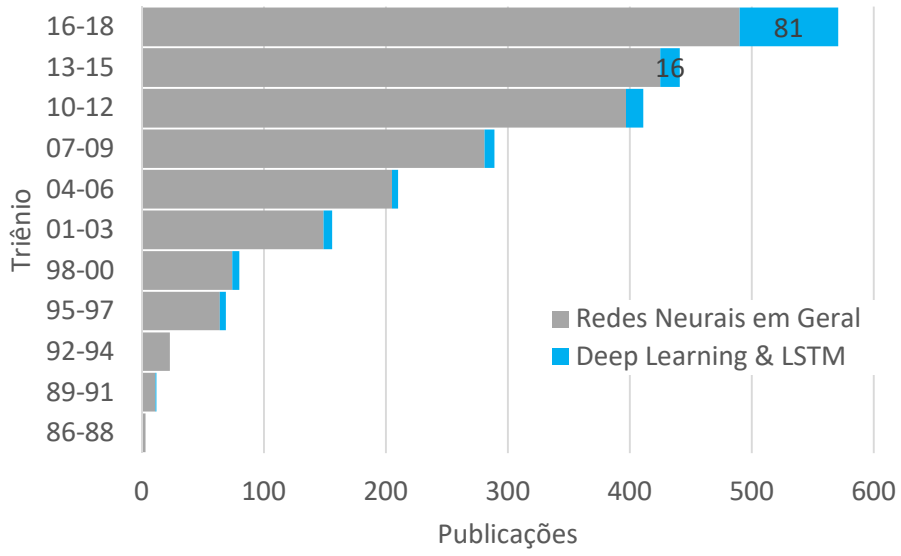
Mais recentemente, (Nelson, Pereira, & Oliveira, 2017) utilizaram RNRs do tipo *Long Short-Term Memory* (LSTM) e obtiveram bons resultados para uma certa seleção de ações da bolsa brasileira. Neste caso, os autores conseguiram níveis de acurácia em torno de 56% em experimentos que buscavam prever se uma certa ação valorizaria ou não de um período para outro. Em particular, o estudo considerou o histórico de cotações e indicadores técnicos derivados do preço dos papéis como entrada do *framework* proposto para operar em períodos de alta frequência.

Apesar destes exemplos de relativo sucesso, é necessário investigar a consistência destes modelos em um período suficientemente grande, com diversidade de ações. Outro ponto importante, como será visto adiante, reside na necessidade de se minimizar o erro de generalização, visando a consistência das previsões futuras. Quando se treina um modelo de classificação, supõe-se que os dados futuros serão similares aos dados de treinamento, logo, minimizar o erro de treinamento significa se aproximar do menor erro de generalização possível (Ben-David & Shalev-Shwartz, 2014).

Há espaço para mais pesquisas desse tipo no Brasil, pois o mercado brasileiro ocupa a 19º posição no ranking de volume financeiro, em dólares, negociado por meio eletrônico, conforme as estatísticas disponíveis no relatório da confederação mundial de bolsas (World Federation of Exchanges, 2017). No mesmo relatório, a bolsa brasileira aparece em 12º lugar em quantidade de negócios, realizados em 2017, sendo um total de aproximadamente 245 milhões de negócios fechados. Portanto, pesquisas que apontem formas consistentes de previsões para a tomada de decisão podem ser implementadas neste mercado.

Nas últimas duas décadas o volume de pesquisas no campo financeiro envolvendo RNAs cresceu substancialmente no mundo. As RNRs profundas, com muitas camadas de neurônios, chamadas *deep learning*, estão ganhando espaço e se destacam entre as aplicações no mercado financeiro internacional. Em particular, a arquitetura LSTM, concebida originalmente por (Hochreiter & Schmidhuber, 1997) com o propósito de atuar com longas sequências de informações, aparece como uma das mais promissoras técnicas para a modelagem de dados que envolvem séries de tempo, como o preço de ações.

Gráfico 1 – Volume de publicações sobre aplicações de RNAs no mercado de capitais



Em levantamento feito na base de dados Scopus, é possível verificar a evolução das publicações na área de mercado de capitais utilizando-se RNAs em âmbito internacional, incluindo o Brasil. Em especial, cresce aceleradamente o número de aplicações deep learning nos últimos anos. Fonte: Scopus Preview. Gráfico elaborado pelo autor.

Esta pesquisa visa analisar **se no mercado acionário brasileiro é possível tomar decisões consistentes no processo de seleção de ações, utilizando-se *deep learning* para prever o retorno das ações** em função de suas cotações históricas, das finanças das empresas emissoras, do cenário macroeconômico e dos preços de commodities.

O desenvolvimento analítico desta pesquisa seguirá a linha desenvolvida por pesquisadores dedicados ao uso de IA nos mercados internacionais que buscam compreender a relevância (i) do histórico de cotações (Fischer & Krauss, 2018; Krauss, Do, & Huck, 2017; Zhang et al.,

2018), (ii) dos indicadores financeiros das empresas e dos principais índices econômicos nacionais (F. A. de Oliveira et al., 2013) e (iii) dos preços das principais *commodities* (Sadorsky, 2014) para a capacidade preditiva dos modelos baseados em RNAs.

As análises serão realizadas sobre o conjunto de ações listadas no Índice Bovespa, que reúne as ações mais negociadas da bolsa de São Paulo, chamada B3. A relevância dos preços de commodities é maior para certos subsetores do mercado. Mais especificamente, àqueles cujos produtos ou serviços, diretamente ligados as commodities, contribuem para a formação das receitas das companhias. Dessa forma, nesta pesquisa será analisada a relevância dos preços de petróleo e minério de ferro para a previsibilidade do modelo proposto.

Esta pesquisa busca entender e apresentar a eficácia dos métodos de aprendizado de máquina como ferramentas de auxílio a tomada de decisões no mercado acionário brasileiro, visto que apenas 6,12% dos artigos publicados em periódicos nacionais referem-se ao mercado de capitais, enquanto a média internacional está acima de 14%. Ainda assim, a academia brasileira passou de uma publicação média de 4 artigos para 22, desde os anos de 1980 até a atualidade, respectivamente, o que mostra o aumento de interesse e relevância crescente do tema, segundo Pinheiro et al. (2018).

Este trabalho também contribui para a formação da base científica, na área de finanças, sobre o processo decisório auxiliado por modelos de aprendizado de máquina. Neste caso, treinados para encontrar padrões complexos nas informações da bolsa brasileira. Está voltada para os agentes do mercado brasileiro de ações, de investidores individuais a grandes fundos, que podem utilizar a metodologia apresentada para ampliar a capacidade de suas estratégias, assim como aos pesquisadores acadêmicos, que terão à disposição evidências sobre a forma e capacidade da IA alcançar desempenho consistente no processo de seleção de carteiras.

1.2.Objetivos da Pesquisa

Neste contexto, são propostos dois objetivos para este trabalho, a saber:

a) verificar a eficiência de arquiteturas simples de redes neurais do tipo LSTM no processo decisório de seleção de ativos no mercado de ações brasileiro, em termos de acurácia e retorno financeiro. Como objetivos secundários desta análise, será investigada a relevância das cotações históricas das ações, dos indicadores financeiros das empresas

subjacentes, dos indicadores macroeconômicos do país, índices futuros e preços de *commodities* para a previsão de retornos das ações que compõem o Índice Bovespa.

No procedimento de análise deste objetivo, serão criadas diversas variáveis independentes baseadas nas informações citadas, assim como uma variável dependente binária relativa ao retorno do ativo num dado período. Com isso, serão treinados modelos de RNAs do tipo LSTM, conforme o método apresentado no Capítulo 3.

b) verificar a eficiência de arquiteturas *deep learning* de redes neurais do tipo LSTM no processo decisório de seleção de ações listadas no Ibovespa, em termos de acurácia e retorno financeiro. Analogamente ao item anterior, todas as informações disponíveis em séries temporais serão incluídas no processo de aprendizado das redes neurais.

Para atingir os objetivos acima, RNAs do tipo LSTM serão desenvolvidas e utilizadas na previsão do retorno das ações do Índice Bovespa, conforme o trabalho de Fischer e Krauss (2018). Os pesos das redes serão calculados com a técnica iterativa *back-propagation* associado a estratégia de otimização de hiperparâmetros *grid search*, proposta por (Bergstra & Bengio, 2012) por ser eficiente em conjuntos de dados complexos. Em seguida, a seleção da carteira teórica de ações se dará em função da probabilidade de alta de cada ação disponível no período, conforme a estratégia *long-short trading*, também utilizado por (Huck, 2009, 2010). Nesse caso, são simuladas a compra de k ações com as maiores probabilidades de alta e a venda das ações com as maiores probabilidades de baixa. As operações são realizadas diariamente e, para fins de comparação, serão considerados três cenários com $k \in \{10, 15, 20\}$. Então, a cada dia do horizonte de testes, computa-se a média de retornos dos papéis e compara-se ao retorno médio das ações listadas no Ibovespa. Os detalhes do procedimento serão apresentados no capítulo 3, Metodologia.

1.3. Metodologia

De acordo com (Martins & Theóphilo, 2009) este estudo, dentro do campo das ciências sociais, pode ser classificado como um trabalho quantitativo, empírico-analítico, bibliográfico, descritivo e estatístico. Por outro lado, segundo (Lakatos & Marconi, 2011) esta pesquisa também segue o método hipotético-dedutivo, visto que apresenta um problema fruto de

evidências e teorias já apresentadas, havendo a partir deste arcabouço, a possibilidade de se formular novas hipóteses e realizar testes com o propósito de refutá-las ou corroborá-las.

Esta pesquisa se encaixa no perfil bibliográfico, segundo (Martins & Theóphilo, 2009), devido ao levantamento de toda a fundamentação teórica, evidências empíricas e principais definições apresentadas no Capítulo 2. Outrossim, esta pesquisa é descritiva, pois em dados momentos são apresentadas descrições dos dados sob análise.

O trabalho é dito quantitativo, empírico-analítico e estatístico em função da utilização de técnicas quantitativas e estatísticas, como classificações e uso de séries temporais numéricas. Técnicas estatísticas também são empregadas para auferir o resultado médio de simulações e métricas de sucesso do método empregado.

1.4. Estrutura do Trabalho

Esse trabalho está organizado de acordo com estrutura a seguir.

No capítulo de Introdução são apresentados os objetivos da pesquisa, devidamente justificados e contextualizados por um panorama sobre o uso das RNAs como ferramentas preditivas de auxílio a tomada de decisões no mercado acionário.

No capítulo 2 de Fundamentação Teórica serão apresentados três assuntos. Primeiro, são apresentados os fundamentos do processo decisório e os principais desafios que o decisor enfrenta no mercado acionário. Em seguida serão apresentados os principais conceitos do aprendizado de máquinas. A última parte dedica-se ao estudo das redes neurais artificiais primárias até as arquiteturas de redes mais avançadas, que implementam o conceito de *deep learning*, além de apresentar algumas aplicações relevantes no mercado acionário.

No capítulo 3 de Metodologia são apresentadas formalmente as hipóteses e a descrição do método de pesquisa, assim como a definição do ambiente computacional disponível para o estudo e execução do protocolo de simulações.

O capítulo 4 de Análise dos Resultados é dividido de acordo com os objetivos da pesquisa, sendo que na primeira parte serão analisados os resultados obtidos com estruturas simples de redes neurais do tipo LSTM, alimentadas por diversas variáveis do mercado financeiro. Na

segunda parte serão analisados os resultados obtidos com estruturas LSTM *deep learning*, mais complexas em termo de parâmetros.

No capítulo 5 encontram-se as Considerações Finais, onde são relatados os principais achados da pesquisa, assim como suas limitações e oportunidades para futuro empreendimento.

2. Fundamentação Teórica

2.1. Tomada de Decisão no Mercado Acionário

2.1.1. Processo da Tomada de Decisão

Para a realização de suas atividades diárias, pessoais e profissionais, as pessoas precisam tomar decisões. Algumas destas decisões são simples, cotidianas e de baixa importância relativa. Nesses casos, as escolhas são feitas naturalmente e pouco afetam o decurso natural da vida. Por outro lado, decisões difíceis e complexas requerem, por parte dos indivíduos, diferentes níveis de responsabilidade, atenção e discernimento (Securato, 2007). Em particular, os investidores do mercado acionário precisam decidir, constantemente, se vão comprar ou vender um certo ativo, ou até mesmo se não farão nada. Ao assumirem riscos em busca dos retornos esperados, estão sujeitos ao sucesso ou fracasso de suas operações e até mesmo de suas carreiras.

Ainda é difusa a compreensão do processo da tomada de decisão no mercado acionário, permeado por investidores individuais e institucionais, que aplicam seu próprio capital ou de terceiros, em ativos de toda natureza. Vários estudos foram realizados e muitas teorias foram propostas ao longo do tempo, resultando em obras ora prescritivas, ora descritivas sobre o tema. Nesta seção serão apresentados alguns conceitos importantes sobre o processo decisório. Inicia-se a discussão pelo *modus operandi* cognitivo e sua relação com o mecanismo neural humano de processamento das informações. Em seguida, aborda-se paradigmas mais conceituais sobre o ato de decidir consciente e inconscientemente. Da ideia de que uma pessoa toma decisões de forma consciente e otimizada emerge a teoria do agente racional, que segue um conjunto de regras que lhe permitem chegar à melhor decisão. Como contraponto a esta vertente, são apresentadas evidências da ação inconsciente do decisor em muitas de suas empreitadas, sujeito aos vieses comportamentais que podem levá-lo para longe da decisão ótima. Por fim, discute-se a ascensão da tecnologia da informação e dos sistemas inteligentes de apoio à tomada de decisões no âmbito financeiro.

Tem-se buscado compreender o processo decisório à luz das principais áreas do conhecimento, num ambiente multidisciplinar, incluindo Finanças, Ciências Econômicas,

Matemática, Neurociência, Psicologia, Filosofia, entre outras. Sob o ponto de vista das Ciências Cognitivas, (Yamamoto & Valentova, 2018, p. 233) definem em linhas gerais que "a tomada de decisão é um processo ativo de escolha entre duas ou mais alternativas possíveis, que pode ser realizado em cenários com as mais diversas configurações". Ainda segundo (Yamamoto & Valentova, 2018), diante de situações que envolvem incertezas, a tomada de decisão é uma importante tarefa, embora afetada por vieses e valores pessoais, empreendida pelos indivíduos visando a "otimização comportamental". Assim, em tese, as pessoas buscam as melhores alternativas para si ou para a sociedade em que vivem.

Estudos neurofisiológicos recentes mostram evidências sobre como o mecanismo neural humano age durante o processo decisório. Conjectura-se que a ação de decidir, com base nas análises da atividade cerebral, seria executada em dois estágios. Num primeiro instante, a valoração das alternativas de escolha é realizada nos setores ventromediais do córtex pré-frontal. No segundo estágio, uma escolha é feita, de acordo com as preferências do indivíduo ponderadas pelo valor de cada item avaliado, envolvendo as áreas pré-frontal lateral e parietal do cérebro. Estas regiões do cérebro são, portanto, responsáveis pelo processamento de informações, pelo raciocínio mais elaborado e pelas tomadas de decisões. Nota-se que a tomada de decisão exige a utilização de áreas do cérebro especializadas, interligadas e acessadas numa ordem pré-definida, como em um modelo algorítmico, de acordo com (Kable & Glimcher, 2009).

As decisões financeiras estão entre as tarefas mais importantes na vida de uma pessoa. Em cada tarefa, o indivíduo utiliza sua capacidade intelectual para processar uma série de informações e tomar uma decisão adequada, de acordo com as alternativas disponíveis. O estudo da base cognitiva das decisões financeiras ganhou tração nos últimos anos, como mostra o trabalho de (Frydman & Camerer, 2016b). Nesta revisão da literatura, os autores reuniram recentes descobertas sobre as formas de influência do processo cognitivo sobre a tomada de decisões financeiras. Tais achados foram viabilizados pelo cruzamento e análise de novas fontes de dados, como buscas no *Google Search*, registros de operações em *Homebrokers* e imagens por Ressonância Magnética Funcional (iRMF).

Vieses comportamentais geralmente influenciam o processo decisório de compra e venda de ações, um dos tipos de decisões financeiras mais desafiadoras para as pessoas. (Frydman & Camerer, 2016a) apresentam evidências de que, na presença de alguns destes vieses, a

mesma região cerebral é ativada durante a ação cognitiva do investidor. Neste contexto, experimentos com iRMF encontraram aumento de atividade neural, mais especificamente na região *Ventral Striatum* (VSt) do cérebro, simultaneamente à observação do efeito de disposição. O efeito de disposição refere-se à tendência do investidor de carregar por mais tempo operações com prejuízo e encerrar rapidamente operações lucrativas. Observou-se também que as atividades neurais na região VSt aumentam sensivelmente durante a ocorrência do viés de recompra, que compreende a preferência de recomprar ativos cujo valor diminuiu ao invés de recomprar ativos cujo valor aumentou após a primeira venda.

Os resultados mostram que estes dois efeitos comportamentais têm forte correlação entre si em pessoas suscetíveis aos dois efeitos citados acima, sugerindo que um mecanismo neural comum, ativado pela ação cognitiva, pode ser o responsável pela geração de ambos os vieses. Ainda nesse trabalho, (Frydman & Camerer, 2016a) especulam sobre a existência de mais mecanismos neurais ligados às atividades cognitivas derivadas de outros processos decisórios, cuja confirmação carece de pesquisas mais específicas.

Passando do domínio neurofisiológico para um nível mais conceitual e abstrato do entendimento do processo decisório, (Kahneman, 2013) discute a existência de dois sistemas mentais, denominados Sistema 1 e Sistema 2, e seus papéis na tomada de decisões das pessoas. Os termos foram originalmente proposto pelos psicólogos Keith Stanovich e Richard West em (Stanovich & West, 2000), para descrever os dois procedimentos mentais, e aparentemente distintos, depreendidos de seus estudos.

Em termos gerais, o pensamento no Sistema 1 refere-se ao processo intuitivo, normalmente rápido, automático, direto, implícito e emocional. A maioria das decisões na vida são feitas usando o pensamento no Sistema 1. Por exemplo, a decisão de como interpretar a linguagem verbal ou a informação visual é automática e inconsciente, como ao se perceber que uma pessoa está com raiva através de seu tom de voz e fisionomia, respectivamente. De outro modo, o Sistema 2 refere-se ao raciocínio mais lento, consciente, racional, explícito e lógico. Com o Sistema 2 em ação, as pessoas desenvolvem uma linha de raciocínio para a solução de problemas mais complexos, como efetuar a multiplicação de dois números grandes. De acordo com (Kahneman, 2013), a principal característica do Sistema 2 é exigir atenção e foco no objetivo da tarefa em execução.

Quanto mais ocupadas e mais apressadas as pessoas estiverem, mais coisas elas têm em sua mente, e maior a probabilidade de que utilizem o pensamento no Sistema 1, justamente pela impossibilidade de manter a atenção. (Buchanan & O'Connell, 2006), ao mencionarem a influência do instinto na tomada de decisão, afirmam que "os pragmáticos agem com base em evidências, já os heróis agem com a coragem." De fato, as decisões instintivas são normalmente tomadas em emergências, quando não há tempo para ponderar argumentos e calcular probabilidades de ocorrência de cada evento. A intuição, argumentam os autores, também é mais aplicável em situações em que não há precedentes e, consequentemente, pouca evidência e disponibilidade de dados.

Por outro lado, a lógica do Sistema 2 deverá influenciar preferencialmente as decisões mais importantes, como a escolha de aplicações financeiras e a execução de grandes projetos empresariais. Segundo (Bazerman & Moore, 2014), o termo racionalidade refere-se ao processo de tomada de decisão que leva ao resultado ideal, dada uma avaliação precisa dos valores e preferências de risco do tomador de decisões.

No entanto, de acordo com (Securato, 2007), é difícil estabelecer um processo claro e robusto para a tomada de decisão, dado a infinidade de situações com as quais o decisor se depara, especialmente em se tratando de fatos inesperados. Ainda assim, identifica-se elementos capazes de influenciar uma decisão e que são, por sua vez, passíveis de ajuste. Segundo o autor, os elementos que influenciam o indivíduo são: experiência, julgamento e ambiente.

A quantidade de exposições de uma pessoa a diferentes tipos de problemas, aos processos de decisão empenhados, ao sentimento de responsabilidade, assim como a verificação dos resultados obtidos são fatores que enriquecem sua experiência. Para o decisor, a capacidade de julgamento é importante para discernir sobre a factibilidade das alternativas à sua mão. Esta característica está estreitamente ligada a experiência e, dessa forma, elas podem evoluir conjuntamente. Por fim, afirma-se que o decisor precisa conhecer bem o ambiente no qual o processo decisório é realizado, pois as características do ambiente podem influenciar a tarefa, como aspectos culturais, ideológicos ou físicos.

Avançando-se em direção ao processo decisório racional, metódico, dirigido pela lógica e atendido pelo conhecimento disponível, (Keeney & Raiffa, 1993) e (Securato, 2007) propõem, de forma similar, cinco etapas que podem ser executadas para se obter uma boa decisão.

Entretanto, advertem que estas etapas não devem ser vistas como uma obrigatoriedade burocrática, havendo flexibilidade para inversão da ordem e duração de cada uma. A seguir são apresentadas tais etapas de forma resumida:

Objetivo: deve ser definido de forma clara e precisa o objetivo da decisão, tal como maximizar o resultado de um investimento ou minimizar os custos operacionais de uma fábrica. Algumas vezes há múltiplos objetivos, que devem ser equalizados e, na medida do possível, atacados individualmente.

Variável-objetivo: a cada objetivo é necessário associar uma variável quantitativa, que receba valores representativos do objetivo e passível de medição direta ou indireta, no caso de informações qualitativas. A variável-objetivo poderá ser utilizada em funções e análises estatísticas.

Alternativas: são as maneiras de se alcançar o objetivo proposto. As alternativas devem ser bem estabelecidas, excludentes e associadas a um valor da variável-objetivo.

Nesse ponto, em condições de certeza, o problema estaria resolvido pela simples observação da alternativa que implica na otimização do valor da variável-objetivo, de acordo com (Securato, 2007). No entanto, as tomadas de decisões que envolvem riscos ou incertezas requerem a elaboração de situações que, hipoteticamente, podem ocorrer no contexto do problema. Só então todos os componentes poderão ser combinados na decisão final.

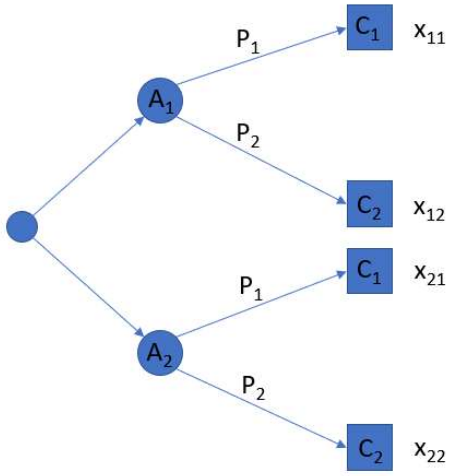
Cenários: são visões estruturadas de situações futuras, obtidas por informações objetivas ou subjetivas, que podem afetar o valor da variável-objetivo. As chances de ocorrência de cada cenário podem ser providas por técnicas preditivas, por especialistas ou outra metodologia aplicável.

Matriz de Decisão: por fim, relacionam-se todos os elementos do processo decisório em uma matriz, proporcionando-se uma clara visualização dos valores da variável-objetivo para cada combinação de cenário e alternativa, como mostra a Figura 1a.

Figura 1 – Organização dos elementos de um processo decisório

Cenários	Chances	Alternativas	
		A_1	A_2
C_1	P_1	x_{11}	x_{12}
C_2	P_2	x_{21}	x_{22}

(a)



(b)

O item (a), conforme (Securato, 2007, p. 30), representa uma matriz de decisão reunindo os cenários C_1 e C_2, suas respectivas probabilidades de ocorrência P_1 e P_2, as alternativas elencadas A_1 e A_2 e, por fim, os valores da variável-objetivo são dados pelos elementos da matriz: x_{11}, x_{12}, x_{21} e x_{22}. (b) com base em (Keeney & Raiffa, 1993, p. 7), a árvore de decisão reorganiza as informações do problema de forma hierárquica. É possível calcular o valor esperado da alternativa A_i tal que $VE(A_i) = P_1 \cdot x_{i1} + P_2 \cdot x_{i2}$. Com isso, a alternativa com melhor valor esperado deve ser a indicação ótima.

De acordo com (Keeney & Raiffa, 1993), nesta fase final do processo decisório, o decisor pode optar pela alternativa que lhe remete ao melhor resultado esperado. Em um caso simples, para se calcular o valor esperado de uma alternativa, basta somar os valores da variável-objetivo de cada cenário ponderando-os pela respectiva chance de ocorrência, como exemplifica a árvore de decisão ilustrada pela Figura 1b.

A proposição de métodos para apoiar a tomada de decisões são comuns na teoria prescritiva, conforme (Bazerman & Moore, 2014). Por outro lado, trabalhos de cunho descritivo procuram analisar a forma como as decisões realmente são tomadas. Na verdade eles buscam descrever como o processo se dá em cada contexto. Além disso, como explica (Buchanan & O'Connell, 2006), é comum estes trabalhos examinarem os fatores que afastam o decisor da racionalidade, mesmo quando há conhecimento e informações suficientes.

