



Universidade Federal  
do Rio de Janeiro  

---

Escola Politécnica

TÉCNICAS DE APRENDIZADO DE MÁQUINA APLICADAS A  
ALGOTRADING NO MERCADO DE AÇÕES

Lucas Schlee de Brito Fernandes

Projeto de Graduação apresentado ao Curso de Engenharia de Controle e Automação da Escola Politécnica, Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Engenheiro.

Orientador: Heraldo Luís Silveira de Almeida

Rio de Janeiro  
Março de 2019

# TÉCNICAS DE APRENDIZADO DE MÁQUINA APLICADAS A ALGOTRADING NO MERCADO DE AÇÕES

Lucas Schlee de Brito Fernandes

PROJETO DE GRADUAÇÃO SUBMETIDO AO CORPO DOCENTE DO CURSO DE ENGENHARIA DE CONTROLE E AUTOMAÇÃO DA ESCOLA POLITÉCNICA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE ENGENHEIRO DE AUTOMAÇÃO.

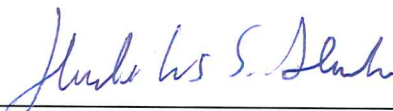
Autor:



---

Lucas Schlee de Brito Fernandes

Orientador:



---

Prof. Heraldo Luís Silveira de Almeida, D.Sc.

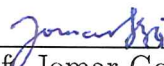
Examinador:



---

Prof. Flávio Luis de Mello, D. Sc.

Examinador:



---

Prof. Jomar Gozzi, D. Sc.

Rio de Janeiro

Março de 2019

Schlee de Brito Fernandes, Lucas

Técnicas de Aprendizado de Máquina aplicadas a AlgoTrading no Mercado de Ações/Lucas Schlee de Brito Fernandes. – Rio de Janeiro: UFRJ/ Escola Politécnica, 2019.

XI, 43 p.: il.; 29, 7cm.

Orientador: Heraldo Luís Silveira de Almeida

Projeto de Graduação – UFRJ/ Escola Politécnica/  
Curso de Engenharia de Controle e Automação, 2019.

Referências Bibliográficas: p. 38 – 40.

1. Aprendizado de Máquina. 2. Mercado Financeiro.  
3. Algotrading. I. Luís Silveira de Almeida, Heraldo. II.  
Universidade Federal do Rio de Janeiro, Escola Politécnica,  
Curso de Engenharia de Controle e Automação. III. Título.

*Aos familiares e amigos.*

# Agradecimentos

Seria injusto começar qualquer agradecimento sem mencionar meus maiores apoiadores, educadores, amigos e companheiros que a vida forçadamente me proporcionou. Agradeço à minha mãe, Magda, por toda paciência e empenho em me tornar uma pessoa melhor a cada dia e ao meu pai, José Oswaldo, por todos os exemplos de empenho e pela generosidade dispensada ao longo da minha jornada. Agradeço também ao meu irmão Antônio que, mesmo com seus 8 anos de idade, é capaz de despertar a cada dia o melhor de mim. Sem vocês a minha vida não seria nada. Aos meus queridos avós, tios e primos, agradeço por vibrarem a cada conquista e por todos os ensinamentos.

Aos meus companheiros de jornada da UFRJ, dispenso os meus mais sinceros agradecimentos. Só nós conhecemos nossos desafios e sabemos o quão difícil foi chegar até o final desse ciclo. Obrigado pela união, companheirismo, generosidade e amizade. Tudo teria sido muito mais complicado sem vocês.

Aos colegas da Chemtech, agradeço por todo apoio, motivação e conselhos ao longo do projeto final. É muito bom poder contar no dia-a-dia com pessoas que eu admiro e me motivam todos os dias a me tornar um melhor profissional.

Ao meu orientador, Heraldo, agradeço por toda a disponibilidade, paciência, dicas e atenção quando foi preciso.

Por fim, agradeço à Carol, por todo apoio e companheirismo. Sei o quanto abrimos mão de programas juntos para a finalização desse trabalho. Muito obrigado por incentivar todos os meus planos e me tornar um homem melhor a cada dia.

Resumo do Projeto de Graduação apresentado à Escola Politécnica/ UFRJ como parte dos requisitos necessários para a obtenção do grau de Engenheiro de Automação.

## TÉCNICAS DE APRENDIZADO DE MÁQUINA APLICADAS A ALGOTRADING NO MERCADO DE AÇÕES

Lucas Schlee de Brito Fernandes

Março/2019

Orientador: Heraldo Luís Silveira de Almeida

Curso: Engenharia de Controle e Automação

A cada dia no mercado de ações é possível identificar milhares de decisões tomadas automaticamente, sem a necessidade de intervenção humana. Apesar disso, essa tendência, considerada irreversível, ainda utiliza estratégias baseadas em regras estritamente definidas. Nesse contexto, o presente trabalho propõe a aplicação de técnicas de aprendizado de máquina para identificar padrões e prever a direção dos preços das ações para negociações de alta frequência.

Abstract of Undergraduate Project presented to POLI/UFRJ as a partial fulfillment of the requirements for the degree of Engineer.

MACHINE LEARNING TECHNIQUES APPLIED TO ALGOTRADING IN  
THE STOCK MARKET

Lucas Schlee de Brito Fernandes

March/2019

Advisor: Heraldo Luís Silveira de Almeida

Course: Automation and Control Engineering

On each day in the stock market it's possible to identify thousands of decisions made automatically, without the need for human intervention. Despite this fact, this irreversible trend still uses strategies based on strictly defined rules. In this context, the present work proposes the application of machine learning techniques to identify patterns and predict the direction of stock prices for high frequency trades.

# Sumário

<b>Lista de Figuras</b>	<b>x</b>
<b>Lista de Tabelas</b>	<b>xi</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Tema . . . . .	1
1.2 Delimitação . . . . .	1
1.3 Justificativa . . . . .	1
1.4 Objetivos . . . . .	2
1.5 Metodologia . . . . .	2
1.6 Descrição . . . . .	3
<b>2 Fundamentos Teóricos</b>	<b>5</b>
2.1 Mercado de Capitais e Bolsa de Valores . . . . .	5
2.2 Algoritmos HFT . . . . .	6
2.3 Trabalhos Relacionados . . . . .	7
<b>3 Pré-processamento e Base de Dados do Modelo</b>	<b>9</b>
3.1 Agrupamento dos dados brutos . . . . .	9
3.2 Atributos do Conjunto de Dados de Entrada . . . . .	10
3.2.1 Indicadores de Análise Técnica . . . . .	11
3.2.2 SMA ( <i>Simple Moving Average</i> ) . . . . .	11
3.2.3 EMA ( <i>Exponential Moving Average</i> ) . . . . .	11
3.2.4 RSI ( <i>Relative Strenght Index</i> ) . . . . .	11
3.2.5 Bandas de Bollinger . . . . .	12
3.2.6 MACD ( <i>Moving Average Convergence Divergence</i> ) . . . . .	12
3.2.7 <i>Aroon Indicator</i> . . . . .	12
3.2.8 ADX Indicator . . . . .	13
3.2.9 CCI ( <i>Commodity Channel Index</i> ) . . . . .	14
3.2.10 CMO ( <i>Chande Momentum Oscilator</i> ) . . . . .	15
3.3 Variáveis do Contexto . . . . .	15
3.3.1 Negociações por intervalo de tempo . . . . .	15



3.3.2	Papéis negociados por intervalo de tempo . . . . .	15
3.4	Manipulações das variáveis do modelo . . . . .	16
<b>4</b>	<b>Estratégias e Geração da Variável Prevista</b>	<b>17</b>
4.1	Janela de Decisão . . . . .	17
4.2	Estratégia de Compra e Venda . . . . .	17
4.3	Geração do sinal de saída . . . . .	19
<b>5</b>	<b>Modelos de Previsão e Técnicas de Validação</b>	<b>21</b>
5.1	Modelos Utilizados . . . . .	21
5.1.1	MLP ( <i>Multi-layer Perceptron</i> ) . . . . .	21
5.1.2	Regressão Logística . . . . .	22
5.2	Técnicas de Validação . . . . .	23
5.2.1	Janelas Deslizantes . . . . .	23
5.2.2	Expansão das Janelas Deslizantes . . . . .	23
5.2.3	Parâmetros de validação . . . . .	24
<b>6</b>	<b>Resultados e Discussões</b>	<b>27</b>
6.1	PETR4 . . . . .	27
6.2	ITUB4 . . . . .	32
<b>7</b>	<b>Conclusões</b>	<b>36</b>
7.1	Conclusões Gerais . . . . .	36
7.2	Trabalhos Futuros . . . . .	37
	<b>Referências Bibliográficas</b>	<b>38</b>
<b>A</b>	<b>Parâmetros de calibração</b>	<b>41</b>
<b>B</b>	<b>Código Base</b>	<b>43</b>

# Lista de Figuras

4.1	Janelas de Tempo e Decisão . . . . .	18
4.2	Exemplo de situação de não-compra . . . . .	19
5.1	MLP com uma camada escondida entre a camada de entrada e a de saída . . . . .	22
5.2	Regressão logística . . . . .	22
5.3	Janelas deslizantes com 4 dias usados para o conjunto de treino e 1 dia para o conjunto de teste . . . . .	24
5.4	Matriz de confusão . . . . .	24
6.1	Mapa de Calor das Correlações entre as Variáveis - PETR4 . . . . .	28
6.2	Performance da Regressão Logística - PETR4 . . . . .	29
6.3	Performance do <i>Multi-layer Perceptron</i> - PETR4 . . . . .	30
6.4	Evolução do lucro dos modelos - PETR4 . . . . .	31
6.5	Mapa de Calor das Correlações entre as Variáveis - ITUB4 . . . . .	33
6.6	Performance da Regressão Logística - ITUB4 . . . . .	34
6.7	Performance do <i>Multi-layer Perceptron</i> - ITUB4 . . . . .	34
6.8	Evolução do lucro dos modelos - ITUB4 . . . . .	35

# Lista de Tabelas

3.1 Layout do arquivo de negociações disponibilizado pela BM&FBOVESPA. . . . .	10
---	----

# Capítulo 1

## Introdução

### 1.1 Tema

O projeto consiste na aplicação de inteligência artificial às negociações de alta frequência no mercado financeiro (*High Frequency Trading*). Nesse contexto, o problema a ser resolvido é o levantamento de padrões de mercado que indiquem uma previsão futura, a curto prazo, de uma variação positiva no preço de um determinado ativo.

### 1.2 Delimitação

Toda a base de dados construída para o presente trabalho foi proveniente dos dados reais negociados na Bolsa de Valores de São Paulo (BM&FBOVESPA), que são de domínio público.

Por conta da granularidade dos dados obtidos pela fonte acima, um posterior trabalho de pré-processamento precisou ser feito para que os dados fossem agrupados em pequenos intervalos de tempo e, dessa forma, pudessem ter maior valor para o modelo.

### 1.3 Justificativa

Algoritmos de negociação vêm ganhando cada vez mais força conforme a globalização se intensifica. Nos Estados Unidos, estima-se que aproximadamente 65% de todo o volume negociado na bolsa de valores seja movimentado por agentes autônomos (robôs) executando algoritmos de HFT (High Frequency Trading). Em 2012, no Brasil, o volume negociado por Acesso Direto ao Mercado, ou seja, plataformas

que conectam o cliente final ao ambiente eletrônico de negociações da bolsa [1], movimentaram R\$104,5 milhões [2].

Apesar do conceito de HFT (*High Frequency Trading*) já estar difundido e vastamente aplicado, grande parte das estratégias desse grupo de algoritmos baseia-se em análises fundamentalistas de mercado, exigindo um conhecimento aprofundado do domínio e um refinamento milimetricamente calibrado na implementação de regras que irão executar a negociação.

Ainda assim, um algoritmo que executa puramente um conjunto de regras não é totalmente seguro para realizar negociações. Dessa forma, a aplicação de técnicas de aprendizado de máquina podem auxiliar no poder preditivo desses algoritmos, identificando padrões passados e gerando sinais para tomadas de decisão mais seguras.

## 1.4 Objetivos

O objetivo do trabalho é desenvolver e modelar estratégias direcionais de negociação de alta frequência com a utilização de técnicas de aprendizado de máquina para indicar se um determinado ativo apresentará alta no período imediatamente após o intervalo em que foi analisado. Sendo assim, se o modelo indicar que haverá alta, um futuro agente autônomo a consumir o modelo receberá um sinal positivo, podendo dessa forma enviar ordens de compra e venda ao mercado.

## 1.5 Metodologia

Com o intuito de otimizar o pré-processamento dos dados brutos da bolsa de valores, inicialmente agrupados em *ticks*, foi implementada uma solução em *C#* utilizando o *framework* multiplataforma *.NET Core*, podendo ser executado em todos os sistemas operacionais. Por ser uma linguagem compilada e performática, obteve-se um bom desempenho na descompactação dos arquivos comprimidos contendo as negociações, no processamento dos dados e na exclusão posterior do arquivo descompactado para economizar memória em disco.

