



Trabalho de Conclusão de Curso

Metodologia para análise de mercado de ações com base em técnicas de aprendizado de máquina

Caio César dos Santos Nunes
ccsn@ic.ufal.br

Orientador:
Prof. Dr. André Luiz Lins de Aquino

Catálogo na fonte
Universidade Federal de Alagoas
Biblioteca Central
Divisão de Tratamento Técnico

Bibliotecário: Marcelino de Carvalho Freitas Neto – CRB-4 - 1767

N972m Nunes, Caio César dos Santos.

Metodologia para análise de mercado de ações com base em técnicas de aprendizado de máquina / Caio César dos Santos Nunes. – 2021.

36 f. : il.

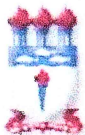
Orientador: André Luiz Lins de Aquino.

Monografia (Trabalho de conclusão de curso em Engenharia de Computação) - Universidade Federal de Alagoas, Instituto de Computação. Maceió, 2020.

Bibliografia: f. 33-36.


1. Aprendizado do computador. 2. Mercado de ações. 3. Redes neurais (Computação). 4. Seleção de atributos. I. Título.

CDU: 004.383.8.032.26



Trabalho de Conclusão de Curso - TCC

Formulário de Avaliação

Nome do Aluno																				
C	A	I	O		C	E	S	A	R		D	O	S		S	A	N	T	O	
S		N	U	N	E	S														
Nº de Matrícula																				
1	2	2	1	2	2	5	1													
Título do TCC (Tema)																				
Metodologia para análise de mercado de ações com base em técnicas de aprendizado de máquina																				
Banca Examinadora																				
André Luiz Lins de Aquino Nome do Orientador						 Assinatura														
Leonardo Viana Pereira Nome do Professor						 Assinatura														
Daniel Gomes de Mello Farias Nome do Professor						 Assinatura														
Data da Defesa								Nota Obtida												
21/12/2020								8,0 (OITO)												
Observações																				
O ALUNO VAI ATENDER À TODOS OS COMENTÁRIOS DA BANCA NA VERSÃO FINAL DO TCC																				
Coordenador do Curso																				
De Acordo																				
 Assinatura																				

Caio César dos Santos Nunes

Metodologia para análise de mercado de ações com base em técnicas de aprendizado de máquina

Monografia apresentada como requisito parcial para obtenção do grau de Bacharel em Engenharia de Computação do Instituto de Computação da Universidade Federal de Alagoas.

Orientador:

Prof. Dr. André Luiz Lins de Aquino

Monografia apresentada como requisito parcial para obtenção do grau de Bacharel em Engenharia de Computação do Instituto de Computação da Universidade Federal de Alagoas, aprovada pela comissão examinadora que abaixo assina.

Prof. Dr. André Luiz Lins de Aquino - Orientador
Instituto de Computação
Universidade Federal de Alagoas

Rian Gabriel Santos Pinheiro - Examinador
Instituto de Computação
Universidade Federal de Alagoas

Ivan César Martins - Examinador
Instituto de Computação
Universidade Federal de Alagoas

Agradecimentos

Minha jornada através do curso de graduação de Engenharia de Computação foi longa e tortuosa, e sem o suporte algumas pessoas especiais essa realização não teria sido possível.

Primeiramente agradeço a minha mãe, como não poderia deixar de ser. Agradeço pelo apoio incondicional, pelos esforços em nome da minha educação e por ter assumido tantas responsabilidades ao longo dos anos. Agradeço também a minha namorada, minha companheira que me deu suporte e me motivou para que mesmo diante das adversidades continuasse perseverando. Homenageio meus amigos, todos aqueles que estão ou já estiveram ao meu lado durante minha jornada na Universidade Federal de Alagoas, que me acolheram e comigo compartilharam amizade. Dedico essa conquista a cada um dos professores cujos quais tive o privilégio de aprender com, agradeço a esses por terem dedicado seu tempo de vida para formarem novos cidadãos, membros da sociedade e profissionais.

Por ultimo mas definitivamente não menos importante, agradeço e dedico todas as minhas conquistas aos que vieram antes de mim, aqueles que construíram essa nação com sangue e suor e que deram suas vidas para que hoje um homem negro pudesse alcançar o ensino superior em uma instituição pública de ensino. Dedico minhas conquistas aos povos negros e indígenas em sua luta imparável por justiça.

Resumo

A predição dos valores de ações é um problema difícil, as séries temporais financeiras possuem comportamento caótico e sendo o “humor” do mercado financeiro uma variável determinante para manobras políticas e econômicas, muitos pesquisadores dedicaram seu tempo para desenvolver meios de prever seu comportamento. Utilizando redes neurais convolucionais e redes neurais de memória a longo prazo construímos modelos capazes de prever o preço da empresa Apple na bolsa de valores. Os resultados obtidos com a metodologia proposta apontam que redes neurais são capazes de generalizar razoavelmente bem o problema, no entanto os resultados obtidos não são positivos o suficiente para justificar o uso dos modelos em ambientes competitivos.

Palavras-chave: Aprendizagem de máquina, Mercado de ações, Redes Neurais Artificiais, Seleção de Atributos.

Abstract

Predicting stock values is a difficult problem, financial time series have chaotic behavior and with the “mood” of the financial market being a determining variable for political and economic maneuvers, many researchers have devoted their time to developing ways to predict their behavior. Using convolutional neural networks and long-term memory neural networks, we build models capable of predicting the price of the company Apple on the stock exchange. The results obtained with the proposed methodology indicate that neural networks are able to generalize the problem reasonably well, however the results obtained are not positive enough to justify the use of models in a competitive environment.

Keywords: Machine learning, Stock market, Artificial Neural Networks, Feature selection.

Sumário

Lista de Figuras	vi
Lista de Tabelas	vii
1 Introdução	1
1.1 Motivação	1
1.2 Trabalhos Relacionados	2
1.3 Objetivo	2
1.4 Contribuições	2
2 Tecnologias e Conceitos	4
2.1 O mercado de ações	4
2.1.1 Análise técnica do mercado de ações	4
2.2 Aprendizagem de máquina	11
2.2.1 Algoritmos supervisionados e tarefa de regressão	11
2.2.2 Função de perda	12
2.2.3 Overfitting	12
2.2.4 Seleção de atributos	12
2.2.5 Árvore de decisão	12
2.2.6 Random Forests	13
2.2.7 Lasso	14
2.2.8 Support vector machines (SVM)	15
2.2.9 Redes Neurais	16
3 Nossa proposta	21
3.1 Aquisição dos dados	21
3.2 Processamento preliminar dos dados	21
3.3 Seleção de atributos	23
3.3.1 Support Vector Regression	23
3.3.2 Random Forest	23
3.3.3 Lasso	23
3.4 Conversão dos dados para o formato supervisionado	25
3.5 Rede LSTM	25
3.6 Rede Neural Convolutacional	26
4 Resultados e Discussões	28
4.1 A Apple	28
4.2 Seleção de atributos	28
4.3 Modelos de predição	29
4.3.1 Métrica de avaliação	29

4.3.2	Rede Neural com células de memória a longo prazo (LSTM)	30
4.3.3	Rede Neural Convolutiva	30
4.3.4	Usabilidade do modelo	30
5	Considerações Finais	32
	Referências bibliográficas	33

Lista de Figuras

2.1	Gráfico gerado com valores de média móvel do preço das ações da empresa Apple.	6
2.2	Gráfico gerado do momento das ações da companhia Google.	7
2.3	Oscilador ROC plotado a partir dos valores da empresa Apple.	8
2.4	RSI plotado com os dados da Microsoft Corporation.	9
2.5	MACD da empresa Amazon no período especificado.	9
2.6	Oscilador estocástico gerado a partir dos registros da empresa Google.	10
2.7	Pseudo-código do algoritmo C4.5 baseado em Quinlan (1993)	13
2.8	Pseudocódigo do algoritmo de Random Forest baseado em Breiman (2001)	14
2.9	Objetos de classes distintas separadas pelo hiperplano H. Lorena and de Carvalho (2007)	15
2.10	Ilustração de rede neural biológica. Freepik (2020)	17
2.11	Ilustração de uma rede neural profunda. DigitalOcean (2018)	17
2.12	Encadeamento de células LSTM. Olah	19
2.13	A figura (b) ilustra a aplicação de Dropout em uma rede neural comum, cada um dos componentes r é responsável por ativar ou não as saídas y da camada interior. Srivastava et al. (2014)	20
2.14	Pseudo-código do otimizador Adam. Kingma and Ba (2017)	20
3.1	Rede LSTM com N camadas seguidas por células de Dropout.	25
3.2	Topologia da rede neural convolucional plotada a partir do pacote Keras .	27
4.1	Contagem da ocorrência dos atributos durante a seleção.	29
4.2	Gráfico de 200 dias de teste, comparação entre o valor real e o valor previsto pela rede convolucional utilizando o conjunto de atributos gerados pelo algoritmo Random Forest.	31

Lista de Tabelas

3.1	Atributos iniciais.	21
3.2	Lista de atributos após a adição de indicadores técnicos de mercado.	22
3.3	Parâmetros do modelo de SVR	23
3.4	Parâmetros do modelo de RF.	24
3.5	Parâmetros do modelo de Lasso.	24
3.6	Formatação da entrada para formato supervisionado.	25
3.7	Parâmetros de criação da rede LSTM por conjunto de atributos.	26
3.8	Parâmetros de configuração do otimizador adam.	26
3.9	Parâmetros de configuração para as camadas da rede convolucional.	26
4.1	Lista de atributos selecionados por cada um dos modelos supervisionados.	29
4.2	Predição Rede LSTM 1 dia a frente	30
4.3	Predição Rede Convolucional 1 dia a frente	30

1

Introdução

Nosso trabalho consiste de uma metodologia para predição de valores diários no mercado de ações. Com uma base de dados coletada a partir do portal [Yahoo](#) extraímos indicadores técnicos de mercado e fizemos uso de algoritmos de aprendizagem supervisionada para selecionar os atributos mais relevantes. Nossos modelos preditivos consistem de duas redes neurais uma de memória a longo prazo (LSTM) e outra convolucional, ambas tiveram seus parâmetros refinados a partir de um processo de busca aleatória.

1.1 Motivação

As bolsas de valores são componentes de fundamental importância dos ecossistemas econômicos ao redor do globo, nelas os emissores transacionam suas ações em busca de financiamento para seus projetos, enquanto investidores as compram visando a obtenção de lucro. Os preços dos ativos listados são influenciados por diversos fatores econômicos e políticos o que torna o mercado de ações extremamente volátil [LABIAD et al. \(2019\)](#).

A predição dos valores de ações é um problema difícil, as séries temporais financeiras possuem comportamento caótico e sendo o “humor” do mercado financeiro uma variável determinante para manobras políticas e econômicas, o estudo de mecanismos de predição é de interesse estratégico para investidores, pesquisadores e participantes do mercado em geral. Tradicionalmente as análises técnica em que o investidor estuda as tendências de preços através de dados históricos e fundamental em que são utilizados dados correlacionados como clima, políticas e notícias são os principais recursos utilizados por operadores para prever valores de ações. Mais recentemente técnicas de aprendizagem de máquina vêm sendo estudadas em busca de resultados mais consistentes [Hegazy et al. \(2013\)](#).