Um experimento realizado por (Junior & Famá, 2002) com estudantes universitários de Administração trouxe evidências de que a "percepção pessoal do risco envolvido em

alternativas de investimento distintas explica melhor a decisão tomada pelas pessoas do que o modelo racional, onde o risco é medido pelo desvio padrão". De acordo com os resultados obtidos pelos autores, há duas características nos tomadores de decisão que se mostram distintas do comportamento esperado pela forma racional, num ambiente de incertezas. Em primeiro lugar, notaram que as pessoas percebem e valorizam as alternativas diferentemente umas das outras, ou seja, com critérios distintos. A segunda característica refere-se à percepção pessoal do risco, que é "significativamente diferente da forma racional com que ele é usualmente mensurado". Em particular, verificou-se por meio de modelos econométricos que, de fato, a percepção pessoal de risco contribui melhor para explicar a tomada de decisão.

Isso posto, reforça-se a importância do discernimento do tomador de decisões, como afirmam (Bazerman & Moore, 2014): o tomador de decisões deve se empenhar para calibrar bem seus parâmetros pessoais, que são utilizados para valorizar as alternativas à sua mão. Em última análise, defendem, que o decisor deve buscar uma boa combinação entre suas crenças privadas e a realidade. No entanto, como mostram uma série de trabalhos na área das Finanças Comportamentais, essa receita é extremamente difícil de se seguir, pois cada indivíduo possui seu próprio ponto de vista, cria suposições simplificadoras e sofre com mecanismos e vieses escondidos no seu inconsciente.

Com todos esses elementos conspiratórios, o investidor pode incorrer em erros comuns ao tomar suas decisões de investimentos, como a negociações excessivas, as decisões impulsivas e a grande tolerância a perdas *vis-à-vis* a realização prematura de ganhos devido a ansiedade. O excesso de confiança, o otimismo sobre retornos positivos acima da média do mercado e o referencial de preço pago inicialmente por uma ação, respectivamente, são os principais indutores destas falhas.

No mercado acionário, todos os retornos esperados dependem, em níveis diversos, de eventos futuros, que naturalmente, são desconhecidos e incertos. Para Malkiel (2015), é justamente este contexto que torna o mundo dos investimentos fascinante e atrai milhões de pessoas ao redor do planeta, pois o sucesso de uma aposta depende da capacidade do investidor prever o futuro. Para Malkiel, um investidor de sucesso é geralmente um indivíduo equilibrado, naturalmente curioso e que explora ao máximo sua vantagem intelectual.

Por sua vez, (Leme, 1983) refletiu sobre o tema da tomada de decisão, considerando que o instante presente representa o ponto em que o indivíduo lança mão de todo o seu conhecimento, obtido no passado, na tentativa de prever o que há de ocorrer no futuro imediatamente a frente deste instante. Segundo Leme, é com base nas melhores previsões do futuro que se tomam decisões e se executam as ações subsequentes visando o resultado previsto. Evidentemente, em um contexto arriscado, há chances de tais previsões não se confirmarem, cenário para o qual o decisor também deve estar preparado. A necessidade de se compreender bem o passado a fim de realizar boas previsões do futuro inspira constantemente a criação de novas tecnologias de apoio à tomada de decisão.

A Tecnologia da Informação foi desenvolvida desde o seu início com o propósito de contribuir para as tomadas de decisões, permitindo a resolução de cálculos e simulação de cenários com eficiência e rapidez. Ainda nas décadas de 1950 e 1960, a comunidade científica dos principais centros de pesquisa dos Estados Unidos, como *Carnagie Institute of Technology*, *Massachusetts Institute of Technology* e *Stanford University* estabeleceram os pilares da computação e os primeiros modelos computacionais da cognição humana, surgindo daí o embrião da Inteligência Artificial. Desde então, a IA evoluiu através dos trabalhos de diversos cientistas, não apenas para ajudá-los a compreender o processo decisório da mente humana, mas também para auxiliar a tomada de decisão de quaisquer indivíduos face aos seus desafios, segundo (Buchanan & O'Connell, 2006).

Atualmente, as técnicas de Aprendizado de Máquina (AM) são protagonistas no âmbito dos processos decisórios de diversos segmentos econômicos, especialmente no mercado acionário. Conforme Goodfellow et al. (2016), tais técnicas são um conjunto de aplicações da IA que realmente viabilizam a resolução de problemas complicados do mundo real. Entre outras possibilidades, o AM pode fornecer ao tomador de decisão previsões acuradas sobre eventos futuros, rápida e consistentemente. Uma vantagem adicional refere-se a sua imunidade aos vieses comportamentais. Mais à frente, após introduzir os principais conceitos do AM e seu estado-da-arte (*deep learning*), a seção 2.3.5 dedica-se especialmente a contextualização e detalhamento de algumas estratégias quantitativas úteis para o processo decisório no mercado acionário, foco deste trabalho.

Como visto acima, o processo de tomada de decisões é uma atividade que demanda atenção, análise de informações, boa capacidade preditiva e julgamento adequado sobre a melhor

alternativa. O decisor, conforme (Securato, 2007, p. 23), capacitado dos recursos necessários, realiza sua tarefa, ora de forma metódica e racional, ora intuitiva e emocionalmente. Observa-se ambas abordagens no mercado acionário e desde meados do século 20 foram propostas e testadas diversas teorias e ferramentas para apoiar a tomada de decisão sobre a escolha de ativos financeiros. A seguir, serão confrontados os principais aspectos entre o comportamento racional esperado e a atitude irracional observada nos investidores.

2.1.2. Escolha de Ativos em Condições de Risco

2.1.4.1. Atitude Racional em Relação à Incerteza

Ao longo do século XX desenvolveu-se o arcabouço teórico sobre a racionalidade do investidor face ao problema da seleção de ativos financeiros. Nos fundamentos da teoria econômica moderna, assume-se que as pessoas escolhem racionalmente dentre as alternativas disponíveis (von Neumann & Morgenstern, 1944), pois são conhecidos tanto a oferta de recursos quanto as restrições orçamentárias. Assim, seria alcançado o equilíbrio em uma economia competitiva (Arrow & Debreu, 1954).

A Moderna Teoria Financeira (MTF) evoluiu substancialmente com o conceito de que os agentes financeiros conhecem as distribuições de probabilidade dos retornos dos ativos e, portanto, os riscos associados a eles. Sendo risco, neste caso, medido pela variância estatística de uma amostra de preços. A seleção de um conjunto de ativos para compor um portfólio é uma das principais tarefas em engenharia financeira e, sobretudo, é fundamental na estratégia de diversificação dos agentes financeiros, de acordo com Freitas et al. (2001).

O conceito de diversificação e a proposição de estratégias de alocação de recursos foram criados por expoentes do assunto ao longo de uma série de trabalhos. Primeiro, Harry Markowitz propôs o método de seleção de carteiras baseado na relação risco *versus* retorno dos ativos, cuja combinação ótima prevê um portfólio com o mínimo de risco, dado um retorno esperado (Markowitz, 1952). Anos depois surgiu o CAPM proposto por William Sharpe, com o objetivo de definir uma teoria consistente para o apreçamento dos ativos (Sharpe, 1963, 1964). Em seguida, intrigado com a dinâmica dos mercados de capitais, Eugene Fama formulou sua Hipótese da Eficiência dos Mercados (HEM), sugerindo que os preços

refletem todas as informações existentes e que, portanto, não seria possível obter vantagens, ou seja, não seria possível obter, consistentemente, retornos acima da média do mercado (Fama, 1970). A ideia de que o investidor é racional, avesso ao risco e que utiliza uma curva de utilidade para maximizar seu bem estar suporta os três conceitos da MTF apresentados acima.

Esse arcabouço teórico orientou muitos agentes do mercado de capitais no processo de seleção de ativos ao longo das últimas décadas, com implementações cada vez mais sofisticadas, na medida em que os recursos computacionais evoluíram e as informações se tornaram mais acessíveis e transmitidas rapidamente desde as fontes até os tomadores de decisão.

Contudo, na prática, o que se observa são investidores buscando vencer o mercado dia após dia, como se a HEM fosse uma falácia. De fato, há diversos estudos e críticas buscando mostrar que o mercado, na realidade, não é tão eficiente. Algumas anomalias são usadas como evidências para refutar a HEM, ao mesmo tempo que novas abordagens buscam explicar como tais anomalias surgem no mercado (Malkiel, 2003). O paradigma que conquistou mais relevância no meio acadêmico estuda o impacto do comportamento do investidor, influenciado por fatores psicológicos, sobre suas próprias decisões e, em última instância, como o agregado de todas as decisões determinam os preços dos ativos. Este paradigma é chamado de Finanças Comportamentais e será o tema da próxima seção.

2.1.4.2. Atitude Irracional em Relação à Incerteza

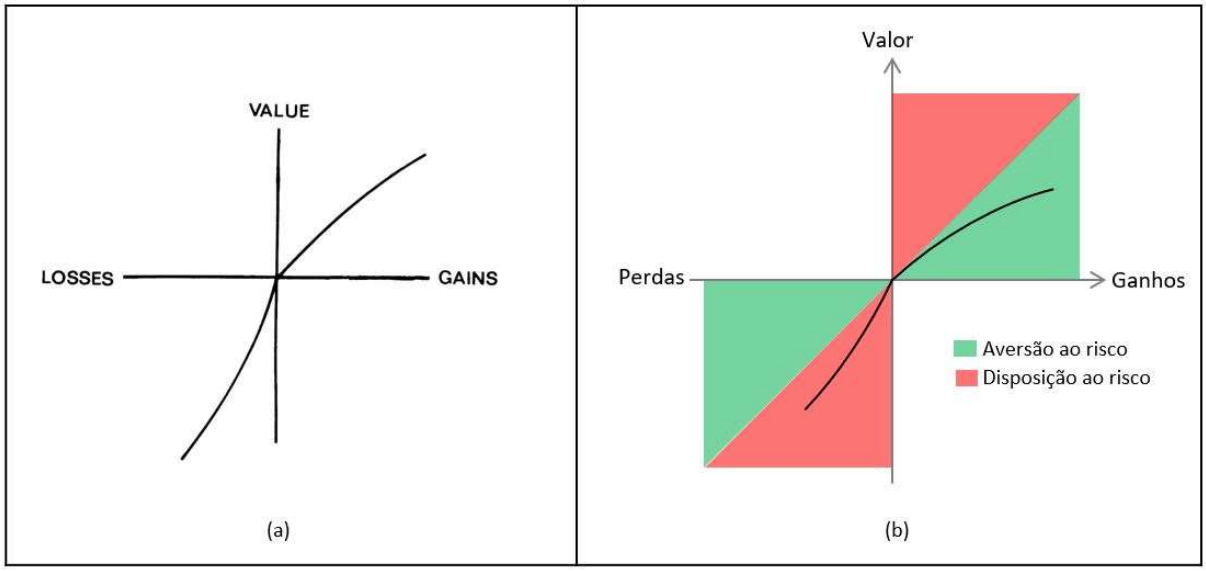
As pesquisas de Daniel Kahneman e Amos Tversky, duas das figuras mais relevantes no estudo da psicologia e escolha do investidor, buscam compreender como se dá o processo de decisão em condições de incerteza. Seus estudos levaram ao desenvolvimento da Teoria dos Prospectos, que procura explicar decisões que são inconsistentes com a avaliação racional das probabilidades e a função de utilidade padrão, sendo, portanto, um contra ponto à clássica Teoria da Utilidade, onde os agentes econômicos tomariam decisões absolutamente racionais de acordo com os resultados esperados (Kahneman & Tversky, 1979).

A Teoria dos Prospectos diz que a curva de utilidade de um investidor possui comportamento assimétrico quando este avalia perdas e ganhos. No primeiro caso, o investidor estaria mais

inclinado a aceitar maiores perdas. Por outro lado, o investidor subestimaria as chances de ganhos elevados e, dessa forma, grandes ganhos teriam menos importância. Essa ideia de função utilidade assimétrica está ilustrada na Figura 2. Em seu artigo, (Shefrin & Statman, 1985) investigaram as razões pelas quais os investidores toleram perdas por mais tempo e realizam ganhos rapidamente. Foram apresentadas evidências, dentro de um *framework* conceitual, de que vários fatores emocionais afetam o processo decisório. O estudo mostra que os principais fatores emocionais envolvidos, além do comportamento assimétrico em relação ao risco, são a contabilidade mental, o arrependimento pelos erros passados e a incapacidade de auto controle.

(Shefrin & Statman, 1985) sugerem ainda que a percepção de impacto negativo no imposto de renda, dadas as decisões possíveis, interage significativamente com os fatores emocionais analisados experimentalmente. Esta percepção influencia negativamente o comportamento do investidor, afastando-o do padrão racional.

Figura 2 – Função de utilidade assimétrica proposta pela Teoria dos Prospectos



O item (a) representa a função utilidade apresentada no artigo original de (Kahneman & Tversky, 1979, p. 279) sobre a Teoria dos Prospectos. À esquerda da linha vertical (neutralidade), o efeito de uma perda seria sobrevalorizado. À direita da linha vertical, o efeito de um ganho seria subvalorizado. No item (b), as regiões de aversão ao risco e disposição ao risco compreendem as partes côncava no domínio dos ganhos e convexa no domínio das perdas, respectivamente, da curva de função utilidade assimétrica, proposta por Kahneman e Tversky.

Após os experimentos pioneiros e Kahneman e Tversky, cuja teoria resultante lhes renderam um Prêmio Nobel, diversos pesquisadores se aprofundaram em pesquisas na área das finanças

comportamentais. Elton et al. (2012) mapearam as principais características emocionais, ou tendenciosidades, sobre o comportamento do investidor analisadas ao longo das últimas décadas. As principais tendenciosidades são apresentadas a seguir de forma resumida:

- **Heurística**: os processos heurísticos, ou "atalhos cognitivos", são tendências do comportamento do investidor que podem limitar a racionalidade da tomada de decisão, pois o indivíduo tende a definir sua estratégia de investimento . As principais heurísticas são: representatividade, ancorar e ajustar, disponibilidade e confiança exagerada.
- **Dissonância cognitiva**: é o conflito interno que ocorre no processo decisório devido à existência de informações ou crenças contraditórias no pensamento do indivíduo. Assim, dado o estado de incoerência, o indivíduo busca reforçar uma cognição em detrimento de outras, a fim de reduzir a situação de dissonância.
- **Contabilidade mental**: é a tendência que os indivíduos possuem de criar contas separadas para cada operação financeira, ao contrário de considerar a totalidade de suas carteiras de investimentos. Este comportamento, segundo a teoria da otimização de portfólios, incorre em decisões subótimas.
- **Humor e emoção**: o afeto, termo psicológico, pode influenciar a tomada de decisão dos investidores, segundo evidências obtidas em experimentos de laboratório, cujas interpretações indicam que as respostas emocionais à incerteza impedem a tomada racional de decisões.
- **Tendenciosidade local**: em contraste às vantagens da diversificação, este comportamento leva o investidor a dar preferência aos ativos locais, de sua região geográfica. Essa tendência se deve a premissa de que o fácil acesso às informações de companhias próximas proporcionaria uma vantagem competitiva ao investidor. No entanto, isso não se verifica na prática, como evidenciado em estudos em diferentes regiões do mundo.
- **Caminho de menor resistência**: neste caso, o investidor possui um comportamento passivo e aceita aplicar seu capital em alternativas pré-determinadas, por exemplo, em fundos de pensão com aplicações automáticas, gerenciadas pelo seu empregador.

- **Diversificação heurística**: este comportamento leva o investidor a aplicar montantes proporcionalmente iguais à quantidade de opções que lhe são dadas, o que nem sempre é o ideal.

Por fim, Elton et al. (2012), com base em todo o arcabouço teórico e empírico produzido nas últimas décadas, refletem sobre a previsibilidade dos erros das pessoas, em se tratando da tomada de decisões em condições de risco:

> "As . . . pesquisas sobre as escolhas do investidor sugerem que muitos investidores tomam decisões que não são ótimas. As pessoas tendem a cometer erros de maneira previsível que refletem o uso da heurística ou de atalhos mentais." (p. 491)

Com tantas evidências cientificas disponíveis sobre a irracionalidade do investidor em ambientes de incerteza *vis-à-vis* a possibilidade de se realizar escolhas racionais, viu-se na aplicação de métodos computacionais inteligentes um meio eficaz para a tomada de decisões coerentes com a MTF. De fato, as máquinas não são influenciadas por fatores comportamentais, não sentem medo ou vergonha, assim como não admitem variações de humor. Programados para sugerir decisões, ou tomá-las diretamente, em linha com as normas e premissas racionais, com base no processamento de um volume de informações intratável para qualquer ser humano, os métodos quantitativos são vistos atualmente como a solução operacional mais viável por um número cada vez maior de agentes do mercado financeiro e pesquisadores, tais como Cavalcante et al. (2016), Deng et al. (2017) e (Fischer & Krauss, 2018). Serão vistos nas próximas seções, tanto o contexto do mercado acionário brasileiro, quanto fatores e ferramentas de AM que podem auxiliar o investidor em seu processo decisório.

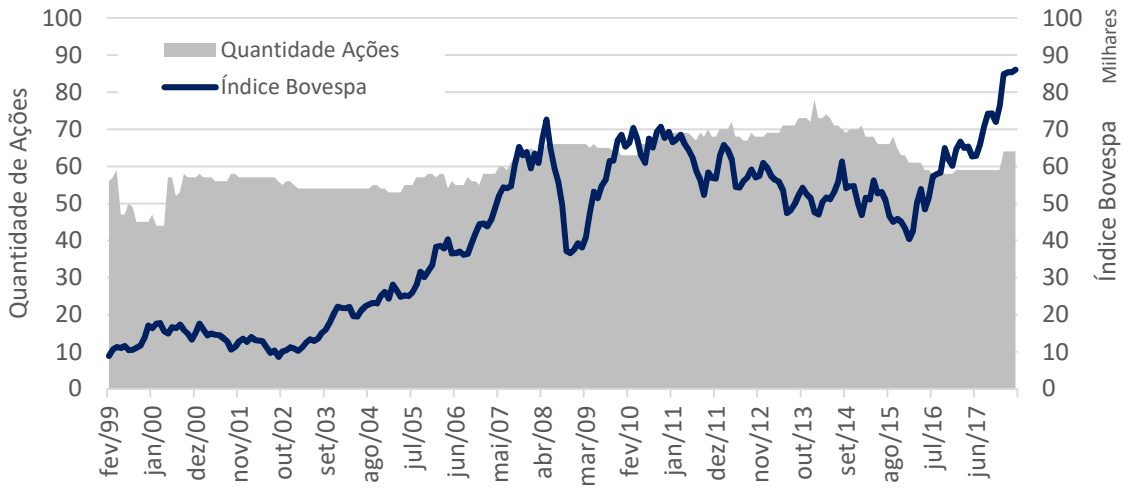
2.1.3. Contexto Decisório no Mercado Acionário Brasileiro

O mercado acionário brasileiro é desafiador. O ambiente de incertezas políticas e dificuldades socioeconômicas contribui para a geração de volatilidade, típica de países emergentes. No entanto, este fato pode gerar oportunidades, visto que as técnicas de previsão mais avançadas podem dar vantagem ao investidor, especialmente em momentos mais turbulentos, como observou (Krauss et al., 2017). A seguir, o leitor entrará em contato com o contexto desta pesquisa.

2.1.4.3. Índice Bovespa

Em 2018 o Índice Bovespa completou 50 anos. É o principal índice do mercado acionário brasileiro, consolidado como referência para os investidores nacionais e internacionais. De acordo com a metodologia adotada pela bolsa de valores B3, o Ibovespa é o resultado de uma carteira teórica de ativos, que atendem aos critérios de inclusão estabelecidos em (B3, 2015). Os principais critérios de inclusão são (i) o índice de negociabilidade, sendo necessário a presença do ativo em mais de 95% dos pregões e (ii) a participação financeira mínima de 0,1% do ativo no período de 12 meses que antecede sua inclusão, ou manutenção, no índice.

Gráfico 2 – Índice Bovespa



Curva do Índice Bovespa de 1999 até 2018 e a quantidade de ações constituintes do índice ao longo dos anos. Fonte: B3. Gráfico elaborado pelo próprio autor.

O Gráfico 2 mostra a evolução do Ibovespa ao longo dos últimos 19 anos. Nesse período, o índice foi composto por 60,9 ações, em média. O auge desse número ocorreu em janeiro de 2014, quando o Ibovespa chegou a possuir 78 ações. O principal índice do mercado brasileiro apresentou uma tendência lateral no início dos anos 2000. A partir do ano 2002, o Ibovespa entrou em um ciclo de alta expressiva, que se encerra em meados de 2008, com a crise do *subprime* originada nos Estados Unidos. A partir desse instante, a bolsa passa por um rápido e agudo declínio, acompanhando os mercados internacionais, e chega em aproximadamente 36 mil pontos em novembro de 2008. Logo em seguida, ocorre rápida recuperação, aproximando o índice novamente da máxima histórica, em meados de 2010. Nota-se até então a grande volatilidade que o índice apresentou. A partir daí a bolsa entrou em um novo

ciclo, com uma tendência de baixa moderada dos papéis, influenciada principalmente pelo cenário de incertezas econômicas e políticas no Brasil, que perdurou por cerca de cinco anos, culminando no processo de *impeachment* da presidente Dilma Rousseff, encerrado em 31 de agosto de 2016. Inicia-se, então, um novo ciclo de alta da bolsa, devido ao ânimo do mercado com a possibilidade de uma retomada da economia brasileira.

Neste ambiente volátil e desafiador, espera-se que sistemas inteligentes de auxílio a tomada de decisão possam, de fato, alavancar os resultados dos investidores, cujos objetivos são proteger o capital e superar o mercado, na medida do possível. Em particular, esta pesquisa busca contribuir para a compreensão do uso de métodos de AM, especialmente de arquiteturas *deep learning* de RNAs. Uma estrutura *deep learning* integra muitos elementos de uma RNA em um sistema capaz de decodificar padrões e calcular probabilidades com precisão. Sua efetividade tem sido observada em diversos trabalhos, abrangendo uma grande quantidade de contextos. As próximas seções estreitarão o conhecimento desta técnica promissora.

2.2. Aprendizado de Máquina

Segundo (Ben-David & Shalev-Shwartz, 2014), Aprendizado de Máquina refere-se à detecção automática de padrões significativos existentes em um conjunto de informações. Aprender remete ao processo de conversão da experiência passada em conhecimento. Para tanto, o algoritmo de aprendizado recebe um conjunto de informações, chamado dados de treinamento, realiza seu processo computacional e exprime o conhecimento obtido em forma de um novo código que pode executar uma determinada tarefa, como prever o preço de uma ação em certo horizonte de tempo.

Em um problema típico, existe um valor quantitativo (*e.g.* preço de ações) ou um evento categórico (*e.g.* retorno positivo ou negativo) que se pretende prever, com base em um conjunto de variáveis (*e.g.* informações financeiras, série histórica de retornos). Define-se o conjunto de dados de treinamento relacionando-se o valor, ou evento, observado com suas respectivas variáveis, para cada objeto de análise (e.g. empresas listadas na bolsa de valores). Então, um modelo preditivo é construído sobre esse conjunto de informações, o qual

permitirá prever o valor de um novo objeto, a partir de suas variáveis (Hastie, Tibshirani, & Friedman, 2008).

Existem duas categorias de métodos de aprendizado de máquina, a saber: aprendizado não-supervisionado e aprendizado supervisionado. Diferentemente do exemplo dado acima, certas ocasiões requerem o agrupamento de elementos similares entre si, visando a criação de grupos homogêneos em termos de suas variáveis, sem a presença de um valor observado para orientar o aprendizado. Por exemplo, o agrupamento de ações em termos da correlação de seus preços ao longo do tempo, onde busca-se reunir ações positivamente correlacionadas e separar ações negativamente correlacionadas. Por outro lado, os métodos de aprendizado supervisionado requerem um valor observado para ser apreendido a partir das informações disponíveis a seu respeito. A seguir serão apresentados os principais aspectos sobre estas duas classes de algoritmos, com maior ênfase aos processos supervisionados, considerando-se o escopo desta pesquisa.

2.2.1. Aprendizado Não-Supervisionado

No aprendizado não-supervisionado, tem-se apenas as variáveis descritivas (*features*) do problema em questão e não há um valor observado (*target*) para orientar o processo de aprendizagem. Na ausência do *target*, não é possível calcular uma medida direta de sucesso. Dessa forma, é difícil validar as inferências feitas pela maior parte dos modelos não-supervisionados. Seguindo (Hastie et al., 2008), apesar desta classe de métodos ser menos desenvolvida na literatura, apresenta-se uma rápida conceitualização de problemas de associação e agrupamento.