A principal função dessa etapa de pré-processamento dos dados brutos é o agrupamento em janelas de tempo para que indicadores de análise técnica, geralmente aplicados em intervalos diários, pudessem ser aplicados em intervalos de minutos. Dessa forma, o intervalo pode conter informações históricas a partir desses indicado-

res, gerando, por exemplo, valores de médias móveis ou indicadores que dependam de intervalos anteriores.

A saída do modelo,  $\mathbf{y}$ , pode assumir valores de 1 e 0, indicando se deve ou não ser enviada uma ordem de compra ou, mais precisamente, se haverá alta ou baixa no preço do ativo no próximo minuto, respectivamente.

Após a etapa de agrupamento dos dados e geração de indicadores, os dados são armazenados em um arquivo *CSV (Comma-Separated Values)* para futuro consumo.

Na etapa de consumo dos dados gerados, foi utilizada a linguagem de programação *Python* e algumas bibliotecas orientadas para a ciência de dados, como o *scikit-learn*, *Pandas*, *Numpy*, *Seaborn* e *Matplotlib*.

Com o auxílio das diversas opções de normalização dos dados oferecidas pelo *scikit-learn*, foram feitas normalizações logarítmicas e de máximo e mínimo para as *features* o conjunto de dados. Dessa forma, todas as variáveis de entrada do modelo possuíam variações entre 0 e 1.

Através do *scikit-learn*, também foram utilizados modelos de aprendizado de máquina supervisionado para a previsão dos dados de teste. Dessa maneira, o MLP (*MultiLayer Perceptron*) e a Regressão Logística foram escolhidos para obter-se uma base comparativa entre modelos de redes neurais e regressão linear.

Por fim, para a validação dos modelos obtidos, foi utilizada uma variação da validação cruzada para séries temporais, de forma que, para um determinado intervalo de tempo, os dados de testes fossem sempre futuros em relação aos dados de treino.

## 1.6 Descrição

No capítulo 2 são levantados alguns fundamentos teóricos sobre o mercado de capitais e bolsa de valores, assim como são negociados seus ativos. Nele, também será feita uma contextualização das negociações de alta frequência e o que são algoritmos HFT (*High Frequency Trading*), assim como os trabalhos presentes na literatura sobre HFT e aprendizado de máquina.

O capítulo 3 introduz o procedimento de aquisição das informações provenientes dos dados brutos da BM&FBOVESPA, assim como as técnicas utilizadas para pré-processá-los. Nele, também serão discutidos os diversos indicadores de análise

técnicas aproveitados como *features* no modelo e como foi realizada a normalização desses indicadores.

O capítulo 4 descreve a estratégia utilizada para a geração da variável de saída do modelo e os diferentes cenários em que um sinal de direção dos preços pode ser ou não disparado.

No capítulo 5, são discutidos os modelos utilizados para a previsão dos dados de teste e as técnicas de validação utilizadas.

O capítulo 6 mostra os resultados e discussões obtidos para uma determinada ação do mercado brasileiro, a PETR4, que representa as ações preferências da Petrobrás.

Por fim, no capítulo 7, são levantadas as conclusões do projeto, indicando as limitações encontradas e sugestões para evoluções do trabalho realizado.

# Capítulo 2

## Fundamentos Teóricos

Neste capítulo, são introduzidos alguns conceitos base para o entendimento e desenvolvimento do projeto. Nas próximas seções, são feitas contextualizações sobre o Mercado de Capitais, Bolsa de Valores e como os algoritmos de HFT se inserem nesse contexto.

### 2.1 Mercado de Capitais e Bolsa de Valores

O Mercado Capitais é um sistema de distribuição de valores imobiliários que permite potencializar o fluxo financeiro entre vários agentes econômicos, aumentando a liquidez dos vários títulos existentes. Esse mercado, formado por corretoras, bancos, bolsa de valores e outras insituições financeiras, facilita o investimento em grandes empreendimentos e permite aos diferenes agentes econômicos a compra de parte da empresa por meio de ações, que são as menores frações do capital social de uma empresa. As ações, por sua vez, são uma forma veloz de atração de investimento para que a empresa consiga reinvestir no seu próprio crescimento [3].

As operações que ocorrem no Mercado de Capitais possuem uma série de regras e procedimentos que garantem uma proteção para ambos os lados interessados.

A bolsa de valores, principal instrumento do Mercado de Capitais, é um mercado organizado onde são negociadas ações de empresas com o capital aberto. Para tornar-se de capital aberto, a empresa precisa passar por uma Oferta Pública Inicial (IPO - *Initial Public Offering*), em que suas ações são vendidas para o público geral numa bolsa de valores pela primeira vez [4]. Ao abrir seu capital, a empresa disponibiliza seus números para que qualquer pessoa possa acompanhar seus balanços e sua evolução financeira ao longo do tempo.



Os acionistas da empresa podem obter lucro a partir de dividendos ou a partir da venda de suas ações, uma vez que estas estejam mais valorizadas do que quando o acionista comprou. A possibilidade de lucrar rapidamente com os movimentos do mercado atrai os especuladores, que apostam ou não na alta ou baixa de um determinado ativo baseado em diversas variáveis. Esses especuladores são importantes para o mercado pois aumentam a liquidez dos papéis negociados e determinam precisamente o preço de um determinado ativo.

Diante da volatilidade do mercado e com a intrínseca característica do lucro a curto prazo, estratégias de negociação de alta frequência começaram a surgir. Com o intuito de automatizar as negociações e, com isso, operar numa velocidade maior do que a humana, algoritmos HFT (*High Frequency Trading*) se popularizaram e passaram a ocupar um espaço crescente nas diversas bolsas de valores.

## 2.2 Algoritmos HFT

Algoritmos de negociação vêm sendo cada vez mais utilizados nas últimas décadas. Segundo ALAIN CHABOUD [5], esses algoritmos classificam-se como uma interface direta entre computadores e plataformas de negociação, sendo capaz de posicionar ordens de compra e venda sem a necessidade de intervenção humana.

O termo HFT (*High Frequency Trading*), por sua vez, é mais recente e trata-se de um subgrupo dos algoritmos de negociação mencionados anteriormente. Esse grupo de algoritmos tem como característica a execução de operações de negociação a nível de milissegundo.

O grupo de algoritmos de negociação e, portanto, as HFT, podem ser vistas como ferramentas para os *traders* ou especuladores que observam parâmetros de mercado em tempo real para produzirem decisões de negociação, compondo a técnica popularmente conhecida como *Algotrading*.

Algoritmos HFT tem como característica uma rápida atualização das ordens enviadas e não permanecem com a posição após o fechamento do pregão. As estratégias visam um lucro pequeno em um montante muito grande negociações, portanto é mais interessante performar em mercados com ativos de alta liquidez. Depois do algoritmo realizar uma ordem, seja esta de compra ou venda, o tempo em que a posição é mantida é curto [6].

Grande parte desses algoritmos se utiliza de indicadores de análise técnica ou puramente derivadas da curva recente do preço para realizar suas operações, aliados a um conjunto de regras e estratégias pré-estabelecidas. Dessa maneira, torna-se interessante a aplicação de aprendizado de máquina na previsão de preços ou direcionamentos futuros para que a tomada de decisão torne-se mais segura e eficiente a longo prazo.

## 2.3 Trabalhos Relacionados

A aplicação de aprendizado de máquina ao mercado financeiro não é recente. Desde 1988, época em que não se podia contar com um poder de processamento elevado para executar previsões, já tinha sido levantada a hipótese de um modelo que considerasse inteligência artificial na determinação dos preços futuros. O estudo foi conduzido por WHITE [7] e tinha como objetivo analisar o retorno diário das ações comuns da IBM ao longo de 4 anos com apoio de algoritmos utilizando redes neurais. Desde então, apoiados pelo avanço na tecnologia e pelas ferramentas de negociações cada vez mais modernas, os investimentos e estudos nesse tema foram aumentando.

Em 2012, com o assunto mais amadurecido, começaram a surgir trabalhos como o de ADEBIYI AYODELE A. e O. [8]. Em seus estudos, utilizou dados públicos disponíveis na Internet para modelar uma estratégia de previsão de preços futuros para alguns ativos do mercado. Baseado em intervalos diários, combinou indicadores de análise técnica e variáveis criadas para formar um modelo híbrido que aplicava redes neurais para a previsão do preço futuro.

Ainda em uma base diária de previsão, destaca-se o trabalho de AUTHORH-ORAN XIE [9]. Ao estruturar sua estratégia, utilizou uma mistura entre notícias de jornais de investimento e modelos de previsão baseadas do livro de ordens com a aplicação de algoritmos de *Support Vector Machines (SVMs)* para gerar um sinal de negociação.

Além dos estudos mencionados acima, outros destacam-se no tema de aprendizado de máquina aplicado a estratégias de longo prazo para uma base diária. Apesar disso, poucos estudos se aventuraram na aplicação de inteligência artificial às negociações de alta frequência, também chamadas de HFT (*High Frequency Trading*) na literatura.

No escopo do curto prazo, estudos como o de DIXON [10] começaram a surgir. Em sua abordagem, ele defende que a volatilidade e liquidez do preço estão intrinsicamente ligadas às ordens de compra e venda de um determinado ativo. Dessa maneira, utiliza-se da profundidade do livro de ofertas para prever a direção dos preços e calcula uma possível alta ou baixa de acordo com o número de compradores ou vendedores interessados, representados pela quantidade e volume das ordens de compra e venda.

Em um aproveitamento maior dos indicadores de análise técnica extensivamente usados pelo mercado financeiro, PUTRA [11] consegue aplicá-los no contexto de HFT ao agrupar os dados da bolsa de valores da Indonésia em janelas de tempo de 15 minutos. Em seu modelo, cada janela carrega consigo preços de abertura e fechamento, além de médias móveis e indicadores de momento e tendência. Sua estratégia está ligada ao retorno futuro obtido, primeiramente sobre a diferença entre preço de abertura e fechamento e posteriormente sobre a diferença entre máximo e mínimo de um intervalo específico, o qual tenta prever com a aplicação de redes neurais artificiais.

No Brasil, estudo semelhante foi aplicado por SILVA [12] em sua tese de mestrado. Utilizando-se também de indicadores de análise técnica aplicados a janelas de tempo, teve como estratégia a geração de um sinal binário indicando se no intervalo seguinte uma determinada ação apresentará uma alta ou baixa dos preços. Seu sinal de direção positiva dos preços é disparado se o máximo preço do intervalo seguinte superar o preço fechamento do intervalo corrente em um determinado limiar. Utiliza-se do modelo *Multi-Layer Perceptron* para o treinamento do modelo e da validação em janelas deslizantes de dias para simular previsões em alguns ativos da BM&FBOVESPA.

O presente trabalho, apesar de utilizar os conceitos de agrupamento de *ticks* em janelas de tempo e validação por janelas deslizantes de dias, diferencia-se por definir um intervalo menor de tempo (3 minutos) para seu agrupamento para identificar mais oportunidades, pela geração de variáveis referentes à volatilidade do mercado e pela elaboração de uma estratégia focada na prevenção de possíveis prejuízos. No decorrer dos capítulos, são apresentadas com mais detalhe as diferenças de estratégia que apresentam maior proteção no cenário de negociações de alta frequência, assim como o raciocínio de como os falsos positivos das estratégias anteriores podem causar perdas maiores do que os possíveis ganhos.

# Capítulo 3

## Pré-processamento e Base de Dados do Modelo

Nesse capítulo são introduzidas as abordagens utilizadas para o pré-processamento e transformação dos dados brutos da BM&FBOVESPA, assim como as estratégias utilizadas para gerar o sinal direcional do mercado.

### 3.1 Agrupamento dos dados brutos

Como fonte de dados para o trabalho, foi utilizada a base de dados brutos oferecida gratuitamente pela BM&BOVESPA. Ao final de cada dia, um arquivo comprimido representando todas as negociações, ordens de compra e ordens de venda do dia anterior é disponibilizado, sendo o arquivo de negociação o de interesse para a geração dos indicadores de mercado. Informações sobre esse arquivo são mostradas conforme a Tabela 3.1.

Cada linha desse arquivo, portanto, representa um *tick*, ou seja, o dado mais granular possível indicando uma operação de negociação, sem informações históricas associadas.

Seguindo a abordagem sugerida por PUTRA [11], os *ticks* foram agrupados em um intervalo determinado de minutos, chamado de janela de tempo. Dessa forma, o dado granular a ser consumido pelo futuro modelo agora possui informações mais densas e pode ser comparado com janelas anteriores para criar indicadores extensivamente usados pelo mercado financeiro, oferecendo um maior poder de previsão. Com os preços de abertura e fechamento de um determinado ativo para cada janela, é possível recriar médias móveis e outros indicadores que necessitem de informações históricas dos preços.

Tabela 3.1: Layout do arquivo de negociações disponibilizado pela BM&FBOVESPA.