1.2 Trabalhos Relacionados

Os estudos de predição de valores focam principalmente na busca por modelos mais adequados a tarefa. [Wang and Wang \(2015\)](#) realizou um estudo comparativo entre arquiteturas de rede neural na previsão da flutuação de preços do mercado de petróleo, seus estudos apontam que a nova arquitetura proposta ST-ERNN é capaz de obter resultados superiores se comparados aos modelos backpropagation neural network (BPNN) e Elman recurrent neural network (ERNN). [Patel et al. \(2015\)](#) propôs um método de previsão em duas etapas, cuja primeira etapa é composta por um modelo de support vector machine (SVM) e a segunda por modelos de redes neurais artificiais (RNAs), Random Forests (RF) e SVMs. Foram utilizados indicadores de análise técnica de mercado além dos dados da série histórica. Seus resultados demonstram o melhor desempenho da abordagem em duas etapas em detrimento de modelos simples. Focando no pré processamento dos dados e em previsões de curto prazo [Chong et al. \(2017\)](#) utilizou redes neurais profundas e métodos auto-regressivos em dados de frequência minuto a minuto para medir a flutuação de preço de títulos listados na bolsa de valores Coreana. Em seu trabalho foram utilizadas as técnicas: principal component analysis (PCA), autoencoder e Restricted Boltzmann machine (RBM) para reduzir a dimensionalidade do problema, seus resultados apontam para o benefício da utilização de técnicas de redução dimensional no problema de predição de valores no mercado financeiro. [Wang et al. \(2018\)](#) fez uso da análise de sentimento de notícias publicadas no The New York Times junto a dados históricos para prever valores na bolsa. Seus resultados apontam para uma redução no erro de predição (erro médio quadrático) quando a análise de sentimento é incorporada ao modelo.

1.3 Objetivo

Utilizando redes neurais convolucionais (CNN) [Lecun et al. \(1998\)](#), redes neurais de memória a longo prazo (LSTM) [Hochreiter and Schmidhuber \(1997\)](#) e dados históricos extraídos do Yahoo! Finances construiremos modelos capazes de prever o preço da empresa Apple na bolsa de valores.

Extrairemos indicadores técnicos de mercado e faremos uso dos algoritmos Random Forest, Lasso e Support Vector Regressor para determinar os atributos mais relevantes a tarefa proposta.

1.4 Contribuições

Construimos modelos de redes neurais com células de memória a longo prazo e redes neurais convolucionais capazes de prever valores no mercado de ações da empresa Apple. A metodologia de predição inclui técnicas de seleção de atributos baseadas nos algoritmos su-

pervisionados Random Forests, Lasso e Support Vector Machine.

Nossos resultados demonstraram a dificuldade de realizar esse tipo de previsão e as limitações do uso exclusivo de elementos da análise técnica de mercado na tarefa proposta.

2

Tecnologias e Conceitos

2.1 O mercado de ações

O mercado de ações é o conjunto de mercados nos quais ocorrem compras, vendas e emissões de ações de companhias abertas. Tais atividades acontecem através de trocas formais institucionalizadas ou mercados de balcão que possuem seu próprio conjunto de regras [Chen \(2019\)](#).

A abertura de capital permite que empresas arrecadem recursos para ampliar seus negócios, enquanto dá a chance de compradores adquirirem ações das mesmas, participando de seus lucros e eventuais prejuízos. As ações representam parcelas de propriedade da empresa, e são listadas em redes de trocas conhecidas como bolsas de valores. A predisposição de compradores e vendedores é o principal fator influenciador no valor dos títulos no mercado de ações. [Anna-Louise, et al. \(2019\)](#).

2.1.1 Análise técnica do mercado de ações

A análise técnica se refere ao estudo das características do próprio mercado, seus números e tendências e não de atributos relacionados aos bens que nele são transacionados. Esse tipo de análise leva em consideração o histórico de volume e preço de títulos para prever seu comportamento futuro. [Edwards and Magee \(2007\)](#) Indicadores técnicos são cálculos heurísticos ou matemáticos utilizados no processo de análise técnica do mercado de ações. [Chen \(2019\)](#)

A seguir destacamos alguns dos principais indicadores utilizados durante a análise técnica.

Volume

O volume representa a quantidade de transações de ações ou contratos de uma empresa realizadas em um dado período de tempo. O volume é um indicador importante na análise técnica de ações por identificar períodos de alta atividade de compradores e vendedores. [Murphy \(1999\)](#)

Preço

Por preço entendemos o preço de mercado de uma determinada ação em um dado período de tempo. Os preços geralmente possuem oscilações diárias devido a mudança na proporção de compradores e vendedores. Scilly (2019)

Média móvel

A média móvel (MA) é um indicador técnico constituído pela média simples dos valores nos últimos n dias, de forma que se quisermos calcular a MA de 30 dias dos valores de fechamento, devemos somá-los e dividí-los por 30. A média móvel é essencialmente um recurso para detecção de tendências e através dela podemos descobrir quando uma nova tendência começa e a antiga termina. Murphy (1999)

$$MA = (A_1 + A_2 + \dots + A_n) / n. \quad (2.1)$$

Sendo:

A_i - O valor da entrada no período n ;

n - O número de períodos contabilizados.

Média móvel com amortecimento exponencial

Derivada da média móvel comum, a média móvel com amortecimento exponencial (EMA) é obtida recursivamente, de maneira que seu valor em um dado período é calculada baseando-se na EMA dos períodos anteriores, o método também permite que o usuário determine a influência dos valores mais recente no cálculo da média, o que elimina o problema de igualdade de pesos presente na média móvel simples. Murphy (1999)

$$EMA_t = [V_t * s / (1 + n)] + EMA_y * [1 - s / (1 + n)] \quad (2.2)$$

$$n = 2 / (n + 1) \quad (2.3)$$

Sendo:

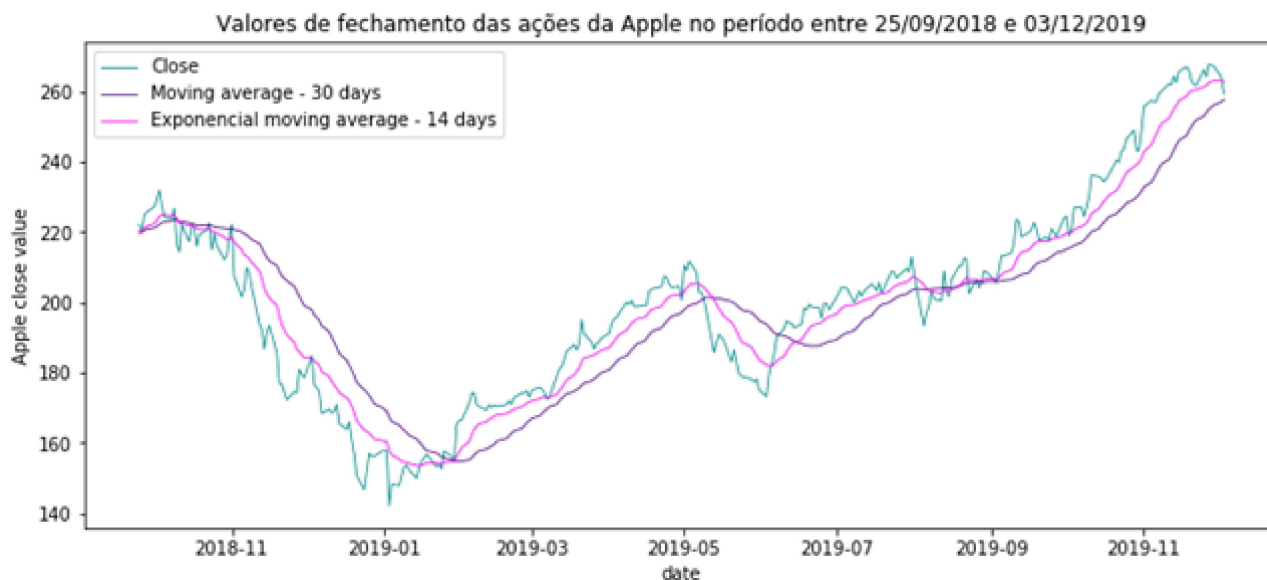
V_t = o valor atual;

s = o coeficiente de amortecimento;

n = o período;

EMA_y = a média móvel com amortecimento exponencial do dia anterior.

Figura 2.1: Gráfico gerado com valores de média móvel do preço das ações da empresa Apple.



Osciladores

Osciladores são indicadores utilizados como uma alternativa aos métodos de visualização de tendências, eles se mostram especialmente úteis quando o preço do título tende a variar de maneira horizontal, o que cria um contexto em que as técnicas baseadas em tendência como a média móvel se mostram pouco eficientes. As três situações mais relevantes no uso de osciladores são:

1. Quando os valores do oscilador estão próximos dos extremos, o que pode indicar uma situação de sobrevenda ou sobrecompra;
2. Quando existe divergência entre os valores mostrados pelo oscilador e o preço das ações;
3. Quando o valor do oscilador cruza o ponto médio, o que pode significar uma mudança na direção da tendência dos preços.

Quando um oscilador beira seu extremo superior, acredita-se que o título tenha sobrevalorizado e esteja vulnerável a uma liquidação, o que caracteriza uma situação de sobrecompra. A sobrevenda se dá quando o oscilador atinge o extremo mais baixo do gráfico, nessa situação é possível que o ativo tenha decaído tanto que esteja sujeito a um salto, uma inversão de tendência. [Murphy \(1999\)](#)

Momento

Pela segunda lei de Newton entendemos momento como sendo a quantidade de movimento de um corpo definida pelo produto da massa pela velocidade. O conceito de momento é um dos mais básicos dentro da análise de osciladores, e refere-se a análise da quantidade de variação do preço ao invés do preço em si, o momento do preço entre dois dias distintos pode ser calculado pela equação 2.4. [Scilly \(2019\)](#)

$$M = V - V_n \quad (2.4)$$

Sendo:

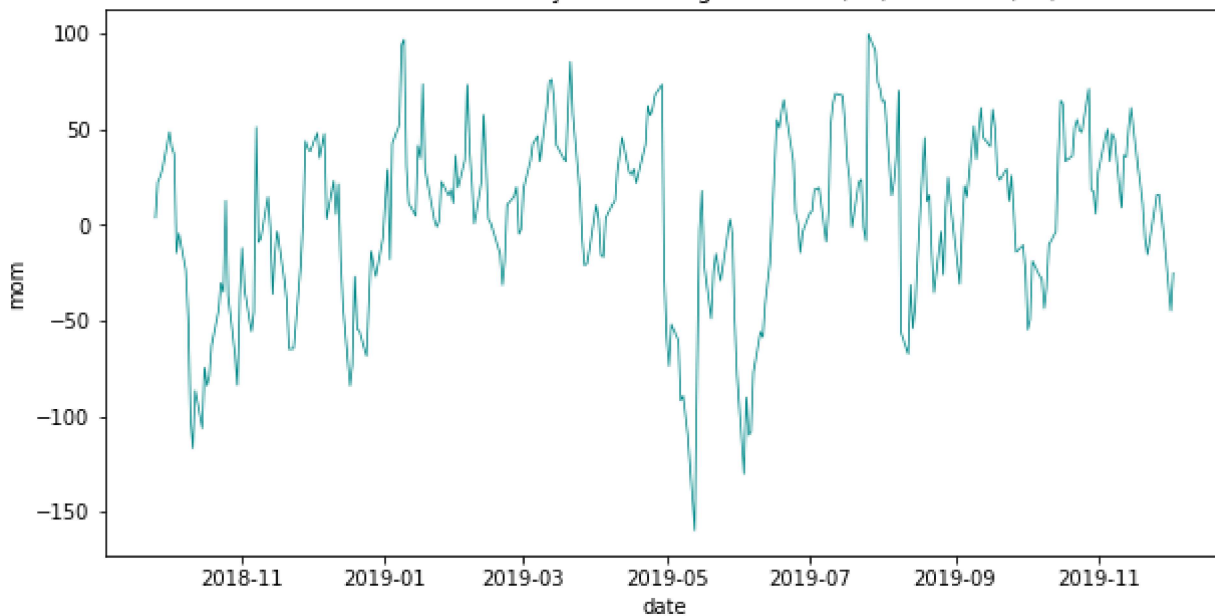
V = o preço mais recente;

V_n = o preço x dias antes.

Momento é um termo genérico que faz referência a uma gama de indicadores e existem essencialmente duas maneiras de interpretá-lo. A primeira é através da observação de séries individuais como a do preço e a segunda é através da manipulação estatística de diferentes componentes de mercado. [Pring \(2015\)](#)

Figura 2.2: Grafico gerado do momento das ações da companhia Google.