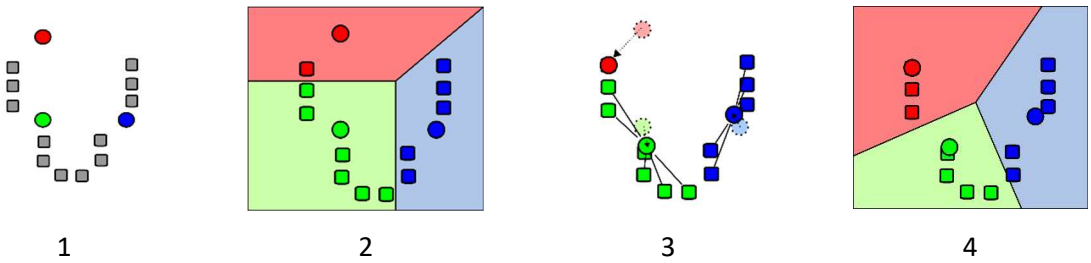
Um caso interessante de problema de associação pode ser visto em um comércio varejista, onde cada consumidor compra uma cesta de produtos, dentre uma lista de itens possíveis. O objetivo é encontrar produtos que aparecem frequentemente juntos. Os produtos podem ser representados por *features* $(X_1, X_2, \dots, X_p)$, onde $X_j \in \{0,1\}$. Então, para cada consumidor i, a *feature* X_j assume um dos valores possíveis; $x_{ij} = 1$ se o produto j foi comprado, caso contrário $x_{ij} = 0$. Assim, aquelas *features* que frequentemente aparecem com valor igual a 1 na mesma compra representam produtos que são vendidos conjuntamente. Este tipo de conhecimento obtido dos dados é útil para campanhas de *marketing*, promoções, *design* de

prateleiras e segmentação de clientes, como visto em (Hastie et al., 2008). Os autores notam que este tipo de problema, quando seus dados atingem dimensões relativamente grandes, são difíceis de resolver, dada sua natureza combinatória.

A análise de agrupamentos, por sua vez, possui uma grande diversidade de objetivos. Todos eles relacionam-se a tarefa de agrupar ou segmentar uma coleção de objetos em subconjuntos (*clusters*), tal que os elementos de um certo *cluster* sejam mais parecidos entre si do que àqueles pertencentes a outro *cluster*. Algumas vezes, busca-se arranjar os *clusters* de uma forma hierárquica natural, reagrupando-os de modo que cada nível contenha subconjuntos similares. Outros problemas demandam uma estatística descritiva para averiguar se existem subconjuntos substancialmente distintos em um conjunto de dados. Em ambos os casos, a questão central refere-se ao conceito de similaridade, ou dissimilaridade, entre os elementos agrupados. Dentre as medidas mais usadas, estão a distância euclidiana e a correlação entre os valores das *features* em análise (Hastie et al., 2008).

O método *k-means*, a técnica mais popular de agrupamento, busca particionar n elementos de um conjunto de dados em k subconjuntos. O processo é inicializado a partir de k pontos médios quaisquer, sendo $k < n$, e então os dois seguintes passos são executados até a convergência: (1) cada elemento (ponto) do conjunto de dados é associado ao ponto médio mais próximo, em termos da distância euclidiana; (2) cada ponto médio é atualizado considerando-se os elementos agrupados no passo anterior. Ao final do processo, cada elemento pertence ao cluster cujo ponto médio esteja mais próximo de si, como ilustra a Figura 3.

Figura 3 – Ilustração do conceito de agrupamento de um conjunto de pontos



O conceito de agrupamento de um conjunto de pontos é simulado em um plano cartesiano, utilizando-se o método K-means. Nesse caso, três clusters são formados: verde, azul e vermelho. 1) k pontos iniciais são gerados aleatoriamente dentro do domínio dos dados. 2) k clusters são criados associando-se cada ponto à média mais próxima. 3) O ponto médio de cada cluster se torna a nova referência. 4) Os passos (1) e (2) são repetidos até a convergência. Fonte: extraído de (Wikipedia, 2019).

Segundo (Jain, Murty, & Flynn, 1999), a análise de agrupamentos (*clustering*) é mais utilizada em etapas preliminares do processo de desenvolvimento de modelos de AM. A exploração de dados para a descoberta de padrões pode ser feita em diversos contextos, embora quanto maior o volume de dados, mais difícil será a resolução do problema. Por sua vez, (Silva et al., 2013) avaliaram uma série de trabalhos relativos a exploração e técnicas robustas e eficientes de agrupamento de massivas e ilimitadas sequencias de informações, geradas continuamente, chamadas *data stream*s.

Grandes *streams* são obtidos em sistemas meteorológicos, através do monitoramento de redes de computadores e em operações do mercado financeiro, entre outras. (Kontaki, Papadopoulos, & Manolopoulos, 2008), como citado em (Silva et al., 2013), aplicaram *clustering* em grandes séries de preços de ações. Nesse caso, a metodologia permitiu obter-se *insights* sobre a evolução das ações ao longo do tempo e, portanto, poderia ajudar na decisão sobre quando é o melhor momento para comprar ou vender uma ação. Para prever, de fato, o que pode ocorrer no futuro, com alguma segurança estatística, recorre-se aos métodos supervisionados, tema da próxima seção.

2.2.2. Aprendizado Supervisionado

Muitos problemas reais possuem um conjunto de situações mensuráveis (*inputs*) que influenciam a ocorrência de certos resultados ou eventos (*outputs*). Assim, de modo geral, o processo de aprendizado supervisionado tem como objetivo utilizar os *inputs* para prever os *outputs*. Para tanto, é preciso encontrar uma função que melhor represente esta relação. Como obter a representação exata é pouco provável, busca-se, na maioria das técnicas, minimizar o erro entre os *outputs* previstos e observados.

A seguir, aborda-se o conceito de aprendizagem supervisionada através de um exemplo de classificação, adaptado de (Alpaydin, 2014, p. 21), construído com os retornos históricos da ação ITUB4, listada no Ibovespa. Como ilustrado na Figura 4a, o preço da ação está consolidado em uma visão semanal. A cada semana o resultado pode ser positivo ou negativo, dependendo se o preço fechou acima ou abaixo do valor do papel na semana anterior, respectivamente. Assim, define-se retorno como $\boldsymbol{r_t = p_t / p_{t-1} - 1}$, onde $\boldsymbol{p_t}$ é o preço da ação na semana $\boldsymbol{t}$, para $\boldsymbol{t = 1, 2, \dots, n}$. Neste exemplo, deseja-se estabelecer uma classificação

preditiva para o resultado da semana atual, y_t, com base nos retornos das últimas duas semanas, r_{t-2} e r_{t-1}.

Posto o problema, representa-se cada *input* histórico como um vetor x com dois valores:

(2.1) $$x_t = \begin{bmatrix} r_{t-2} \\ r_{t-1} \end{bmatrix},$$

e seu *output* como:

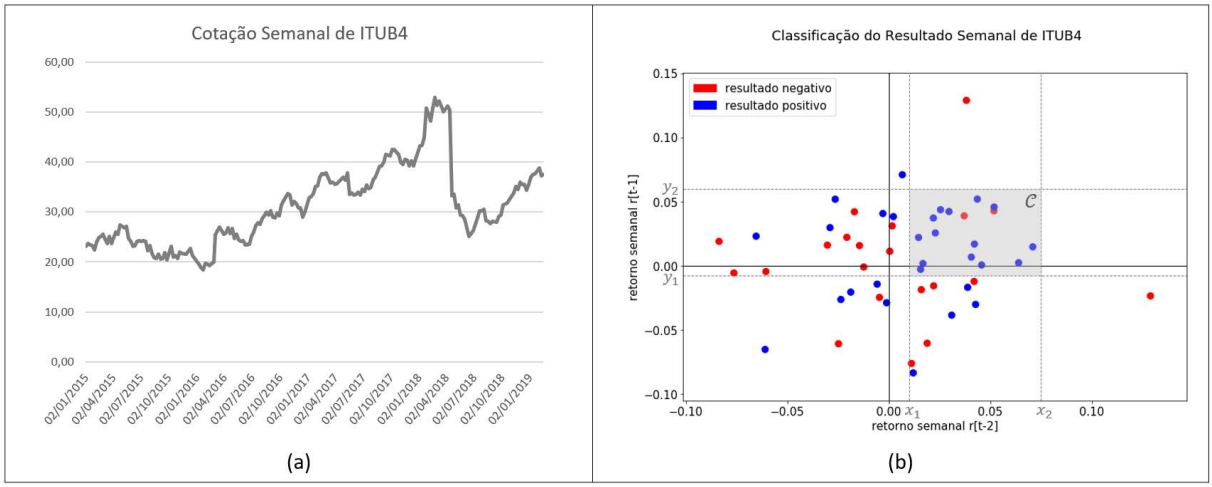
(2.2) $$y_t = \begin{cases} 1, & se\ r_t \geq 0, \text{resultado positivo;} \\ 0, & se\ r_t < 0, \text{resultado negativo.} \end{cases}$$

Assim, a cada semana, tem-se um par (x, y). O conjunto de treinamento $\mathcal{X}$ é composto por n exemplos, tal que:

(2.3) $$\mathcal{X} = \{x_t, y_t\}_{t=1}^{n}.$$

Para visualização dos dados deste conjunto de treinamento, pode-se dispor no plano cartesiano, como no Gráfico 3b, cada instância t referente a um ponto com coordenadas (r_{t-2}, r_{t-1}). Nesse gráfico ilustrativo, cada ponto exibe uma cor, azul ou vermelho, referente a classe que representa, resultado positivo ou negativo, respectivamente.

Figura 4 – Ilustração da análise de classificação dos retornos de ITUB4



O item (a) apresenta o gráfico da cotação semanal de ITUB4 abrangendo o período entre janeiro de 2015 a fevereiro de 2019. O item (b) mostra o gráfico da dispersão dos retornos semanais defasados de ITUB4. O retorno da semana t é dado por $r_t = p_t/p_{t-1} - 1$, onde p_t é o preço da ação na semana t. O eixo x refere-se aos retornos ocorridos há duas semanas (r_{t-2}). O eixo y refere-se aos retornos ocorridos há uma semana (r_{t-1}). A cor do ponto do gráfico diferencia os resultados positivos (azul) e negativos (vermelho) ocorridos na semana atual, de acordo com o seu respectivo retorno (r_t). O retângulo $\mathcal{C}$, em cinza, destaca-se com uma taxa de 87,5% de retornos positivos, enquanto o espaço complementar possui apenas 44,1% de pontos azuis. Fonte das cotações: Economatica. Gráficos elaborados pelo autor.

Após analisar a Figura 4b, é possível supor que, para haver maiores chances de retornos positivos na semana t, os retornos das duas últimas semanas devem pertencer a um certo intervalo:

(2.4) $(x_1 \leq r_{t-2} \leq x_2)\ e\ (y_1 \leq r_{t-2} \leq y_2)$,

definido por valores adequados de x_1, x_2, y_1 e y_2. Assim, a equação (2.4) corresponde ao retângulo $\mathcal{C}$ no espaço $r_{t-2} \times r_{t-1} \in \mathbb{R}^2$.

Conforme (Alpaydin, 2014), na teoria de Aprendizado de Máquina, a equação (2.4) determina uma **classe de hipóteses $\mathcal{H}$** (o conjunto de todos retângulos do plano cartesiano), que dá origem a $\mathcal{C}$ (um retângulo específico). O processo de aprendizagem, portanto, deve encontrar uma hipótese específica $h \in \mathcal{H}$, definida pelos valores $(x_1,\ x_2,\ y_1,\ y_2)$, que se aproxima de $\mathcal{C}$ ao máximo. Ou seja, o objetivo do aprendizado é determinar os valores dos quatro parâmetros que determinam o retângulo h.

Considera-se, então, uma hipótese h que realiza predições para um *input* x, tal que:

(2.5) $$h(x_t) = \begin{cases} 1, & \text{se } h \text{ classifica } x_t \text{ como um resultado positivo;} \\ 0, & \text{se } h \text{ classifica } x_t \text{ como um resultado negativo.} \end{cases}$$

A real distribuição de $\mathcal{C}(x_t)$ é desconhecida, pois tem-se apenas uma amostra observada $\mathcal{X}$ de resultados, que é por sua vez um subconjunto de todos os resultados possíveis de x_t em um horizonte de tempo t ilimitado. Contudo, é possível calcular o **erro empírico** do problema, dado pela proporção de instâncias do conjunto de treinamento em que a predição de $h(x_t)$ difere do resultado observado y_t. Assim, o erro da hipótese h no conjunto de treinamento é dado por:

(2.6) $E(h \mid \mathcal{X}) = \sum_{t=1}^{n} \mathbb{I}(h(x_t) \neq y_t)$,

onde $\mathbb{I}(\cdot)$ é a função identidade, com valor igual a 1 se $h(x_t) \neq y_t$ e valor igual a 0 caso contrário. Então, o processo de aprendizagem supervisionada consiste em encontrar um conjunto de parâmetros $(x_1,\ x_2,\ y_1,\ y_2)$ que especificam h e que minimizam o erro empírico dado pela equação (2.6). No exemplo do Gráfico 3b, uma boa aproximação para o subespaço $\mathcal{C}$ é dada pelos pares $(0.01,\ 0.075)$ no eixo r_{t-2} e $(-0.008,\ 0.06)$ no eixo r_{t-1}. Neste caso, o erro empírico é igual a 0.125 ou 12,5%.

Analogamente, para problemas de regressão são definidas classes de hipóteses $\mathcal{G}$ para a construção de preditores, assim como a função que especifica o erro empírico, a ser minimizado no processo de otimização que determina a solução ótima dos parâmetros da hipótese específica $g \in \mathcal{G}$. Conforme (Hastie et al., 2008), as classes de hipóteses possuem um conjunto de parâmetros θ que podem ser moldados para acomodar da melhor forma possível os dados do conjunto de treinamento. Por exemplo, em modelos lineares da forma $f(x) = x^T\beta$, tem-se $\theta = \beta$. Outra classe de hipóteses comum são as expansões lineares do tipo:

(2.7) $$f_\theta(x) = \sum_{k=1}^{K} h_k(x)\theta_k,$$

onde h_k são funções de x e θ_k são os parâmetros a serem encontrados. Dentre as mais tradicionais, tem-se as funções polinomiais e trigonométricas, como x^2, $x_1x_2^2$ e $\cos(x)$, entre outras. Segundo (Hastie et al., 2008), a função sigmoide é muito utilizada em modelos de redes neurais, sendo esta pertencente à uma classe de hipóteses não-lineares e geralmente escrita como:

(2.8) $$h_k(x) = \frac{1}{1+\exp(-x^T\beta_k)}.$$

Em qualquer caso de regressão, como os citados acima, o erro empírico pode ser escrito como uma soma do quadrado das diferenças entre os valores preditos e observados, tal que:

(2.9) $$E(\theta) = \sum_{i=1}^{n}\left(y_i - f_\theta(x_i)\right)^2.$$

A minimização dos erros residuais da equação (2.9) leva aos parâmetros ótimos θ^* de f_θ, como segue:

(2.10) $$\theta^* = \underset{\theta}{\operatorname{argmin}}\, E(\theta).$$

De acordo com (Hastie et al., 2008), para modelos lineares e algumas outras formas funcionais mais simples, existem soluções analíticas fechadas para resolver a equação (2.10). Caso contrário, necessita-se aplicar métodos iterativos ou otimização numérica, como a técnica do gradiente descendente.

Um princípio mais geral, com abordagem estatística, e muito utilizado em problemas de AM é a estimativa por máxima verossimilhança (*maximum likelihood estimation*). Supõe-se que os *outputs* y_i, para $i = 1, 2, \dots, N$ sejam variáveis aleatórias, com distribuição de densidade

$\mathrm{P}_\theta(y)$, em função de algum parâmetro θ. Assim, o logaritmo da probabilidade da amostra de dados é dado por:

(2.11) $L(\theta) = \sum_{i=1}^{N} \log \mathrm{P}_\theta(y_i)$.

O princípio da máxima verossimilhança assume que o valor ótimo do parâmetro θ é aquele que maximiza o valor de (2.11). Assim como nos exemplos anteriores, muitas formas funcionais de distribuições de probabilidade podem ser utilizadas para definir a equação *log-likelihood* de um problema. A título de aplicabilidade nesta pesquisa, destaca-se a função multinomial de verossimilhança para uma distribuição $P(Y \mid X)$ de um certo *output* Y que assume categorias distintas (*e.g.* $Y \in \{A, B, C, \dots\}$). Esta função dá origem a chamada *binary cross-entropy likelihood*, muito utilizada em problemas de classificação com *outputs* binários, onde $Y \in \{0, 1\}$.

Nesse caso, supõe-se que existe uma distribuição de probabilidade condicional da forma $\mathrm{P}_\theta(Y = \mathcal{Y}_k \mid X = x) = p_{k,\theta}(x)$ para a probabilidade de ocorrência de cada classe de Y dado a variável aleatória X, com parâmetro θ. Então, a função *log-likelihood* (também chamada *cross-entropy*) é escrita da seguinte forma:

(2.12) $L(\theta) = \sum_{i=1}^{N} \log p_{g_i,\theta}(x_i)$,

e quando maximizada fornece o parâmetro ótimo θ^* que melhor ajusta os dados do problema.

No caso em que a distribuição de probabilidade condicional se refere a um *output* binomial, $Y \in \{0, 1\}$, a equação (2.12) pode ser reescrita tal que:

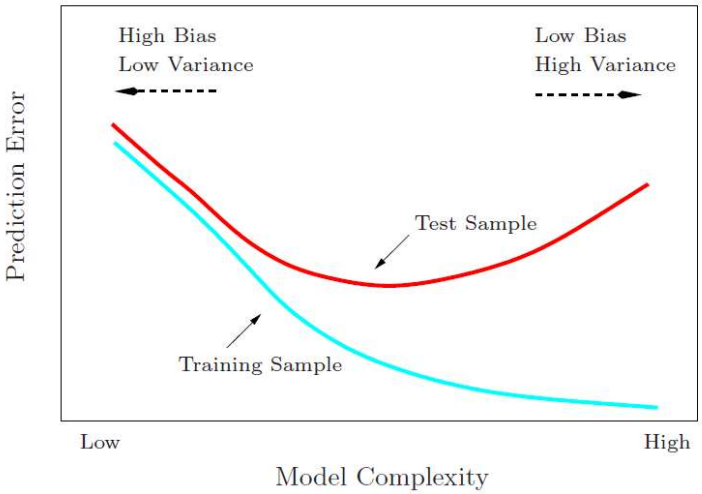
(2.13) $L(\theta) = -\frac{1}{N} \sum_{i=1}^{N} y_i \log\big(p(y_i)\big) + (1 - y_i) \log\big(1 - p(y_i)\big)$.

Na equação (2.13), verifica-se que o termo $\log\big(p(y_i)\big)$ é adicionado ao valor total da função $L(\theta)$ caso o valor observado de y_i seja igual a 1. Caso contrário, adiciona-se a $L(\theta)$ o termo $\log\big(1 - p(y_i)\big)$. As funções $L(\theta)$ são denominadas função de perda (*loss function*), por definirem o erro (perda) do processo de predição. (Hastie et al., 2008, p. 261) e (Alpaydin, 2014, p. 251) apresentam detalhadamente a formulação matemática das funções de perda do tipo *binary cross-entropy*.

Todos os modelos e algoritmos citados, entre outros abordados nas demais referências, possuem parâmetros que determinam sua **complexidade**, em termos de ponderadores das

variáveis, tempo de processamento e quantidade de dados. Um modelo com alta complexidade, quando não tratado corretamente, pode implicar em resultados indesejados, inclusive a falta de precisão de predições futuras.

Gráfico 3 – Curva do erro de predição *versus* a complexidade de um modelo



Curva típica do erro de predição em função da complexidade de um modelo de aprendizado supervisionado. Fonte: (Hastie et al., 2008, p. 38)

Em (Hastie et al., 2008, p. 37), se introduz o conceito da capacidade de **generalização** do modelo de aprendizado supervisionado, quando aplicado em uma nova amostra de dados, diferente da base de treinamento. Um modelo com boa capacidade de generalização, apresenta níveis satisfatórios de erros em uma base de testes. De modo geral, o erro na amostra de teste ocorre devido aos vieses da base de treinamento e à própria variância amostral. Este erro tende a aumentar conforme o modelo se torna mais complexo. Por outro lado, tanto o erro de treinamento quanto o erro de teste são grandes quando o modelo permanece muito simples relativamente à estrutura de dados em análise, caso em que não se aprende a relação entre *input* e *output*.

O Gráfico 3 refere-se a uma ilustração clássica deste fenômeno. A curva referente ao erro de treinamento (azul claro) decresce na medida em que a complexidade aumenta e o modelo se ajusta perfeitamente aos dados. Já a curva vermelha, que apresenta um formato côncavo, representa o erro de teste. Por um lado, na medida em que a complexidade aumenta, o modelo perde sua capacidade de generalização. Por outro lado, enquanto o modelo

permanece simples, também generaliza mal. Assim, há um ponto de equilíbrio que minimiza o erro de teste, onde ocorre a melhor generalização.

Existem ainda uma série de métricas muito utilizadas em aplicações de AM, que buscam inferir a capacidade do modelo de prever corretamente o evento em questão. Dentre as métricas mais aplicadas, destacam-se a medida de acurácia do modelo, a precisão e o *recall* do modelo. Em particular, existe um trade-off entre as duas últimas medidas, pois conforme uma cresce a outro necessariamente se reduz, conforme (Géron, 2017, p. 84). Para definir tais métricas, primeiramente se faz necessário conhecer o conceito de matriz de confusão.

A matriz de confusão é uma tabela que mostra a frequência de classificações para cada evento do problema. No caso de eventos binários, o valor do *output* pode ser 0 ou 1. Da mesma forma, o método de classificação pode associar ao mesmo registro um valor (previsto) igual a 0 ou 1. O cruzamento dos resultados previstos com os eventos reais gera uma matriz como ilustra a Tabela 1.

Tabela 1 – Matriz de confusão

		Predito	
		0	**1**
Observado	**0**	VN	FP
	1	FN	VP

Ilustração de uma matriz de confusão, onde as células da matriz se referem às frequências (quantidade) de valores obtidos ao comparar o valor real com o predito por um certo modelo. VN = verdadeiro negativo, FP = falso positivo, FN = falso negativo e VP = verdadeiro positivo. Fonte: elaborado pelo autor, com base em (Géron, 2017, p. 88)

Assim, as definições de cada métrica são tais que:

- Acurácia = $\frac{\text{Quantidade de predições corretas}}{\text{Total de predições}} = \frac{\text{VN + VP}}{\text{VN+FP+FN+V}}$
- Precisão = $\text{Quantidade de previsões positivas corretas} = \frac{\text{VP}}{\text{FP+V}}$
- *Recall* = $\text{Quantidade de positivos identificados corretamente} = \frac{\text{VP}}{\text{VP+F}}$

Foram apresentados acima os conceitos fundamentais referentes ao aprendizado supervisionado. A partir deste arcabouço desenvolvem-se diversas metodologias de classificação, agrupamento e regressão, como apresentado nas obras de (Hastie et al., 2008),

(Alpaydin, 2014) e (Ben-David & Shalev-Shwartz, 2014). Em particular, destacam-se as Redes Neurais Artificiais e toda a família de estruturas de RNAs desenvolvida nas últimas décadas, que possibilitam resolver tarefas consideradas impossíveis para uma máquina. A seção a seguir detalha as RNAs, desde sua estrutura fundamental até grandes arquiteturas concebidas para apreender padrões não-lineares complexos dos dados, chamadas de *deep learning*.

2.3. *Deep Learning*

Atualmente, as máquinas podem aprender praticamente qualquer coisa a partir da experiência passada, registrada nos mais diversos conjuntos de informações, e auxiliar o processo decisório, inclusive no mercado acionário. Segundo (Goodfellow et al., 2016), ao abordar problemas muito complexos, e até mesmo intuitivos, é necessário criar uma representação hierárquica do mundo real, onde estruturas abstratas e simples são definidas e interligadas entre si através de relações básicas. O paradigma mais utilizado para modelar tais estruturas são as RNAs. Então, com essa representação estruturada, possibilita-se à máquina aprender automaticamente fenômenos complexos a partir de uma série de conceitos simples. O termo *deep learning* origina-se, portanto, deste empilhamento hierarquizado de muitos níveis de conceitos simples que leva ao aprendizado.

De acordo com (Goodfellow et al., 2016), o *deep learning* se popularizou a partir de 2007, quando os trabalhos de (Hinton, Osindero, & Teh, 2006), (Bengio, Lamblin, Popovici, & Larochelle, 2007) e (Ranzato, Poultney, Chopra, & LeCun, 2007) apresentaram técnicas eficientes para o treinamento de arquiteturas demasiadamente grandes de RNAs. A partir de então, muitas aplicações de *deep learning* surgiram para desafiar a performance dos melhores sistemas de IA baseados em modelos de AM tradicionais. Além disso, o sucessivo aumento da capacidade computacional também contribuiu para alavancar as aplicações de grandes RNAs em problemas envolvendo bases de dados massivas, como dados de voz, imagens, informações de sistemas meteorológicos e registros de transações financeiras.

Nesta seção serão apresentados os conceitos fundamentais das RNAs, assim como uma das principais arquiteturas de Redes Neurais Recorrentes (RNRs), chamada *Long Short-Term Memory*, que possui, em tese, a capacidade de modelar as relações existentes dentro de uma sequências de dados (e.g. série histórica de preços de uma ação) e entre sequências distintas

(*e.g.* série de preços históricos de empresas distintas). O modelo LSTM foi aplicado no mercado americano por (Fischer & Krauss, 2018), observando-se bons resultados financeiros em *backtests*. Com isso, completa-se a fundamentação teórica para essa pesquisa, que aplicará a técnica LSTM no mercado acionário brasileiro.