Coluna	Descrição
Data da Sessão	Data em que ocorreu a sessão
Símbolo do Instrumento	Ativo que foi negociado
Número de Negócio	Número sequencial do negócio
Preço do negócio	Preço negociado
Quantidade	Quantidade de ações negociada
Hora	Hora negociada com precisão de nanossegundos
Ind. Anulação	Indicador de Anulação: 1 - ativo / 2 - cancelado
Data Oferta Compra	Data da oferta de compra
Seq.Oferta Compra	Número sequencial da oferta de compra
GenerationID - Of.Compra	Número de geração (GenerationID) da Oferta de compra
Condição Oferta de Compra	0 - Oferta Neutra; 1 - Oferta Agressora; 2 - Oferta Agredida
Data Oferta Venda	Data da oferta de venda
Seq.Oferta Venda	Número sequencial da oferta de venda
GenerationID - Of.Venda	Número de geração (GenerationID) da Oferta de venda
Condição Oferta de Venda	0 - Oferta Neutra; 1 - Oferta Agressora; 2 - Oferta Agredida
Indicador de Direto	1 - Intencional / 0 - Não Intencional
Corretora de Compra	Identificador da corretora de Compra
Corretora de Venda	Identificador da corretora de venda

Com o intuito de realizar o agrupamento e pré-processamento dos dados brutos, foi desenvolvido um *software* especificamente para o projeto. Para tal, foi escolhida a linguagem de programação C#, compilada e orientada a objetos, rodando sobre o *framework* multiplataforma .NET Core. Dessa forma, alcançou-se uma performance satisfatória na geração dos dados agrupados.

O *software*, compatível com qualquer sistema operacional em uma máquina convencional, recebe o caminho de um diretório contendo todos os arquivos de negociação da BM&FBOVESPA comprimidos. Ao ser executado, descomprime cada arquivo, filtra pelo ativo determinado no seu dicionário de parâmetros, descritos no apêndice A, e agrupa os *ticks* para formar janelas de tempo com minutos de intervalo também pré-determinados no dicionário. Tais janelas de tempo possuem, por sua vez, seus preços de abertura e fechamento, valores máximos e mínimos, total de negociações e quantidade total de papéis em cada negociação. Dessa maneira, é possível calcular indicadores técnicos do mercado financeiro para uma base de intervalos em minutos.

## 3.2 Atributos do Conjunto de Dados de Entrada

Nessa seção são discutido os atributos utilizados para montar o conjunto de dados de entrada do modelo.

### 3.2.1 Indicadores de Análise Técnica

Vários dos indicadores comumente usados para análise técnica no mercado financeiro foram reaproveitados e usados nas janelas de tempo resultantes do agrupamento dos dados. Utilizaram-se boa parte dos indicadores sugeridos em SILVA [12] que empiricamente obtiveram a melhor correlação com a saída do modelo do trabalho corrente.

### 3.2.2 SMA (*Simple Moving Average*)

Médias móveis simples tem como objetivo identificar tendência de preço e um potencial de movimentação de acordo com uma tendência já estabelecida. O jeito mais simples de utilizar o identificador é observando se o preço atual encontra-se abaixo ou acima da curva de tendência [13]. Pode ser calculada pela fórmula abaixo:

$$SMA = \frac{A_1 + A_2 + \dots + A_N}{N} \quad (3.1)$$

### 3.2.3 EMA (*Exponential Moving Average*)

Assim como as médias móveis simples, médias móveis exponenciais tem como objetivo identificar tendência nos preços, porém insere um peso maior aos preços mais recentes. Pode ser calculada pela seguinte fórmula:

$$EMA_x = EMA_{x-1} + K \cdot CP_x - SMA_{x-1} \quad (3.2)$$

Onde:

$$K = \frac{2}{N + 1} \quad (3.3)$$

E:

$$EMA_0 = SMA_0 \quad (3.4)$$

E CP equivale ao preço de fechamento do período, SMA é a média móvel simples e N é o número de períodos calculados.

### 3.2.4 RSI (*Relative Strenght Index*)

O índice de força relativa é um oscilador de momento que tem como objetivo mensurar velocidade e mudança de movimento nos preços [14]. Pode ser calculado pela seguinte fórmula:

$$RSI = 100 - \frac{100}{1 + RS} \quad (3.5)$$

Onde:

$$RS = \frac{AG}{AL} \quad (3.6)$$

Sendo AG e AL a média dos ganhos e das perdas para os últimos N períodos de interesse, respectivamente.

### 3.2.5 Bandas de Bollinger

As Bandas de Bollinger são indicadores úteis na medição da volatilidade dos preços, pois combinam a média móvel simples com o desvio padrão, gerando 2 curvas com a seguinte fórmula:

$$UpperBand = SMA + 2 \cdot SD \quad (3.7)$$

E

$$LowerBand = SMA - 2 \cdot SD \quad (3.8)$$

Onde SD é o desvio padrão, dado pelo fórmula:

$$SD = \sqrt{\frac{\sum_{i=1}^N (x_i - M_A)^2}{N}} \quad (3.9)$$

E  $M_A$  é a média móvel dos N períodos de interesse.

O estreitamento das bandas indica que a volatilidade dos últimos períodos foi baixa, enquanto uma banda larga indica que houve maior volatilidade [15].

### 3.2.6 MACD (*Moving Average Convergence Divergence*)

O indicador MACD é um indicador de momento e seguidor de tendências, caracterizando-se por relacionar 2 médias móveis de períodos diferentes. O MACD pode ser calculado subtraindo-se uma média móvel exponencial de longo prazo de uma média móvel exponencial de curto prazo, usualmente de 26 e 12 períodos [16]. Pode ser calculado segundo a fórmula:

$$MACD = EMA[12] - EMA[26] \quad (3.10)$$

### 3.2.7 *Aroon Indicator*

O indicador *Aroon* é usado para mensurar variações de tendência em um preço de um determinado ativo, assim como medir a força dessa tendência. Em síntese,

esse indicador verifica o tempo de diferença para a última alta e para a última baixa sobre um número determinado de períodos. Esse indicador é composto por 2 curvas, uma medindo a tendência de alta e outra medindo a tendência de baixa. As curvas podem ser calculadas pela seguinte equação:

$$Arron_{Up} = \frac{N - P_{MAX}}{N} \cdot 100 \quad (3.11)$$

$$Arron_{Down} = \frac{N - P_{MIN}}{N} \cdot 100 \quad (3.12)$$

Combinando as 2 curvas:

$$Aroon_{indicator} = Aroon_{Up} - Aroon_{Down} \quad (3.13)$$

Sendo  $P_{MAX}$  e  $P_{MIN}$  o número de períodos desde a última máxima e última mínima no intervalo de N períodos especificado, respectivamente.

### 3.2.8 ADX Indicator

O indicador ADX trata-se de um indicador que mede a força de uma tendência e, quando combinado com indicadores direcionais, ditam também sua direção. É um dos indicadores mais complexos de serem calculados, pois sua fórmula por si só já define outros indicadores de dependência para o seu cálculo [17]. Pode ser calculado pela seguinte fórmula:

$$ADX = 100 \cdot EMA_{\delta} \quad (3.14)$$

Onde  $\delta$ :

$$\delta = \frac{|DI_{+} - DI_{-}|}{DI_{+} + DI_{-}} \quad (3.15)$$

Sendo DI (*Directional Indicators*):

$$DI_{+} = 100 \cdot \frac{EMA_{DM+}}{ATR} \quad (3.16)$$

$$DI_{-} = 100 \cdot \frac{EMA_{DM-}}{ATR} \quad (3.17)$$

Dado ATR (*Average True Range*) [18]:



$$ATR = \frac{1}{n} \sum_{i=1}^n TR_i \quad (3.18)$$

Sendo TR (*True Range*):

$$TR = \max[(Alta - Baixa), |Alta - Fechamento_{n-1}|, |Baixa - Fechamento_{n-1}|] \quad (3.19)$$

Onde DM:

$$DM_+ = \begin{cases} Move_{Up}, & \text{if } Move_{Up} \geq Move_{Down} \\ 0, & \text{caso contrário} \end{cases} \quad (3.20)$$

$$DM_- = \begin{cases} Move_{Down}, & \text{if } Move_{Down} \geq Move_{Up} \\ 0, & \text{caso contrário} \end{cases} \quad (3.21)$$

E, por fim:

$$Move_{Up} = Alta_N - Alta_{N-1} \quad (3.22)$$

$$Move_{Down} = Baixa_N - Baixa_{N-1} \quad (3.23)$$

### 3.2.9 CCI (*Commodity Channel Index*)

Esse indicador foi desenvolvido com o objetivo de apontar novas tendências e condições extremas, baseado em um comportamento cíclico dos preços. Em suma, o CCI mede a relação do preço atual de um determinado ativo com a média dos preços de um período especificado.

Quando os preços encontram-se acima da média, o CCI é alto indicando que o ativo está sendo mais comprado do que deveria, ou seja, os preços tenderão a baixar nos próximos períodos. Quando o CCI é baixo, da mesma maneira, pode-se inferir que o ativo está sendo mais vendido do que deveria e, portanto, os preços subirão [19]. O indicador pode ser calculado da seguinte forma:

$$CCI = \frac{TP - SMA(TP)}{0.015 \cdot SD(TP)} \quad (3.24)$$

Onde:

$$TP = \frac{P_A + P_B + P_F}{3} \quad (3.25)$$

Sendo  $P_A$ ,  $P_B$  e  $P_F$  os preços de alta, baixa e fechamento do intervalo avaliado, respectivamente. A média móvel simples (SMA) e o desvio padrão (SD) podem ser consultados pelas fórmulas 3.1 e 3.9 expostas anteriormente.

### 3.2.10 CMO (*Chande Momentum Oscillator*)

O indicador CMO também é um indicador de momento, ou seja, aponta se um determinado ativo está sendo mais comprado ou mais vendido do que realmente deveria, assim como o indicador RSI (3.2.4). O CMO não suaviza os resultados, disparando as mais frequentes penetrações de sobrecompra e sobrevenda [20]. Pode ser calculado por:

$$CMO = 100 \cdot \frac{Up - Down}{Up + Down} \quad (3.26)$$

Onde  $Up$  refere-se ao preço de todos os intervalos de alta e  $Down$  aos intervalos de baixa dentro do período especificado.

## 3.3 Variáveis do Contexto

Com o objetivo de avaliar o volume de negociações ocorridas dentro de um intervalo, foram criados mais 2 indicadores a partir dos dados brutos para futuro consumo do modelo, previamente descritos conforme a tabela 3.1. As variáveis são mostradas abaixo:

### 3.3.1 Negociações por intervalo de tempo

Esse indicador tem como objetivo mensurar a velocidade de ações negociadas no intervalo de tempo especificado. Seu valor é puramente definido pelo número de negociações ocorridas num determinado intervalo.

### 3.3.2 Papéis negociados por intervalo de tempo

Através dessa variável é possível verificar o volume de papéis sendo negociados em um intervalo de tempo. Pode ser calculado pela seguinte forma:

$$V_{k,k+n} = \sum_{t=k}^{k+n} Q_t \quad (3.27)$$

Sendo  $V$  o volume de papéis negociados no intervalo contido entre o tempo  $k$  e  $k + n$  e  $Q$  a quantidade de papéis presentes em cada negociação do intervalo.

### 3.4 Manipulações das variáveis do modelo

Com o intuito de otimizar a assimilação das variáveis escolhidas pelo modelo, foram utilizadas algumas técnicas de normalização e dimensionamento.

De todas as variáveis que representam o valor de preço (SMA, EMA e Bandas de Bollinger), subtraiu-se o preço de fechamento do intervalo correspondente. Dessa maneira, foi possível centralizar a variação dos preços para um futuro redimensionamento dos valores, guardando a precisão de cada movimento.

Dada a grande gama de valores das variáveis que representam o volume das negociações e papéis negociados, foi necessária a realização de uma transformação logarítmica para reduzir a escala dos dados. Para isso, utilizou-se o pacote científico *Numpy* junto com a linguagem de programação *Python*, extensivamente usada após o inicial agrupamento dos dados. A função *numpy.log* foi a escolhida para tal transformação.

Após a normalização inicial dos dados, padronizou-se a escala de todas as variáveis do modelo para o intervalo entre 0 e 1. Para aplicar esse dimensionamento, foi instanciada a classe *MinMaxScaler*, oferecida pela biblioteca de pré-processamento do pacote *scikit-learn* para realizar a transformação dos dados, seguindo a fórmula:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (3.28)$$

No capítulo seguinte será discutida a estratégia utilizada um sinal de compra, variável a ser prevista pelo modelo.

## Capítulo 4

# Estratégias e Geração da Variável Prevista

Nesse capítulo são introduzidas as abordagens e estratégias utilizadas para gerar a variável de previsão do modelo.