Oscilador de momento das ações da Google entre 25/09/2018 a 03/12/2019



Taxa de mudança (ROC)

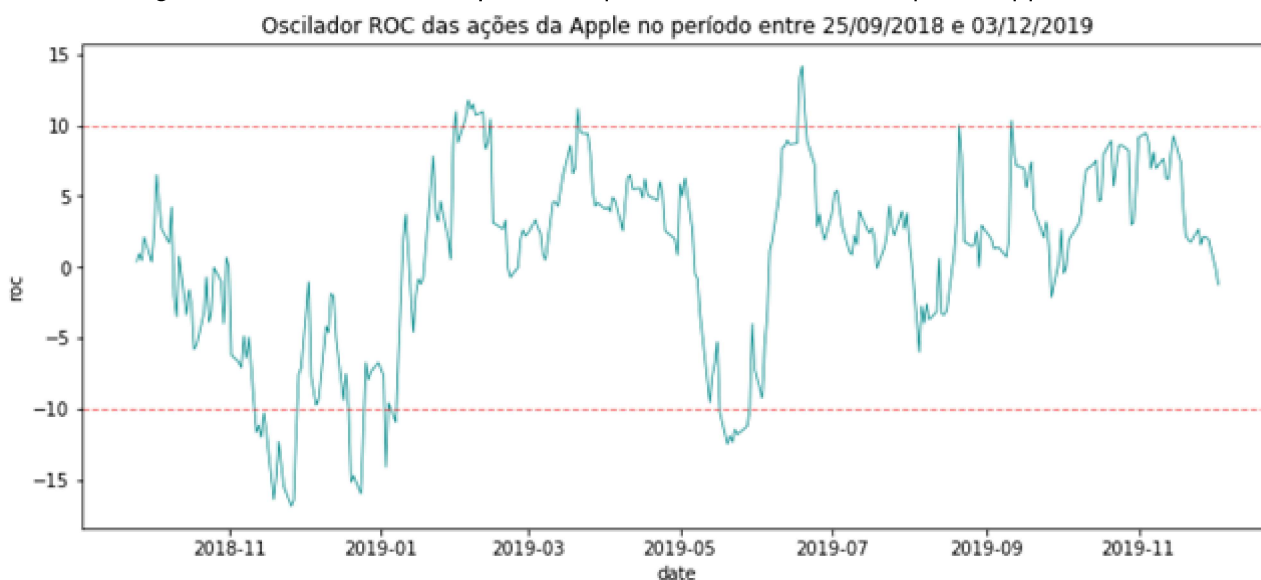
A Taxa de Mudança é um oscilador calculado através da diferença entre o preço de fechamento atual e o preço de fechamento de n períodos anteriores, gerando assim uma curva que ilustra a variação da taxa de mudança do preço em função do tempo. Os pontos do oscilador são plotados em formato de porcentagem, valores extremamente altos ou baixos podem ser indicativos de mudanças na tendência do preço. A equação 2.5 define o cálculo da taxa de mudança. [Pring \(2015\)](#)

$$ROC = [(Close_t - Close_{t-n}) / Close_{t-n}] * 100 \quad (2.5)$$

Sendo:

Close - O valor de fechamento da ação e consequentemente $Close_t$ o valor de no período t .

Figura 2.3: Oscilador ROC plotado a partir dos valores da empresa Apple.



Indicador de força relativa (RSI)

O indicador de força relativa (RSI) foi apresentado por Wells Wilder em 1978 e é um indicador que ilustra a força de um título em relação a ele mesmo. O RSI busca resolver o problema de movimento errático da linha de momento convencional (obtida da diferença entre dois valores). O cálculo do RSI é dado pela equação 2.6. Para obter a média dos n dias de fechamento em alta, devemos somar o valor ganho durante os dias de fechamento em alta dentro do intervalo de n dias e dividi-lo por n . Analogamente podemos calcular a média dos n dias de fechamento em baixa somando as perdas dos dias de fechamento em baixa e dividindo o valor obtido por n . O intervalo de tempo usual é de 14 dias, quanto menor o intervalo de tempo mais volátil se torna o oscilador. [Murphy \(1999\)](#)

$$RSI = 100 - 100/(1 + RS) \quad (2.6)$$

$$RS = MH/ML \quad (2.7)$$

Sendo:

MH - A média dos valores nos n dias com fechamento em alta;

ML - A média dos valores nos n dias com fechamento em baixa.

Para o período padrão de 14 dias valores de RSI menores do que 30 são indicadores de sobrevenda enquanto indicadores maiores do que 70 indicam uma situação de sobrecompra.

[Pring \(2015\)](#)

Figura 2.4: RSI plotado com os dados da Microsoft Corporation.

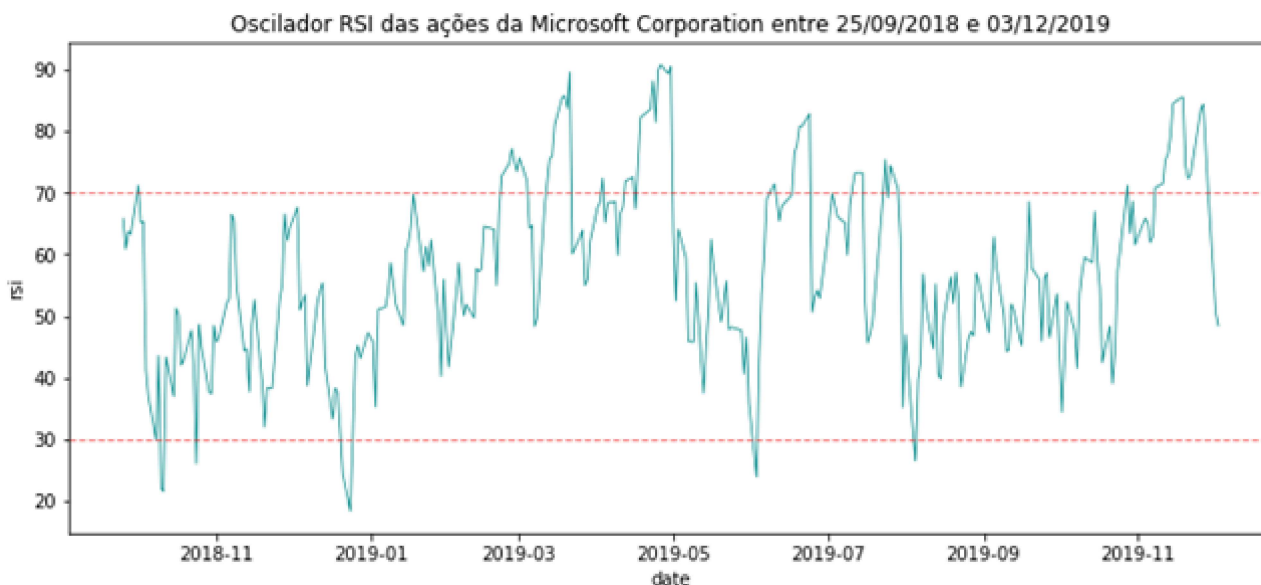
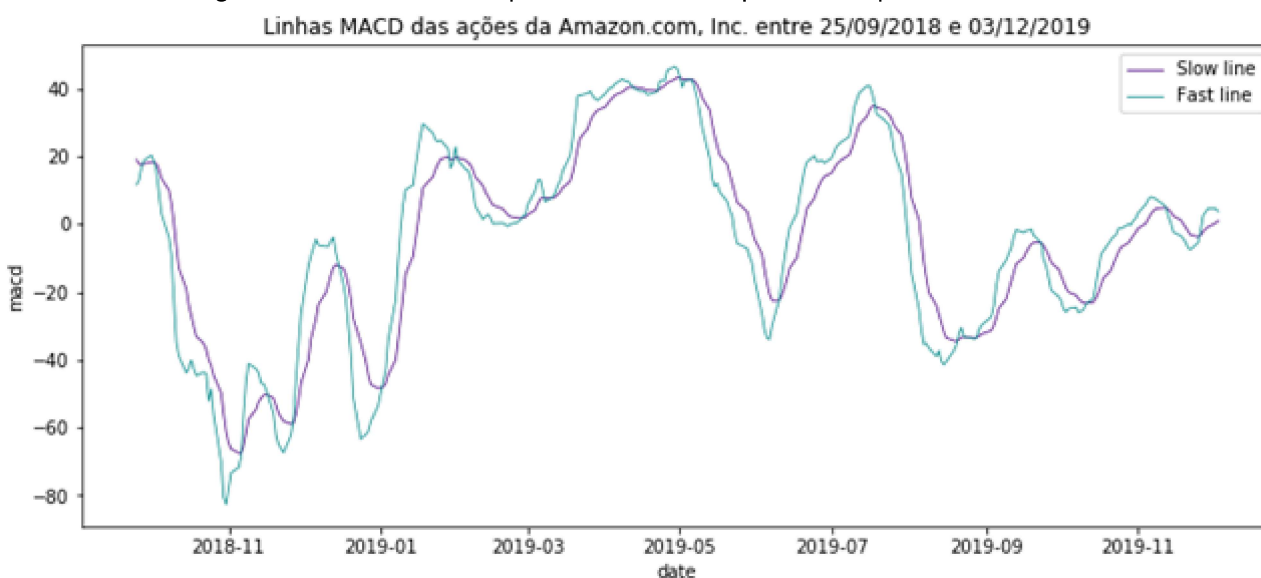


Figura 2.5: MACD da empresa Amazon no período especificado.



Convergência e divergência de média móvel (MACD)

A convergência e divergência de média móvel é uma métrica baseada em duas linhas. A linha rápida é a diferença entre duas EMA dos valores de fechamento com períodos de 12 e 26, enquanto a segunda é conhecida como linha lenta e é uma EMA com período de 9 da própria linha rápida. Os sinais de compra e venda são gerados a partir dos pontos de intersecção das duas linhas. Um cruzamento da linha MACD mais rápida acima da mais lenta é um sinal de compra. Um cruzamento da linha mais rápida abaixo da lenta é um sinal de venda. [Murphy \(1999\)](#)

o assunto, recomenda que sinais de compra sejam construídos em um gráfico diário com uma combinação de parâmetros 8, 17 e 9, porém ele sente que sinais de venda são mais confiáveis quando acionados com parâmetros 12, 25 e 9. Por outro lado, o popular programa MetaStock define os valores padrão como sendo 12 e 26 com a linha de sinal em 9. [Pring \(2015\)](#)

Estocástico

O estocástico é um oscilador popularizado por George Lane. O indicador é constituído por duas linhas assim como o MACD, a linha rápida %K e a linha lenta %D que é a média móvel com período igual a 3 de %K. Os limiares para sobrecompra e sobrevenda são 80 e 20 respectivamente. O sinal de compra é ativado quando %K ultrapassa %D e ambas estão abaixo de 20, enquanto o sinal venda é dado quando %D ultrapassa %K e ambas estão acima de 80. A linha %K pode ser calculada através da fórmula 2.8. [Murphy \(1999\)](#)

$$\%K = 100 * [(C - L14) / (H14 - L14)] \quad (2.8)$$

Sendo:

C = o último valor de fechamento;

L14 = o menor preço de baixa dos últimos 14 dias;

H14 = o maior preço de alta dos últimos 14 dias.

Figura 2.6: Oscilador estocástico gerado a partir dos registros da empresa Google.



2.2 Aprendizagem de máquina

Para resolver problemas utilizando computadores, fazemos uso de conjuntos de instruções que definem os passos para a realização de uma tarefa, os algoritmos. Os algoritmos de ordenação, por exemplo, ordenam sequências de dados através de um roteiro conhecido de operações e de fato diversos métodos já foram desenvolvidos para a execução dessa tarefa. Por outro lado, algumas tarefas como a detecção de spams na caixa de email não possuem algoritmo definido, ou seja: não existe uma sequência de operações programáveis capaz de avaliar efetivamente se um email recebido é legítimo ou não, e para tais problemas é desejável que o computador descubra algoritmos capazes de solucioná-los. [Alpaydin \(2010\)](#)

O termo aprendizagem de máquina ou aprendizagem automatizada faz referência ao ramo da inteligência artificial que estuda programas de computador capazes de aprender através de experiências. Processos de aprendizagem de máquina são comparáveis a processos no mundo animal onde organismos vivos resolvem problemas presentes utilizando conhecimento adquiridos no passado. [Shalev-Shwartz and Ben-David \(2014\)](#)

Quando ratos encontram alimentos com aparência ou cheiros novos, eles primeiro comem quantidades muito pequenas e a alimentação subsequente dependerá do sabor do alimento e de seus efeitos fisiológicas. Se o alimento produz um efeito prejudicial, ele frequentemente será associado com um mal-estar e, posteriormente, os ratos não a comerão. Claramente, existe um mecanismo de aprendizado acontecendo aqui - o animal usou experiências passadas com alguns alimentos para adquirir perícia na detecção da segurança destes. Se a experiência passada com a comida foi avaliada negativamente, o animal prevê que um efeito negativo também ocorrerá no futuro. [Shalev-Shwartz and Ben-David \(2014\)](#)

Programas de aprendizagem de máquina são úteis em casos onde a solução exata não é conhecida, a resolução por programação convencional é muito extensa ou uma solução adaptativa é necessária. [Géron \(2017\)](#)

Nas sessões subsequentes apresentaremos conceitos relacionados a problemas e algoritmos de aprendizagem de máquina.