2.3.1. Redes Neurais

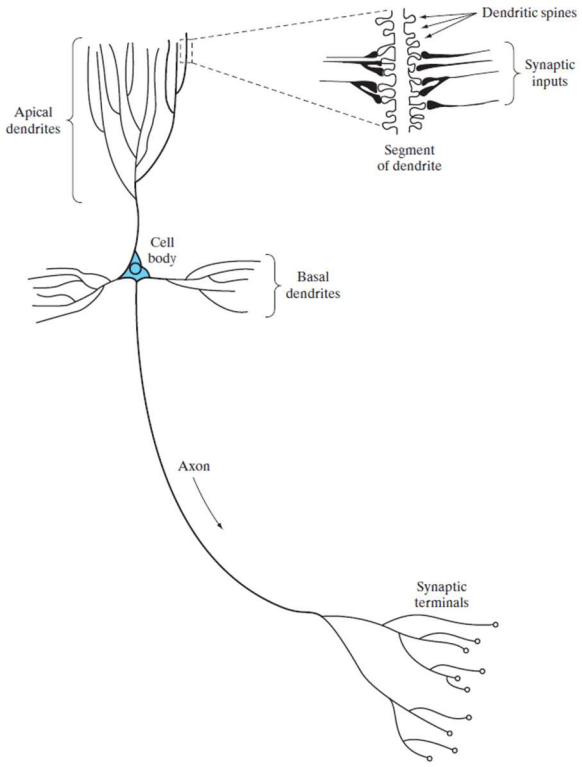
As redes neurais artificiais surgiram da ideia de se imitar as estruturas cerebrais com o objetivo de compreender o funcionamento do intelecto e, posteriormente, para resolver problemas tais quais humanos fazem. As RNAs podem ser implementadas fisicamente através de sistemas eletrônicos ou pela programação de estruturas virtuais em computadores digitais. De acordo com (Haykin, 2001), uma rede neural é uma máquina adaptativa projetada para modelar a maneira como o próprio cérebro humano realiza uma determinada tarefa. De forma mais específica, Haykin afirma que:

> "Uma rede neural é um processador maciçamente paralelamente distribuído constituído de unidades de processamento simples, que têm a propensão natural para armazenar conhecimento experimental e torná-lo disponível para o uso." (p. 28)

O conhecimento é expresso na rede neural através de um processo de aprendizagem supervisionado, onde os parâmetros da rede são ajustados aos dados de entrada. Segundo Haykin, há diversos benefícios associados ao uso de redes neurais, entre eles destacam-se a sua adaptabilidade a problemas reais; a possibilidade de se criar estruturas não-lineares que se enquadram perfeitamente bem a mecanismos físicos geradores de informações não lineares; e a tolerância a falhas, devido à natureza distribuída do conhecimento armazenado na rede.

De fato, a inspiração para o desenvolvimento das redes neurais artificiais provém do modelo neuronal encontrado no cérebro humano, como ilustra a Figura 5. As RNAs são formadas por elementos interligados entre si. Cada elemento possui uma característica específica e efetivamente contribui para o processamento de uma determinada tarefa. Em termos matemáticos, tais elementos representam variáveis, parâmetros ou funções e são integrados como um sistema de equações.

Figura 5 – Concepção de um neurônio cerebral

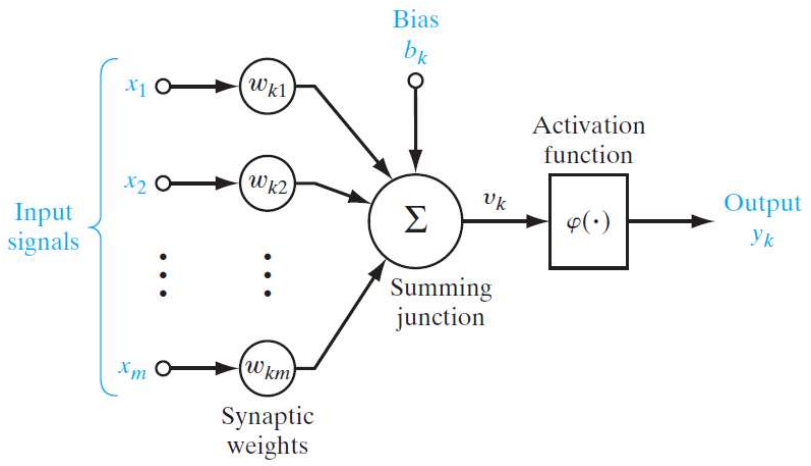


Os verdadeiros neurônios cerebrais motivam o design de neurônios artificiais e, consequentemente, o desenvolvimento de grandes redes de neurônios artificiais interligados. Fonte: (Haykin, 2008, p. 8)

O neurônio é a unidade de processamento básica de uma rede neural. O diagrama de blocos da Figura 6 mostra o modelo de um neurônio. Identifica-se neste modelo três itens elementares que permeiam praticamente todos os projetos de RNAs.

Primeiro, tem-se um conjunto de **pesos sinápticos** caracterizados por valores w_{jk} que representam a ligação das variáveis de entrada x_j, para $j = 1, 2, \dots, m$, com o núcleo do neurônio k. O **integrador** implementa, de fato, a combinação das variáveis de entrada ponderadas pelos respectivos pesos sinápticos. Por fim, a **função de ativação** determina o domínio do valor de saída y_k do neurônio. O modelo neuronal básico também pode apresentar um parâmetro de viés b_k, encarregado por aumentar ou reduzir a entrada total de informação na função de ativação.

Figura 6 – Diagrama de blocos de um neurônio artificial



Fonte: (Haykin, 2008, p. 11)

De acordo com o (Haykin, 2008), o neurônio da Figura 6 pode ser descrito matematicamente através do seguinte par de equações:

(2.14) $v_k = \sum_{j=1}^{m} w_{kj}\, x_j$

e

(2.15) $y_k = \varphi(u_k + b_k)$

onde $x_1, x_2, \dots, x_m$ são as variáveis de entrada (*input signals*); $w_{k1}, w_{k2}, \dots, w_{km}$ são os respectivos pesos sinápticos; v_k é a soma ponderada dos sinais de entrada e denomina-se campo local induzido; b_k é o parâmetro relativo ao viés adicionado no modelo; $\varphi(\cdot)$ é a função de ativação e, finalmente, y_k é a saída do neurônio, um valor calculado pela função de ativação. Algumas funções são especialmente úteis nos projetos de redes neurais, devido as suas propriedades aritméticas e equivalência a problemas reais. Seguem exemplos abaixo:

(2.16) $\varphi(v) = \begin{cases} 1, & \text{se } v \geq 0 \\ 0, & \text{se } v < 0 \end{cases}$ Função de limiar (*threshold*)

(2.17) $\varphi(v) = \begin{cases} 0, & \text{se } v \leq -\frac{1}{2} \\ v, & \text{se } -\frac{1}{2} < v \leq +\frac{1}{2} \\ 1, & \text{se } v > \frac{1}{2} \end{cases}$ Função linear por partes

(2.18) $\varphi(v) = \frac{1}{1+\exp(-av)}$ Função sigmoide

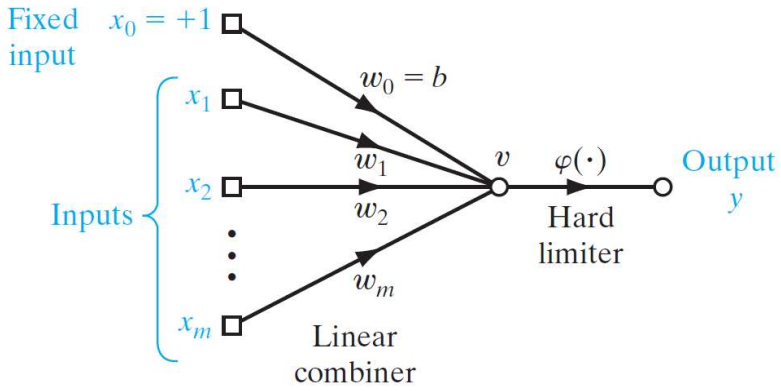
(2.19) $\varphi(v) = \tanh(v)$ Função tangente hiperbólica

Não sendo o objetivo desta pesquisa detalhar toda a matemática que fundamenta o desenvolvimento das RNAs, mas entretanto, permitir ao leitor conectar os principais elementos das RNAs à matéria do trabalho, que é predizer o retorno diário (positivo ou negativo) das ações negociadas no Ibovespa, deixa-se a critério do leitor o aprofundamento no tema através das obras consolidadas de (Haykin, 2008) e (Goodfellow et al., 2016). De forma conceitual, serão abordadas a seguir as estruturas de RNAs que tem por finalidade embasar o modelo avaliado nesta pesquisa. Discute-se, com esse propósito, as redes mais simples de uma camada, em seguida são apresentadas as redes de múltiplas camadas e, finalmente, as redes recorrentes, que dão origem, de fato, as redes LSTM.

2.3.2. Perceptrons de Camada Única

O perceptron de Rosenblatt, também conhecido na literatura como modelo neuronal McCulloch–Pitts, em tributo aos pesquisadores e idealizadores do assunto, é um caso especial de rede neural de uma camada, uma função de ativação e consequentemente de um neurônio, como ilustra a Figura 7. Este modelo, embora simples, tem a especial capacidade de resolver problemas de classificação de padrões linearmente separáveis. De acordo com (Haykin, 2001, p. 143), "Rosenblatt provou que se os vetores usados para treinar o perceptron são retirados de duas classes linearmente separáveis, então o algoritmo do perceptron converge" e sua demonstração é conhecida como o teorema de convergência do perceptron. A partir daí estendeu-se toda uma teoria sobre as RNAs.

Figura 7 – Grafo do fluxo de sinal em um perceptron



Fonte: (Haykin, 2008, p. 50)

Neste caso, o processo de aprendizagem supervisionada busca minimizar os erros e_i entre o resultado observado d_i e o *output* y_i previsto pelo modelo neuronal, tal que $e_i = d_i - y_i$, onde $i = 1, 2, \dots, n$ refere-se a quantidade de registros na base de treinamento. A função de perda (*loss function*) do problema é definida em função dos parâmetros $w_j, j = 1, 2, \dots, m$, tal que $\mathcal{E} = \mathcal{E}(w)$. Logo, busca-se encontrar w^* que minimiza o erro total, ou seja, a função de perda com menor valor possível $\mathcal{E}(w^*) \leq \mathcal{E}(w)$.

Dentre as técnicas de otimização irrestrita deste problema de minimização, destaca-se o método da **descida mais íngreme**, que utiliza o gradiente da função $\mathcal{E}(w)$; o método de **Newton**, que utiliza a matriz Hessiana de $\mathcal{E}(w)$; e o método de **Gauss-Newton**, que faz uso da matriz Jacobiana do erro total $\mathcal{E}(w)$ (Haykin, 2001).

Embora o perceptron de camada única tenha grande valia histórica por ter aberto o caminho para o desenvolvimento da teoria das RNAs, foi demonstrado por (Minsky & Papert, 1969) que este modelo é incapaz de generalizar algumas aplicações, com base no aprendizado local (amostra de treinamento). Na época esta obra lançou dúvidas, inclusive, sobre a performance das RNAs em geral. No entanto, ao longo das décadas seguintes, a capacidade computacional das redes neurais avançaram e há atualmente uma coleção de arquiteturas robustas e eficientes (Haykin, 2001).

2.3.3. Perceptrons de Múltiplas Camadas e o Método *Back-Propagation*

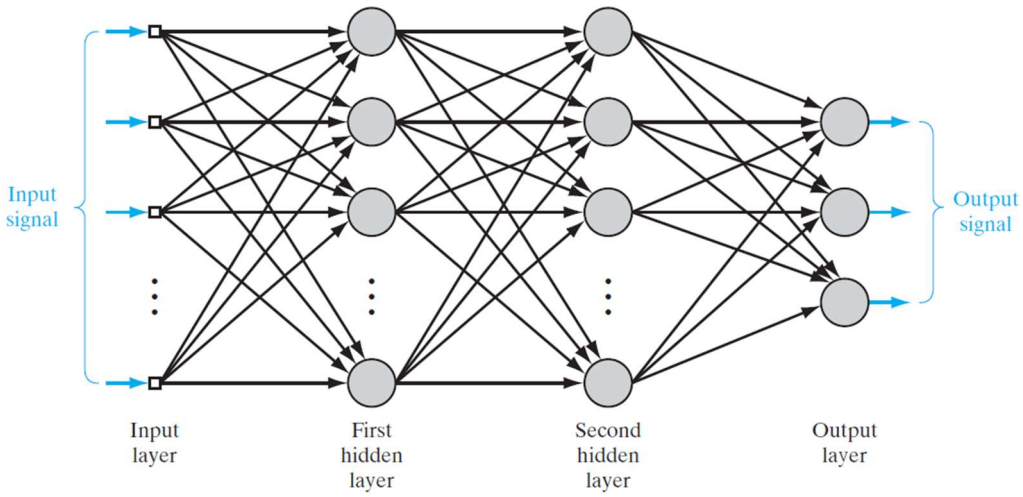
Discute-se nesta seção as redes neurais de múltiplas camadas, constituídas por uma camada de entrada de informações, por uma ou mais camadas de nós computacionais e uma última camada de saída, também formada por nós computacionais. De acordo com (Haykin, 2008) este conceito de RNAs tem sido aplicadas com sucesso na resolução de problemas complexos. (Goodfellow et al., 2016) ressalta a importância do desenvolvimento teórico sobre os projetos de múltiplas camadas para o alcance de soluções viáveis no âmbito das redes de *deep learning*.

Segundo (Haykin, 2001), apesar do pessimismo com relação a performance dos perceptrons, introduzido por Minsky e Papert em 1969, o modelo de múltiplas camadas ganhou projeção prática ao se utilizar o algoritmo de retropropagação de erro (*error back-propagation*) para a calibragem dos pesos sinápticos da rede. O algoritmo *back-propagation* "representa um

marco nas redes neurais, pois fornece um método computacional eficiente" para o processo de aprendizagem supervisionada do modelo (Haykin, 2001, p. 184).

Segundo (Haykin, 2008), os perceptrons de múltiplas camadas são caracterizadas por três aspectos: (i) cada nó computacional (neurônio) possui uma função de ativação suave e não-linear, como a função sigmoide (eq. 2.18). Este requisito é importante para satisfazer os critérios de diferenciação do algoritmo *back-propagation*; (ii) a rede é capacitada para aprender tarefas complexas pela extração progressiva de padrões significativos dos dados de entrada através de suas camadas ocultas (unidades neuronais entre as entradas e saídas); e (iii) exibe alta densidade de conexões sinápticas envolvendo os nós disponíveis no modelo. A Figura 8 ilustra um projeto típico de rede neural de múltiplas camadas.

Figura 8 – Perceptron de múltiplas camadas



Grafo arquitetural de um perceptron de múltiplas camadas com duas camadas ocultas. Fonte: (Haykin, 2008, p. 124)

Assim como as redes de uma única camada, nas arquiteturas maiores define-se a função de perda, que mapeia os pesos sinápticos à saída da rede e, consequentemente, mede-se o erro de predição *vis-à-vis* o real *output* do sistema. Nas obras de (Goodfellow et al., 2016; Haykin, 2008; Russell & Norvig, 2013) estão consolidados todos os passos do processo de aprendizagem das redes de múltiplas camadas. Mas convém, neste texto, explanar o funcionamento do método *back-propagation*, seguindo a ideia original de (Rumelhart, Hinton, & Williams, 1986) e posteriormente detalhado por (Haykin, 2001) e (Nielsen, 2015).

Em termos simples, o método ajusta recursivamente os pesos sinápticos de toda a rede propagando o erro $e_j(n)$, medido em cada neurônio j na camada de saída, para as camadas

internas do modelo. Por isso se diz que o erro é retropropagado, pois flui no sentido inverso do fluxo de informações (*i.e.* da saída para a entrada da rede). Assim, a cada iteração do algoritmo, um certo peso sináptico é atualizado visando-se minimizar o erro medido à sua direita (*i.e.* para frente, no sentido do fluxo de informação da rede neural). Frequentemente, utiliza-se o método da descida mais íngreme para calcular o valor do ajuste. Neste caso, é importante a utilização de funções de ativação diferenciáveis, como a função sigmoide, pois o processo de retropropagação do erro exige o cálculo de derivadas parciais, com relação a cada peso sináptico do neurônio em questão, para obtenção dos gradientes e respectivos fatores de ajuste de cada peso. Conforme (Haykin, 2001), o algoritmo pode ser resumido da seguinte forma:

1. Inicialização: atribuição inicial aleatória de pesos sinápticos de acordo com uma distribuição apropriada;
2. Exemplos de treinamento: apresentação dos dados de treino à rede, com os quais serão executados os passos 3 e 4;
3. Propagação: este passo refere-se aos cálculos para frente, no sentido do fluxo de informações, das funções de ativação de cada neurônio e do sinal de erro dos nós de saída. Os principais cálculos executados nessa etapa, para cada elemento n do conjunto de treinamento, para o neurônio j da camada l são os seguintes:
    a. campo local induzido: $v_j(n) = \sum_{i=0}^{m} w_{ij}^{(l)}(n)\, y_i^{(l-1)}(n)$, sendo $y_i^{(l-1)}(n)$ um dos sinais de entrada do neurônio j proveniente de um neurônio $i \in \{1, 2, \dots, m\}$, conjunto de nós da camada anterior $l-1$ e $w_{ij}^{(l)}(n)$ o seu respectivo peso sináptico;
    b. função de ativação: $y_j^{(l)} = \varphi_j\left(v_j(n)\right)$, para cada neurônio das camadas ocultas l. Se o neurônio estiver na camada de entrada, utiliza-se $y_j^{(0)} = x_j(n)$. Se for um neurônio de saída (*i.e.* $l = L$, em que L é dito a **profundidade da rede**), calcula-se $y_j^{(L)} = o_j(n)$ e o respectivo sinal de erro $e_j(n) = d_j(n) - o_j(n)$, que é a diferença entre o real valor de saída do sistema e o valor previsto pelo neurônio j.

4. Retropropagação: agora o sentido do fluxo de informações é invertido e os erros são propagados para trás (do final para o início da rede) a fim de se atualizar os pesos sinápticos. Para isso, calcula-se os gradientes locais $\delta_j^{(l)}(n)$ da rede tal que
   a. $\delta_j^{(l)}(n) = e_j^{(L)}(n)\varphi_j'\left(v_j^{(L)}(n)\right)$ para o neurônio j da camada L;
   b. $\delta_j^{(l)}(n) = \varphi_j'\left(v_j^{(l)}(n)\right)\sum_k \delta_k^{(l+1)}(n)w_{kj}^{(l+1)}(n)$ para o neurônio j da camada oculta l. Nesse caso, $\varphi_j'(\cdot)$ refere-se à diferenciação da função φ com relação ao argumento.

   Por fim, os valores dos pesos sinápticos da rede são ajustados pela chamada **regra delta generalizada**:

   (2.20) $$w_{ij}^{(l)}(n+1) = w_{ij}^{(l)}(n) + \alpha\left[w_{ij}^{(l)}(n-1)\right] + \eta\delta_j^{(l)}(n)y_i^{(l-1)}(n), \forall\, i, j$$

   onde α é a **constante de momento** e η é a **taxa de aprendizagem**.
5. Iteração: os passos 3 e 4 são repetidos sobre o conjunto de dados de treinamento até a satisfação de certos critérios de convergência (*e.g.* número de iterações N = 100 ou incremento na função objetivo menor que um valor ϵ pequeno).

A partir da concepção deste método, possibilitou-se o treinamento de grandes estruturas e as RNAs tornaram-se cada vez mais atrativas para o público que utiliza técnicas de AM para a resolução de problemas de tomada de decisão. Conforme (Wlodarczak, Soar, & Ally, 2015), citado em (Möws, 2016), apesar de receber críticas devido a seu apelo de *marketing* em detrimento aos métodos estatísticos já amplamente aplicados, a metodologia *Deep Learning*, definida como a estruturação de redes neurais com elevado número de neurônios distribuídos em muitas camadas, está estabelecida na comunidade acadêmica nas mais diversas frentes de pesquisa.

De acordo com (Möws, 2016), que estudou a aplicação de *Deep Learning* no mercado acionário, a principal vantagem de tais arquiteturas refere-se a larga exploração das não linearidade intrínsecas do problema pelo modelo, que permite a identificação automática de padrões complexos nos dados. Para (Glorot & Bengio, 2010), um dos motivos para se utilizar *Deep Learning* baseia-se em sua capacidade de extrair *features* automaticamente a partir dos dados de entrada originais, por meio de suas camadas intermediárias (ocultas). Isso reduz a necessidade de se criar variáveis manualmente ou por meio de processos adicionais de *feature*

engineering. As redes neurais recorrentes apresentadas a seguir são um arranjo especial, em termos do fluxo de informações, das redes vistas acima.

2.3.4. Redes Neurais Recorrentes

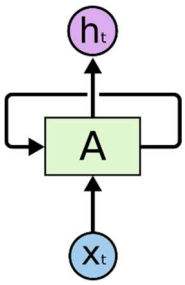
Uma Rede Neural Recorrente (RNR) é um tipo de rede neural especializada no processamento de dados sequenciais da forma $x^{(1)}, x^{(2)}, \dots, x^{(\tau)}$, conforme (Goodfellow et al., 2016, p. 363). Em geral, o conjunto $\mathrm{T} = \{(1), (2), \dots, (\tau)\}$ de índices da variável de entrada x representa uma sequência de escala temporal (*e.g.* dias, horas ou minutos referentes a sequência de preços de uma ação) ou ordinal (*e.g.* a posição sequencial das palavras que formam uma frase). As RNRs permitem que os padrões relevantes, encontrados em algum intervalo da sequência $\{x\}_{\mathrm{T}}$, persistam ao longo do processo de aprendizado. Há diversas possibilidades de arquiteturas de RNRs, assim como seu treinamento e uso prático. Em aplicações reais, são comuns as configurações de múltiplas camadas com elevado grau de profundidade.

Um dos principais aspectos relacionados a aprendizagem de uma RNR refere-se ao compartilhamento do vetor de parâmetros ao longo de diferentes níveis do modelo *vis-à-vis* a especificação de um vetor de parâmetros para cada nível da rede. Este diferencial se deve ao uso da recorrência no *design* da rede. A recorrência permite fazer com que o *output* $h^{(t)}$ do modelo seja função do *output* $h^{(t-1)}$ anterior combinado com uma determinada *feature* $x^{(t)}$, tal que:

(2.21) $$h^{(t)} = f\left(h^{(t-1)}, x^{(t)}; \theta\right),$$

onde $h^{(t)}$ também é chamada de **estado** do sistema.

Figura 9 – Neurônio recorrente



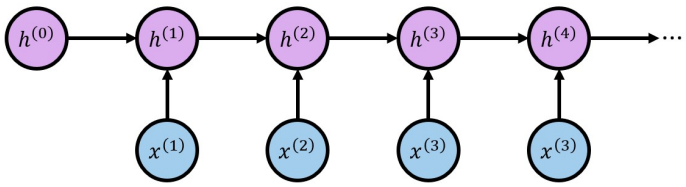
Grafo representativo de um perceptron recorrente. Fonte: (Olah, 2015).

Tipicamente, as arquiteturas utilizando RNRs adicionam uma camada extra à rede com o objetivo de se fazer predições a partir do estado $h^{(t)}$. A Figura 9 ilustra o conceito da equação (2.21), onde o bloco A representa a função $f(\cdot)$.

A função de $h^{(t)}$ pode ser desdobrada para um número finito de ***time steps*** (elementos do conjunto de índices T), aplicando-se a definição (2.21) por $\tau - 1$ vezes. Assim, é possível observar uma rede pura, sem o laço de recorrência, como ilustra a Figura 10. Este encadeamento de unidades de processamento revela a relação natural das RNRs com sequências de valores ou símbolos ordenados.

Como dito, as RNRs são capacitadas para extrair o conhecimento do passado, dentro de uma sequência de informações, persistir tal conhecimento ao longo dos ciclos e utilizá-lo na tarefa de predição do estado presente. No entanto, como demonstrado por (Bengio, Simard, & Frasconi, 1994), em aplicações práticas, a performance dos modelos cai na medida em que as sequências apresentadas à rede neural se tornam demasiadamente grandes.

Figura 10 – Representação de um neurônio recorrente desdobrado



Representação de um neurônio recorrente desdobrado, possuindo quatro *time steps*, sem a camada final preditiva de saída. Fonte: elaborado pelo autor com base em (Goodfellow et al., 2016, p. 366)

Em termos simples, conforme (Goodfellow et al., 2016, p. 390), durante o treinamento de RNRs submetidas a sequências muito grandes de informação, os parâmetros calculados no processo de atualização pelo método do gradiente (descida mais íngreme) podem ser levados à zero ou se tornarem infinitamente grandes, devido a composição de fatores causada pela recorrência. Este fato implica no descarte de qualquer padrão existente na sequência relacionado aos parâmetros mal condicionados. Nota-se ainda que o processo de treinamento utiliza a metodologia *back-propagation*, como discutido na seção 2.3.3, com adaptações inerentes ao caráter serial do problema.