### 4.1 Janela de Decisão

Com o objetivo de se obter lucro, o modelo deve ser capaz de prever movimentações de preço para que ordens de compra e venda possam ser enviadas no momento certo. Dada essa característica intrínseca do modelo, a variável de saída deve ser gerada a partir de um determinado intervalo futuro, de modo que exista alguma sequência de compras e vendas que resultem em uma diferença positiva de preço.

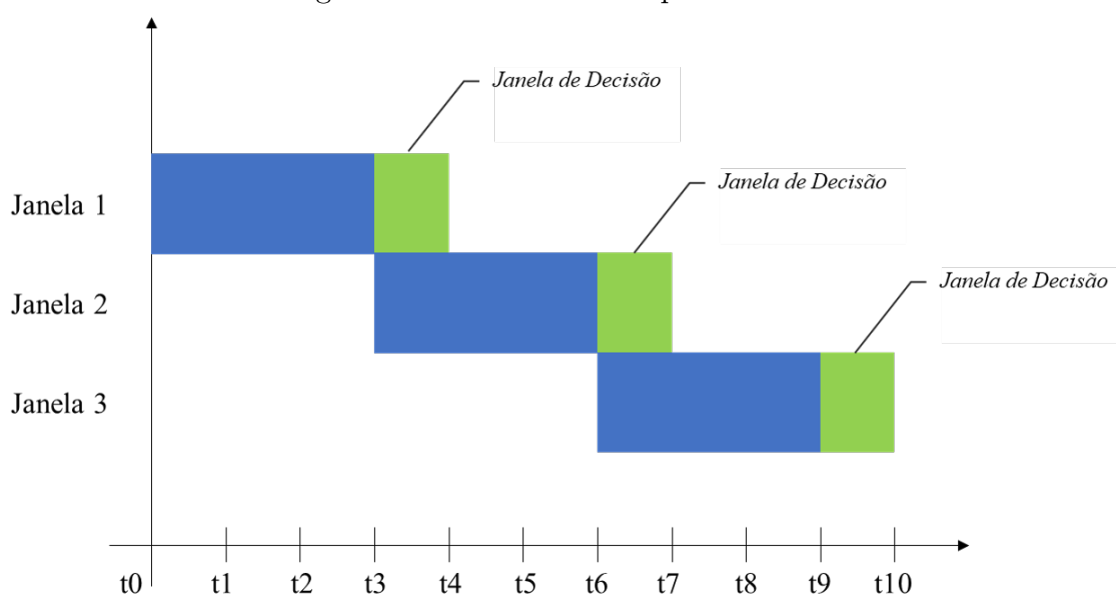
Considerando que as janelas de tempo em que os dados brutos foram agrupados possuem um intervalo de tempo  $t$ , o interesse está, portanto, no intervalo de tempo contido entre  $t$  e  $t + t'$ , chamado de janela de decisão, sendo  $t'$  preferencialmente menor do que  $t$  para que se obtenha um maior poder de previsão a curto prazo.

A imagem 4.1 ilustra o cenário descrito. Cada intervalo de tempo  $t$  utiliza-se do intervalo verde  $t'$  imediatamente posterior, que se inicia no mesmo momento da segunda janela usada para gerar o segundo intervalo.

### 4.2 Estratégia de Compra e Venda

De modo a facilitar a tomada de decisão e diminuir as premissas tomadas pelo agente autônomo, optou-se por uma estratégia que desconsidera a quantidade de ações possuídas. A função do robô reduz-se, portanto, a detectar movimentações de

Figura 4.1: Janelas de Tempo e Decisão



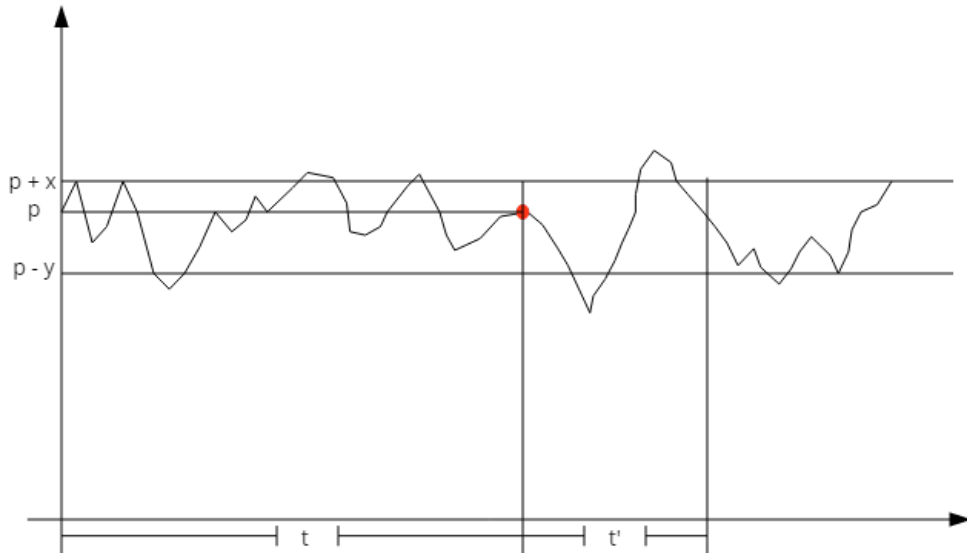
alta dos preços para que se possa obter lucro comprando e vendendo, necessariamente nessa ordem, sem guardar excedentes de ações ou papéis de uma janela de tempo para outra.

Uma das possíveis estratégias para a geração de um indicador direcional de alta dos preços é assumir que se a diferença entre o máximo preço negociado na janela de previsão e o preço de fechamento da janela de tempo superar um limite  $x$ , o agente autônomo deve enviar uma ordem de compra no período  $t$  e uma ordem de venda no período  $t+t'$ , assim como proposto em SILVA [12]. O principal problema dessa abordagem é que, para um resultado positivo a longo prazo, o modelo precisa garantir uma precisão apurada para que os ganhos superem as possíveis perdas, classificadas como falsos positivos pelo modelo. Esses falsos positivos, ou seja, baixas de preço erroneamente classificadas como altas, podem gerar um prejuízo desproporcional se ordens de compra e venda forem enviadas a mercado no início e no fim de cada janela de previsão para um cenário de queda abrupta dos preços.

Com o objetivo de contornar os possíveis cenários de perda descritos acima, foi necessária a elaboração de uma estratégia que considerasse outros fatores na variação de preço dentro da janela de previsão. Dessa maneira, para gerar o indicador de compra, é verificado se a primeira variação positiva de  $x$  entre preço negociado e preço de fechamento ocorreu antes da primeira variação negativa  $y$ . Se ocorrer essa situação, será feita a sinalização de compra para o modelo. No cenário de um falso positivo, nessa abordagem, estabelece-se que uma ordem de venda a mercado será

enviada se a diferença entre o preço de fechamento e a última negociação superar o valor  $y$ . Se, por acaso, o preço não ultrapassar o limite entre  $x$  e  $-y$ , a ordem de venda a mercado será enviada ao final da janela de previsão.

Figura 4.2: Exemplo de situação de não-compra



Para ilustrar uma situação de não-compra, observa-se a imagem 4.2, que representa a variação dos preços ao longo do tempo. Mesmo o preço tendo ultrapassado o limite superior  $p + x$  ao fim do intervalo  $t'$ , a saída indicando o sinal de compra não é gerada porque o limite inferior  $p - y$  foi ultrapassado anteriormente nesse mesmo intervalo.

Destaca-se, nessa abordagem, que o maior prejuízo que pode ser obtido por um falso positivo é de  $y$ . É válido mencionar também que nem todo falso positivo, nessa perspectiva, causará prejuízo. Na situação em que o preço de fechamento supera o valor de abertura em  $z$ , sendo  $z < x$ , é possível obter lucro sem que o sinal direcionador de compra seja disparado.

### 4.3 Geração do sinal de saída

Dada a estratégia definida na seção anterior, conclui-se que trata-se de um problema de classificação: se o preço tiver variação positiva maior do que  $x$ , tem-se a saída do modelo com o valor 1, o que significa que ordens de compra e venda serão enviadas conforme definido na seção anterior. Se a variação de preço for positiva mas não ultrapassar o limite superior  $x$  ou se a variação for negativa, tem-se a saída com o valor 0.

Dessa forma, o problema reduz-se a um classificador binário, em que temos somente 2 classes possíveis para serem previstas.

# Capítulo 5

## Modelos de Previsão e Técnicas de Validação

Nesse capítulo são definidos os modelos utilizados para o treinamento e previsão, assim como as técnicas de validação utilizadas para o levantamento de precisão e das oportunidades de compra corretamente aproveitadas pelo preditor.

### 5.1 Modelos Utilizados

São definidos abaixo os modelos utilizados para treinamento e previsão do conjunto de dados.

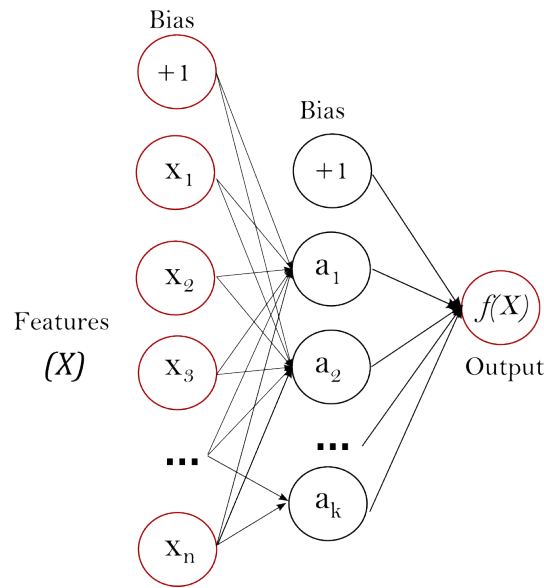
#### 5.1.1 MLP (*Multi-layer Perceptron*)

Segundo a definição de PEDREGOSA *et al.* [21], essa técnica trata-se de um algoritmo de aprendizado profundo supervisionado que é capaz de aprender uma função  $f(\cdot) : R^m \rightarrow R^o$  pelo treinamento sobre um conjunto de dados, sendo  $m$  o número de dimensões de entrada (variáveis do conjunto de treino) e  $o$  o número de dimensões da saída. Dadas uma combinação de variáveis de entrada  $X = x_1, x_2, \dots, x_m$  e um alvo de saída  $y$ , o modelo é capaz de aprender uma função de aproximação não-linear tanto para regressão quanto para classificação, sendo esse último o problema a ser resolvido nesse projeto. Entre a camada de entrada e a camada de saída podem existir uma ou mais camadas não lineares, chamadas de camadas escondidas. Ilustra-se o MLP a partir da figura 5.1:

Cada círculo na camada de entrada contém o que é chamado de neurônio  $\{x_i | x_1, x_2, \dots, x_m\}$  e cada círculo da camada escondida representa um neurônio da camada escondida e transforma os valores da camada anterior baseado em uma soma com pesos  $w_1x_1 + w_2x_2 + \dots + w_mx_m$ , seguido por uma função de ativação não linear



Figura 5.1: MLP com uma camada escondida entre a camada de entrada e a de saída

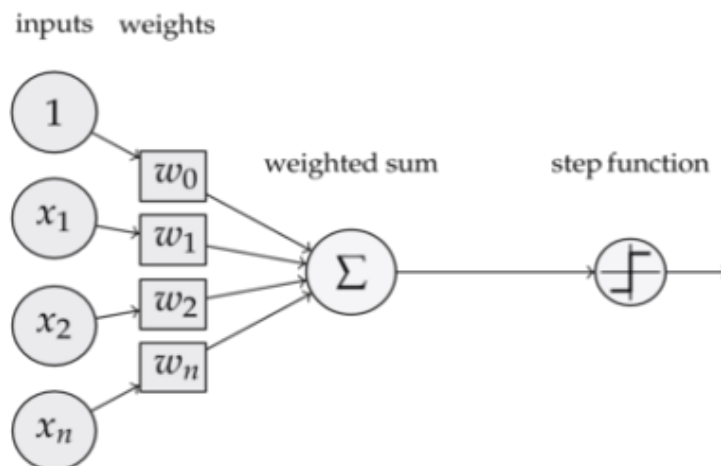


$g(\cdot) : R \rightarrow R$ . A camada de saída recebe os valores da última camada escondida e transforma nos valores de saída [22].

### 5.1.2 Regressão Logística

A regressão logística pode ser definida como uma técnica para calcular os parâmetros de um modelo logístico, ou seja, um modelo que se utiliza de funções logísticas para modelar o valor de uma variável binária dependente. Dessa maneira, essa técnica é vastamente utilizada para problemas de classificação cujas classes a serem previstas podem assumir 2 valores possíveis, como uma decisão de compra e não-compra de ações no mercado financeiro, escopo delimitado pelo projeto. A imagem 5.2 ilustra os passos para a previsão da saída pelo algoritmo.