2.2.1 Algoritmos supervisionados e tarefa de regressão

São chamados algoritmos de aprendizagem supervisionada aqueles responsáveis por prever uma variável alvo y através de variáveis de entrada x . Dentro das tarefas realizadas por modelos supervisionados a regressão busca encontrar uma relação entre os atributos x e uma variável de interesse numérica (real) y , para assim prever seus valores em amostras futuras. A predição de preços na bolsa a partir de dados históricos e indicadores técnicos é um exemplo de tarefa regressiva. [Shalev-Shwartz and Ben-David \(2014\)](#)

2.2.2 Função de perda

A função de perda ou função de utilidade é a métrica de desempenho do modelo proposto. Para tarefas de classificação por exemplo, a função de perda pode ser mais complexa do que a simples contabilização de instâncias classificadas incorretamente, poderíamos utilizar uma pontuação que atribuísse importâncias diferentes a falsos negativos e falsos positivos, o que pode fazer mais sentido em alguns problemas. [Bishop \(2016\)](#)

2.2.3 Overfitting

Na aprendizagem supervisionada o problema de overfitting se dá quando o modelo é incapaz de , generalizar o conhecimento através dos dados conhecidos, demonstrando ótimo desempenho em entradas já conhecidas e performance ruim em exemplos desconhecidos. O overfitting é um dos principais problemas relacionados a esse tipo de modelo. [Ying \(2019\)](#)

2.2.4 Seleção de atributos

Durante a resolução de problemas com aprendizado de máquina é preciso escolher quais variáveis serão utilizadas para alimentar o modelo, alguns atributos podem não ser significativos na resolução do problema e estes tendem a introduzir ruídos na solução. Adicionalmente a redução da dimensionalidade auxilia o modelo preditivo a não sofrer por overfitting. Conhecimentos do domínio do problema podem ser utilizados para escolher os atributos mais relevantes assim como soluções computacionais. [Guyon and Elisseeff \(2003\)](#)

Na seleção conjunta um algoritmo supervisionado é utilizado para atribuir rankings ou pontuações aos atributos de entrada de acordo com a sua importância. Enquanto na seleção iterativa um modelo supervisionado é treinado iterativamente com diferentes subgrupos de atributos em busca da combinação com melhor desempenho em relação a função de objetivo. [Bachu and Anuradha \(2019\)](#)

2.2.5 Árvore de decisão

Árvores de decisão são preditores que representam a relação entre as variáveis de entrada X e os objetivos Y através de um conjunto de regras arranjado em forma de árvore. No processo de treinamento, cada regra criada divide os dados em parcelas menores que serão divididas novamente até que o conjunto gerado seja totalmente puro ou uma outra condição de parada seja satisfeita. [Shalev-Shwartz and Ben-David \(2014\)](#)

O C4.5 é um algoritmo gerador de árvores de decisão proposto por Ross Quinlan em 1993 e é uma versão melhorada de outro algoritmo de mesma finalidade, o ID3. O C4.5 se difere do ID3 por sua habilidade em operar com valores contínuos, restrição do ID3. [Hssina et al. \(2014\)](#)

Figura 2.7: Pseudo-código do algoritmo C4.5 baseado em [Quinian \(1993\)](#)

Função C4.5:

Entrada: Conjunto de dados S , com atributos R

Árvore = {}

Se D for puro ou outra condição de parada for satisfeita

terminar

Para todos os atributos R em S fazer

calcular $G(R)$ e guardar em α atributo que gera maior ganho de informação

Árvore = Nó de decisão que separa as amostras de acordo com o valor de α

S' = Subconjuntos gerados pela separação causada por α

Para cada subconjunto em S'

Árvore' = C4.5 (subconjunto)

Anexar Árvore' ao ramo correspondente em Árvore

Retornar Árvore

Dado um conjunto de dados S e seus atributos R , o algoritmo utiliza uma métrica G de ganho de informação para obter o atributo α que proporciona a melhor separação dos dados. Uma nova regra é adicionada a árvore e o conjunto S é então particionado em conjuntos S' de acordo com os valores de α . A função é chamada recursivamente para cada um dos subconjuntos enquanto as árvores geradas se tornam ramos da árvore principal. Para tarefas de regressão (predição de valores reais) o erro médio quadrático é uma escolha comum para função G . [Raileanu and Stoffel \(2004\)](#)

2.2.6 Random Forests

A Random Forest é uma técnica de aprendizado conjunto, que utiliza múltiplas árvore de decisão treinadas com subconjuntos distintos dos dados de treinamento. Os subconjuntos são amostrados com reposição seguindo uma mesma distribuição aleatória. Para tarefas de classificação a variável de interesse pode ser determinada por um processo de votação, enquanto para tarefas regressivas o veredicto é usualmente extraído através média dos resultados. Por definição, Random Forests são uma implementação modificada do algoritmo de Bagging utilizando árvores de decisão. [Breiman \(2001\)](#)

O algoritmo de Bagging foi proposto em 1996 por Breiman, sendo um dos primeiros e mais intuitivos métodos de aprendizado conjunto. Dado um conjunto de treino com M amostras, o

algoritmo proposto por Breiman treina de maneira paralela modelos do mesmo algoritmo utilizando subconjuntos da partição de treino original, os subconjuntos tem tamanho fixo N e as amostras são extraídas através de um processo aleatório de amostragem com reposição. Em problemas de classificação a solução final é dada pela resposta mais recorrente entre as retornadas pelos modelos, enquanto em problemas de regressão a solução é o valor médio de todas as respostas. O método de Bagging funciona melhor com algoritmos mais sensíveis a mudanças nos dados de entrada, nesses casos o método será capaz de produzir hipóteses mais diversificadas. [Arbib \(2003\)](#)

Figura 2.8: Pseudocódigo do algoritmo de Random Forest baseado em [Breiman \(2001\)](#)

```

Função Criar Random Forest:
Entrada: Conjunto de dados  $S$ , número de estimadores  $N$ 
Floresta = {}
 $i = 0$ 
Enquanto  $i < N$ , fazer
     $S'$  = subconjunto extraído aleatoriamente de  $S$ 
     $T$  = árvore gerada a partir de  $S'$ 
    Inserir  $T$  em Floresta
     $i = i + 1$ 
Retornar Floresta
  
```

2.2.7 Lasso

Lasso (least absolute shrinkage and selection operator) é uma técnica de regressão linear com característica de estreitamento que reduz a zero os coeficientes relacionados a atributos menos importantes, retornando modelos mais interpretáveis e com menor variância. O modelo pode ser descrito pela equação 2.9 onde o parâmetro t é um regulador que limita a magnitude dos coeficientes β . O Lasso é uma técnica linear e pode ser solucionada por algoritmos de otimização computacional. [Tibshirani \(1996\)](#)

$$(\hat{\alpha}, \hat{\beta}) = \underset{\alpha, \beta}{\operatorname{argmin}} \sum_{i=1}^N (y_i + \alpha - \sum_j \beta x_{ij})^2 \quad \text{para} \quad \sum_j |\beta_j| \leq t \quad (2.9)$$

Sendo:

α = bias da função linear;

β_j = coeficiente associado a variável j ;

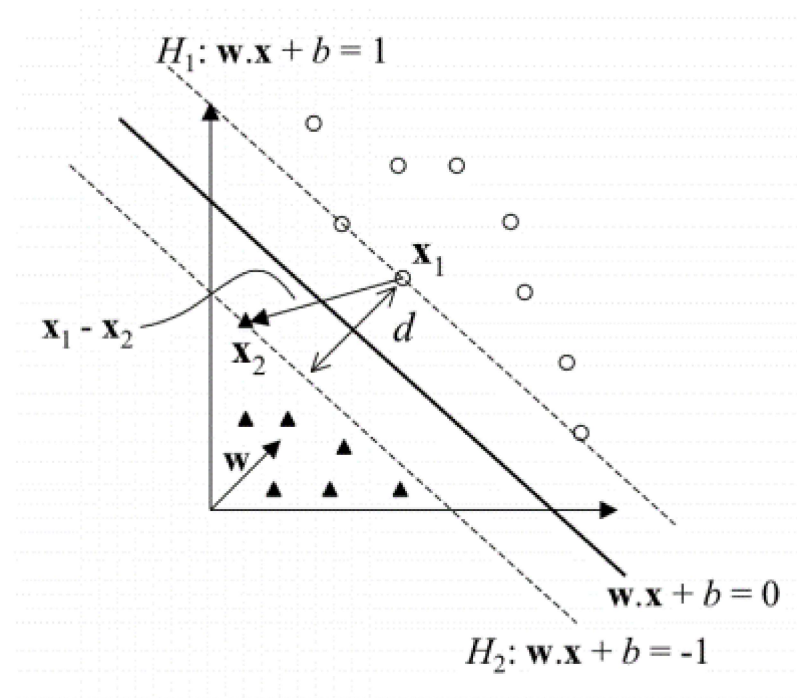
x = atributos;

y = variáveis de interesse;

t = termo de regularização de estreitamento.

2.2.8 Support vector machines (SVM)

Figura 2.9: Objetos de classes distintas separados pelo hiperplano H . [Lorena and de Carvalho \(2007\)](#)



As SVMs são algoritmos de aprendizado supervisionado que utilizam kernels para maximizar a distância entre amostras de categorias distintas. Os kernels lineares são definidos pela equação 2.10. [Lorena and de Carvalho \(2007\)](#)

$$f(x) = wx + b \quad (2.10)$$

Para tarefas de regressão, o algoritmo busca minimizar a distância entre y e $f(x)$, procurando um hiperplano tal que $|y - f(x)| = 0$, caso tal plano não exista é então feita uma flexibilização tal que a função de perda seja definida por 2.11.

$$f(n) = \begin{cases} 0, & \text{if } |y - f(x)| \leq \varepsilon \\ |y - f(x)| - \varepsilon, & \text{caso contrário} \end{cases} \quad (2.11)$$

Onde ε é a tolerância de erro do algoritmo. [Mathworks \(2020\)](#)

2.2.9 Redes Neurais

As redes neurais artificiais (RNAs) são modelos computacionais inspirados no cérebro humano, elas são compostas por uma coleção de modelos mais simples (neurônios) conectados por uma rede capaz de resolver problemas complexos. As RNAs foram propostas em meados do século XX e vêm obtendo sucesso na resolução de problemas no campo da inteligência artificial, seu formato pode ser ilustrado por grafos direcionados em que os nós ou neurônios são os vértices e as ligações são as arestas. O tipo mais básico de RNA, o feedforward não possui ciclos e foi o primeiro a ser desenvolvido, subsequentemente surgiram diferentes variações de topologia especializadas em resolver diferentes tipos de problemas. [Shalev-Shwartz and Ben-David \(2014\)](#)