Algumas abordagens foram propostas para diminuir a dificuldade do aprendizado das dependências em longas sequências de dados. Uma estratégia é criar **reservatórios computacionais,** através de unidades ocultas de processamento (neurônios), que guardam a representação de aspectos relevantes dos dados de entrada. Outra técnica consiste em definir **múltiplas escalas** sobre a série de dados e atuar diferentemente sobre escalas mais amplas e escalas mais granulares. Com isso, é possível incluir **saltos** do passado para o presente, **atrasos** (*delays*) entre as variáveis de entrada da sequência $\{x\}_{\mathrm{T}}$ e até mesmo a **remoção** de certos intervalos da sequência. Estes artifícios reduzem o número de *time steps* e consequentemente mitigam o risco de mal condicionamento dos parâmetros do modelo, devido a multiplicação de fatores, cujo resultado tende a zero ou infinito. Há também a possibilidade de criar neurônios especializados em armazenar o histórico de informações (*e.g.* $\mu^{(t)} = \alpha\mu^{(t-1)} + (\alpha - 1)v^{(t)}$, onde $\mu^{(t)}$ refere-se a média ao longo dos *time steps* t de um certo valor $v^{(t)}$) dando mais importância para os registros mais atuais ($\alpha \approx 0$) ou mais antigos ($\alpha \approx 1$) (Goodfellow et al., 2016, pp. 390–396).

Além das ideias acima, (Hochreiter & Schmidhuber, 1997) propôs equipar as RNRs com a capacidade de memorizar informações por grandes períodos ao longo da sequência de dados de entrada. O método LSTM apresentou ótimos resultados em problemas de reconhecimento de voz (Graves, Jaitly, & Mohamed, 2013), tradução automática (Sutskever, Vinyals, & Le, 2014), captura de imagens (Xu et al., 2015) e previsão de retorno de ações (Fischer & Krauss, 2018), entre outros. Para criar esta memória, utiliza-se uma combinação especial de neurônios que compõem cada bloco de processamento da estrutura recorrente das RNRs, como será visto a seguir.

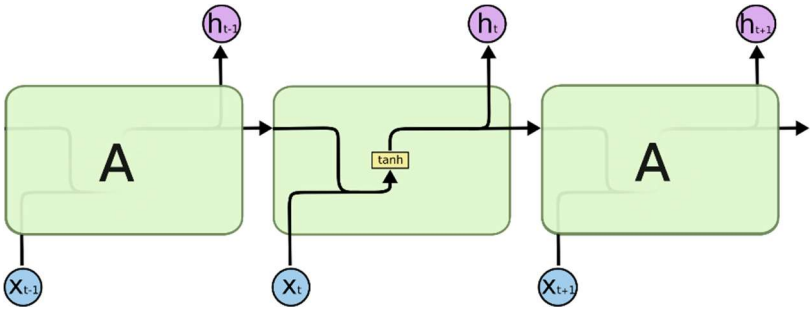
2.3.4.1. Redes Neurais Recorrentes do Tipo *Long Short-Term Memory*

As redes neurais LSTM são projetadas especificamente para aprender padrões ao longo de grandes sequências de dados. Como observado por (Sak, Senior, & Beaufays, 2014), estes modelos são capazes de superar o principal problema inerente às RNRs: a degeneração ou explosão dos gradientes durante o processo de aprendizagem.

Como visto anteriormente, as RNRs possuem uma estrutura de neurônios recorrentes relativamente simples. Cada nó computacional possui uma única função que processa os

sinais de entrada e envia para frente um valor de saída, como a função $tanh$ ilustrada na Figura 11.

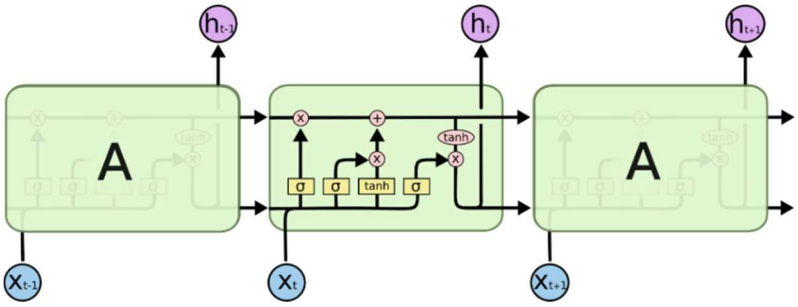
Figura 11 – Sequência de blocos de processamento de uma RNR comum



Sequência de blocos de processamento de uma RNR comum, constituída por uma única função $tanh$ em seu neurônio interno. Fonte: (Olah, 2015).

Por outro lado, uma LSTM típica possui uma combinação específica de neurônios, que interagem entre si recursivamente, responsáveis pelo controle das informações que fluem pelo **estado do sistema** (*cell state*). Estes neurônios (*gates*) abrem ou fecham o fluxo de sinal, reforçando ou mitigando, respectivamente, uma determinada característica presente ao longo da sequência de dados. Na formulação padrão são observados os seguintes elementos: ***input gate***, ***forget gate*** e ***output gate***, sendo cada um responsável pela execução de um controle específico. A seguir, será apresentada a representação matemática do mecanismo ilustrado na Figura 12.

Figura 12 – Estrutura conceitual de uma unidade LSTM



Fonte: extraído de (Olah, 2015).

Conforme (Goodfellow et al., 2016, p. 339), o principal componente da rede LSTM é a unidade *cell state* $s_i^{(t)}$, que possui uma recorrência linear própria. Neste caso, o peso associado à recorrência é controlado pela unidade *forget gate* $f_i^{(t)}$ do sistema, para cada *time step* t e bloco i da rede, cujo valor varia entre 0 e 1 conforme a saída da função sigmoide definida:

(2.22) $f_i^{(t)} = \sigma\left(b_i^f + \sum_j U_{i,j}^f x_j^{(t)} + \sum_j W_{i,j}^f h_j^{(t-1)}\right),$

onde $x^{(t)}$ é o *input* referente ao *time step* t e $h^{(t)}$ é o vetor contendo o *output* das células da rede. Além disso, os parâmetros b^f, U^f e W^f representam o viés, o peso associado ao *input* $x^{(t)}$ e o peso associado à recorrência $h^{(t-1)}$, respectivamente. Assim, o estado interno do sistema é atualizado, considerando-se o valor obtido em (2.22), tal que:

(2.23) $s_i^{(t)} = f_i^{(t)} s_i^{(t-1)} + g_i^{(t)} \sigma\left(b_i + \sum_j U_{i,j} x_j^{(t)} + \sum_j W_{i,j} h_j^{(t-1)}\right),$

onde b, U e W são o viés, o peso do *input* e o peso do valor recorrente da rede, respectivamente. Analogamente, a função $g_i^{(t)}$ referente ao *input gate* é calculada utilizando-se uma função sigmoide tal que:

(2.24) $g_i^{(t)} = \sigma\left(b_i^g + \sum_j U_{i,j}^g x_j^{(t)} + \sum_j W_{i,j}^g h_j^{(t-1)}\right).$

O *output gate* $q_i^{(t)}$ é responsável pelo controle do fluxo de $h_i^{(t)}$. Estes dois componentes são descritos pelo par de equações:

(2.25) $h_i^{(t)} = tanh\left(s_i^{(t)}\right) q_i^{(t)}$

e

(2.26) $q_i^{(t)} = \sigma\left(b_i^o + \sum_j U_{i,j}^o x_j^{(t)} + \sum_j W_{i,j}^o h_j^{(t-1)}\right),$

onde, mais uma vez, b, U e W são parâmetros a serem determinados no processo de treinamento.

Resumidamente, são estes controles dentro da célula recorrente da rede neural que permitem à rede LSTM memorizar um certo aspecto relevante dentro de uma sequência e esquecer aspectos irrelevantes, ou ruídos, para o problema de previsão.

Finalmente, uma função de perda deve ser associada à estrutura recorrente visando medir o resultado do mapeamento da sequência de *inputs* $x^{(t)}$ aos valores *target* $y^{(t)}$ no processo de aprendizagem. Utilizando-se a função σ(·), tal como na equação (2.18), para representar a distribuição de probabilidade da variável binária $y^{(t)}$ através do output $h^{(t)}$, tal que:

(2.27) $\hat{y}^{(t)} = \sigma\left(h^{(t)}\right).$

Dessa forma, de acordo com a equação (2.13), a função de perda do tipo *binary cross-entropy* pode ser escrita, para uma rede LSTM, tal como:

(2.28) $$L(\theta) = -\frac{1}{N}\sum_{t=1}^{\tau} y^{(t)} \log\left(\hat{y}^{(t)}\right) + \left(1 - y^{(t)}\right) \log\left(1 - \hat{y}^{(t)}\right).$$

Assim como nos modelos RNRs comuns, o processo de otimização dos parâmetros de uma rede neural LSTM se faz através do método *back-propagation*. A depender da escala do volume de informações do conjunto de treinamento, diferentes abordagens computacionais podem ser utilizadas na prática, visando a melhoria do tempo de processamento e convergência da solução, conforme (Goodfellow et al., 2016).

2.3.5. *Deep Learning* no Mercado Acionário

Os sistemas inteligentes de negociação ganharam espaço no mercado acionário a partir da década de 1980, com o desenvolvimento de programas de computador, ainda rudimentares, que realizavam cálculos automatizados. Os *trading systems* evoluíram e se tornaram mais robustos e eficientes. Atualmente, são desenvolvidos em um contexto de integração de diversos sistemas, acesso a informações em tempo real e uso de algoritmos avançados de AM.

O crescente poder computacional junto com o advento da internet culminou na chamada era do *Big Data*. De acordo com o (Real Time Statistics Project, 2019), existem cerca de 1,7 bilhões de *sites* ativos na internet; mais de 4,3 bilhões de pessoas, 56% da população mundial, tem acesso à internet; e são processadas 3,5 bilhões de pesquisas diárias pelo Google. Somam-se a isso todo o tráfego de dados pelas redes sociais: a cada dia são enviados cerca de 500 milhões de tweets, são trocados 290 bilhões de e-mails e cerca de 4 *petabytes* de dados são criados no Facebook, entre outros exemplos. No âmbito financeiro, são executadas milhões de transações por minuto e milhares de notícias são veiculadas pelas mídias especializadas. Estima-se que o volume de dados armazenados no mundo seja de 44 zettabytes e que a geração diária de dados atinja a marca de 463 exabytes em 2025 (Racounteur, 2019).

Para tangibilizar o valor desse enorme volume de dados, é preciso robustez no processamento e análise automatizada das informações. É exatamente nesse ponto que convergem o poder computacional disponível atualmente e o AM, compreendendo um conjunto de técnicas que permite que um sistema inteligente, um programa de computador, tome uma decisão com

base na experiência obtida nos dados (Goodfellow et al., 2016). Um modelo baseado em AM pode detectar automaticamente padrões nas informações e, em seguida, usá-los para prever eventos futuros ou executar outros tipos de tomada de decisão sob incerteza (Murphy, 2012).

Como visto, a máquina busca otimizar um critério de desempenho, através da calibração dos parâmetros de um certo modelo matemático. Os parâmetros do modelo são apreendidos dos dados de treinamento que representam a experiência passada. No instante da tomada de decisão, portanto no presente, o modelo é então utilizado para realizar previsões do futuro ou para calcular a probabilidade de ocorrência de um certo evento. Em certas aplicações, a eficiência do algoritmo é tão importante quanto sua precisão preditiva, pois o tempo de execução deve ser coerente com o volume de dados do problema (Alpaydin, 2014).

Os *players* do mercado acionário estão utilizando todo o potencial das novas tecnologias da informação, incluindo os métodos de AM discutidos anteriormente, e estabeleceram um novo paradigma de negociações. A aplicação de técnicas de aprendizado de máquinas para apoiar a tomada de decisão do investidor profissional se tornou uma prática mandatória. No mercado americano, cerca de 90% das operações de compra e venda de ações são baseadas em algoritmos de sistemas inteligentes, segundo estimativas de (J.P.Morgan, 2017).

Seguindo a tendência americana, praticamente todos os demais mercados internacionais relevantes, inclusive o brasileiro, intensificaram as aplicações de sistemas inteligentes, especialmente no âmbito das operações de alta frequência, como investigou (Lopes, 2018). No Brasil, estima-se que aproximadamente 40% das operações realizadas no mercado acionário sejam realizadas por algoritmos, em ambientes computacionais conectados diretamente na bolsa de valores, segundo (Paraná, 2016).

Porém, de acordo com (Krauss et al., 2017), devido aos direitos proprietários, os modelos mais avançados e complexos desenvolvidos na indústria financeira, e seus resultados práticos, não são acessíveis aos pesquisadores, quanto menos para o público em geral. No entanto, há uma grande variação de abordagens, desde simples negociação de pares de ações correlacionadas, estratégia denominada *pair trading*, até o uso de métodos preditivos sofisticados, como *random forest*, *gradient boosting* e *deep learning*. Mesmo assim, alguns estudos emergem de centros econômicos relevantes e podem ser listados como exemplos.

(Fischer & Krauss, 2018) analisaram a performance operacional de métodos deep learning, aplicando-se modelos LSTM, no mercado norte americano representado pelo conjunto de ações pertencentes ao índice S&P, obtendo resultados satisfatoriamente acima da média de referência. No mercado asiático, Pang et al. (2018) propuseram uma estrutura com duas camadas de redes neurais LSTM, com alto nível de profundidade. Os testes apresentaram melhor resultado para a previsão do índice composto *Shanghai A-shares* do que a performance da previsão de ações individuais. Por sua vez, (Jeong & Young, 2019) estudaram a rentabilidade de *trading systems*, baseados em *deep learning*, aplicados em índices do mercado americano (S&P 500), mercado europeu (EuroStoxx50), mercado sul-coreano (KOSPI) e mercado chinês (HSI). Em todos os casos, a rentabilidade das operações simuladas foi maior do que a obtida com estratégias tradicionais.

No mercado brasileiro, alguns pesquisadores abordaram o uso de IA baseada em RNAs como ferramenta de apoio a tomada de decisões no âmbito da bolsa de valores. Em sua pesquisa, Parreiras e Vicente (2007) aplicaram redes neurais para a previsão do preço de ações e a partir daí estabeleceram uma série de estratégias de arbitragem estatística para a seleção de cestas de ações, com resultados positivos. Oliveira, Montini e Bergmann (2008) compararam a performance preditiva de modelos Arima-Garch e RNAs para ações de diferentes setores econômicos, como financeiro, industrial, alimentício e de serviços. No caso, ambos modelos apresentaram resultados similares e, conforme os autores, “podem ser utilizadas como ferramentas complementares ... para aumentar a capacidade de decisão” do investidor. Como mencionado no capítulo introdutório, Freitas et al. (2001), Oliveira et al. (2013), Paiva (2014) e, em particular, Nelson et al. (2017) que utilizaram RNRs do tipo LSTM, são exemplos de destaque na aplicação de RNAs para apoiar a tomada de decisão na área de finanças e mercado de capitais.

Por fim, começam surgir evidências de saturação no mercado acionário, no sentido de não mais haver grandes oportunidades de ganhos para aqueles que utilizam modelos sofisticados. Os modelos criados por Fischer e Krauss (2018) apresentam resultados medianos a partir de 2010. Os autores especulam que tais resultados se devem a intensificação do uso de IA em operações de compra e venda de ações e, portanto, este fato estaria mitigando a vantagem competitiva dos pioneiros. Ao explorar um histórico de 22 anos das 300 maiores empresas americanas listadas no índice S&P500, reunindo à base de dados aproximadamente 600

variáveis descritivas, Huck (2019) obteve conclusões similares ao trabalho de Fischer e Krauss: altos ganhos no período de análise compreendido até 2010, e logo após uma queda no poder discriminante dos modelos, embora ainda existam resultados satisfatórios. Tratando-se de mercados maduros, como o americano, a alta competitividade pode atuar em prol da HEM. Nesse sentido, como está evoluindo o mercado brasileiro? Algumas evidências sobre a performance de modelos de AM, em particular, de redes LSTM nesse mercado serão discutidas mais adiante, nesse trabalho.

3. Metodologia

3.1. Introdução

Esse trabalho se configura como uma pesquisa aplicada, desenvolvida para oferecer melhor compreensão sobre o potencial das técnicas de *deep learning* no mercado acionário brasileiro. Também é uma pesquisa de caráter confirmatório, pois já existem evidências empíricas na literatura e, portanto, busca confrontar os resultados obtidos com os resultados já existentes. De acordo com Martins e Theóphilo (2009), essa pesquisa é de caráter quantitativo, pois tem por base a coleta, processamento e análise de dados numéricos a fim de verificar a capacidade do método abordado proporcionar boas tomadas de decisão.

3.2. Hipóteses

Visto que o principal objetivo dessa pesquisa é verificar a eficiência das redes neurais do tipo LSTM como ferramentas de previsão e auxílio à tomada de decisão no mercado acionário, define-se as hipóteses a seguir, que servirão como guias nesse processo de análise. Tais hipóteses são afirmações baseadas em evidências encontradas na literatura e carecem de confirmação no âmbito da bolsa de valores brasileira.

Hipótese 1: é possível superar consistentemente o retorno médio diário do mercado acionário brasileiro com o auxílio de redes neurais do tipo LSTM, desenvolvidas sobre uma adequada seleção de variáveis relacionadas às ações de interesse.

Hipótese 2: os modelos *deep learning* baseados no conceito de RNAs do tipo LSTM proporcionam maior assertividade, e consequentemente, maior rentabilidade, aos portfólios criados com base em suas previsões acerca do retorno diário das ações do Ibovespa.

Tais hipóteses serão analisadas com o apoio da metodologia apresentada a seguir e se espera que os resultados estejam em linha com as pesquisas prévias.

3.3. Método de Pesquisa

O método para investigar a validade das hipóteses estabelecidas seguirá o roteiro proposto por (Kaastra & Boyd, 1996). Esta abordagem, foi concebida especialmente para a realização

de aplicações de redes neurais em dados de séries temporais financeiras. O esquema do Quadro 1 mostra as oito etapas do procedimento. Vale ressaltar o uso deste *framework* nas pesquisas realizadas por (Ashwood, 2013; Fischer & Krauss, 2018; Nelson et al., 2017). Em todas elas, a estratégia de criação de sequências de mesmo tamanho proposta por (Kaastra & Boyd, 1996) se integra perfeitamente ao *design* das redes neurais do tipo LSTM, devido a capacidade que este tipo de modelo possui para lidar com a interdependência temporal dos valores de entrada.

Quadro 1 – Oito passos para o treinamento de redes neurais

Passo 1: Definição de variáveis
Passo 2: Coleta dos dados
Passo 3: Pré-processamento dos dados
Passo 4: Conjuntos de treino, validação e teste
Passo 5: Paradigma da rede neural
- Número de camadas
- Número de neurônios
- Funções de ativação

Passo 6: Critérios de avaliação
Passo 7: Treinamento da rede neural
- Número de iterações
- Learning rate e momentum
- *Time steps*

Passo 8: Implementação

Oito passos do roteiro proposto por (Kaastra & Boyd, 1996) para o treinamento de redes neurais com dados de séries de tempo financeiras. Fonte: Kaastra e Boyd (1996). Adaptado pelo autor.

3.3.1 Definição de Variáveis

A escolha das **variáveis explicativas**, ou fatores que podem contribuir para a previsibilidade do retorno das ações sob análise, seguirá os trabalhos de (Ashwood, 2013), (Fischer & Krauss, 2018), (Krauss et al., 2017), (Zhang et al., 2018) e (Nelson et al., 2017). Estes trabalhos possuem o desafio comum de desenvolver modelos de AM, incluindo RNAs do tipo LSTM, que permitam a previsão do movimento dos preços de ações, além de cotações de índices, em mercados variados como Austrália, Europa, Estados Unidos, China e Brasil, respectivamente. Também foi levado em consideração as sugestões e a experiência dos membros da banca de

qualificação, do orientador da pesquisa, bem como do próprio autor. O item (a) do Quadro 2 apresenta de forma resumida o conceito das fontes de informações.

É comum chamar a **variável independente** de um modelo de classificação ou regressão como variável resposta, ou apenas *target*. Nesse caso, busca-se modelar a probabilidade do retorno diário de uma determinada ação ser maior que zero, conforme Nelson et al. (2017). De modo alternativo, a variável *target* pode ser definida de acordo com Fischer e Krauss (2018), que utilizaram a mediana da distribuição de retornos como ponto de corte. Portanto, um problema de classificação binária deverá ser resolvido, onde a variável *target* $\boldsymbol{y_t^s}$ é definida como segue.

Alternativa 1: seja $p^s = (p_t^s)_{t \in T}$ a sequência de preços de uma ação s em um intervalo de tempo T, com $s \in \Omega_i = \{1,2, \dots, n_i\}$, o conjunto de n_i ações disponíveis no período de estudo $i = 1, 2, \dots, p$. Então, o retorno diário r_t^s de uma certa ação s é definido como:

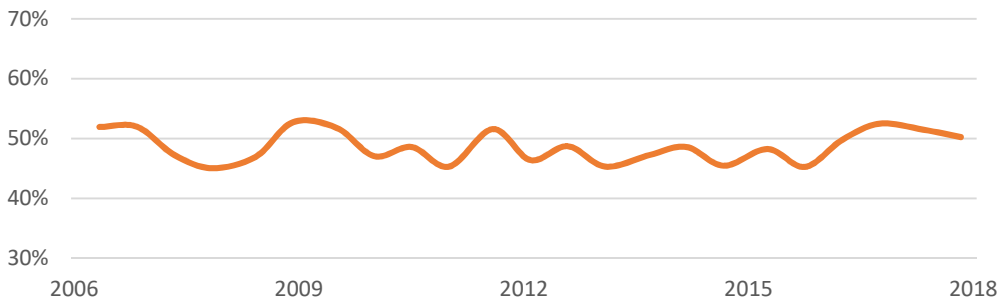
(3.1) $$r_t^s = \frac{p_t^s}{p_{t-1}^s} - 1$$

Com isso, define-se y_t^s no domínio $\{0,1\}$ tal que:

(3.2) $$y_t^s = \begin{cases} 1, & se\ r_t^s \geq 0 \\ 0, & se\ r_t^s < 0 \end{cases}, \text{ para qualquer } t \in T \text{ e } s \in \Omega_i, \forall\ i.$$

O Gráfico 4 mostra que a taxa média de eventos $y_t^s = 1$ ao longo do tempo gira em terno de 50%.

Gráfico 4 – Taxa média de eventos da variável



Taxa média de eventos da variável binária $\boldsymbol{y_t^s}$, ao longo de um período de aproximadamente dez anos. Fonte: dados dos preços das ações coletados no Economática e gráfico elaborado pelo autor.

Alternativa 2: para a definição das duas classes, ordena-se os retornos r_t^s dados pela equação (3.1) de todas as ações s em cada dia t. Com base no retorno mediano dessa lista, define-se a variável *target* tal que:

$$y_t^s = \begin{cases} 1, & se\ r_t^s \geq \text{retorno mediano do dia } t \\ 0, & se\ r_t^s < \text{retorno mediano do dia } t \end{cases}, \text{ para qualquer } t \in T \text{ e } s \in \Omega_i, \forall\, i. \tag{3.3}$$

3.3.2 Coleta dos Dados

As informações utilizadas nesta pesquisa são de natureza secundária. Todas as fontes dos dados estão listadas no item (a) do Quadro 2. As informações foram extraídas do sistema Economatica e dos respectivos sites das fontes, como B3, IPEA e Portal Investing, disponíveis gratuitamente na internet.

Quadro 2 – Catálogo de fontes de informações

Fonte	Informação
Aplicativo Economatica	Preço de fechamento de ações ROE ROIC Índice de força relativa Índice de sharpe
Portal Investing.com	Índice diário Standard & Poor's 500 Índice diário Dow Jones Industrial Índice Ibovespa futuro Preço fechamento de dólar Preço fechamento de petróleo Preço fechamento de minério de ferro
IPEADATA	Índice EMBI+ Risco-Brasil
B3	Composição de ações do Ibovespa

(a)

codigo, data, fechamento
VALE3, 20161129 18:00:00.000, 27.4643
VALE3, 20161130 18:00:00.000, 26.4102
VALE3, 20161201 18:00:00.000, 26.7960
VALE3, 20161202 18:00:00.000, 27.9191
VALE3, 20161205 18:00:00.000, 28.0233
VALE3, 20161206 18:00:00.000, 28.4012
…

(b)

(a) Relação das fontes de informações utilizadas nessa pesquisa, incluindo as principais referências, confirme aplicações semelhantes. (b) Exemplo de informação de preço de fechamento de ações da Vale S.A., formatadas como uma tabela de banco de dados relacional. Fonte: elaborado pelo próprio autor.

Cada informação coletada foi armazenada, catalogada e formatada em formato de banco de dados relacional, possuindo como campo-chave o código da ação e a data de referência da informação, como exemplificado no item (b) do Quadro 2.

3.3.3 Pré-processamento dos Dados

Todos os dados de entrada e saída utilizados devem ser pré-processados a fim de garantir a melhor performance do algoritmo e eficiência no processo de aprendizagem. A variável *target* deve ser tratada contra possíveis valores vazios, chamados *missings values*. Os períodos em que a variável *target* permanece com valores constantes também devem ser excluídos, pois não agregam nenhuma informação à rede neural, conforme (Ashwood, 2013, p. 79).