Figura 5.2: Regressão logística



Assim como no MLP (5.1), a Regressão Logística aplica pesos às variáveis de entrada para encontrar uma soma calculada das mesmas. Em seguida, ao invés de passar por camadas escondidas, essa soma passa por uma função de atachamento que retorna uma probabilidade associada. Essa probabilidade, por fim, passa por uma função degrau que classifica a saída baseado em um limiar [23].

No modelo logístico, o logaritmo de acerto (referente às saídas classificadas como "1") é uma combinação linear de uma ou mais variáveis independentes binárias ou contínuas, que são as entradas do modelo. A probabilidade correspondente ao valor "1", pode variar continuamente entre 0 e 1 e a função que traduz essa probabilidade para a classe final é chamada de função logística.

## 5.2 Técnicas de Validação

Nessa seção são descritas as técnicas de validação usadas para o projeto.

### 5.2.1 Janelas Deslizantes

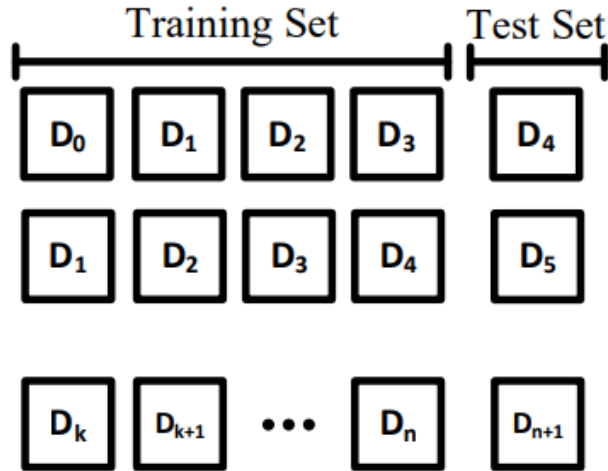
Como o problema é de séries de tempo, a divisão do conjunto de dados entre treinamento e teste foi feita de modo que, obrigatoriamente, o conjunto de testes fosse composto por dados obtidos cronologicamente depois dos dados de treino. Dessa forma, foi possível obter previsões mais próximas do cenário real, em que a entrada do modelo obedece uma ordenação temporal.

Com o objetivo de obter o maior aproveitamento possível de dias utilizados para treino e previsão do conjunto de dados, foi utilizada a técnica de janelas deslizantes de treino e previsão. Nessa estratégia, utiliza-se um número  $N$  de dias para formar o conjunto de treino e um número  $N'$  para o conjunto de testes para realizar a validação. Na próxima iteração, utiliza-se o mesmo número de dias para as duas janelas, com a diferença que a janela de treino e de previsão tem início e fim do início e fim anteriores. A imagem 5.3 pode ser usada para ilustrar a explicação.

### 5.2.2 Expansão das Janelas Deslizantes

Com o intuito de encontrar o número ótimo de dados de treino necessários para o modelo, foi feita uma iteração tal que, ao final do ciclo de janelas deslizantes de  $N$  dias de treino, fosse iniciado um outro ciclo com  $N + 1$  dias para o conjunto de treino. Foi utilizada a linguagem de programação *Python* para a implementação dessa lógica.

Figura 5.3: Janelas deslizantes com 4 dias usados para o conjunto de treino e 1 dia para o conjunto de teste



### 5.2.3 Parâmetros de validação

Para cada previsão feita pelo modelo, cria-se uma matriz de confusão (5.4) antes de extrair os parâmetros utilizados para mensurar a performance do modelo.

Figura 5.4: Matriz de confusão

		Valor Verdadeiro	
		Positivo	Negativo
Valor Previsto	Positivo	Verdadeiros Positivos	Falsos Positivos
	Negativo	Falsos Negativos	Verdadeiros Negativos

Com a matriz de confusão, pode-se visualizar de forma mais direta os valores previstos em relação aos valores verdadeiros. Dessa forma, são de interesse para a performance do modelo o número de verdadeiros positivos e verdadeiros negativos em relação ao resto do conjunto.

Um verdadeiro positivo é quando uma classe classificada como "1" no conjunto de testes de fato possui o valor verdadeiro "1". Um falso positivo, por sua vez, refere-se

a uma classe que é classificada como "1" enquanto possui "0" como valor verdadeiro. Pensamento análogo pode ser aplicado aos verdadeiros negativos e falsos negativos.

Para o domínio do trabalho, tem-se como interesse óbvio um modelo que seja capaz de obter mais acertos do que erros para obter lucro e, além disso, identifique o maior número possível de oportunidades de compra. Dessa maneira, a matriz de confusão pode ser reaproveitada para gerar os parâmetros de validação abaixo.

### **Revocação (Sensibilidade)**

A revocação é definida como a quantidade de valores classificados corretamente como verdadeiros sobre a quantidade total de verdadeiros do conjunto de teste. Pode ser calculada pela fórmula abaixo:

$$R = \frac{VP}{VP + FN} = \frac{VP}{TP} \quad (5.1)$$

Onde R é a revocação, FN é a quantidade de amostras classificadas como falso negativas e TP é o total de valores positivos do conjunto de teste.

### **Precisão**

A precisão é definida como a quantidade de valores classificados corretamente como verdadeiros sobre a quantidade total de previsões classificadas como verdadeiras no conjunto de teste. Pode ser calculada pela fórmula abaixo:

$$P = \frac{VP}{VP + FP} \quad (5.2)$$

Onde P é a precisão e VP e FP são classificadas como as amostras classificadas como verdadeiros positivos e falsos positivos, respectivamente.

Para cada ciclo de uma determinada janela de treinamento, é calculada a média das precisões e das realocações obtidas de forma a obter a melhor combinação de dias de treino. Quanto mais alta a média das precisões, mais alta é a taxa de acerto obtida para futuras amostras. Se a realocação estiver alta, significa que o modelo está identificando grande parte das oportunidades de compra. Nesse ponto, é possível escolher a estratégia a ser seguida: manter um modelo mais seguro, que tenha menos sensibilidade para a hora de entrar no mercado, porém com maiores chances de acerto; manter um modelo mais agressivo, que seja mais sensível para identificar oportunidades, porém menos preciso em sua previsão.

Estendendo a análise da precisão aplicada na estratégia definida na seção 4.2, é possível identificar imediatamente um mínimo de precisão segura, baseado nos limites superior  $x$  e inferior  $y$ , que o modelo deve alcançar para o cenário de prejuízos máximos. Um cenário de prejuízo máximo, como discutido no capítulo anterior, se caracteriza quando todas as previsões classificadas como falso positivas alcançam o limite inferior  $y$ , causando um prejuízo de  $N \cdot Y$ , sendo  $N$  o número de papéis negociados.

Definindo que, para obter lucro, o volume de verdadeiros positivos vezes o lucro precisa ser maior do que o volume de falsos positivos vezes a perda, tem-se a seguinte relação:

$$V \cdot P \cdot x \geq V \cdot (1 - P) \cdot y \quad (5.3)$$

Onde  $V$  é o volume de papéis negociados,  $P$  é a precisão de acerto e  $x$  e  $y$  referem-se aos limites superiores e inferior, respectivamente. Estabelecendo que a relação de  $x$  e  $y$  ocorre por  $K = \frac{x}{y}$ , a equação pode ser reduzida a:

$$K > \frac{1}{P} - 1 \quad (5.4)$$

E:

$$K + 1 > \frac{1}{P} \quad (5.5)$$

Então:

$$P > \frac{1}{K + 1} \quad (5.6)$$

Que é equivalente a:

$$P > \frac{y}{x + y} \quad (5.7)$$

Em suma, a partir da relação acima, pode ser definida uma precisão mínima que nunca causa prejuízos nos falsos positivos a partir dos limites  $x$  e  $y$ . Destaca-se o cenário de que, mesmo obtendo uma precisão menor do que a mínima segura, ainda é possível obter lucro pelo modelo se houver uma parcela de falsos positivos que não ultrapassem a margem inferior  $y$ .

# Capítulo 6

## Resultados e Discussões

Com o objetivo de puramente validar a aplicação dos modelos de aprendizado de máquina, o treinamento utilizou os modelos de Regressão Logística e *Multi-Layer Perceptron* para a geração dos resultados obtidos. Por conta da complexidade associada ao cálculo do MLP, a varredura de treino para os ciclos de dias totalizou aproximadamente de 4 a 5 horas.

Em relação aos parâmetros de calibração para os modelos de aprendizado, foram utilizados os padrões definidos pelo próprio pacote do *scikit-learn* [21] para ambos os modelos.

### 6.1 PETR4

Como estudo de caso para o projeto, foi escolhida a análise sobre a PETR4, ações preferências da Petrobrás, empresa que voltou a ocupar em outubro de 2018 o título de mais valiosa do Brasil, chegando a bater a marca de R\$358 bilhões [24].

Para isso, foram importados todos os dados diários negociados da BM&FBOVESPA de julho de 2018 até janeiro de 2019 e feito o pré-processamento descrito na seção 3.1.

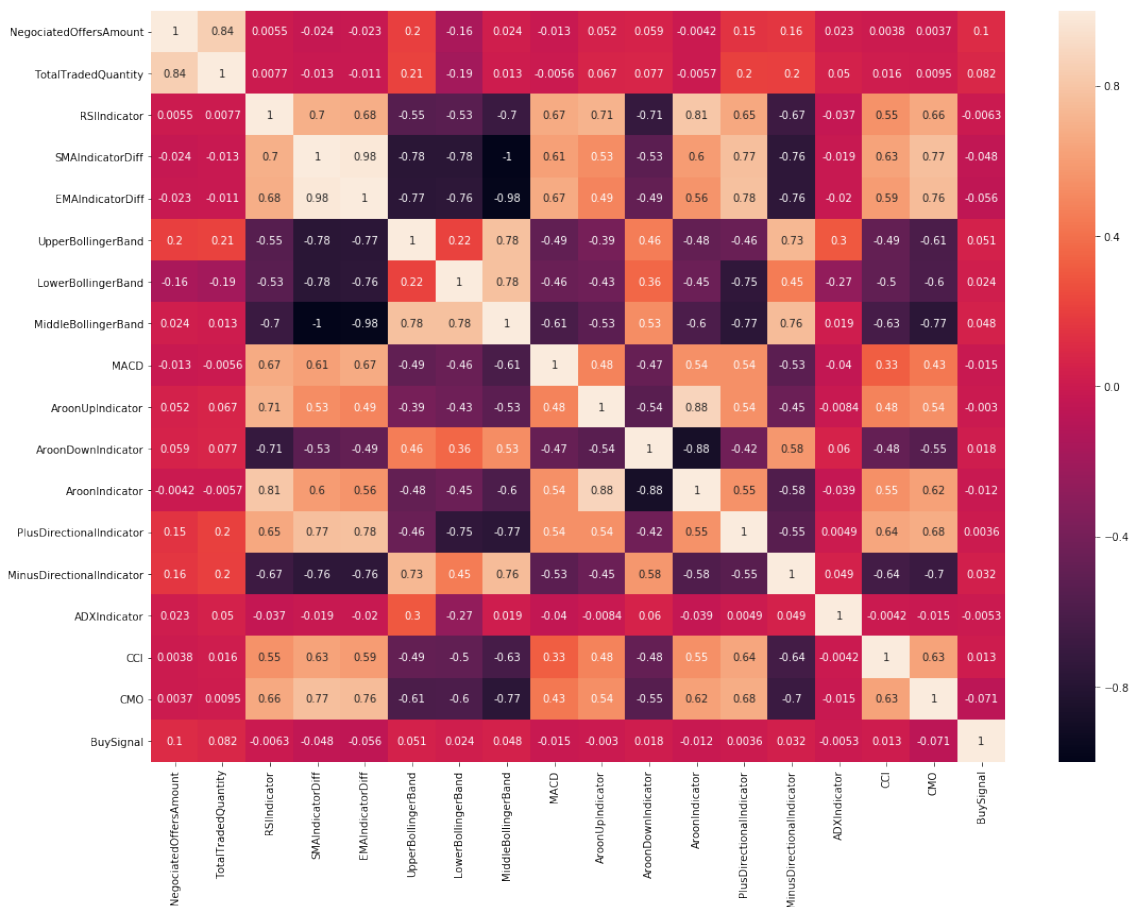
Após a importação dos dados brutos, escolheu-se empiricamente uma janela de tempo de 3 minutos para a geração os indicadores de análise técnica com a saída determinada pela estratégia descrita na seção 4.2 para o intervalo futuro de tempo de 1 minuto a partir do tempo de fim da janela de tempo.

Para o limite superior e inferior dos preços, também utilizados na estratégia de formação da variável de saída, foram considerados o valores  $x = 0,01$  e  $y = 2x = 0,02$ , respectivamente. Com a escolha desses limites, os dados ficaram distribuídos

com aproximadamente 66,7% como classe "1" e, para o treinamento do modelo, foram excluídos aleatoriamente dados dessa classe para que a distribuição ficasse equilibrada. A distribuição das classes próxima de 50% é essencial, pois reduz o viés que pode ser obtido decorrente de uma distribuição desproporcional dos dados. O detalhamento de outros parâmetros como períodos de tempo para formação de cada indicador está especificado no apêndice A. Especificados os parâmetros de pré-processamento, o software C# inicia o agrupamento dos dados para cada dia coletado na BM&FBOVESPA.