Neurônios

De um ponto de vista biológico os neurônios são células que constituem o cérebro e se conectam de maneira a formar uma rede responsável pelas funções neurológicas dos organismos vivos, seus receptores são alimentados por impulsos provenientes dos ambientes internos e externo, enquanto seus transmissores são responsáveis por propagar a informação através da rede. No processo de propagação dos pulsos certos caminhos entre neurônios podem ser ativados ou desativados por reações químicas que acontecem com mais ou menos frequência a depender da experiência do indivíduo. McCulloch e Pitts propuseram um modelo de neurônio computacional que realiza a soma ponderada das entradas e ativa ou não a saída de acordo com um limiar estabelecido. Este modelo inicial foi generalizado de diversas formas, principalmente pela utilização de funções de ativação no lugar de um limiar. [Jain et al. \(1996\)](#)

Redes feedforward

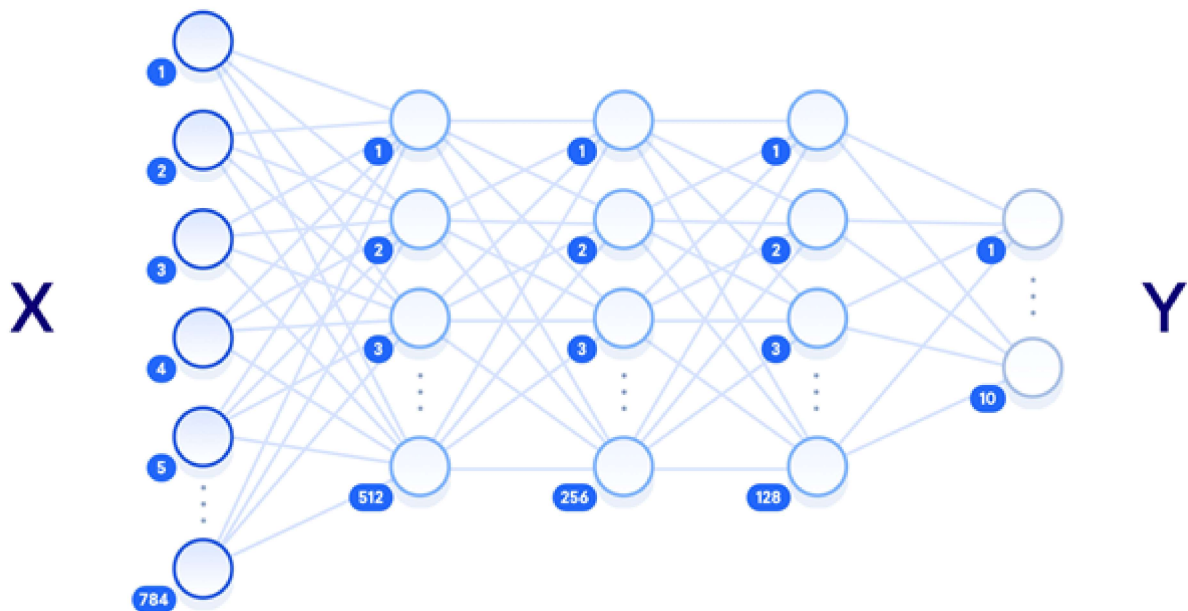
A idéia por trás do funcionamento de uma rede neural é de que diversos neurônios combinem suas saídas de maneira a encontrar relações representativas entre dados de entrada e as saídas fornecidas. Redes neurais feedforward podem ser entendidas como grafos direcionais sem ciclos, conjuntos de neurônios divididos em camadas de maneira que as saídas dos neurônios da camada anterior atuam como entradas nos neurônios da camada subsequente. Na estrutura descrita a primeira camada é chamada de camada de entrada, a última de camada de saída e as intermediárias de camadas ocultas. [Shalev-Shwartz and Ben-David \(2014\)](#)

Para que as RNAs possam ser utilizadas na prática se faz necessário um método de atualização dos pesos da rede. Em 1986 D. E. Rumelhart et al., propôs o backpropagation, uma técnica que possibilitou o treinamento de redes neurais feedforward. Para cada uma das instâncias de treinamento, o backpropagation primeiramente calcula o erro na camada de saída através de uma função de perda e analisa a contribuição de cada uma das ligações pelo cálculo do gradiente da função de ativação, o algoritmo então se repete para os neurônios da camada

Figura 2.10: Ilustração de rede neural biológica. Freepik (2020)



Figura 2.11: Ilustração de uma rede neural profunda. DigitalOcean (2018)



anterior em um processo de retropropagação do erro. Os pesos das ligações são atualizados de acordo com o inverso do gradiente obtido, numa técnica conhecida como Gradiente Descendente. É importante ressaltar que o algoritmo descrito anteriormente somente se aplica a redes com função de ativação deriváveis. Géron (2017)

De acordo com Arbib (2003) para uma rede feedforward com k variáveis alvo e função de erro definida por $E = \sum_k (t_k - o_k)^2$, a atualização de um peso w_{ij} é dada pela equação 2.12:

$$\Delta w_{ij} = 2 * \sum_k (t_k - o_k) \partial o_k / \partial w_{ij} \quad (2.12)$$

Sendo:

t_k = o valor real da variável de interesse;

o_k = o valor previsto pelo modelo;

$\partial o_k / \partial w_{ij}$ = a derivada parcial de o_k em função do peso w_{ij} .

Redes neurais recorrentes e LSTMs

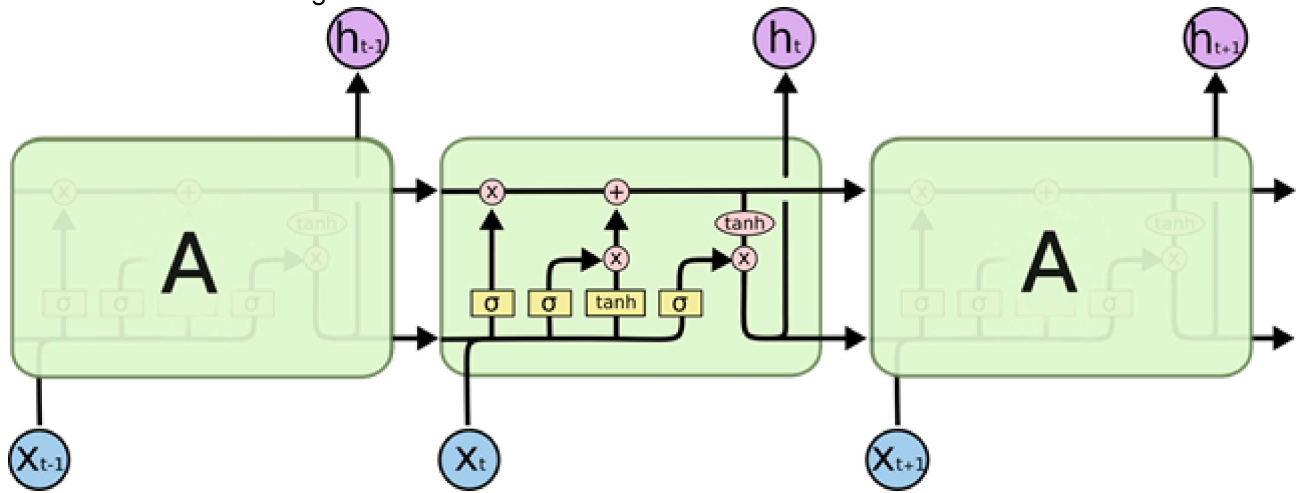
As redes neurais feedforward avançaram o estado da arte da aprendizagem de máquina encontrando aplicações em diversas áreas do conhecimento, porém, o modelo de uma RNA sem ciclos funciona apenas como uma função de mapeamento não linear entre um conjunto de entradas x e variáveis de interesse y . No mundo real os seres humanos tomam decisões baseadas em informações disponíveis no presente e em experiências de aprendizagem passadas, com o intuito de modelar essa capacidade e expandir o poder das redes neurais surge o modelo recorrente. [Arbib \(2003\)](#)

O modelo recorrente opera de maneira semelhante a uma rede feedforward, com a diferença de que a camada oculta para além dos impulsos da camada de entrada também recebe as saídas de suas próprias funções de ativação na interação anterior. Esta pequena mudança possibilita que as redes recorrentes operem com o conceito de memória e assim trabalhem de maneira mais eficiente com dados históricos. [Graves et al. \(2014\)](#)

Apesar de terem bom desempenho em aplicações sequenciais os dados na memória das redes recorrentes comuns são gradualmente deteriorados pelas consecutivas transformações e a cada interação a contribuição de contextos mais antigos se torna menos relevante, isso faz com que relações de dependência a longo prazo não sejam representadas eficientemente. Propostas por Sepp Hochreiter em 1997 as células de memória a longo prazo ou long short-term memory (LSTM) são unidades de memória capazes guardar informações de contexto por longas sequências de interações, sendo uma melhoria natural das redes recorrentes. [Géron \(2017\)](#)

A chave para o funcionamento das células de memória de longo prazo é o mecanismo de estados, representado pelo barramento mais superior na figure 2.12. Para além do vetor h_t de resultados, cada célula também passa a frente seu vetor de estado C_t que determina quais informações devem ser mantidas ou descartadas durante a execução, a cada interação t o valor de C é atualizado por uma rede interna que tem como entradas h_{t-1} e x_t . O resultado h_t é determinado por uma subrede que tem h_{t-1} , x_t e C_t como entradas. [Hochreiter and Schmidhuber \(1997\)](#)

Figura 2.12: Encadeamento de células LSTM. Olah



Redes Convolucionais

Usadas principalmente no processamento de imagens e sinais de áudio, camadas convolucionais realizam operações de convolução através de dados organizados em formatos de matriz, transmitindo o resultado da operação para a camada seguinte, tal abordagem potencialmente diminui a quantidade de variáveis de entrada e capta relações de dependência entre variáveis próximas. Uma camada convolucional é definida pelas dimensões do máscara aplicada a matriz, o número de filtros e o passo do filtro. Cada filtro representa um filtro espacial aplicado a convolução, diferentes filtros tendem a ajudar a rede a perceber diferentes padrões no problema. [Albawi et al. \(2017\)](#)

Camadas de Dropout

Para tratar o problema de overfitting em redes neurais [Srivastava et al. \(2014\)](#) propôs uma camada intermediária conhecida como Dropout, a camada de Dropout ativa ou desativa a saída de um neurônio de acordo com uma probabilidade p . A ideia da abordagem é desmotivar interdependências durante o processo treinamento e assim melhorar a capacidade generalização do modelo. [Srivastava et al. \(2014\)](#)

Otimizador Adam

O algoritmo Adam é uma alternativa ao gradiente descendente estocástico comum, proposto por P. Kingma e Lei Ba em 2015, a técnica atribui diferentes taxas de aprendizagem não fixas para cada variável. De maneira mais específica, o algoritmo computa a média móvel exponencial de cada gradiente e de seu quadrado. [Kingma and Ba \(2017\)](#)

Figura 2.13: A figura (b) ilustra a aplicação de Dropout em uma rede neural comum, cada um dos componentes r é responsável por ativar ou não as saídas y da camada interior. [Srivastava et al. \(2014\)](#)

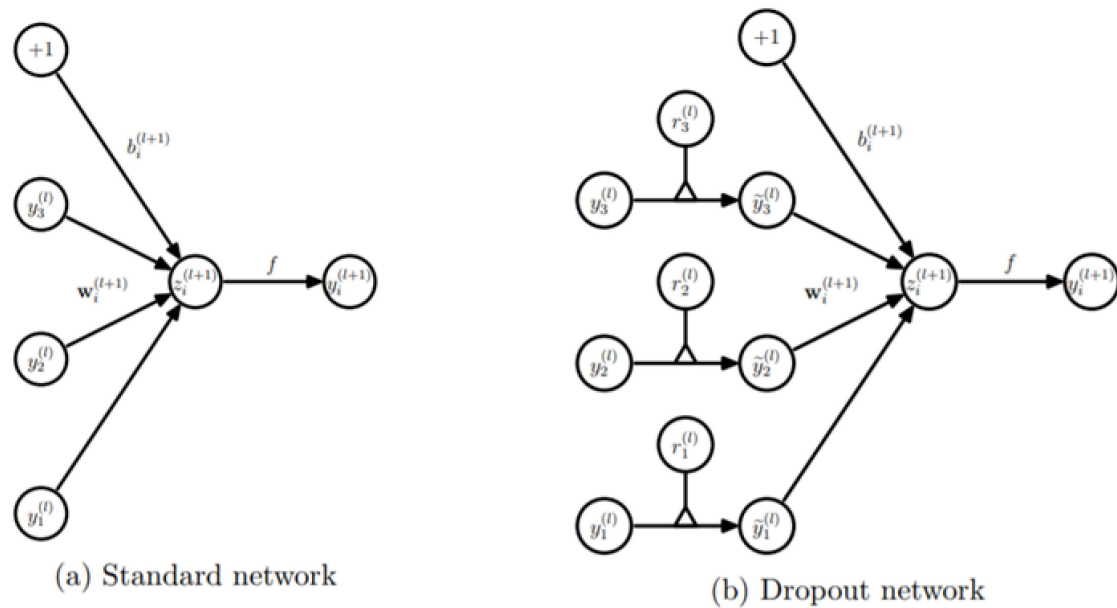


Figura 2.14: Pseudo-código do otimizador Adam. [Kingma and Ba \(2017\)](#)