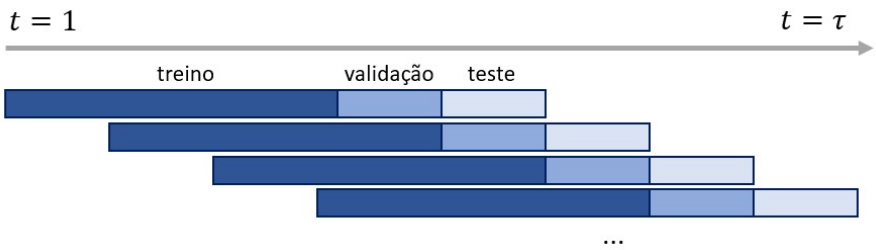
Cada uma das variáveis explicativas deve ser escalonada, em função de seus valores mínimo e máximo, para uma nova escala entre -1 e 1. Este procedimento é recomendado para o treinamento de redes neurais visando a adequação de diferentes magnitudes de variáveis. Por exemplo: o retorno anual sobre o investimento de uma empresa está em uma escala, em geral, bem menor que o nível do Índice Bovespa ou a cotação do câmbio (real *versus* dólar). Outra razão para o escalonamento das variáveis de entrada neste tipo de rede refere-se à função de ativação sigmoide.

Todos os registros com dados ausentes são tratados, substituídos por zero ou pela média da variável (*imputation*),dependendo do contexto da informação, ou excluídos se necessário.

3.3.4 Conjuntos de Treino, Validação e Teste

A arquitetura especial da LSTM exige uma estrutura de dados em formato de sequências. Seguindo (Ashwood, 2013), (Bao, Yue, & Rao, 2017) e (Fischer & Krauss, 2018), o conjunto de dados foi particionado em três subconjuntos, a saber: treino, validação e teste, conforme o esquema da Figura 13. O tamanho das sequências de treino, validação e teste pode ser variado a fim de encontra a melhor configuração para o problema em questão. Todas as ações compreendidas em um certo intervalo de tempo são consideradas, desde que satisfaçam as condições de pré-processamento dos dados de entrada.

Figura 13 – Conjuntos de treino, validação e teste



Separação dos dados de entrada em bases de treino, validação e teste. Fonte: elaborado pelo próprio autor, adaptado de (Fischer & Krauss, 2018).

3.3.5 Paradigma da Rede Neural

Encontrar a arquitetura ideal da rede neural do tipo LSTM para o contexto brasileiro requer o teste de diversos cenários, em termos da configuração dos principais elementos do modelo, tais como a quantidade de camadas (*layers*), a quantidade de neurônios por camada (*neurons*) e as funções de ativação dos neurônios ocultos e de saída da rede.

Dessa forma, serão avaliadas combinações dos seguintes conjuntos de parâmetros:

- *Layers*: 1, 2, 3, 4 e 5
- *Neurons*: 5, 10, 25, 50 e 100

A função sigmoide, tal como apresentada na equação (2.18), será atribuída a todas as funções de ativação dos neurônios da rede, seguindo o modelo LSTM proposto por (Fischer & Krauss, 2018) para previsão de retornos de ações do mercado americano.

3.3.6 Critérios de Avaliação

Os critérios de avaliação visam analisar a performance preditiva, assim como o potencial financeiro dessa metodologia. Sendo assim, a **acurácia do modelo** será analisada como medida de capacidade preditiva. Além disso, como um critério secundário de avaliação, será verificado a **precisão do modelo**, tal como apresentado na seção 2.2.2. Também será calculada a **rentabilidade diária do portfólio** selecionado com uma estratégia *long-short* simples, seguindo (Krauss et al., 2017) e (Fischer & Krauss, 2018).

A estratégia *long-short* se dará da seguinte maneira: para todo modelo, (i) calcula-se a probabilidade predita $\hat{y}_{t+1}^{s}$ de cada ação s obter um retorno positivo no dia $t + 1$, utilizando-

se apenas as informações (*features*) disponíveis até o dia t; (ii) lista-se todas as ações disponíveis para o dia $t + 1$ em ordem descendente dessas probabilidades. O topo dessa lista corresponde às ações com maior potencial de valorização, enquanto no final da lista encontram-se as ações com menor probabilidade de valorização (maior chance de desvalorização); (iii) então simula-se a compra de k ações com a maior probabilidade de retorno positivo (k primeiras ações da lista) e a venda de k ações com menor chance de valorização (k últimas ações da lista). Este processo será repetido ao longo dos períodos de teste do modelo ao longo do horizonte de simulação, compreendendo os anos de 2007 até 2018.

3.3.7 Treinamento da Rede Neural

O treinamento do modelo se dará por meio das bibliotecas Keras e TensorFlow, que permitem configurar o paradigma da rede de forma amigável, além de otimizar o processo de aprendizado. Os principais elementos de aprendizagem referem-se ao método de otimização da função de perda, e consequentemente, dos demais parâmetros do modelo; ao uso do mecanismo de *dropout regularization* na camada de recorrência, conforme (Gal & Ghahramani, 2016); e a manutenção (*stateful*) ou reinicialização (*stateless*) da memória *cell state* da rede LSTM ao longo das iterações (*epochs*), sendo que a configuração stateless é mais apropriada para problemas que não recebem dados *on-line*, de acordo com (Brownlee, 2019).

Tratando-se de um problema de classificação binária, de acordo com o domínio da variável *target* definido em (3.2), será buscada a minimização da função de perda (*loss function*) do modelo, que segue o conceito apresentado na equação (2.28). Esta função é convexa e pode ser minimizada utilizando o método do gradiente descendente, além de atender os critérios de diferenciabilidade do método *back-propagation* para obtenção dos demais parâmetros da rede (Hastie et al., 2008).

Em particular, para a otimização dos parâmetros da rede, serão testados os algoritmos RMSprop e Adam. O primeiro é sugerido por (Chollet, 2019) como um boa escolha para modelos baseados em RNRs e utilizado por (Fischer & Krauss, 2018), enquanto o segundo algoritmo é eficiente e tem apresentado melhores resultados quando comparada a outros métodos estocásticos, segundo (Kingma & Ba, 2015).

Por último, também serão avaliados diferentes comprimentos de sequência dos dados de entrada (*time steps*) para o treino dos modelos. O mesmo comprimento utilizado no treinamento será aplicado nos períodos de validação e teste.

3.3.8 Implementação

Adota-se a linguagem de programação Python para a implementação do projeto. Todos os códigos estão disponíveis para consulta e utilização no portal Github, em https://github.com/felipetshr/lstm_ibov. No mesmo endereço estão disponíveis os dados de entrada e todos os resultados obtidos nas simulações realizadas. Os códigos também estão disponíveis no Apêndice B ao final deste trabalho.

3.4. Ambiente Computacional

O *framework* computacional criado para o desenvolvimento dessa pesquisa compreende uma combinação de bibliotecas *open source* disponíveis em linguagem Python 3.6. A interface utilizada para o desenvolvimento de códigos se chama Spyder, programa integrante do conjunto de aplicações Anaconda, atualmente o mais popular ambiente de ciência de dados (Anaconda, 2018).

Com o foco no desenvolvimento de RNAs do tipo LSTM, as bibliotecas Keras (Keras, 2018), TensorFlow (Google, 2018), NVIDIA CUDA Toolkit (NVIDIA, 2018a), NVIDIA CUDA® Deep Neural Network (NVIDIA, 2018b) foram utilizadas, como em (Fischer & Krauss, 2018) e (Nelson et al., 2017). Para a etapa de pré-processamento dos dados, foram utilizadas as bibliotecas sklearn, pandas e nympy, todas disponíveis no ambiente Anaconda.

O equipamento utilizado possui as seguintes especificações relevantes para o treinamento de redes neurais: CPU Intel i7 5500U com 2 núcleos de 2,40 GHz cada, memória RAM DDR3 com 8GB e placa de vídeo NVIDIA GeForce 940M com uma GPU de 384 núcleos de processamento paralelo (*cuda cores*). Esta configuração permite a execução em paralelo do algoritmo de treinamento dos modelos.

4. Análise dos Resultados

A seguir serão apresentados os principais resultados identificados nessa pesquisa. Primeiramente, serão apresentados os resultados obtidos com modelos LSTM simples, desenvolvidos para prever o retorno diário de ações constituintes do Ibovespa. Em seguida, serão apresentados os resultados obtidos com modelos LSTM mais complexos, em termos de profundidade e quantidade de neurônios da rede neural, chamados *deep learning*, também desenvolvidos para prever o retorno diário das ações do Ibovespa. Em ambos os casos, verifica-se o potencial dos modelos para aplicações práticas no mercado acionário.

4.1. Modelos LSTM Simples para Ações do Ibovespa

Seguindo-se a metodologia proposta, uma série de experimentos foram realizados para se determinar o *design* apropriado de uma rede neural simples, com apenas uma camada oculta formada por neurônios recorrentes do tipo LSTM. Em cada experimento, foi medido a performance preditiva do modelo e seu potencial econômico, em termos da rentabilidade dos portfólios selecionados diariamente ao longo do horizonte de simulação. Foram definidos portfólios simulando-se a compra de k ações, assim como a venda de k ações. Com isso, foi possível analisar o resultado consolidado, em termos da estratégia *long-short* apresentada previamente, assim como os resultados individuais para compra (*long*) e venda (*short*).

Foram testados diversos cenários de configuração das RNAs do tipo LSTM, conforme listagem apresentada no Apêndice C. Em cada cenário, incluiu-se de forma incremental as variáveis discutidas anteriormente, visando identificar a melhor combinação de fatores preditivos.

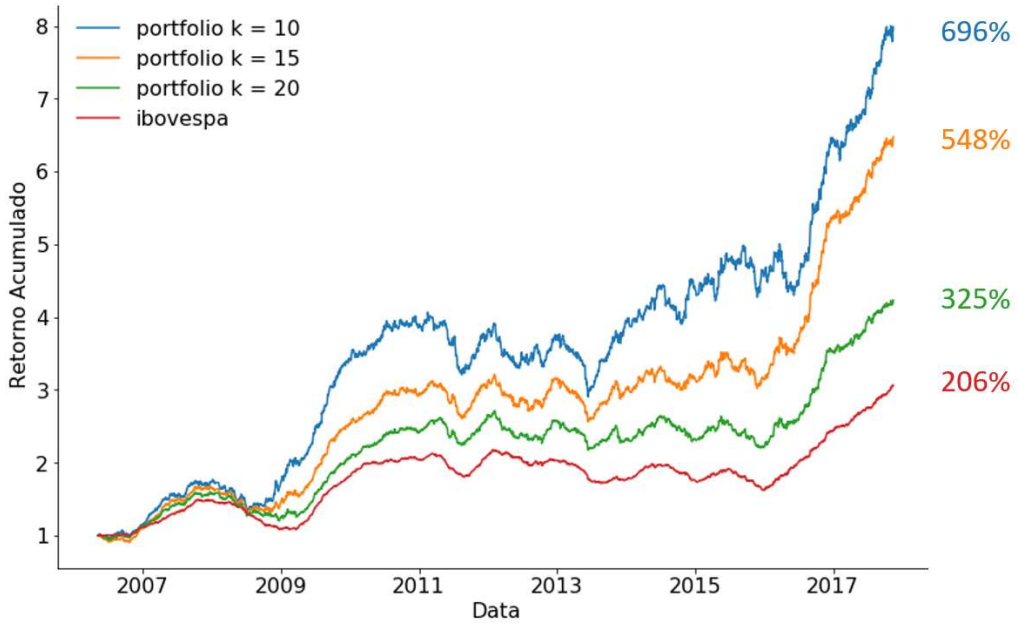
O melhor desempenho, em termos de uma estratégia de compra de ações, foi obtido pelo modelo baseado no paradigma constituído por uma camada oculta de 25 unidades neuronais e sequências compostas por 21 *time steps* (cenário lstm_simples_15_1). Nessa configuração, foram disponibilizadas 6 variáveis explicativas para o processo de aprendizagem do modelo. Todas as variáveis possuem atualização diária, sendo elas: preço de fechamento da ação, retorno da ação, índice S&P500, índice DJI, índice Ibovespa e preço do dólar comercial.

Nota-se que, em média, foram utilizados 378 dias em cada batelada de simulação. Para o conjunto de dados de treinamento, foram utilizados 202 dias para se construir as sequências

de *inputs* da rede LSTM; para a validação do modelo, 50 dias; e por fim, o período de teste compreendeu 126 dias.

O Gráfico 5 mostra a evolução do retorno acumulado das carteiras teóricas formadas pela simulação de compra de $k = 10, 15$ ou 20 ações, considerando-se apenas os dados de teste do modelo. Em primeiro lugar, destaca-se a superioridade do método proposto, em termos da rentabilidade acumulada final obtida pelos portfólios criados, frente ao índice médio de mercado.

Gráfico 5 – Retorno acumulado por portfólios *long position*: LSTM simples 1 x 25



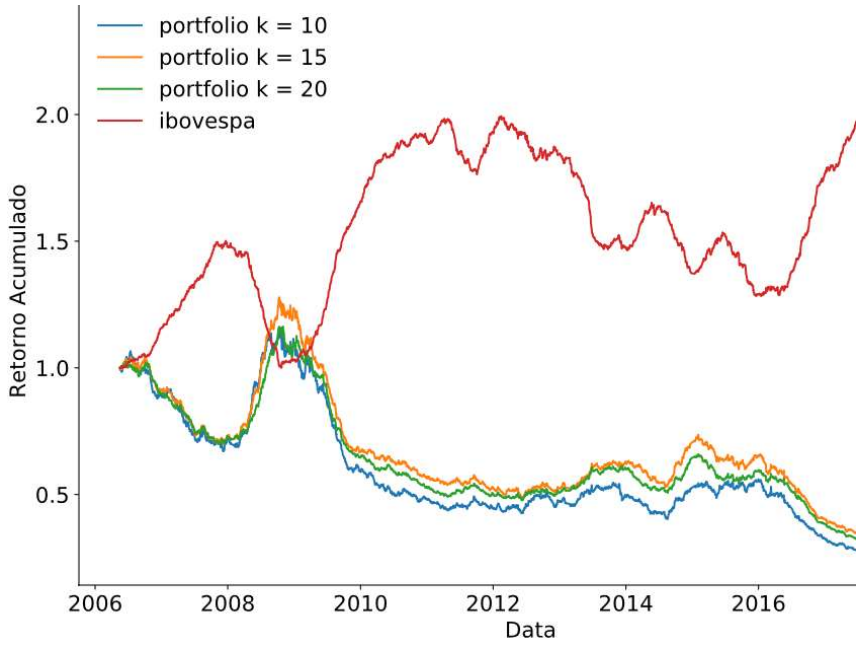
Retorno acumulado de carteiras teóricas compostas por k = 10, 15 e 20 ações ao longo de 10 anos. Os retornos referem-se à simulação de compra (*long position*) de ações. A curva portfolio = 10, em azul, apresenta o maior retorno acumulado, enquanto o portfólio = 20, em verde, se aproxima do Ibovespa, em vermelho, que representa o retorno médio do mercado naquele período. Fonte: elaborado pelo autor.

Observa-se ainda que o retorno acumulado de todas as carteiras é consistentemente maior que o retorno acumulado do índice brasileiro, ao longo de praticamente todo o período de simulação. Embora exista uma pequena vantagem para o Ibovespa nos primeiros meses, a partir daí o excesso de retorno dos portfólios evoluem positivamente. Com isso, há uma evidência sobre a consistência da metodologia proposta e sobre a sua capacidade de extrair informações preditivas das variáveis de entrada, típicas séries temporais, mesmo se considerando uma arquitetura mais simples.

Mais especificamente, o portfólio gerado diariamente pela compra de 10 ações apresentou o melhor resultado no longo prazo, entregando um retorno acumulado de 696% em pouco mais de 10 anos. O retorno médio desta carteira teórica de ações foi de 22,3% a.a., sem considerar o desconto de taxas (*e.g.* corretagem e emolumentos) e impostos. Este resultado é 102% maior que o retorno médio anual do Ibovespa, no mesmo período, conforme os dados sumarizados na Tabela 2.

A carteira composta por 15 ações apresentou resultado um pouco menor, chegando a um retorno acumulado de 548% ao final do horizonte de simulação. Já o portfólio gerado pela simulação de compra de 20 ações apresentou um retorno mais próximo do índice médio do mercado. Esse fato é coerente com a expectativa de que, na medida em que o número de ativos de uma certa carteira se aproxima da quantidade de ativos totais disponíveis, as médias de seus retornos tendem a se igualarem.

Gráfico 6 – Retorno acumulado por portfólios *short position*: LSTM simples 1 x 25



Retorno acumulado de carteiras teóricas compostas por k = 10, 15, 20 ações ao longo de 10 anos. Os retornos referem-se à simulação de venda (*short position*) de ações. A curva portfolio = 10, em azul, apresenta o pior retorno acumulado, enquanto o portfólio = 15, em amarelo, apresenta um retorno ligeiramente melhor. Todos os portfólios apresentaram retorno final muito ruim, perdendo mais que 50% do valor. Fonte: elaborado pelo autor.

Por outro lado, com uma abordagem análoga, porém utilizando as menores probabilidades de alta (*i.e.* maiores chances de queda dos preços), verificou-se que a estratégia de venda de ações não apresentou bons resultados. Como pode ser observado no Gráfico 6, todos os

portfólios que simularam a venda de ativos perderam mais de 50% de seu valor ao longo do horizonte de simulação.

No entanto, nos períodos de queda generalizada dos preços, como pode ser observado na trajetória da linha referente ao Ibovespa (vermelha), nota-se boa assertividade do modelo preditivo, pois há uma clara reação dos retornos acumulados dos portfólios. Especula-se que o motivo desse resultado negativo se relaciona ao fato de haver poucos ativos listados no Ibovespa, comparando-se aos resultados obtidos por (Fischer & Krauss, 2018), no contexto do S&P500, onde existem 500 ativos disponíveis. No caso brasileiro, a maioria dos ativos tendem a acompanhar o movimento do índice e, no longo prazo, uma estratégia de compra é mais favorecida, pois nem sempre haverá uma quantidade razoável de ações (10, 15 ou 20) se desvalorizando.

Tabela 2 – Estatísticas dos portfólios de compra e venda de ações: LSTM simples 1 x 25

Portfólio *Long*	**k = 10**	**k = 15**	**k = 20**	**Ibovespa**
Quantidade	2731	2731	2731	2731
Média	0.0008	0.0007	0.0005	0.0004
Desvio Padrão	0.0086	0.0070	0.0060	0.0034
Mínimo	-0.0429	-0.0345	-0.0301	-0.0154
Q1	-0.0040	-0.0031	-0.0029	-0.0015
Mediana	0.0007	0.0007	0.0006	0.0006
Q3	0.0056	0.0046	0.0039	0.0024
Máximo	0.0612	0.0613	0.0398	0.0209
Retorno Mensal	0.0169	0.0150	0.0116	0.0088
Retorno Anual	0.2226	0.1952	0.1478	0.1102

Portfólio Short	**k = 10**	**k = 15**	**k = 20**	**Ibovespa**
Quantidade	2731	2731	2731	2731
Média	-0.0007	-0.0005	-0.0004	0.0004
Desvio Padrão	0.0084	0.0069	0.0059	0.0034
Mínimo	-0.0664	-0.0461	-0.0372	-0.0154
Q1	-0.0051	-0.0042	-0.0037	-0.0015
Mediana	-0.0005	-0.0004	-0.0005	0.0006
Q3	0.0040	0.0034	0.0028	0.0024
Máximo	0.0516	0.0389	0.0360	0.0209
Retorno Mensal	-0.0136	-0.0099	-0.0083	0.0088
Retorno Anual	-0.1518	-0.1124	-0.0952	0.1102

Estatísticas sobre os resultados simulados dos portfólios de compra (*long*) e venda (*short*) de ações ao longo de 10 anos, utilizando-se uma estrutura simples de rede neural do tipo LSTM, com uma camada de 25 neurônios. Fonte: elaborado pelo autor

Dessa forma, o desempenho simulado da carteira *long-short*, que consolida os resultados diários das posições de compra e de venda de ativos, foi pior que o retorno acumulado do Ibovespa, impactado pelas posições vendidas, como mostra a Tabela 3.

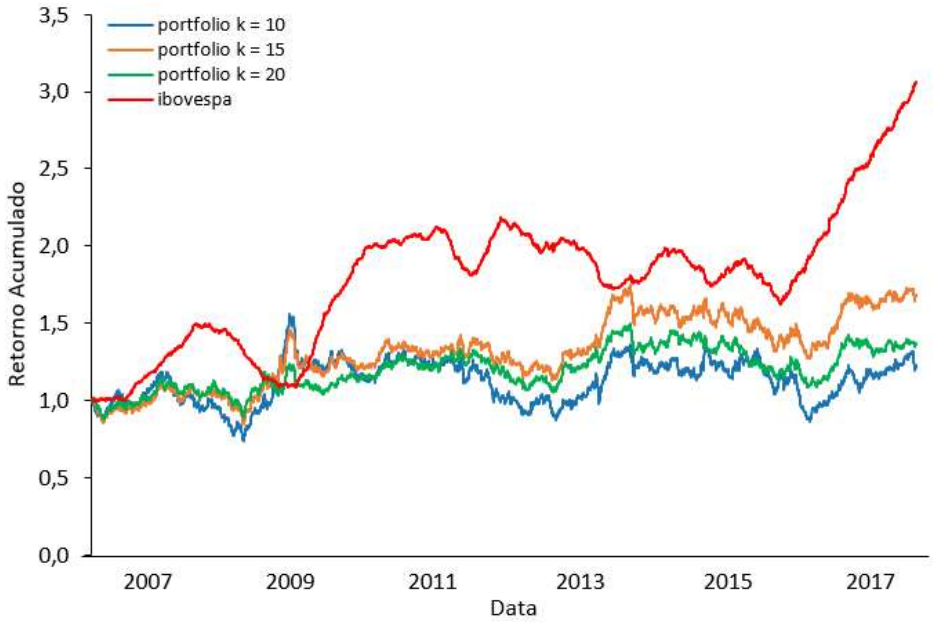
Tabela 3 – Resultados do portfólio *long-short*: LSTM simples 1 x 25

Portfólio *Long-Short*	**k=10**	**k=15**	**k=20**	**Ibovespa**
Retorno Diário Médio	0.01%	0.02%	0.01%	0.04%
Retorno Mensal Médio	0.30%	0.50%	0.31%	0.88%
Retorno Anual Médio	3.72%	6.11%	3.84%	11.02%
Retorno Acumulado	21.42%	67.25%	36.73%	206.16%

Resultados médios obtidos pela consolidação de operações de compra (*long*) e venda (*short*) de ações ao longo de 10 anos, utilizando-se uma estrutura simples de rede neural do tipo LSTM, com uma camada de 25 neurônios. Fonte: elaborado pelo autor.

No Gráfico 7 é possível verificar que ao longo de todo o período de testes, a carteira consolidada superou o índice de referência apenas em um breve intervalo, justamente no qual o portfólio de venda obteve melhor resultado. Vale notar que em todos os outros cenários simulados a mesma situação foi observada, não sendo encontrada uma situação vantajosa para a estratégia de venda de ações utilizando-se redes LSTM simples.

Gráfico 7 – Retorno acumulado por portfólios long-short position: LSTM simples 1 x 25



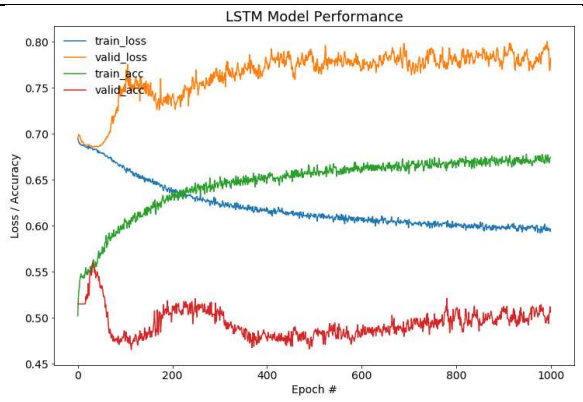
Retorno acumulado referente às carteiras *long-short* teóricas compostas por k = 10, 15, 20 ações, selecionadas diariamente ao longo de 10 anos. Os retornos referem-se ao resultado composto de simulações de compra (*long position*) e venda (*short position*) de ações. Fonte: elaborado pelo autor.

A título de exemplificação, o Quadro 3 exibe uma série de elementos obtidos no processo de aprendizagem e simulação das estratégias de compra e venda de ações com base nas previsões do modelo. Nesse caso, os dados do quadro se referem a um período de desvalorização, em geral, das ações do Ibovespa.