Após o pré-processamento dos dados e normalização das variáveis, observou-se a correlação entre as variáveis de entrada e a variável de saída pelo mapa de calor representado pela figura 6.1. Observa-se que, apesar das variáveis independentes de entrada possuírem uma correlação forte entre si em vários casos, a correlação dessas variáveis com a variável dependente de saída é fraca, alcançando seu maior valor com a variável que representa o número de negociações dentro do intervalo especificado.

Figura 6.1: Mapa de Calor das Correlações entre as Variáveis - PETR4

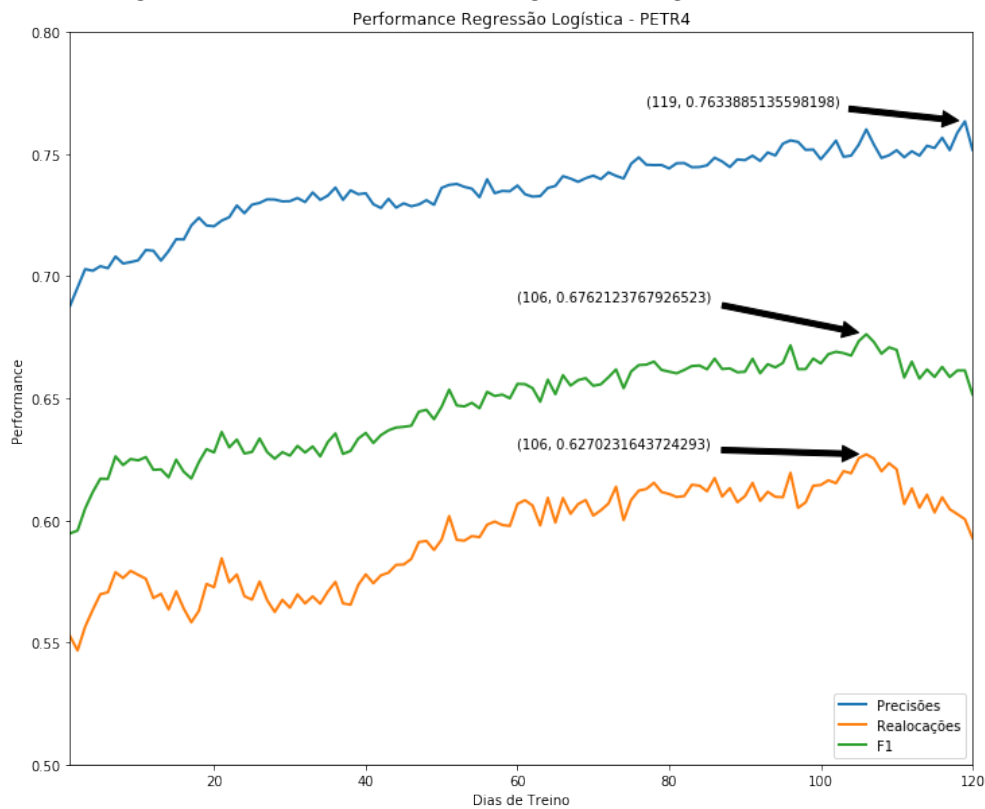


São aplicados, então, os modelos de Regressão Logística e *Multi-layer Perceptron* descritos em 5.1 para janela de treino, como detalhado em 5.2. As figuras 6.2 e 6.3 mostram a performance de cada modelo com a medida das precisões e realocações

sobre a quantidade de dias de treino. Dessa forma, é possível acompanhar a evolução das medidas até o máximo de cada métrica.

Em ambos os gráficos, nota-se que tanto as realocações quanto as precisões apresentam uma curva ascendente até alcançarem o seu máximo, depois apresentam uma curva descendente. Na Regressão Logística, a máxima realocação observada foi de aproximadamente 62,70% com 106 dias de treino e a máxima precisão foi de aproximadamente 76,34%, alcançada com 119 dias. Já no MLP, foi observada uma realocação máxima de aproximadamente 67,88% com 104 dias, enquanto que sua maior precisão foi de 75,37% com 119 dias de treino.

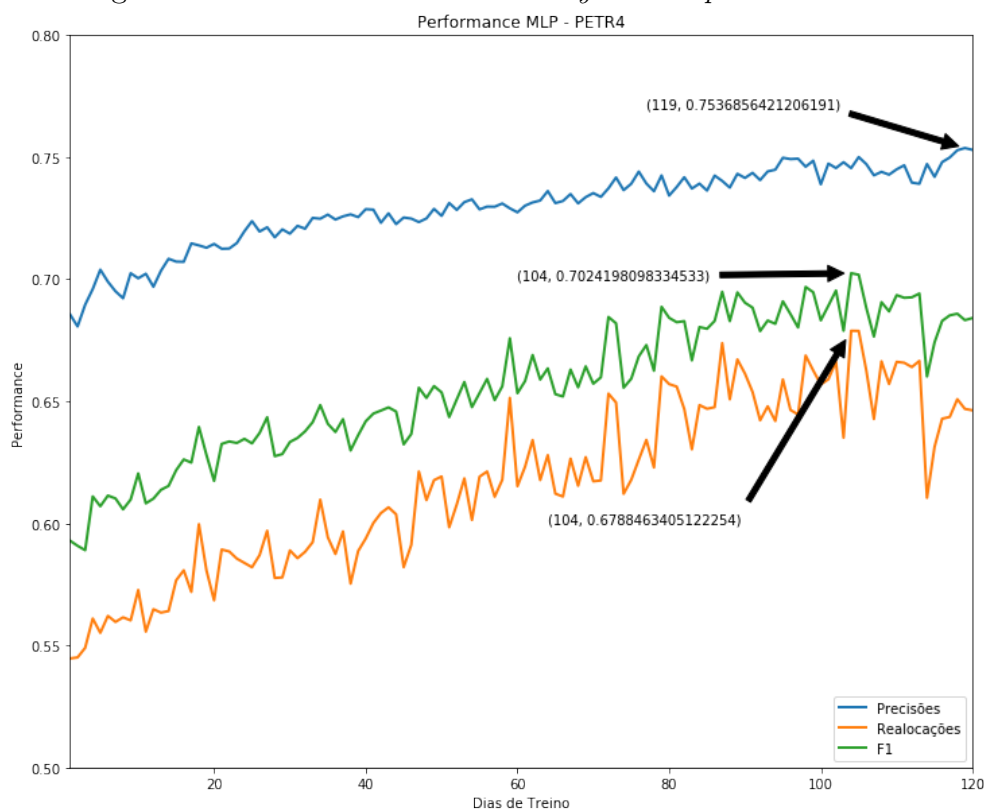
Figura 6.2: Performance da Regressão Logística - PETR4



Ao observar a curva de precisão em ambos os modelos, nota-se que conforme os dias de treino aumentam, essa medida tende a crescer. A curva de sensibilidade ou realocação, apesar de ter apresentado um comportamento ascendente durante a maior parte da janela, apresentou uma queda após o seu máximo, indicando um limite de dados ótimos para a identificação de oportunidades de compra. A curva F1, representada em verde, indica uma média harmônica entre a precisão e a sensibilidade, acompanhando a evolução da performance das 2 medidas, estando sempre entre as mesmas.



Figura 6.3: Performance do *Multi-layer Perceptron* - PETRA4



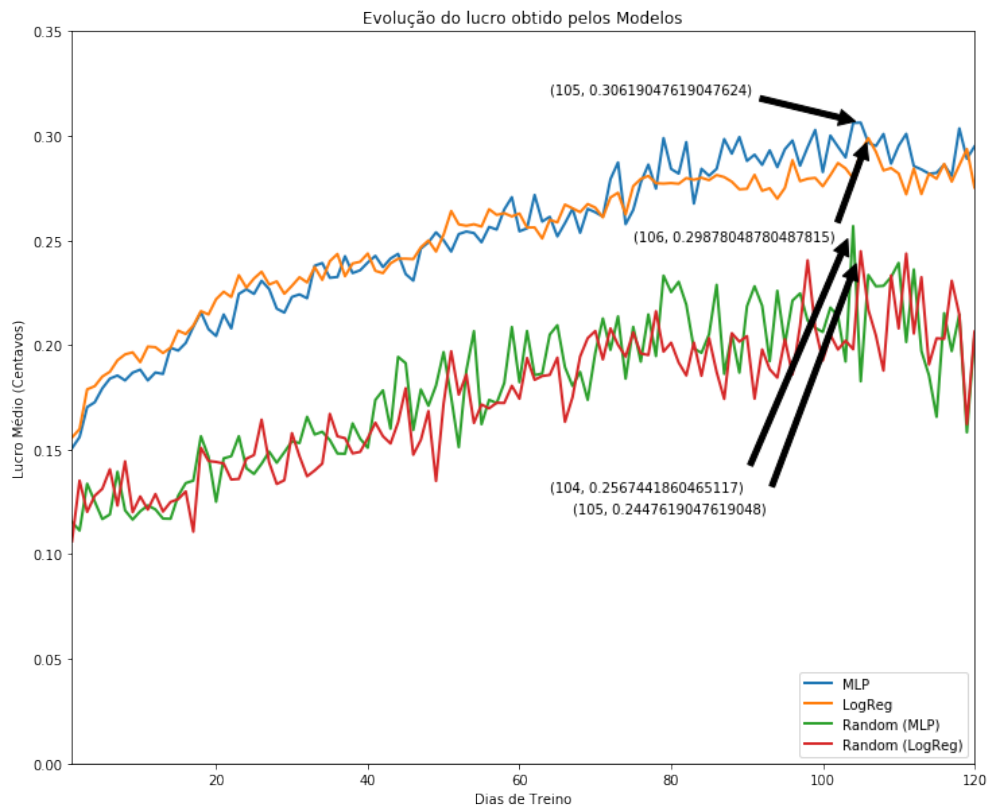
Comparando os máximos valores de precisão obtidos no *Multi-Layer Perceptron* e na Regressão Logística com a mínima precisão mínima segura definida na equação 5.7, nota-se que em ambos os modelos o mínimo de 66,7% é superado. Espera-se, dessa forma, que sucessivas ordens de compra e venda obtenham lucro se enviadas respeitando a previsão do modelo.

Apesar das medidas de precisão serem um indicador positivo na direção dos preços, elas não refletem necessariamente que o modelo obterá lucro. Em uma futura estratégia de compra e venda por um agente autônomo, ordens de compra e venda serão enviadas à mercado, isto é, terão de se adequar a um limiar acima e abaixo do valor nominal do preço para que sejam imediatamente realizadas.

Com o objetivo de simular as movimentações automáticas e prever o lucro diário, o livro de ofertas com as ordens de compra e venda à mercado foi reconstruído com os dados de negociações a partir da última agressão de compra ou venda de um intervalo específico. Dessa forma, uma ordem de compra à mercado precisa atender necessariamente a última negociação em que a ordem de venda foi a agressora e vice-versa.

Levantados os lucros reais obtidos para cada janela de previsão, foi possível classificar a média de lucro dos dias utilizados para o treinamento do modelo. Para isso, somaram-se todas as previsões positivas, ou seja, verdadeiros positivos e falsos positivos de cada dia previsto para cada ciclo de dias de treinamento. A evolução dos lucros sobre a quantidade de dias de treino pode ser observada pela figura 6.4. O eixo  $y$  do gráfico indica a soma dos ganhos por unidade de ação diariamente enquanto que o eixo  $x$  indica o número de dias de treino.

Figura 6.4: Evolução do lucro dos modelos - PETR4



Para uma base comparativa do resultado dos modelos com um modelo puramente aleatório, foram adicionadas 2 curvas no gráfico mostrando a evolução do lucro de um modelo puramente aleatório que enviase o mesmo número de ordens de compra e venda que os modelos de aprendizado de máquina utilizados. Observa-se que, mesmo para ordens completamente aleatórias, foi possível observar um lucro positivo, indicando que o ativo sendo analisado apresentou uma alta constante durante o período de amostragem. Mesmo assim, pode-se distinguir claramente a diferença do lucro de movimentações aleatórias com o lucro dos modelos de aprendizado de máquina, sempre maiores durante todo o intervalo.

Pela evolução dos lucros obtidos, nota-se que existe um lucro crescente que acompanha a performance de ambos os modelos. Para a Regressão Logística, o máximo

lucro obtido coincidiu com o número de dias necessários para obter a maior performance de sensibilidade (realocação), 106 dias. O lucro máximo para o MLP, apesar de não coincidir com o número de dias de treino para o máximo de precisão e sensibilidade, também esteve próximo do máximo de sensibilidade encontrado.