Require: α : Stepsize

Require: $\beta_1, \beta_2 \in [0, 1)$: Exponential decay rates for the moment estimates

Require: $f(\theta)$: Stochastic objective function with parameters θ

Require: θ_0 : Initial parameter vector

$m_0 \leftarrow 0$ (Initialize 1st moment vector)

$v_0 \leftarrow 0$ (Initialize 2nd moment vector)

$t \leftarrow 0$ (Initialize timestep)

while θ_t not converged **do**

$t \leftarrow t + 1$

$g_t \leftarrow \nabla_{\theta} f_t(\theta_{t-1})$ (Get gradients w.r.t. stochastic objective at timestep t)

$m_t \leftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$ (Update biased first moment estimate)

$v_t \leftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$ (Update biased second raw moment estimate)

$\hat{m}_t \leftarrow m_t / (1 - \beta_1^t)$ (Compute bias-corrected first moment estimate)

$\hat{v}_t \leftarrow v_t / (1 - \beta_2^t)$ (Compute bias-corrected second raw moment estimate)

$\theta_t \leftarrow \theta_{t-1} - \alpha \cdot \hat{m}_t / (\sqrt{\hat{v}_t} + \epsilon)$ (Update parameters)

end while

return θ_t (Resulting parameters)

3

Nossa proposta

Neste trabalho propomos uma metodologia de predição de valores da bolsa de valores baseada em técnicas de aprendizagem e máquina.

3.1 Aquisição dos dados

Foram coletados dados históricos da cotação da empresa Apple utilizando o portal especializado Yahoo! Finances [Yahoo]. Coletamos registros do período de 31/08/10 a 31/08/20, totalizando 2519 amostras. Os atributos registrados estão listados na tabela 3.1.

3.2 Processamento preliminar dos dados

Utilizando a linguagem de programação Python e o pacote [Pandas](#), carregamos os arquivos no programa no formato Dataframe que auxilia a manipulação dos dados. A seguir, utilizamos o pacote Financial Technical Analysis (FINTA) para calcular os indicadores de análise técnica de mercado: média móvel (SMA), média móvel com amortecimento exponencial (EMA), momento

Tabela 3.1: Atributos iniciais.

Atributo	Descrição
Open	valor de abertura da ação
Close	valor fechamento da ação
High	maior valor alcançado durante o dia
Low	menor valor alcançado durante o dia
Volume	volume de transações
Adj. Close	valor de fechamento ajustado
Date	data em que a amostra foi coletada

Tabela 3.2: Lista de atributos após a adição de indicadores técnicos de mercado.

Atributo	Descrição	Tipo
Close	valor fechamento da ação	numérico
High	maior valor alcançado durante o dia	numérico
Low	menor valor alcançado durante o dia	numérico
Volume	volume de transações	numérico
Sma(15,30,45,60)	média móvel simples com períodos de 15,30,45 e 60	numérico
Ema(15,30,45,60)	média móvel exponencial com períodos de 15,30,45 e 60	numérico
Mom	curva de momento	numérico
Roc	oscilador ROC	numérico
Rsi	oscilador RSI	numérico
Macd	linha rápida do oscilador MACD	numérico
macd-signal	linha lenta do oscilador MACD	numérico
stoch-k	linha %k do oscilador estocástico	numérico
stoch-d	linha %d do oscilador estocástico	numérico

(MOM), estocástico (STOCH), convergência e divergência de média móvel (MACD), indicador de força relativa (RSI) e taxa de mudança (ROC). Para as variáveis SMA e EMA foram utilizados valores de $n = [15, 30, 45, 60]$, todos os outros indicadores não necessitaram de parâmetros.

A seguir, separamos os dados do período entre 31/08/10 e 12/08/2019 como grupo de treinamento e os entre 13/08/2019 e 31/08/20 como teste.

Os valores foram então escalados de acordo com a equação 3.1, o escalonamento de variáveis frequentemente melhora a convergência de técnicas de aprendizagem de máquina pois através dele o algoritmo é capaz de entender que inicialmente todas as variáveis possuem a mesma importância ou magnitude.

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (3.1)$$

Sendo:

X_{scaled} = o valor escalado;

X = valor da variável;

X_{min} = o menor valor no eixo;

X_{max} = o maior valor no eixo.

Uma vez escalados os valores, a variável de interesse (o valor de fechamento da ação no dia seguinte) é separada em um vetor Y , que servirá como referência alvo para os algoritmos supervisionados.

Tabela 3.3: Parâmetros do modelo de SVR

Parâmetro	Valor	Descrição
Kernel	Linear	O tipo de Kernel utilizado
Tol	0.001	Tolerância para a condição de convergência
C	1	Peso do fator de regularização
Epsilon	0.1	O valor da variável ϵ no algoritmo de SVR
Shinking	True	Se deve ser utilizada uma heurística de estreitamento
Verbose	0	Nível de informação exibido durante o treinamento do modelo

3.3 Seleção de atributos

Utilizamos os algoritmos supervisionados Support Vector Regression (SVR), Random Forest (RF) e Lasso para gerar seleções de atributos para o modelo preditivo.

3.3.1 Support Vector Regression

Foi criado um modelo de Support Vector Regressor de Kernel linear utilizando o pacote [Scikit-learn](#) segundo os parâmetros da tabela 3.3.

Assim como o recomendado em [Guyon and Elisseeff \(2003\)](#) foi feita uma seleção recursiva de atributos, mantendo o conjunto de 10 atributos com o melhor desempenho.

3.3.2 Random Forest

Utilizando o pacote [Scikit-learn](#) criamos um modelo de regressor Random Forest, com os parâmetros especificados na tabela 3.4. O modelo supervisionado utilizou o valor futuro da variável `close` como alvo.

Selecionamos as todas as variáveis com relevância acima da mediana das pontuações geradas pelo algoritmo, assim como estabelecido pelo padrão do método "Select From Model" da biblioteca [Scikit-learn](#).

3.3.3 Lasso

Também utilizando o pacote [Scikit-learn](#), geramos um modelo de regressor Lasso segundo os parâmetros especificados na tabela 3.5. Os atributos com maior coeficiente compuseram o conjunto 'Lasso', todos os atributos com pontuação menor do que $1e-5$ foram desconsiderados seguindo o padrão do método "Select From Model" do pacote [Scikit-learn](#).

Tabela 3.4: Parâmetros do modelo de RF.

Parâmetro	Valor	Descrição
n_estimators	150	Número de árvores na floresta
Criterion	MSE	A função de ganho de informação
max_depth	None	Profundidade máxima dos nós nas árvores
min_samples_split	2	Número mínimos de amostras para que seja feita uma separação
min_samples_leaf	1	Número mínimo de amostras para que uma folha seja considerada válida
min_weight_fraction_leaf	0	O peso relativo mínimo para que uma folha seja considerada válida
max_features	Auto	Número máximo de atributos a serem considerados no processo de geração de folhas
max_leaf_nodes	None	Número máximo de folhas em cada árvore
min_impurity_decrease	0	Ganho mínimo de informação para que uma separação seja feita
min_impurity_split	None	Limiar da função de ganho de informação para que a árvore pare de crescer
Bootstrap	True	Utilização da técnica de bootstrap
oob_score	False	Utilização de amostras fora da bolsa de bootstrap para avaliar a performance em amostras não vistas
n_jobs	None	Número de tarefas do algoritmo rodando em paralelo
random_state	None	Semente de aleatoriedade para o algoritmo
Verbose	0	Nível de informação exibido durante o treinamento do modelo
warm_start	False	Utilização do resultado do ultimo treinamento como estado inicial do treinamento seguinte
ccp_alpha	0	Valor da constante de regularização CCP
max_samples	None	O número de amostras a ser utilizado no algoritmo de bootstrap

Tabela 3.5: Parâmetros do modelo de Lasso.

Parâmetro	Valor	Descrição
alpha	0.01	Valor da constante alpha
fit_intercept	True	Define se a distribuição será considerada centralizada ou não
normalize	False	Se o modelo normalizará as entradas automaticamente
precompute	False	Se o modelo usará a matrix de Gram pre-calculada para acelerar o desempenho
copy_x	True	Se a entrada X será copiada, caso contrário a variável original pode ser substituída
max_iter	1000	Número máximo de interações
tol	0.0004	Otimização mínima para parar de iterar
warm_start	False	Utilização do resultado do ultimo treinamento como estado inicial do treinamento seguinte
positive	False	Se os coeficientes devem ser todos positivos
random_state	None	Semente de aleatoriedade para o algoritmo
selection	Cyclic	Método de seleção de coeficientes a serem atualizados (aleatório ou cíclico).

Tabela 3.6: Formatação da entrada para formato supervisionado.

Entrada N			
Atributo(1) Entrada(N-1)	Atributo(1) Entrada(N-2)	...	Atributo(1) Entrada(N -tamanho_da_janela)
Atributo(2) Entrada(N-1)	Atributo(2) Entrada(N-2)	...	Atributo(2) Entrada(N -tamanho_da_janela)
...
Atributo(n_de_atributos) Entrada(N-1)	Atributo(n_de_atributos) Entrada(N-2)	...	Atributo(n_de_atributos) Entrada(N -tamanho_da_janela)

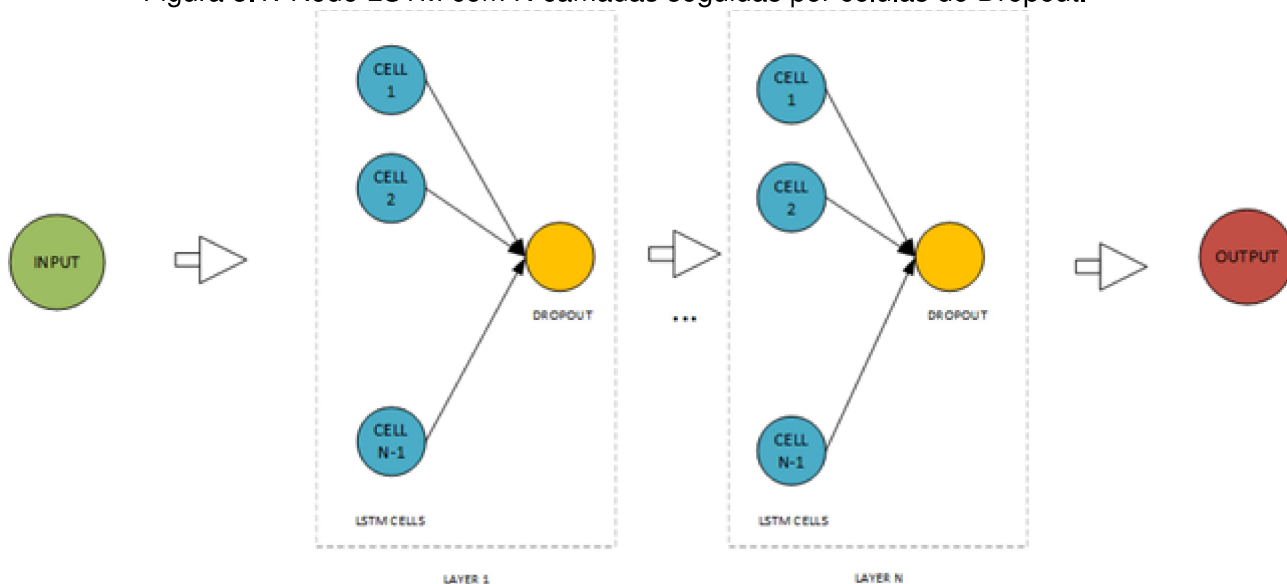
3.4 Conversão dos dados para o formato supervisionado

Para alimentar a rede de memória a longo prazo, convertemos nossos dados de maneira que para cada saída ou dia registrado houvessem $TamanhoDaJanela * NDeAtributos$ entradas. De maneira que a formatação da entrada para uma dada saída N ficasse na tabela 3.6.