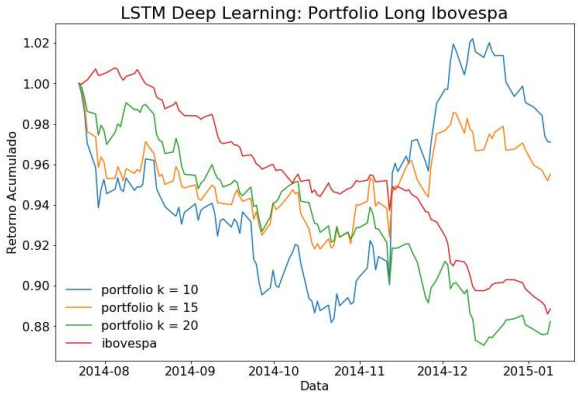
Quadro 3 – Previsões com LSTM simples em tempos de desvalorização

Data início treino	2013-08-27
Data fim treino	2014-07-16
Data início teste	2014-07-22
Data fim teste	2015-01-08
Quantidade de variáveis	6
Quantidade de ações	68
Loss treino	0.6808
Loss teste	0.7540
Gap de Generalização	0.0732
Acurácia treino	0.5701
Acurácia teste	0.5450
Taxa de Eventos Treino	0.4838
Taxa de Eventos Teste	0.4550
Retorno Portfolio k = 10	-0.0180
Retorno Portfolio k = 15	-0.0366
Retorno Portfolio k = 20	-0.1086
Retorno Ibovespa	-0.1025

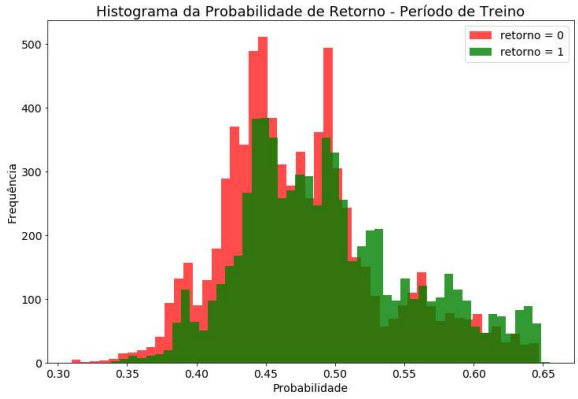
(a)



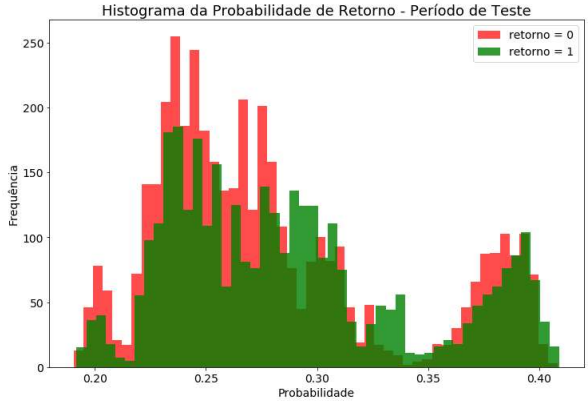
(b)



(c)



(d)



(e)

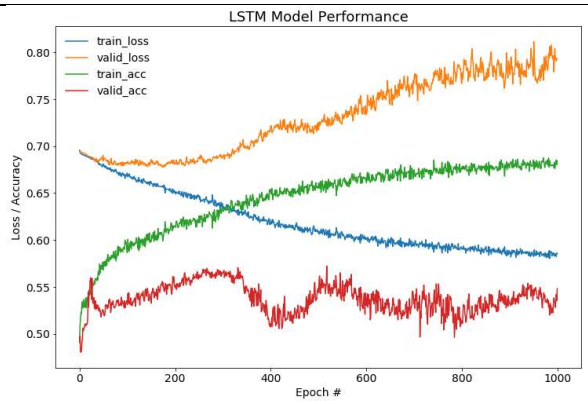
Previsões com LSTM simples em tempos de desvalorização: resultados do modelo treinado em um período de baixa da bolsa de valores. Em (a), são consolidados os principais indicadores de performance obtidos no período de treino e teste. Em (b), observa-se a evolução dos indicadores de performance de treino e validação ao longo das iterações do algoritmo. (c) apresenta o retorno acumulado de carteiras teóricas comparadas ao Ibovespa. (d) e (e) apresenta as distribuições de probabilidade do modelo, no período de treino e teste, respectivamente. Fonte: elaborado pelo autor.

Analogamente, o Quadro 4 apresenta elementos referentes a um período de valorização generalizada das principais ações da bolsa brasileira. Em particular, é notável como, em ambos os casos, o modelo apresenta melhor desempenho quando a tendência do movimento dos preços dos ativos é mais acentuada e persistente.

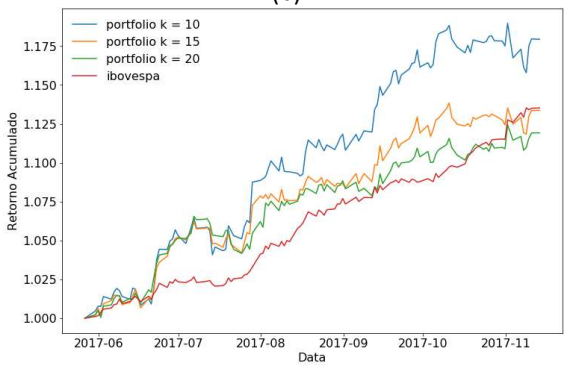
Quadro 4 – Previsões com LSTM simples em tempos de valorização

Data início treino	2016-07-12
Data fim treino	2017-05-24
Data início teste	2017-05-26
Data fim teste	2017-11-14
Quantidade de variáveis	6
Quantidade de ações	56
Loss treino	0.6066
Loss teste	0.8635
Gap de Generalização	0.2569
Acurácia treino	0.6678
Acurácia teste	0.5049
Taxa de Eventos Treino	0.5142
Taxa de Eventos Teste	0.5028
Retorno Portfolio k = 10	0.1505
Retorno Portfolio k = 15	0.1151
Retorno Portfolio k = 20	0.1042
Retorno Ibovespa	0.1112

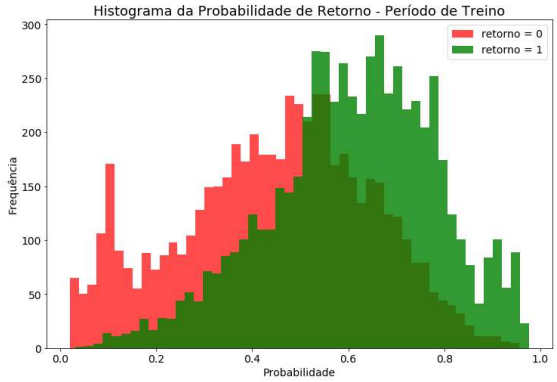
(a)



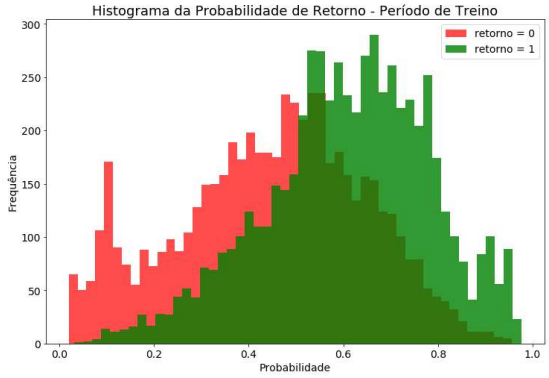
(b)



(c)



(d)

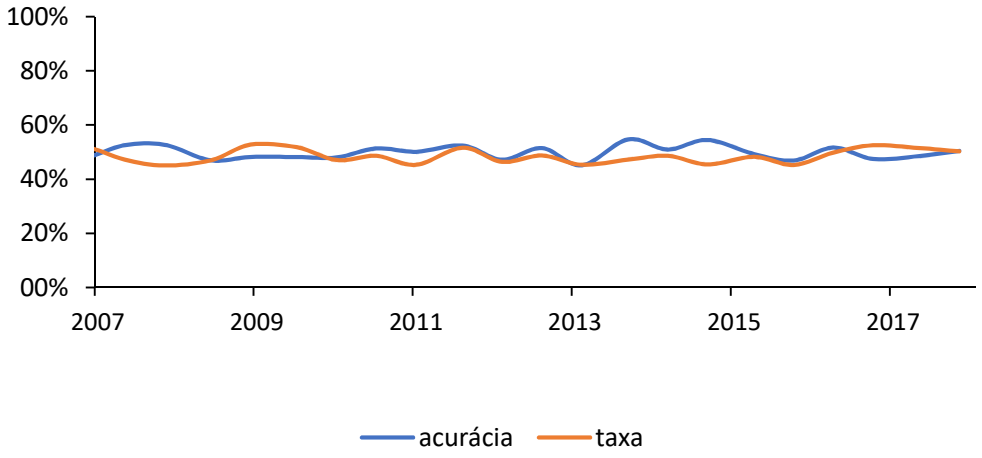


(e)

O quadro apresenta os resultados do modelo treinado em um período de alta da bolsa de valores. Em (a), são consolidados os principais indicadores de performance obtidos no período de treino e teste. Em (b), observa-se a evolução dos indicadores de performance de treino e validação ao longo das iterações do algoritmo. (c) apresenta o retorno acumulado de carteiras teóricas comparadas ao Ibovespa. (d) e (e) apresenta as distribuições de probabilidade do modelo, no período de treino e teste, respectivamente. Fonte: elaborado pelo autor.

Por fim, os resultados econômicos obtidos pelo modelo apresentado acima, especialmente no que tange à estratégia de compra de ações, se justifica pela relativa estabilidade da acurácia das previsões ao longo do tempo. Embora não tenha sido possível obter um índice elevado, situação comum em problemas de previsão do retorno de ações, o modelo acerta em torno de 52% das vezes, consistentemente, como evidenciado no Gráfico 8.

Gráfico 8 – Acurácia do modelo LSTM simples



A curva azul refere-se à acurácia do modelo ao longo dos períodos de testes do modelo e a curva laranja refere-se à taxa de eventos da variável *target*, construída utilizando-se a alternativa 1. Fonte: elaborado pelo autor.

Complementarmente, os mesmos experimentos foram realizados utilizando a definição alternativa da variável *target*, criada com base no retorno mediano diário das ações. Em geral, os resultados obtidos são similares aos analisados acima. Alguns cenários se mostraram bastante rentáveis, como é o caso de uma rede neural construída com uma camada de 50 unidades LSTM, como resume a Tabela 4 a seguir.

Todos os dados coletados, processados e os respectivos resultados obtidos nos mais diversos experimentos podem ser consultados em https://github.com/felipetshr/lstm_ibov. As demais tabelas e análises comparativas foram omitidas aqui em detrimento da clareza e simplicidade de apresentação dos principais fatos. A seguir, serão apresentados e analisados os principais resultados obtidos com estruturas mais complexas de RNAs do tipo LSTM, comumente chamadas de modelos *deep learning*.

Tabela 4 – Performance e rentabilidade referente à variável *target* 2

(a) Performance preditiva		Função de Perda	Acurácia	Precisão
Treino	Média	0.7025	0.5040	0.5102
	Desvio Padrão	0.0052	0.0099	0.0059
	Mínimo	0.6966	0.4969	0.5047
	Q1	0.7006	0.4983	0.5071
	Mediana	0.7046	0.4998	0.5095
	Q3	0.7055	0.5075	0.5130
	Máximo	0.7064	0.5153	0.5165
Teste	Média	0.7068	0.5077	0.5088
	Desvio Padrão	0.0200	0.0135	0.0041
	Mínimo	0.6931	0.4967	0.5043
	Q1	0.6953	0.5002	0.5071
	Mediana	0.6976	0.5036	0.5098
	Q3	0.7136	0.5132	0.5111
	Máximo	0.7297	0.5228	0.5124

(b) Rentabilidade diária		k = 10	k = 15	k = 20
Portfólio Long-Short	Média	0.0011	0.0008	0.0006
	Desvio Padrão	0.0175	0.0143	0.0124
	Mínimo	-0.1049	-0.0871	-0.0687
	Q1	-0.0073	-0.0063	-0.0060
	Mediana	0.0001	0.0006	0.0001
	Q3	0.0083	0.0069	0.0064
	Máximo	0.1003	0.0752	0.0610
Portfólio Long	Média	0.0012	0.0010	0.0008
	Desvio Padrão	0.0234	0.0222	0.0215
	Mínimo	-0.0994	-0.1031	-0.1010
	Q1	-0.0090	-0.0089	-0.0083
	Mediana	0.0010	0.0011	0.0009
	Q3	0.0113	0.0097	0.0096
	Máximo	0.1792	0.1717	0.1596
Portfólio Short	Média	0.0000	-0.0002	-0.0002
	Desvio Padrão	0.0207	0.0203	0.0197
	Mínimo	-0.0832	-0.1173	-0.1142
	Q1	-0.0110	-0.0102	-0.0097
	Mediana	-0.0010	-0.0008	-0.0009
	Q3	0.0106	0.0099	0.0092
	Máximo	0.0921	0.0978	0.1025

(c) Rentabilidade anual	k = 10	k = 15	k = 20
Portfólio *Long-Short*	33.5%	22.7%	16.1%
Portfólio *Long*	35.1%	28.2%	22.4%
Portfólio *Short*	-1.2%	-4.3%	-5.2%

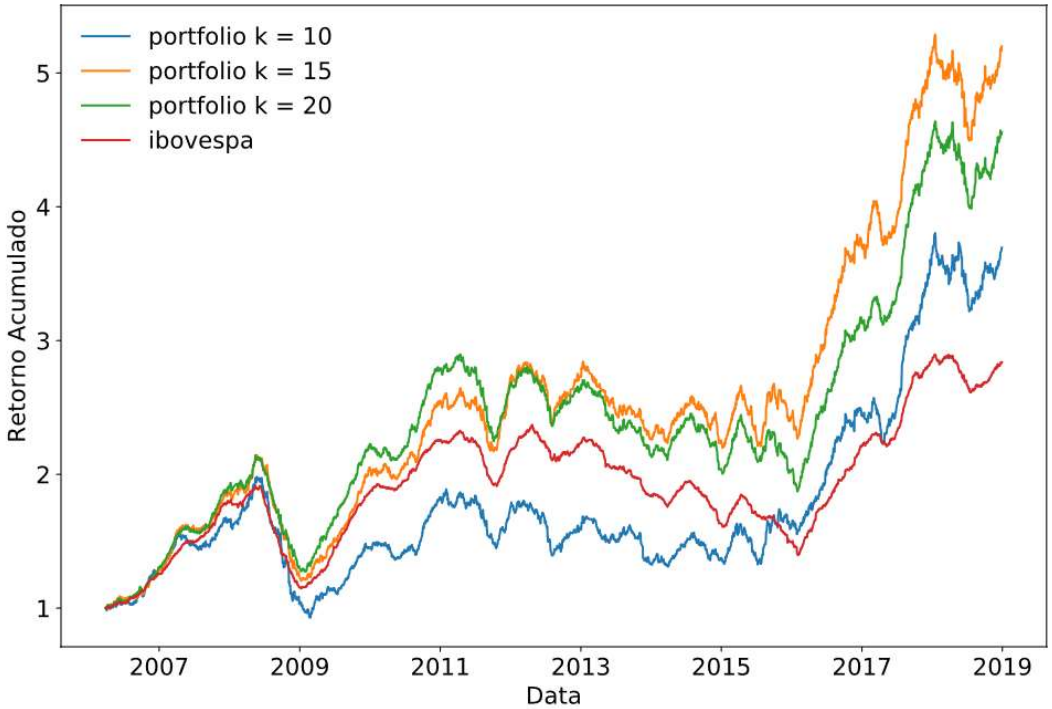
A tabela apresenta (a) os resultados de performance nos períodos de treino e teste de um modelo LSTM de uma camada com 50 neurônios; (b) estatísticas sobre a rentabilidade diária simulada dos portfólios de compra (long) e venda (short) de ações ao longo de 10 anos, além do resultado consolidado *long-short* e (c) a rentabilidade anual média para cada um dos portfólios simulados. Fonte: elaborado pelo autor.

4.2. Modelos LSTM *Deep Learning* para Ações do Ibovespa

Como esperado, os resultados obtidos pela aplicação de uma estrutura LSTM com duas ou mais camadas foram satisfatórios, com rentabilidade acima da média do mercado, em termos gerais. Porém, inesperadamente, os retornos acumulados ao longo do período de avaliação se mostram aquém daqueles obtidos com uma rede mais simples. A seguir serão analisados dois exemplos de modelos *deep learning,* treinados a partir das mesmas premissas utilizadas no desenvolvimento dos modelos mais simples.

O Gráfico 9 mostra o retorno acumulado de portfólios selecionados com base nas previsões de retorno das ações integrantes do Ibovespa, geradas por uma rede neural multicamadas, composta por três camadas e 32 unidades LSTM por camada. Analogamente aos experimentos anteriores, foi simulado a compra de $k = 10, 15$ ou 20 ações, diariamente. Diferentemente do caso anterior, observa-se algumas inversões das curvas de retornos ao longo do tempo. Além disso, a curva de retorno do portfólio constituído por 10 ações passou grande parte do tempo abaixo do Ibovespa.

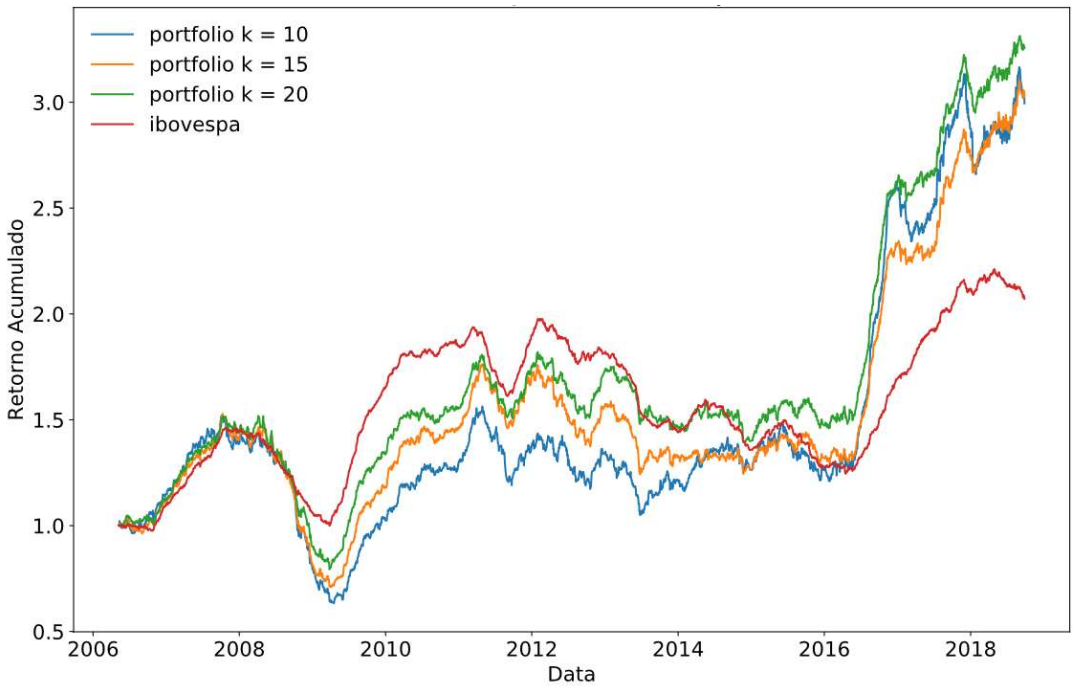
Gráfico 9 – Retorno acumulado por portfólios *long position*: LSTM *deep learning* 3 x 32



Rentabilidade obtida a partir do uso de uma rede neural LSTM com três camadas e 32 neurônios por camada. O retorno acumulado de carteiras teóricas compostas por k = 10, 15 e 20 ações ao longo de 10 anos, é representado pelas curvas azul, laranja e verde, respectivamente. Os retornos de cada portfólio referem-se à simulação de compra (*long position*) de ações. A curva vermelha descreve a trajetória do retorno do Ibovespa nesse período. Fonte: elaborado pelo autor.

De modo semelhante, o Gráfico 10 mostra as curvas de retorno de portfólios selecionados com base nas previsões de uma rede neural multicamadas, desenvolvida com duas camadas e 100 unidades LSTM em cada uma. Este modelo apresentou uma rentabilidade ainda menos vantajosa, embora acima do mercado, em comparação com os modelos prévios.

Gráfico 10 – Retorno acumulado pelos portfólios *long* position via *deep learning* 2 x 100



Rentabilidade obtida a partir do uso de uma rede neural LSTM com duas camadas e 100 neurônios por camada. O retorno acumulado de carteiras teóricas compostas por k = 10, 15 e 20 ações ao longo de 10 anos, é representado pelas curvas azul, laranja e verde, respectivamente. Os retornos de cada portfólio referem-se à simulação de compra (*long position*) de ações. A curva vermelha descreve a trajetória do retorno do Ibovespa nesse período. Fonte: elaborado pelo autor.

Vale a pena ressaltar duas considerações feitas a partir da análise destes dois casos, além dos demais experimentos. Em primeiro lugar, nota-se que na medida em que o número de neurônios cresce, a performance dos modelos, em termos das métricas avaliadas, diminui. Essa questão está em linha com os relatos encontrados em (Fischer & Krauss, 2018; Krauss et al., 2017), que alertam para a necessidade de uma quantidade suficientemente grande de pontos amostrais para a calibração de redes neurais com muitos parâmetros (*e.g.* 2752 parâmetros em um modelos para o S&P500). Apesar do modelo ser provido por uma gama razoável de variáveis, não há uma quantidade grande de ações para o treino de redes demasiadamente grandes, como é o caso de mercados mais dinâmicos como o americano e até mesmo o chinês.

No caso brasileiro, ao longo do tempo o Índice Bovespa foi composto por apenas 60 ações, em média. Isso limita a disponibilidade de dados para o processo de aprendizagem e, consequentemente, pode impactar a capacidade de detecção de padrões robustos para a predição ou até mesmo incorrer em *overfitting*, tal como alertado por Fischer e Krauss.

O segundo ponto de destaque dos modelos *deep learning*, refere-se à capacidade de identificação de fortes tendências do mercado. Analistas e operadores experientes sempre buscam trabalhar a favor de tais tendências, aproveitando as claras oportunidades de negócios. Nota-se, por exemplo, de forma bastante contundente que o modelo mais complexo, com aproximadamente 200 neurônios recorrentes do tipo LSTM, reconheceu duas importantes tendências de alta ao longo do período de testes. A primeira ocorreu a partir de meados de 2009, após a crise americana do *subprime*. Tempos depois, a guinada para o lado positivo da bolsa brasileira, após a transição de governo decorrente do processo de *impeachment* de 2016, foi apreendida e, com previsões corretas, os portfólios puderam comprar as melhores ações.

Evidentemente, experimentos mais robustos e abrangendo uma gama mais diversificada de mercados ao redor do mundo devem ser conduzidos, em trabalhos futuros, a fim de se desenvolver uma teoria mais sólida sobre esse tema. Ainda assim, tais evidências contam a favor do uso das redes neurais, compostas por estruturas do tipo LSTM, para auxiliar a tomada de decisão de compra de ações no mercado acionário brasileiro.

5. Considerações Finais

Os estudos apresentados nessa dissertação contribuem para a crescente lista de publicações acadêmicas que buscam compreender de que forma as técnicas mais modernas de aprendizado de máquina podem ser aplicadas eficientemente no mercado acionário. Além do conhecimento teórico, tais aplicações se justificam, na prática, pela demanda dos agentes econômicos por ferramentas inteligentes de auxílio à tomada de decisão, capazes de gerar lucros consistentes no contexto altamente competitivo, baseado na ampla disponibilidade de informações, tanto históricas quanto em tempo real, e poder de processamento computacional, da atualidade. Nesse ambiente, um investidor, limitado pelas mais diversas condições psicológicas e cognitivas, pode obter um diferencial competitivo quando devidamente auxiliado por uma inteligência artificial.

Com o intuito de experimentar o potencial das RNAs do tipo LSTM no mercado acionário brasileiro, à luz de trabalhos similares desenvolvidos no âmbito internacional, foi proposta uma metodologia para treinar essa classe de modelos preditivos visando estimar a probabilidade de valorizações diárias das ações constituintes do Ibovespa, no período entre 2007 e 2018. Um conjunto especial de variáveis, que já se mostraram eficientes para o processo de treinamento em trabalhos prévios, foi definido para alimentar as RNAs. A partir das estimativas dadas pelas melhores arquiteturas neurais obtidas, simulou-se a seleção de portfólios diários para inferir o potencial econômico do método.

Em sintonia com os estudos previamente analisados, os modelos LSTM apresentaram resultados bastante satisfatórios. Em linhas gerais, as arquiteturas mais simples, com apenas uma camada de poucas dezenas de neurônios do tipo LSTM obtiveram melhor performance. Por outro lado, grandes estruturas LSTM tiveram resultados inferiores às arquiteturas menores, além de apresentarem maior volatilidade nas curvas de retorno dos portfólios gerados a partir de suas previsões.

Além disso, observou-se que o conjunto de variáveis preditoras, inclusive o próprio preço dos ativos, melhoraram o aprendizado e a acurácia dos modelos, sobretudo os índices de mercado, como o índice S&P e o índice futuro do Ibovespa, e a taxa de câmbio. Vale destacar que, em termos econômicos, o retorno médio anual de uma carteira simulada de compras diárias de 10 ações atingiu a marca de 22,3%, sem considerar o desconto de taxas de

corretagem e impostos. Nesse caso, o resultado superou em mais de 100% o retorno médio das ações constituintes do Ibovespa, no mesmo período.