## 6.2 ITUB4

De forma similar às ações da PETR4, foi feita a análise para as ações do banco Itaú, ITUB4.

Como janela de tempo para aplicação dos indicadores de análise técnica, foi utilizado o mesmo intervalo de 3 minutos. Para os limites superiores e inferiores da janela de previsão,  $x$  e  $y$ , foram utilizados os valores de 0,02 e 0,03, respectivamente. Dessa forma, tem-se a mínima precisão segura de 60%, como descrita na fórmula 5.7.

Após o pré-processamento dos dados brutos, a distribuição entre as classes obtida foi de aproximadamente 49,96% para a classe "1", indicando pouca perda no posterior equilíbrio para alcançar 50% no treinamento. A correlação entre as variáveis de entrada e a de saída podem ser observadas pela figura 6.5.

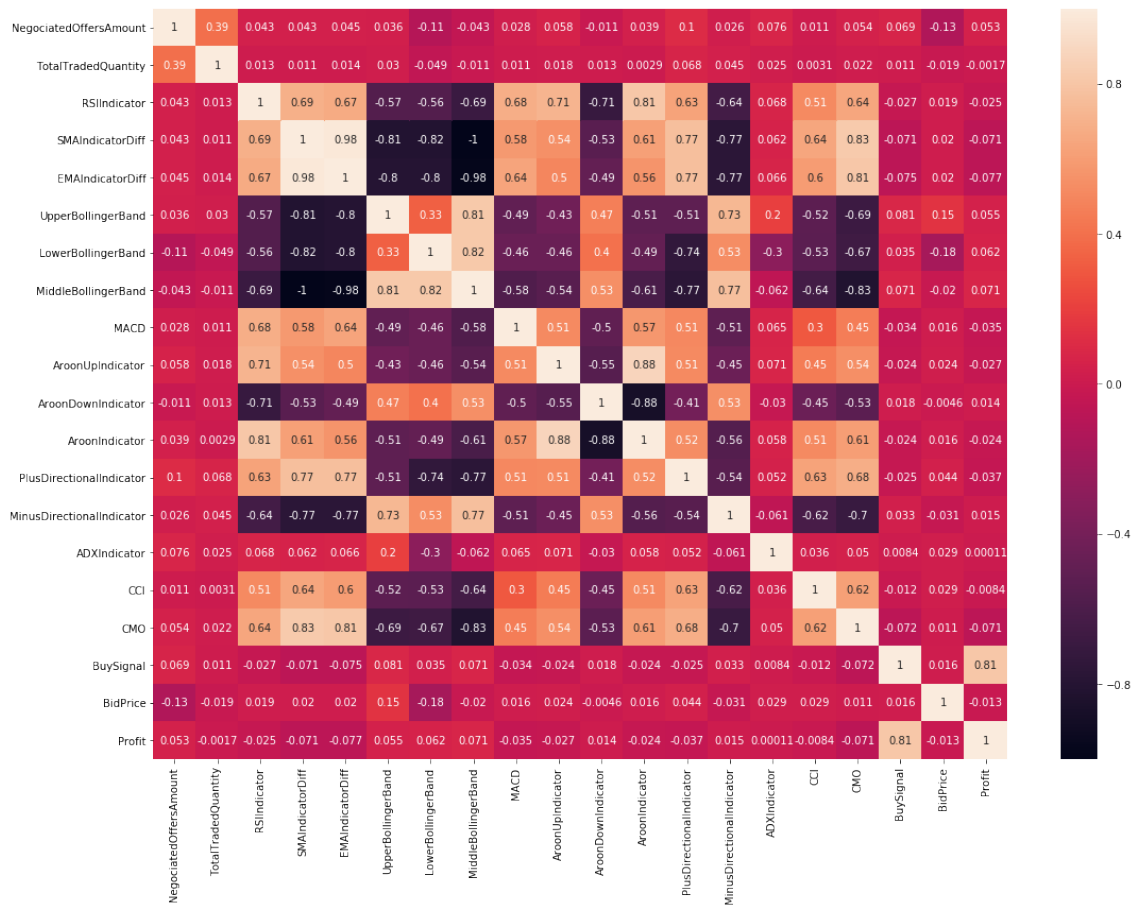
Como pode ser observado pelas figuras 6.6 e 6.7, a curva de sensibilidade (realocação) apresentou uma curva ascendente até o final da janela, indicando que o modelo ainda pode identificar mais oportunidades se treinado com mais dias. A curva de precisão, apesar da pouca variação no intervalo, apresentou uma curva ligeiramente crescente até alcançar o seu máximo antes de decrescer.

Pelo MLP, foram necessários menos 12 dias para alcançar o máximo de precisão, 56,43%, do que pela Regressão Logística. Esse último, por sua vez, alcançou 56,60% de precisão com 75 dias de treinamento. Para as realocações, ambos os modelos aparentaram ser capazes de progredirem suas performances, apresentando máximo somente ao fim da janela de dias de treino.

Ao comparar a suavidade das curvas, nota-se que a Regressão Logística apresentou muito menos ruído do que as curvas do MLP, tanto para a precisão quanto para a realocação.

Para acompanhar a evolução dos lucros simulados dos modelos para ITUB4, foi gerada a figura 6.8. Nota-se que, mesmo a máxima precisão alcançada ter ficado abaixo

Figura 6.5: Mapa de Calor das Correlações entre as Variáveis - ITUB4



da mínima precisão segura, o máximo lucro obtido pelos modelos para ITUB4, de 0,33, foi maior do que o lucro obtido para as ações da PETR4, de aproximadamente 0,31, que obteve uma taxa de precisão maior do que a segura. Em comparação com um modelo que envia aleatoriamente o mesmo número de ordens de compra e venda nos modelos de aprendizado, nota-se que a curva desses últimos foram sempre maiores. Assim como a PETR4, a ITUB4 também apresentou uma constância de altas no período de análise devido ao lucro positivo dos modelos aleatórios durante o intervalo de amostras.

Figura 6.6: Performance da Regressão Logística - ITUB4

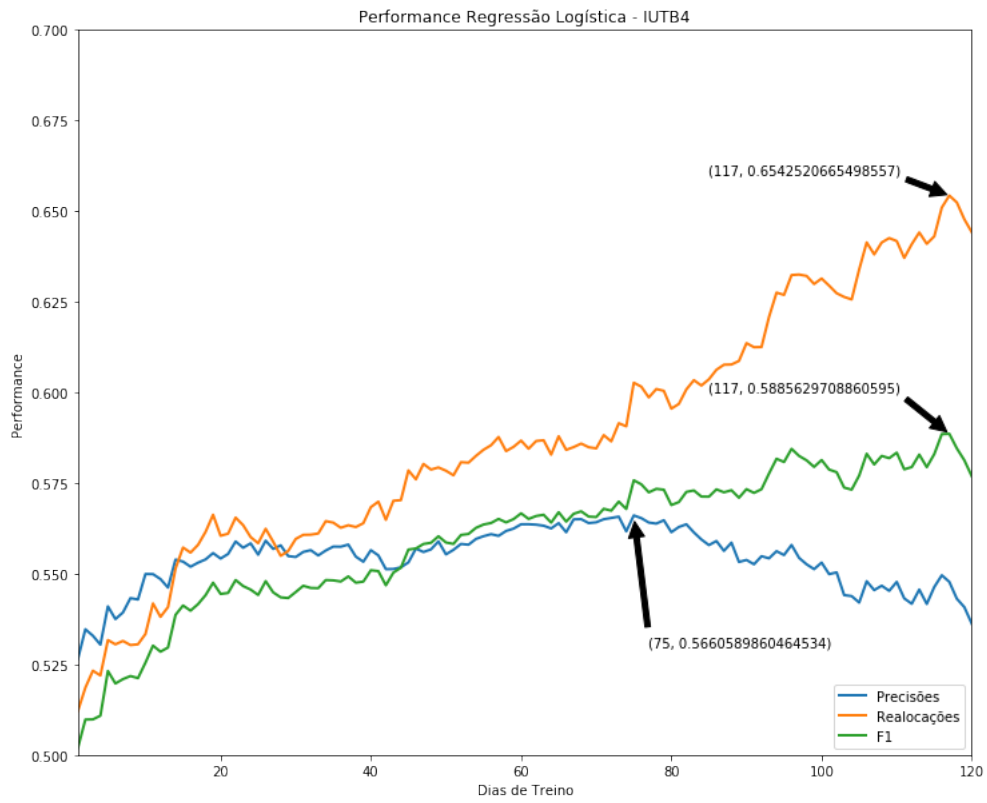


Figura 6.7: Performance do *Multi-layer Perceptron* - ITUB4

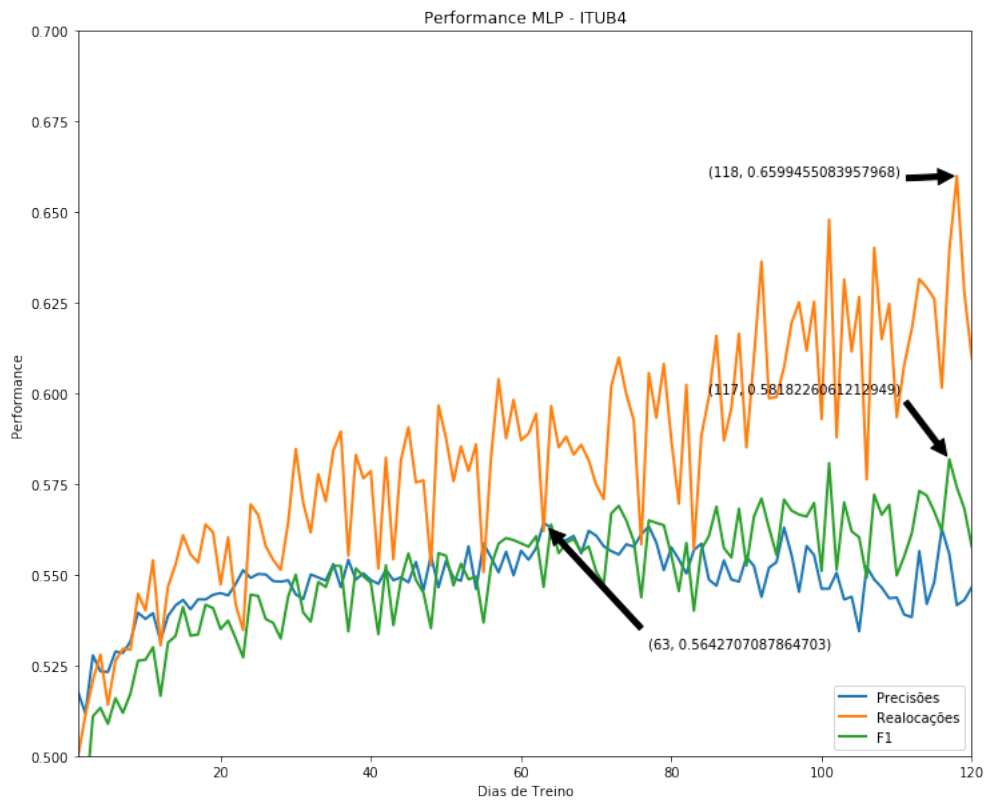
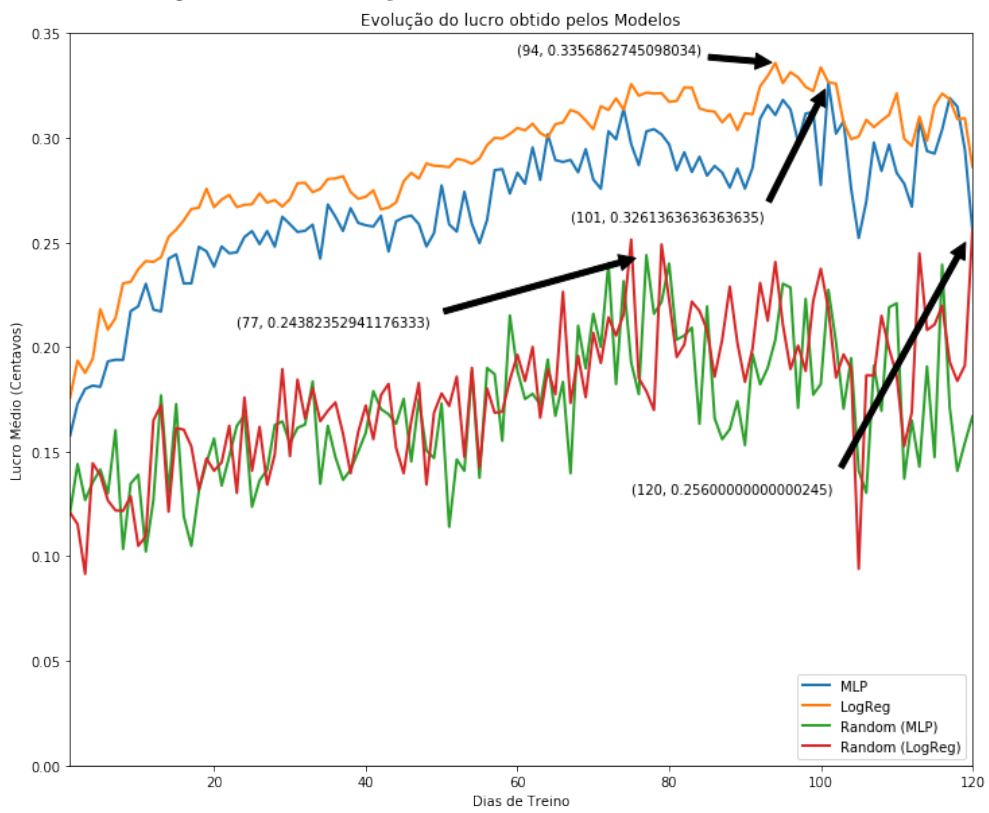


Figura 6.8: Evolução do lucro dos modelos - ITUB4



# Capítulo 7

## Conclusões

### 7.1 Conclusões Gerais

O presente trabalho teve como objetivo aplicar técnicas de aprendizado de máquina para gerar modelos de previsão de direção dos preços. A partir da validação utilizando testes históricos das séries temporais, foi possível verificar que a tomada de decisão apoiada por algoritmos de inteligência artificial pode ser muito mais eficaz do que uma simples análise técnica tradicional.