3.5 Rede LSTM

Com o intuito de prever os valores da bolsa de valores da empresa Apple, criamos um modelo de rede neural de memória de longo prazo (LSTM) utilizando a biblioteca [Keras](#). Seguindo a topologia ilustrada na imagem 3.1.

Figura 3.1: Rede LSTM com N camadas seguidas por células de Dropout.



O número de camadas, a taxa de dropout, o número de neurônios por camada e o tamanho da janela de entrada foram determinados seguindo um processo de busca aleatória. Para cada um dos conjuntos de atributos gerados foi gerada uma coleção de parâmetros.

A função de ativação usada foi a tangente hiperbólica e as demais variáveis de configuração foram mantidas fixas no valor estabelecido pela biblioteca, para a atualização dos pesos da rede usamos o otimizador adam. Para cada um dos conjuntos a rede foi treinada por 100 interações.

Tabela 3.7: Parâmetros de criação da rede LSTM por conjunto de atributos.

Conjunto	Número de camadas	Neurons por Camada	Taxa de Dropout	Tamanho da janela
SVR	1	200	0.2	90
Random Forest	1	25	0.2	30
LASSO	1	25	0.2	90

Tabela 3.8: Parâmetros de configuração do otimizador adam.

Parâmetro	Valor	Descrição
learning_rate	0.001	Valor do passo.
beta_1	0.9	Parâmetro β_1 do algoritmo Adam.
beta_2	0.999	Parâmetro β_2 do algoritmo Adam.
epsilon	1e-07	Constante adicionada para estabilidade numérica.

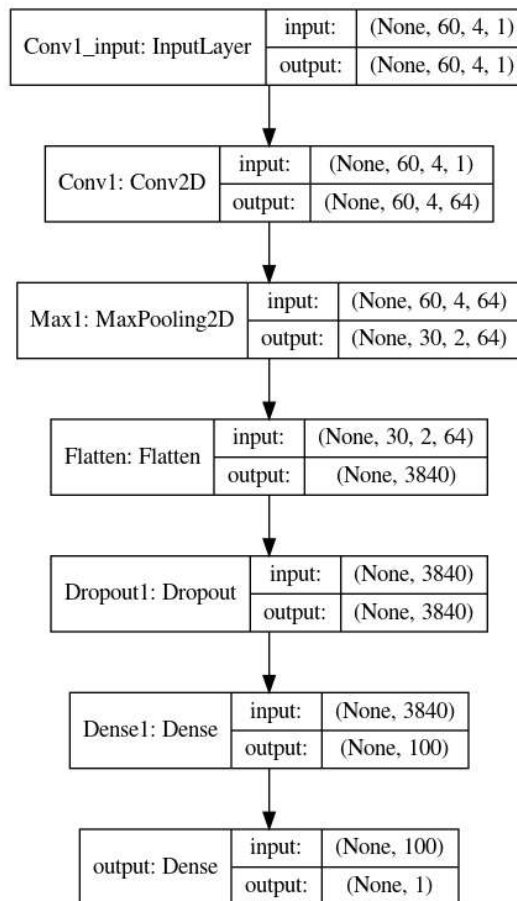
3.6 Rede Neural Convolutiva

Construímos um modelo de rede neural convolutiva com o intuito de prever os valores da empresa no mercado de ações utilizando a capacidade das camadas convolutivas de captar interdependências nos dados de entrada. A rede foi construída com a biblioteca [Keras](#) e segue a topologia explicitada na [figure 3.2](#).

Analogamente a rede LSTM, utilizamos um processo de busca aleatória para determinar os melhores parâmetros para o modelo, os parâmetros podem ser vistos na [tabela 3.9](#). O otimizador utilizado foi o adam que também segue as configurações especificadas na [tabela 3.8](#). Para cada um dos conjuntos de atributos a rede foi treinada por 100 iterações.

Tabela 3.9: Parâmetros de configuração para as camadas da rede convolutiva.

Camada	Parâmetro	Valor	Descrição
Conv1	filters	64	Quantidade de filtros aplicados durante a convolução
	padding	same	Determina se a entrada deve ser preenchida com zeros para se tornar um múltiplo exato do kernel
	kernel_size	(5,5)	Tamanho do Kernel de Convolução
Max1	pool_size	(2,2)	Tamanho do kernel para que seja aplicada a função Max
Dropout	rate	0.2	Probabilidade de desativar a saída de um neurônio.
Dense1	neurons	100	Quantidade de neurônios na camada

Figura 3.2: Topologia da rede neural convolucional plotada a partir do pacote [Keras](#).



Resultados e Discussões

4.1 A Apple

A Apple Inc. é uma empresa norte-americana de tecnologia especializada em produtos eletrônicos e softwares, foi fundada por Steve Jobs e Steve Wozniak, e é atualmente a empresa mais rentáveis do mundo. [Uol \(2019\)](#)

A empresa que começou como uma startup de garagem, hoje conta com 80 mil empregados nos Estados Unidos além de 450 mil em suas subsidiárias. Ela protagonizou a revolução dos computadores pessoais e liderou a disputa no mercado dos smartphones. [Rawlinson \(2017\)](#)

Tim Cook (Presidente da Apple): Acreditamos que estamos na terra para fazer ótimos produtos e isso não está mudando. Estamos constantemente focados em inovar. Acreditamos no simples, não no complexo. Acreditamos que precisamos possuir e controlar as principais tecnologias por trás dos produtos que fabricamos e participar apenas de mercados onde podemos dar uma contribuição significativa. [Wanderley \(2020\)](#)

Em nossa metodologia utilizamos dados da Apple para medir a capacidade dos nossos modelos de prever corretamente valores de ações. Os dados foram extraídos do portal [Yahoo](#) no formato de planilha que segue a formatação especificada na sessão 3.1

4.2 Seleção de atributos

Na metodologia proposta selecionamos conjuntos de atributos baseados em pontuações atribuídas por algoritmos supervisionados, a figure 4.1 mostra a frequência em cada cada atributo apareceu nos conjuntos. É visível que: os atributos primários (os que não foram gerados por

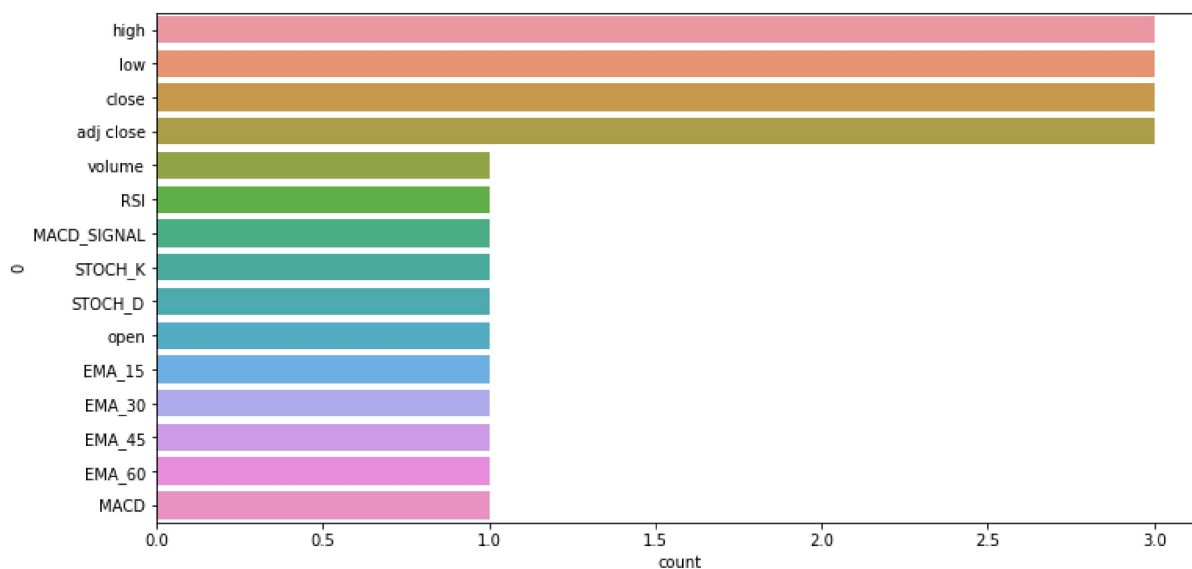
Tabela 4.1: Lista de atributos selecionados por cada um dos modelos supervisionados.

Conjunto	Atributos
Support Vector Machine (SVM)	['high', 'low', 'open', 'close', 'adj close', 'EMA_15', 'EMA_30', 'EMA_45', 'EMA_60', 'MACD']
Lasso	['high', 'low', 'close', 'volume', 'adj close', 'RSI', 'MACD_SIGNAL', 'STOCH_K', 'STOCH_D']
Random Forest (RF)	[high, low, close, adj close]

tecnicas de análise de mercado) tendem a aparecer mais frequentemente, já que as variáveis high, low, close e adj close apareceram em 100% dos conjuntos.

Por outro lado nenhum dos atributos gerados a partir das formulas de análise técnica de mercado apareceu mais do que uma vez, o que pode indicar que estes não são muito úteis na tarefa proposta.

Figura 4.1: Contagem da ocorrência dos atributos durante a seleção.



4.3 Modelos de predição

4.3.1 Métrica de avaliação

Com o intuito de mensurar o desempenho das redes treinadas e comparar o resultado entre os modelos e conjuntos gerados utilizaremos o erro médio percentual (MAPE) que é calculado segundo a fórmula 4.1.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - y_{pred_i})^2 \quad (4.1)$$

Tabela 4.2: Predição Rede LSTM 1 dia a frente

Conjunto	MAPE
Support Vector Machine (SVM)	5.82833
Lasso	3.20247
Random Forest (RF)	5.35987

Tabela 4.3: Predição Rede Convolutacional 1 dia a frente

Conjunto	MAPE
Support Vector Machine (SVM)	4.28463
Lasso	10.32160
Random Forest (RF)	3.12442

Sendo:

n = o número de amostras;

y_i = o valor real da variável de interesse;

y_{pred_i} = o valor previsto da variável de interesse.

4.3.2 Rede Neural com células de memória a longo prazo (LSTM)

Os resultados da rede com memória a longo prazo podem ser vistos na tabela 4.2, o conjunto de atributos gerado pelo algoritmo Lasso que selecionou mais indicadores de mercado do tipo oscilador obteve melhor desempenho. Podemos também observar que os outros conjuntos obtiveram resultado ruim, as razões para essa disparidade precisam ser avaliadas mais a fundo.

4.3.3 Rede Neural Convolutacional

Na tabela 4.3 podemos ver o desempenho do modelo convoluiconal criado. O grupo Random Forest (RF) obteve 3.1% de erro, seguindo pelo grupo Support Vector Machine (SVM) com 4.2% enquanto o Lasso teve performance ruim. O que pode indicar que os osciladores selecionados pelo algoritmo Lasso pioram o desempenho da rede convolutacional.