Um dos pontos mais interessantes notados nessa pesquisa é a consistência da curva de rentabilidade proporcionada pelo portfólio obtido com base nas previsões de um modelo relativamente simples. Observou-se que a curva de retorno acumulado da carteira foi sempre maior que a curva obtida pelo Ibovespa. Há, portanto, evidências positivas sobre (i) a consistência temporal da metodologia proposta; (ii) sobre a sua capacidade de extrair fatores preditivos dos dados de entrada; e consequentemente, (iii) sobre o potencial de aplicação das redes neurais LSTM no mercado brasileiro.

Outra característica de destaque diz respeito aos modelos *deep learning* desenvolvidos com as redes LSTM, que se revelaram ótimas ferramentas de apoio no processo decisório de seleção de ações havendo condições claras de tendência na direção do mercado. Tanto em tendência de baixa quanto de alta, as simulações mostraram ótimos desempenhos, em termos econômicos, especialmente em portfólios associados a estratégia de compra de ações.

Algumas limitações importantes nessa pesquisa devem ser expostas, pois necessitam ser abordadas em trabalhos mais específicos futuramente. Uma destas limitações está relacionada ao desconto das taxas de corretagem (pagas para corretoras), de custódia e emolumentos (pagas para a bolsa de valores) e de imposto de renda (pago ao governo) que são cobradas em um ambiente normal de operações financeiras e que afetam a rentabilidade líquida das aplicações. Com isso, os resultados encontrados nesse trabalho seriam minimizados e deve-se considerar, futuramente, uma estrutura de custos integrada ao processo de simulação de carteiras.

Outra limitação é a disponibilidade dos dados de cotações das ações. Nesse caso, trabalhou-se apenas com informações diárias. No entanto, o mercado acionário é muito mais dinâmico e grande parte dos recursos são movimentados em operações que são iniciadas e encerradas no mesmo dia, chamadas operações *intra-day* ou *day-trades*. É justamente para esta linha de operações de alta frequência que os esforços das frentes de desenvolvimento de sistemas inteligentes estão sendo direcionados. Assim, espera-se que este trabalho possa ser aprimorado para o contexto *intra-day*, incluindo regras mais assertivas de negociação e controles de risco mais efetivos.

O mercado acionário brasileiro ainda é pequeno, em termos da quantidade de ações listadas na sua bolsa e volume financeiro negociado, quando comparado com os mercados internacionais mais avançados, como o mercado americano, europeu e asiático. Porém, espera-se uma evolução gradual desse cenário, em função da retomada do crescimento econômico do Brasil, devido às reformas estruturais que entraram na agenda governamental a partir de 2019. Com isso, o mercado de capitais deve ganhar mais escala, como um todo, e o uso de ferramentas inteligentes de apoio à tomada de decisão no âmbito das negociações de ações deve acompanhar esse crescimento.

Acompanhando a tendência internacional de pesquisas no que tange as aplicações de métodos de IA, essa pesquisa é relevante ao evidenciar o potencial do *deep learning* para o mercado acionário brasileiro, assim como por mostrar que variáveis simples e acessíveis a qualquer investidor, proporcionam modelos de boa performance. Essa pesquisa também contribui para disseminar o uso das técnicas de AM para o público do setor financeiro, exibindo a facilidade com que suas técnicas têm sido disponibilizadas para uso, inclusive em plataformas computacionais gratuitas e confiáveis. Logo, cada vez mais investidores, de todas as categorias, poderão utilizá-las em seus negócios.

Por último, acredita-se que a operação de títulos dos mercados futuros e de *commodities* sejam um grande campo de aplicação desta metodologia e que este tema deva ser apropriadamente abordado em pesquisas futuras.

Bibliografia

Alpaydin, E. (2014). *Introduction to Machine Learning*. The MIT Press.

Anaconda. (2018). Anaconda Distribution. Retrieved July 6, 2018, from https://www.anaconda.com/distribution/

Arrow, K. J., & Debreu, G. (1954). Existence of an equilibrium for a copetivite economy. *Econometrica*, *22*(3), 265–290.

Ashwood, A. J. (2013). *Portfolio Selection Using Artificial Intelligence*. Queensland University of Technology.

Bao, W., Yue, J., & Rao, Y. (2017). A deep learning framework for financial time series using stacked autoencoders and long- short term memory. *PLoS ONE*, *12*(7), 1–24.

Bazerman, M. H., & Moore, D. (2014). *Processo Decisório* (8ª edição). Rio de Janeiro: Elsevier.

Ben-David, S., & Shalev-Shwartz, S. (2014). *Understanding Machine Learning: From Theory to Algorithms*. *Cambridge University Press*. New York, NY, USA: Cambridge University Press.

Bengio, Y., Lamblin, P., Popovici, D., & Larochelle, H. (2007). Greedy Layer-Wise Training of Deep Networks. *Advances in Neural Information Processing Systems*.

Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, *5*(2), 157–166.

Bergstra, J., & Bengio, Y. (2012). Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research*, *13*, 281–305.

Brownlee, J. (2019). Stateful and Stateless LSTM for Time Series Forecasting with Python. Retrieved from https://machinelearningmastery.com/stateful-stateless-lstm-time-series-forecasting-python/

Buchanan, L., & O'Connell, A. (2006). A Brief History of Decision Making. *Harvard Business Review, January*, 32–41.

Catmull, E. E. (2014). *Creativity, Inc.: overcoming the unseen forces that stand in the way of true inspiration*. New York, NY, USA: Random House.

Cavalcante, R. C., Brasileiro, R. C., Souza, V. L. F., Nobrega, J. P., & Oliveira, A. L. I. (2016). Computational Intelligence and Financial Markets: A Survey and Future Directions. *Expert Systems with Applications*, *55*, 194–211.

Chollet, F. (2019). Keras Documentation. Retrieved from https://github.com/keras-team/keras

Chong, E., Han, C., & Park, F. C. (2017). Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. *Expert Systems with Applications*, *83*, 187–205.

Deng, Y., Bao, F., Kong, Y., Ren, Z., & Dai, Q. (2017). Deep Direct Reinforcement Learning for Financial Signal Representation and Trading. *IEEE Transactions on Neural Networks and Learning Systems*, *28*(3), 653–664.

Elton, E. J., & Gruber, M. J. (1997). Modern portfolio theory, 1950 to date. *Journal of Banking and Finance*, *21*(11–12), 1743–1759.

Elton, E. J., Gruber, M. J., Brown, S. J., & Goetzmann, W. N. (2012). *Moderna Teoria de Carteiras e Análise de Investimentos*. Rio de Janeiro: Elsevier.

Fabozzi, F. J., Gupta, F., & Markowitz, H. M. (2002). The legacy of modern portfolio theory. *New York*, 7–22.

Fama, E. F. (1970). Efficient Capital Markets : A Review of Theory and Empirical Work. *American Finance Association*, *25*(2).

Fama, E. F. (1995). Random Walks in Stock Market Prices. *Financial Analysts Journal*, *51*(1), 75–80.

Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, *0*, 1–16.

Freitas, F. D. de, Souza, A. F. de, Gomes, F. J. N., & Almeida, A. R. de. (2001). Portfolio Selection with Predicted Returns Using Neural Networks. *IASTED International Conference on Artificial Intelligence and Applications*, 99–103.

Frydman, C., & Camerer, C. (2016a). Neural evidence of regret and its implications for investor behavior. *Review of Financial Studies*, *29*(11), 3108–3139.

Frydman, C., & Camerer, C. F. (2016b). The Psychology and Neuroscience of Financial Decision Making. *Trends in Cognitive Sciences*, *20*(9), 661–675. https://doi.org/10.1016/j.tics.2016.07.003

Gal, Y., & Ghahramani, Z. (2016). A Theoretically Grounded Application of Dropout in Recurrent Neural Networks. *30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.*, 135–139.

Géron, A. (2017). *Hands-on Machine Learning with Scikit-Learn & TensorFlow*.

Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, *9*, 249–256.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*.

Google. (2018). TensorFlow. Retrieved from https://www.tensorflow.org/

Graves, A., Jaitly, N., & Mohamed, A. R. (2013). Hybrid speech recognition with Deep Bidirectional LSTM. *IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU 2013 - Proceedings*, 273–278.

Hastie, T., Tibshirani, R., & Friedman, J. (2008). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (Second Edi). Springer.

Haykin, S. (2001). *Redes neurais: princípios e práticas* (2ª edição). Porto Alegre: Editora Artmed.

Haykin, S. (2008). *Neural Networks and Learning Machines*. *Pearson Prentice Hall* (Third Edit). New Jersey, USA.

Hinton, G. E., Osindero, S., & Teh, Y.-W. (2006). A Fast Learning Algorithm for Deep Belief Nets. *Neural Computation*, *18*(7), 1527–1554.

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780.

Huck, N. (2009). Pairs selection and outranking: An application to the S&P 100 index. *European Journal of Operational Research*, *196*(2), 819–825.

Huck, N. (2010). Pairs trading and outranking: The multi-step-ahead forecasting case. *European Journal of Operational Research*, *207*(3), 1702–1716.

Huck, N. (2019). Large data sets and machine learning: Applications to statistical arbitrage. *European Journal of Operational Research*, *278*(1), 330–342.

J.P.Morgan. (2017). *Innovations in Finance with Machine Learning, Big Data and Artificial Intelligence*.

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data Clustering: A Review. *ACM Computing Surveys*, *31*(3), 264–323.

Jeong, G., & Young, H. (2019). Improving financial trading decisions using deep Q-learning: Predicting the number of shares , action strategies , and transfer learning. *Expert Systems With Applications*, *117*, 125–138.

Johannes, M., Korteweg, A., & Polson, N. (2014). Sequential learning, predictability, and optimal portfolio returns. *Journal of Finance*, *69*(2), 611–644.

Junior, F. H. F. de C., & Famá, R. (2002). As Novas Finanças e a Teoria Comportamental no Contexto da Tomada de Decisão sobre Investimentos. *Caderno de Pesquisas Em Administração*, *9*(2), 25–35.

Kaastra, I., & Boyd, M. (1996). Designing a neural network for forecasting financial and economic time series. *Neurocomputing*, *10*(3), 215–236.

Kable, J. W., & Glimcher, P. W. (2009). The Neurobiology of Decision: Consensus and Controversy. *Neuron*, *63*(6), 733–745.

Kahneman, D. (2013). *Thinking, Fast and Slow*. Farrar, Straus, and GirouxDate.

Kahneman, D., & Tversky, A. (1979). Prospect Theory: An Analysis of Decision under Risk. *Econometrica*, *47*(2), 263–292.

Keeney, R. L., & Raiffa, H. (1993). *Decisions with multiple objectives: preferences and value tradeoffs*. Cambridge University Press.

Keras. (2018). Keras. Retrieved from https://keras.io/

Kingma, D. P., & Ba, J. (2015). Adam: A Method for Stochastic Optimization. *Published as a Conference Paper at the 3rd International Conference for Learning Representations, San Diego*, 1–15.

Kolm, P. N., Tütüncü, R., & Fabozzi, F. J. (2014). 60 Years of portfolio optimization: Practical challenges and current trends. *European Journal of Operational Research*, *234*(2), 356–371.

Kontaki, M., Papadopoulos, A. N., & Manolopoulos, Y. (2008). Continuous Trend-Based Clustering in Data Streams. *Data Warehousing and Knowledge Discovery. Lecture Notes in Computer Science*, *5182*.

Krauss, C., Do, X. A., & Huck, N. (2017). Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500. *European Journal of Operational Research*, *259*(2), 689–702.

Lakatos, E. M., & Marconi, M. de A. (2011). *Fundamentos de metodologia científica*. *Editora Atlas S. A.* https://doi.org/10.1590/S1517-97022003000100005

Leme, R. A. da S. (1983). Por que prever? *Revista de Administração*, *18*(3), 96–97.

Lopes, R. S. (2018). *Aplicação de estratégias de high frequency trading no mercado brasileiro de dólar futuro*. Universidade de São Paulo.

Malkiel, B. G. (2003). The Efficient Market Hypothesis and Its Critics. *Journal OfEconomic Perspectives*, *17*(1), 59–82.

Malkiel, B. G. (2015). *A Random Walk Down Wall Street* (Eleventh E). W. W. Norton & Company.

Markowitz, H. (1952). Portfolio Selection. *The Journal of Finance*, *7*(1), 77–91.

Martins, G. de A., & Theóphilo, C. R. (2009). *Metodologia da investigação científica para ciências sociais aplicadas* (2ª edição). São Paulo - SP: Editora Atlas.

Merton, R. C. (1986). A simple model of capital market equilibrium with incomplete information. *Journal of Finance*, *42*(3), 483–510.

Minsky, M. L., & Papert, S. (1969). *Perceptrons an introduction to computational geometry*. Cambridge, MA: Massachusetts Institute of Technology.

Möws, B. (2016). Deep Learning for Stock Market Prediction: Exploiting Time-Shifted Correlations of Stock Price Gradients.

Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. The MIT Press.

Nelson, D. M. Q., Pereira, A. C. M., & Oliveira, R. A. de. (2017). Stock market's price movement prediction with LSTM neural networks. *Proceedings of the International Joint Conference on Neural Networks*, *2017-May*(Dcc), 1419–1426.

Nielsen, M. A. (2015). *Neural Networks and Deep Learning*. Determination Press.

NVIDIA. (2018a). CUDA. Retrieved from https://developer.nvidia.com/cuda-toolkit-archive

NVIDIA. (2018b). cuDNN. Retrieved from https://developer.nvidia.com/cudnn

Olah, C. (2015). Understanding LSTM Networks. Retrieved July 7, 2018, from http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Oliveira, F. A. de, Nobre, C. N., & Zárate, L. E. (2013). Applying Artificial Neural Networks to prediction of stock price and improvement of the directional prediction index - Case study of PETR4, Petrobras, Brazil. *Expert Systems with Applications*, *40*(18), 7596–7606.

Oliveira, M., Montini, A., & Bergmann, D. (2008). Previsão de retornos de ações dos setores

financeiro, de alimentos, industrial e de serviços, por meio de RNA e modelos ARIMA-GARCH. *Revista de Administração Mackenzie*, *9*(1), 130–156.

Paiva, F. D. (2014). *Redes neurais para decisões no mercado de ações brasileiro*. Universidade Federal de Lavras. Lavras - MG.

Pang, X., Zhou, Y., Wang, P., Lin, W., & Chang, V. (2018). An innovative neural network approach for stock market prediction. *The Journal of Supercomputing*, (February).

Paraná, E. (2016). *A Finança Digitalizada: Capitalismo Financeiro e Revolução Informacional* (1ª edição). Florianópolis: Editora Insular.

Parreiras, L. P. R. de F., & Vicente, R. (2007). Arbitragem Estatística e Inteligência Artificial. São Paulo.

Pinheiro, J. L., Tomaz, C. M., Bueno, N. X., & Silva, W. A. C. (2018). Half century of academic production in Capital Market in Brazil in high impact journals. *Brazilian Business Review*, *15*(3), 1–17.

Racounteur. (2019). *Future of Data*. London.

Ranzato, M., Poultney, C., Chopra, S., & LeCun, Y. (2007). Efficient learning of sparse representations with an energy-based model. *NIPS*.

Real Time Statistics Project. (2019). Internet Live Stats. Retrieved July 4, 2019, from https://www.internetlivestats.com/

Ross, S. A. (1976). The arbitrage theory of capital asset pricing. *Journal of Economic Theory*, *13*(3), 341–360.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, *323*(9), 533–536.

Russell, S., & Norvig, P. (2013). *Inteligência Artificial*. (Elsevier, Ed.) (3rd ed.). Rio de Janeiro.

Sadorsky, P. (2014). Modeling volatility and correlations between emerging market stock prices and the prices of copper, oil and wheat. *Energy Economics*, *43*, 72–81.

Sak, H., Senior, A., & Beaufays, F. (2014). Long Short-Term Memory Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition. *ArXiv*.

Securato, J. R. (2007). *Decisões Financeiras Em Condições de Risco* (2ª edição). São Paulo - SP: Saint Paul.

Sharpe, W. F. (1963). A Simplified Model for Portfolio Analysis. *Management Science*, *9*(2), 277–293.

Sharpe, W. F. (1964). Capital Asset Prices: A Theory of Market Equilibrium Under Conditions of Risk. *The Journal of Finance*, *19*(3), 425–442.

Shefrin, H., & Statman, M. (1985). The Disposition to Sell Winners Too Early and Ride Losers Too Long: Theory and Evidence. *The Journal of Finance*, *40*(3), 777–790.

Silva, J. A., Faria, E. R., Barros, R. C., Hruschka, E. R., Carvalho, A. C. P. L. F. de, & Gama, J. (2013). Data Stream Clustering: A Survey. *ACM Computing Surveys*, *46*(1), 1–31.

Stanovich, K. E., & West, R. F. (2000). Individual differences in reasoning: Implications for the rationality debate? *Behavioral and Brain Sciences*, *23*(5), 645–665.

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to Sequence Learning with Neural Networks. *Advances in Neural Information Processing Systems*, *27*, 1–9.

von Neumann, J., & Morgenstern, O. (1944). *Theory of Games and Economic Behavior*. *Princeton University Press*. Princeton, NJ.

Wikipedia. (2019). k-means clustering. Retrieved July 14, 2019, from https://en.wikipedia.org/wiki/K-means_clustering

Wlodarczak, P., Soar, J., & Ally, M. (2015). Multimedia data mining using deep learning. *5th International Conference on Digital Information Processing and Communications, ICDIPC*, 190–196.

World Federation of Exchanges. (2017). *WFE Annual Statistics Guide*. London.

Xu, K., Ba, J. L., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., … Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. In *32nd International Conference on Machine Learning* (Vol. 37, pp. 169–176). Lille, France.

Yamamoto, M. E., & Valentova, J. V. (2018). *Manual de Psicologia Evolucionista*. *Manual de Psicologia Evolucionista*. Natal: EDUFRN.

Zhang, J., Cui, S., Xu, Y., Li, Q., & Li, T. (2018). A novel data-driven stock price trend prediction system. *Expert Systems with Applications*, *97*, 60–69.

Apêndice

Apêndice A – Ações constituintes do Ibovespa entre fevereiro de 1999 e abril de 2018

Código	Dados?	Código	Dados?	Código	Dados?	Código	Dados?	Código	Dados?	Código	Dados?	Código	Dados?
ABEV3	SIM	BRTP3	SIM	DASA3	SIM	GRSU3	NÃO	PCAR4	SIM	SMLS3	SIM	TRPL4	SIM
ACES4	SIM	BRTP4	SIM	DTEX3	SIM	GRSU6	NÃO	PCAR5	SIM	SUBA3	SIM	TSEP4	SIM
AEDU3	SIM	BTOW3	SIM	DURA4	SIM	HGTX3	SIM	PDGR3	SIM	SUZB5	SIM	TSPC3	SIM
AELP3	SIM	BVMF3	SIM	EBEN4	NÃO	HYPE3	SIM	PETR3	SIM	TAEE11	SIM	TSPC6	NÃO
AGEI3	SIM	CCPR3	SIM	EBTP3	SIM	IGTA3	SIM	PETR4	SIM	TAMM4	SIM	TSPP	NÃO
ALLL11	SIM	CCRO3	SIM	EBTP4	SIM	INEP4	SIM	PLIM4	NÃO	TBLE3	SIM	TSPP3	NÃO
ALLL3	SIM	CESP4	NÃO	ECOD3	SIM	ITAU4	SIM	POMO4	SIM	TCOC4	SIM	TSPP4	NÃO
AMBV4	SIM	CESP5	SIM	ECOR3	SIM	ITSA4	SIM	PRGA3	SIM	TCSL3	SIM	UBBR11	SIM
ARCE3	SIM	CESP6	SIM	EGIE3	SIM	ITUB4	SIM	PRML3	SIM	TCSL4	SIM	UGPA3	SIM
ARCZ6	SIM	CEVA4	NÃO	ELET3	SIM	JBSS3	SIM	PRTX3	SIM	TCSP3	NÃO	UGPA4	SIM
BBAS3	SIM	CGAS5	SIM	ELET6	SIM	KLAB4	NÃO	PSIM4	NÃO	TCSP4	NÃO	UNIP6	SIM
BBAS4	NÃO	CIEL3	SIM	ELPL4	SIM	KLBN11	SIM	PTIP4	SIM	TDBH4	SIM	USIM3	SIM
BBDC3	SIM	CLSC6	SIM	ELPL5	SIM	KLBN4	SIM	QUAL3	SIM	TEPR12	NÃO	USIM5	SIM
BBDC4	SIM	CMET4	SIM	ELPL6	SIM	KROT3	SIM	RADL3	SIM	TEPR4	NÃO	USIM6	NÃO
BBSE3	SIM	CMIG3	SIM	EMAE4	SIM	LAME4	SIM	RAIL3	SIM	TERJ4	NÃO	VAGR3	SIM
BELG3	NÃO	CMIG4	SIM	EMBR3	SIM	LIGH3	NÃO	RCTB30	NÃO	TIMP3	SIM	VALE3	SIM
BELG4	NÃO	COGU4	NÃO	EMBR4	SIM	LIGT3	SIM	RCTB31	NÃO	TLCP4	SIM	VALE5	SIM
BESP11	NÃO	CPFE3	SIM	ENBR3	SIM	LIPR3	SIM	RCTB40	NÃO	TLPP11	NÃO	VCPA3	NÃO
BESP4	SIM	CPLE6	SIM	EPTE4	SIM	LLXL3	SIM	RCTB41	NÃO	TLPP12	NÃO	VCPA4	SIM
BISA3	SIM	CPNE5	NÃO	EQTL3	SIM	LREN3	SIM	RDCD3	SIM	TLPP3	SIM	VIVO4	SIM
BMTO4	SIM	CPSL3	SIM	ERIC4	SIM	MGLU3	SIM	RENT3	SIM	TLPP4	SIM	VIVT4	SIM
BNCA3	SIM	CRGT5	NÃO	ESTC3	SIM	MMXM3	SIM	RLOG3	SIM	TLSP3	NÃO	VVAR11	SIM
BRAP4	SIM	CRTP5	SIM	EVEN3	SIM	MRFG3	SIM	RSID3	SIM	TLSP4	NÃO	WEGE3	SIM
BRDT4	NÃO	CRUZ3	SIM	FIBR3	SIM	MRVE3	SIM	RUMO3	SIM	TMAR5	SIM	WHMT3	SIM
BRFS3	SIM	CSAN3	SIM	FLRY3	SIM	MULT3	SIM	SALM4	SIM	TMAR6	SIM		
BRHA4	NÃO	CSNA3	SIM	GEPA4	SIM	NATU3	SIM	SANB11	SIM	TMCP4	SIM		
BRKM5	SIM	CSTB4	SIM	GETI4	SIM	NETC4	SIM	SAPR11	NÃO	TNEP4	SIM		
BRML3	SIM	CTAX3	NÃO	GFSA3	SIM	OGXP3	SIM	SBSP3	SIM	TNLP3	SIM		
BRPR3	SIM	CTAX4	SIM	GGBR4	SIM	OIBR3	SIM	SDIA4	SIM	TNLP4	SIM		
BRTO3	SIM	CTIP3	SIM	GOAU4	SIM	OIBR4	SIM	SHAP4	NÃO	TPRC6	NÃO		
BRTO4	SIM	CYRE3	SIM	GOLL4	SIM	PALF3	SIM	SMLE3	SIM	TRJC6	NÃO		

Apêndice B – Medidas de volume de dados nos sistemas computacionais

Unidade	Valor	Tamanho
b - bit	0 ou 1	1/8 de byte
B - byte	8 bits	1 byte
KB - kilobyte	1.000 bytes	1.000 bytes
MB - megabyte	1.000^2 bytes	1.000.000 bytes
GB - gigabyte	1.000^3 bytes	1.000.000.000 bytes
TB - terabyte	1.000^4 bytes	1.000.000.000.000 bytes
PB - petabyte	1.000^5 bytes	1.000.000.000.000.000 bytes
EB - exabyte	1.000^6 bytes	1.000.000.000.000.000.000 bytes
ZB - zettabyte	1.000^7 bytes	1.000.000.000.000.000.000.000 bytes
YB - yottabyte	1.000^8 bytes	1.000.000.000.000.000.000.000.000 bytes

Apêndice C – Relação de paradigmas experimentais para redes LSTM com l camadas.

Experimento	Layers	Neurons	Time steps
lstm_simples_01_l	l	5	7
lstm_simples_02_l	l	5	14
lstm_simples_03_l	l	5	21
lstm_simples_04_l	l	5	42
lstm_simples_05_l	l	5	84
lstm_simples_06_l	l	5	168
lstm_simples_07_l	l	10	7
lstm_simples_08_l	l	10	14
lstm_simples_09_l	l	10	21
lstm_simples_10_l	l	10	42
lstm_simples_11_l	l	10	84
lstm_simples_12_l	l	10	168
lstm_simples_13_l	l	25	7
lstm_simples_14_l	l	25	14
lstm_simples_15_l	l	25	21
lstm_simples_16_l	l	25	42
lstm_simples_17_l	l	25	84
lstm_simples_18_l	l	25	168
lstm_simples_19_l	l	50	7
lstm_simples_20_l	l	50	14
lstm_simples_21_l	l	50	21
lstm_simples_22_l	l	50	42
lstm_simples_23_l	l	50	84
lstm_simples_24_l	l	50	168
lstm_simples_25_l	l	100	7
lstm_simples_26_l	l	100	14
lstm_simples_27_l	l	100	21
lstm_simples_28_l	l	100	42
lstm_simples_29_l	l	100	84
lstm_simples_30_l	l	100	168