Ao aplicar a estratégia definida para os casos de uso reais das ações da Petrobrás (PETR4) e do Itaú (ITUB4), foi possível observar que, mesmo para um grupo mínimo de dias de treino, o modelo apresentou uma precisão consideravelmente maior do que a de um modelo aleatório e um lucro máximo substancial. Ao comparar os resultados da PETR4 com os da ITUB4, é possível perceber que não necessariamente a precisão do modelo precisa estar acima da mínima segura para a geração de lucro. Devido à natureza do ativo e aos limites superior e inferior escolhidos para as ações da ITUB4, obtiveram-se lucros maiores do que para PETR4, mesmo essa última apresentando uma precisão segura. Dessa forma, já é possível a utilização do presente modelo para uma estratégia mais arrojada puramente baseada em aprendizado de máquina.

Mesmo restringindo os casos de usos à análise de dois ativos específicos da BM&FBOVESPA, o agrupamento dos dados foi implementado com o objetivo de ser generalizado para qualquer ativo do mercado financeiro em uma base diária de *ticks* que defina o mesmo *layout* da tabela 3.1, incluindo mini-índices e opções. A flexibilização pensada no *software* implementado para a geração dos indicadores de análise técnica também oferece a possibilidade de customização e adaptação para ativos com outra natureza de flutuação, permitindo análises adequadas aos mais diversos fatores que regem a oscilação dos preços de uma ação.

## 7.2 Trabalhos Futuros

Em relação ao conjunto de dados de entrada do modelo de predição, nota-se uma predominância muito maior de indicadores técnicos aplicados puramente ao dados provenientes do conjunto de dados negociados. Com o objetivo de aprofundar a análise para considerar a volatilidade de um ativo, torna-se interessante a consideração das informações do livro de ofertas para a criação de variáveis com informações sobre a quantidade de ordens de compra e venda em um intervalo, o valor dessas ordens e outras características.

Em relação ao agrupamento dos dados, foi utilizada uma abordagem por separação de janelas de tempo e previsão de forma que uma janela de tempo nunca colidissem com a outra. Por causa disso, a implementação de janelas deslizantes no tempo podem ser úteis para aumentar a quantidade de dados adquiridos para o treinamento do modelo.

Sobre os modelos de aprendizado aplicados, foram utilizadas calibrações padrão do *scikit-learn*. Em um cenário real, torna-se necessária a otimização dos parâmetros de calibração de cada modelo em seu contexto específico, de modo a encontrar a melhor performance possível. Dessa maneira, sugere-se que seja feita programaticamente uma varredura de performance com várias combinações possíveis de entrada para que haja melhor adequação do modelo escolhido para cada situação.

Para consumo diário de um agente autônomo responsável pelas negociações, ainda é necessária a aquisição manual dos dados nos servidores públicos da BM&FBOVESPA e posterior execução do *software* desenvolvido para o pré-processamento. Neste caso, uma solução de agendamento que recorrentemente verificasse a disponibilização de novos arquivos de negociação seguida de um pré-processamento e agrupamento dos mesmos poderia automatizar esse esforço.

Após treinado o modelo com os dados agrupados, torna-se necessária a exposição de um serviço que retorne o sinal de previsão do próximo intervalo baseado nas entradas de indicadores de análise técnica descritos ao longo do trabalho. Para garantir a disponibilidade desse serviço, torna-se interessante a hospedagem do mesmo em algum provedor de nuvem como *Amazon Web Services* ou *Google Cloud Computing*. A lógica a ser executada a cada requisição, dessa maneira, seria a predição desses dados de entrada em um modelo já treinado, salvo em um arquivo binário e carregado em memória para executar imediatamente as previsões.



# Referências Bibliográficas

- [1] INFOMONEY, P. “Trader, não deixe que te enganem sobre DMA”. <https://www.infomoney.com.br/conteudo-patrocinado/xpi/noticia/6946952/trader-nao-deixe-que-enganem-sobre-dma>, 2017. Acessado em Fevereiro/2019.
- [2] MORENO, F. “Robôs operando na velocidade da luz: conheça o High Frequency Trading”. <https://www.infomoney.com.br/mercados/noticia/2376689/robos-operando-velocidade-luz-conheca-high-frequency-trading>, 2012. Acessado em Fevereiro/2019.
- [3] BARRETO, R. “Você sabe a importância do mercado de capitais?” <https://www.infomoney.com.br/blogs/economia-e-politica/economia-com-renata-barreto/post/5454164/voce-sabe-importancia-mercado-capitais>, 2016. Acessado em Fevereiro/2019.
- [4] BRASIL, P. “Como abrir o capital da sua empresa no Brasil (IPO)”. <http://vemprabolsa.com.br/wp-content/uploads/2016/06/Guia-abertura-de-capital-%E2%80%93-93-BMFB0VESPA-e-PricewaterhouseCoopers.pdf>, 2015. Acessado em Fevereiro/2019.
- [5] ALAIN CHABOUD, BENJAMIN CHIQUOINE, E. H. C. V. “Rise of the Machines: Algorithmic Trading in the Foreign Exchange Market”, *Federal Reserve System International Finance Discussion Papers*, v. 1, n. 980, pp. 1–3, out. 2009.
- [6] PETER GOMBER, BJÖRN ARNDT, M. L. T. U. “High-Frequency Trading”, v. 1, n. SSRN 1858626, pp. 13–15, 2011.
- [7] WHITE, H. “ECONOMIC PREDICTION USING NEURAL NETWORKS: THE CASE OF IBM DAILY STOCK RETURNS”, 1988.

- [8] ADEBIYI AYODELE A., AYO CHARLES K., A. M. O., O., O. S. “Stock Price Prediction using Neural Network with Hybridized Market Indicators”, *Journal of Emerging Trends in Computing and Information Sciences*, v. 1, n. 3, jan. 2012.
- [9] AUTHORHAORAN XIE, X. L. D. Z. W. “An intelligent market making strategy in algorithmic trading”, v. 8, n. 4, ago. 2014.
- [10] DIXON, M. F. “Deep learning for spatio-temporal modeling: Dynamic traffic flows and high frequency trading”, jul. 2017.
- [11] PUTRA, E. F. “Application of Artificial Neural Networks To Predict Intraday Trading Signals”, .
- [12] SILVA, E. J. D. *MODELAGEM E APLICAÇÃO DE TÉCNICAS DE APRENDIZADO DE MÁQUINA PARA NEGOCIAÇÃO EM ALTA FREQUÊNCIA EM BOLSA DE VALORES*. Dissertação de mestrado, Universidade Federal de Minas Gerais, Belo Horizonte, 2015.
- [13] HAYES, A. “Simple Moving Average - SMA Definition”. <https://www.investopedia.com/terms/s/sma.asp>, 2019. Acessado em Fevereiro/2019.
- [14] CHARTS, S. “Relative Strength Index (RSI)”. [https://stockcharts.com/school/doku.php?id=chart\\_school:technical\\_indicators:relative\\_strength\\_index\\_rsi](https://stockcharts.com/school/doku.php?id=chart_school:technical_indicators:relative_strength_index_rsi). Acessado em Fevereiro/2019.
- [15] LANA, M. “Como usar Bandas de Bollinger?” <https://smartttbot.com/blog/como-usar-bandas-de-bollinger/>, 2016. Acessado em Fevereiro/2019.
- [16] HAYES, A. “Moving Average Convergence Divergence - MACD Definition”. <https://www.investopedia.com/terms/m/macd.asp>, 2019. Acessado em Fevereiro/2019.
- [17] CHARTS, S. “Average Directional Index (ADX)”. [https://stockcharts.com/school/doku.php?id=chart\\_school:technical\\_indicators:average\\_directional\\_index\\_adx](https://stockcharts.com/school/doku.php?id=chart_school:technical_indicators:average_directional_index_adx), 2018. Acessado em Fevereiro/2019.
- [18] HAYES, A. “Average True Range - ATR Definition”. <https://www.investopedia.com/terms/a/atr.asp>, 2019. Acessado em Fevereiro/2019.
- [19] CHARTS, S. “Commodity Channel Index (CCI)”. [https://stockcharts.com/school/doku.php?id=chart\\_school:technical\\_indicators:commodity\\_channel\\_index\\_cci](https://stockcharts.com/school/doku.php?id=chart_school:technical_indicators:commodity_channel_index_cci), 2019. Acessado em Fevereiro/2019.

- [20] CHEN, J. “Chande Momentum Oscillator Definition”. <https://www.investopedia.com/terms/c/chandemomentumoscillator.asp>, 2019. Acessado em Fevereiro/2019.
- [21] PEDREGOSA, F., VAROQUAUX, G., GRAMFORT, A., et al. “Scikit-learn: Machine Learning in Python”, *Journal of Machine Learning Research*, v. 12, pp. 2825–2830, 2011.
- [22] PEDREGOSA, F., VAROQUAUX, G., GRAMFORT, A., et al. “1.17. Neural network models (supervised)”. [https://scikit-learn.org/stable/modules/neural\\_networks\\_supervised.html](https://scikit-learn.org/stable/modules/neural_networks_supervised.html). Acessado em Fevereiro/2019.
- [23] FACURE, M. “Regressão Logística”. <https://matheusfacure.github.io/2017/02/25/regr-log/>, 2017. Acessado em Fevereiro/2019.
- [24] MIOZZO, J. “Como comprar as ações da Petrobras, passo a passo”. <https://www.infomoney.com.br/onde-investir/acoes/noticia/7652938/como-comprar-as-acoes-da-petrobras-passo-a-passo>, 2018. Acessado em Fevereiro/2019.

# Apêndice A

## Parâmetros de calibração

Na lista abaixo, estão presentes todos os parâmetros de podem ser calibrados na geração de dados agrupados. Além dos períodos para os indicadores de análise técnica, também é possível selecionar o tamanho das janelas de tempo e previsão e limites superiores e inferiores para estruturar a estratégia de compra e venda.

```

1      {
2          "negFolderPath": "C:\\MarketData\\Bovespa\\",
3          "buyFolderPath": "C:\\MarketData\\AllData\\",
4          "sellFolderPath": "C:\\MarketData\\AllData\\",
5          "negFilePrefix": "NEG_",
6          "buyFilePrefix": "OFER_CPA_",
7          "sellFilePrefix": "OFER_VDA_",
8          "dateSuffixFormat": "yyyyMMdd",
9          "instrumentSymbol": "PETR4",
10         "slidingWindowMinutes": 3,
11         "RSIPeriods": 14,
12         "SMAPeriods": 14,
13         "BollingerBandsPeriods": 14,
14         "BuyTimeHold": 1,
15         "CsvCharSeparator": ",",
16         "PeriodsToNormalize": 28,
17         "OutputCsvPath": "C:\\Projects\\GitHub\\stock-analysis\\",
18         "ShortMACDPeriods": 12,
19         "LongMACDPeriods": 26,
20         "AroonIndicatorPeriods": 14,
21         "ConsiderOrderFiles": false,
22         "InterpolateWindows": false,
23         "IndexStockCodeVariation": false,
24         "ATRPeriods": 14,
25         "CMOPeriods": 14,
26         "ROCPeriods": 1,
27         "MinimumVariationOfInterest": 0.01
28     }

```

Listing 1: Arquivo JSON com propriedades de calibração dos indicadores de análise técnica

# Apêndice B

## Código Base

Todo o código base desenvolvido para o trabalho foi hospedado no repositório público *GitHub* e pode ser acessado na página do autor <https://github.com/lshlee>. Dessa maneira, o código pode ser reutilizado e evoluído por qualquer pessoa que quiser contribuir e evoluir com o presente trabalho.