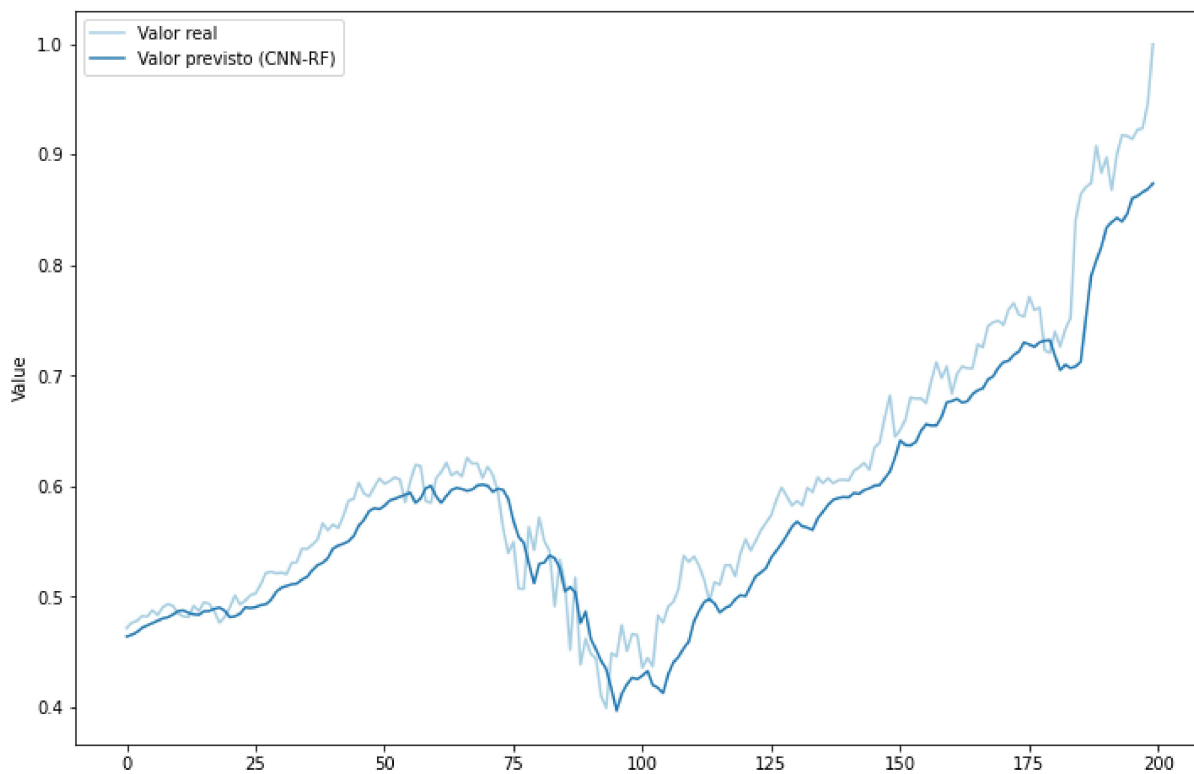
4.3.4 Usabilidade do modelo

Os modelos tiveram resultados similares, e de certa forma generalizaram o problema de predição de valores da bolsa para a Apple como podemos observar no gráfico 4.2. Por outro lado o resultado ainda é insuficiente para uso não experimental. Nos últimos 10 anos a flutuação diária dos valores da empresa na bolsa é de em média 3.45% e desta forma os modelos propostos, com o erro obtido não são adequados para a resolução do problema.

Por outro lado, nosso método dá indícios de que pode ser refinado, a escolha do método de seleção de atributos se mostrou importante e a adição de novas variáveis pode dar ao mo-

delo a informação necessária para ir além. Adicionalmente, novas topologias rede podem ser exploradas também no intuito de reduzir a imprecisão na tarefa de predição.

Figura 4.2: Gráfico de 200 dias de teste, comparação entre o valor real e o valor previsto pela rede convolucional utilizando o conjunto de atributos gerados pelo algoritmo Random Forest.



5

Considerações Finais

Em nosso trabalho foi possível observar a relevância da escolha de métodos adequados de seleção de variáveis no processo de treinamento de modelos preditivos baseados em aprendizagem de máquina. Os experimentos realizados mostram que: no problema de previsão dos valores de mercado da companhia Apple o conjunto selecionado de variáveis de treinamento constantemente envolve os atributos high, low, close e adj close.

Notamos que a utilidade da adição de métricas de análise técnica de mercado tem utilidade controversa no problema abordado. Os resultados obtidos com os modelos criados apontam que redes neurais são capazes de generalizar razoavelmente bem este problema de regressão, no entanto os resultados obtidos não são positivos o suficiente para justificar o uso dos modelos em um ambiente competitivo.

Acreditamos que nossos resultados podem ser melhorados através da exploração de novas variáveis que não estão necessariamente ligadas aos valores registrados pelas bolsas. O mercado de ações é altamente susceptível a mudanças no cenário sociopolítico e por isso é possível que a incorporação de atributos obtidos através de técnicas de análise de sentimento e áreas correlacionadas influenciem positivamente os resultados do preditor. Adicionalmente, é necessário explorar mais a fundo as possíveis topologias de redes neurais a fim de obter resultados ainda melhores.

Referências bibliográficas

S. Albawi, T. A. Mohammed, and S. Al-Zawi. Understanding of a convolutional neural network. In *2017 International Conference on Engineering and Technology (ICET)*, pages 1–6, 2017. DOI [10.1109/ICEngTechnol.2017.8308186](https://doi.org/10.1109/ICEngTechnol.2017.8308186).

Ethem Alpaydin. *Introduction to machine learning*. The MIT Press, 2010.

Michael A. Arbib. *The handbook of brain theory and neural networks*. MIT Press, 2003.

Venkatesh Bachu and J. Anuradha. A review of feature selection and its methods. *Cybernetics and Information Technologies*, 19:3, 03 2019. DOI [10.2478/cait-2019-0001](https://doi.org/10.2478/cait-2019-0001).

Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer New York, 2016.

Leo Breiman. Random forests. *Machine Learning*, 45:5–32, 2001.

James Chen. Technical Indicator Definition.

www.investopedia.com/terms/t/technicalindicator.asp, apr 2019. Online, acessado em maio de 2020.

Eunsuk Chong, Chulwoo Han, and Frank C. Park. Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. *Expert Systems with Applications*, 83:187 – 205, 2017. ISSN 0957-4174.

DOI <https://doi.org/10.1016/j.eswa.2017.04.030>. URL

<http://www.sciencedirect.com/science/article/pii/S0957417417302750>.

DigitalOcean. Como construir uma rede neural para reconhecer dígitos manuscritos com o tensorflow, Dec 2018. URL <https://www.digitalocean.com/community/tutorials/como-construir-uma-rede-neural-para-reconhecer-digitos-manuscritos-com-o-tensorfl>

Robert D. Edwards and John Magee. *Technical Analysis of Stock Trends*. Taylor Francis Group, 9th edition, 2007.

Freepik. Graphic resources for everyone, Nov 2020. URL <https://www.freepik.com/>.

Alex Graves, Greg Wayne, and Ivo Danihelka. *Neural Turing machines*, 2014.

- Isabelle Guyon and Andre Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, Mar 2003.
- Aurélien Géron. *Hands-On Machine Learning with Scikit-Learn and TensorFlow*. O'Reilly Media, 2017.
- Osman Hegazy, Omar S. Soliman, and Mustafa Abdul Salam. A machine learning model for stock market prediction. *International Journal of Computer Science and Telecommunications*, 4:17–23, 12 2013.
- Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9: 1735–80, 12 1997. DOI [10.1162/neco.1997.9.8.1735](https://doi.org/10.1162/neco.1997.9.8.1735).
- Badr Hssina, Abdelkarim Merbouha, Hanane Ezzikouri, and Mohammed Erritali. A comparative study of decision tree id3 and c4.5. *International Journal of Advanced Computer Science and Applications*, 4(2), 2014. DOI [10.14569/specialissue.2014.040203](https://doi.org/10.14569/specialissue.2014.040203).
- A. K. Jain, Jianchang Mao, and K. M. Mohiuddin. Artificial neural networks: a tutorial. *Computer*, 29(3):31–44, 1996. DOI [10.1109/2.485891](https://doi.org/10.1109/2.485891).
- Keras. URL <https://keras.io/>.
- Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.
- B. LABIAD, A. Berrado, and L. Benabbou. Intelligent system for intraday stock market forecasting. In *2019 5th International Conference on Optimization and Applications (ICOA)*, pages 1–6, 2019. DOI [10.1109/ICOA.2019.8727658](https://doi.org/10.1109/ICOA.2019.8727658).
- Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. DOI [10.1109/5.726791](https://doi.org/10.1109/5.726791).
- Ana Carolina Lorena and André de Carvalho. Uma introdução às support vector machines. *Revista de Informática Teórica e Aplicada*, 14(2), 2007.
- Mathworks. Support vector machine, 2020. URL <https://www.mathworks.com/help/stats/understanding-support-vector-machine-regression.html>.
- John J. Murphy. *Technical analysis of the financial markets: a comprehensive guide to trading methods and applications*. New York Institute of Finance, 1999.
- Christopher Olah. Understanding lstm networks. URL <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>.
- Pandas. Pandas. URL <https://pandas.pydata.org/>.

Jigar Patel, Sahil Shah, Priyank Thakkar, and K Kotecha. Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques.

Expert Systems with Applications, 42(1):259 – 268, 2015. ISSN 0957-4174.

DOI <https://doi.org/10.1016/j.eswa.2014.07.040>. URL

<http://www.sciencedirect.com/science/article/pii/S0957417414004473>.

Martin J. Pring. *Technical analysis explained ; the successful investors guide to spotting investment trends and turning points*. JMcGraw-Hill Education, 2015.

Ross Quinian. *C4.5: programs for machine learning*. Morgan Kaufmann, 1993.

Laura Raileanu and Kilian Stoffel. Theoretical comparison between the gini index and information gain criteria. *Annals of Mathematics and Artificial Intelligence*, 41:77–93, 05 2004. **DOI** [10.1023/B:AMAI.0000018580.96245.c6](https://doi.org/10.1023/B:AMAI.0000018580.96245.c6).

Nik Rawlinson. Apple was 41 years old in april, here's some history, Apr 2017. URL <https://www.macworld.co.uk/feature/history-of-apple-steve-jobs-mac-3606104/>.

Scikit-learn. Scikit-learn. URL <https://scikit-learn.org/stable/>.

Martin Scilly. What Is Meant by the Market Price of a Stock?

www.ifinance.zacks.com/meant-market-price-stock-5731.html, apr 2019. Online, acessado em abril de 2020.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: from theory to algorithms*. Cambridge University Press, 2014.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. 15(1):1929–1958, January 2014. ISSN 1532-4435.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

DOI [10.1111/j.2517-6161.1996.tb02080.x](https://doi.org/10.1111/j.2517-6161.1996.tb02080.x).

Uol. Apple é marca mais valiosa do mundo pelo 7º ano; disney desbanca facebook, Oct 2019.

URL <https://economia.uol.com.br/noticias/redacao/2019/10/17/>

[apple-google-e-amazon-sao-as-marcas-mais-valiosas-do-planeta-diz-estudo.htm](https://economia.uol.com.br/noticias/redacao/2019/10/17/apple-google-e-amazon-sao-as-marcas-mais-valiosas-do-planeta-diz-estudo.htm).

Mônica Wanderley. Tim cook completa 9 anos como líder da apple, Aug 2020. URL

<https://computerworld.com.br/negocios/>

[tim-cook-completa-9-anos-como-lider-da-apple/](https://computerworld.com.br/negocios/tim-cook-completa-9-anos-como-lider-da-apple/).

- Jie Wang and Jun Wang. Forecasting stock market indexes using principle component analysis and stochastic time effective neural networks. *Neurocomputing*, 156:68 – 78, 2015. ISSN 0925-2312. DOI <https://doi.org/10.1016/j.neucom.2014.12.084>. URL <http://www.sciencedirect.com/science/article/pii/S0925231215000090>.
- Z. Wang, S. Ho, and Z. Lin. Stock market prediction analysis by incorporating social and news opinion and sentiment. In *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 1375–1380, 2018. DOI [10.1109/ICDMW.2018.00195](https://doi.org/10.1109/ICDMW.2018.00195).
- Yahoo. Yahoo finance - stock market live, quotes, business finance news. URL <https://finance.yahoo.com/>.
- Xue Ying. An overview of overfitting and its solutions. *Journal of Physics: Conference Series*, 1168:022022, 02 2019. DOI [10.1088/1742-6596/1168/2/022022](https://doi.org/10.1088/1742-6596/1168/2/022022).