

UNIVERSIDADE FEDERAL DO PARANÁ
DEPARTAMENTO DE ENGENHARIA ELÉTRICA
SETOR DE TECNOLOGIA

ADRIANO DE CASTRO BENATTO PAUL

**UTILIZAÇÃO DE FILTROS DIGITAIS E MÉTODOS
ESTOCÁSTICOS PREDITIVOS EM ESTRATÉGIAS DE
NEGOCIAÇÃO DE AÇÕES NO MERCADO FINANCEIRO**

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA
2018

ADRIANO DE CASTRO BENATTO PAUL

**UTILIZAÇÃO DE FILTROS DIGITAIS E MÉTODOS
ESTOCÁSTICOS PREDITIVOS EM ESTRATÉGIAS DE
NEGOCIAÇÃO DE AÇÕES NO MERCADO FINANCEIRO**

Trabalho de Conclusão de Curso apresentado ao Setor de Tecnologia da Universidade Federal do Paraná, como requisito parcial para a obtenção do título de Bacharel em Engenharia Elétrica.

Orientador: Luis Henrique A. Lolis
Universidade Federal do Paraná

CURITIBA
2018

Aos pais, pelo exemplo de abnegação, dedicação e resiliência.

Aos amigos, pela amizade pura e simples.

Aos professores, pela dedicação ao trabalho de formar irmãos de profissão.

”Forecasting is the art of saying what will happen, and then explaining why it didn’t! ”

Anonymous

RESUMO

PAUL, Adriano. UTILIZAÇÃO DE FILTROS DIGITAIS E MÉTODOS ESTOCÁSTICOS PREDITIVOS EM ESTRATÉGIAS DE NEGOCIAÇÃO DE AÇÕES NO MERCADO FINANCEIRO. 2018. 54 f. Trabalho de Conclusão de Curso – Setor de Tecnologia, Universidade Federal do Paraná. Curitiba, 2018.

Este trabalho buscou integrar métodos de processamento digital de sinais e técnicas estocásticas preditivas em estratégias de negociação para o mercado acionário. Diferentes estratégias são implementadas em código e a simulação dos retornos é realizada para diferentes empresas que compõem a S&P 500 - a bolsa composta pelas 500 maiores empresas do mercado norte-americano - em um período de quatro anos (2014 a 2018). A combinação de filtros EWMA apresentou melhores resultados em relação aos filtros MAV, e a performance das simulações preditivas apresentaram resultados positivos em papéis cujo retorno acumulado no período obtiveram prejuízo. São também discutidas outras sugestões de implementação de filtros e técnicas, além de considerações para trabalhos futuros que possam expandir e integrar novas estratégias de negociação a estes métodos.

Palavras-chave: Filtros de Médias Móveis, Simulações de Monte Carlo, Mercado de Ações

ABSTRACT

PAUL, Adriano. USE OF DIGITAL FILTERS AND PREDICTIVE STOCASTIC METHODS IN STOCK MARKET TRADING STRATEGIES. 2018. [54](#) f. Trabalho de Conclusão de Curso – Setor de Tecnologia, Universidade Federal do Paraná. Curitiba, 2018.

This final paper aimed to integrate different digital signals processing methods and predictive stochastic techniques in stock markets trading strategies. Different strategies are implemented in code and the returns are simulated for different companies that comprise the S&P 500 - the stock market comprised of the 500 biggest companies of the north american market - in a 4 year period of observations (ranging from 2014 to 2018). The combination of EWMA filters showed better results than those obtained with MAV filters, and the predictive simulations performance showed positive results in stocks that accumulated losses in the study period. Different filters and techniques are also suggested for study, along with other considerations to expand and integrate novel trading strategies in future works.

Keywords: Moving Average Filters, Monte Carlo Simulations, Stock Markets

LISTA DE FIGURAS

Figura 1 – Distribuição do retorno diário de mercados financeiros diversos.	7
Figura 2 – Retorno acumulado de uma ação ao longo do tempo, em relação ao investimento inicial.	8
Figura 3 – Passeio aleatório com 150 passos e dez iterações.	10
Figura 4 – Implementação de dois filtros de médias móveis simples com sensibilidades à variação diferentes.	17
Figura 5 – Comparação de desempenho entre os filtros do tipo MA e EWMA de mesma janela de entrada.	18
Figura 6 – Filtros de Médias Móveis Simples com diferentes janelas de entrada de dados.	19
Figura 7 – Fluxograma do algoritmo de estratégia de negociação de ativos com utilização de médias móveis.	20
Figura 8 – Interação entre filtros de médias móveis sinalizando pontos para compra e venda do ativo, de acordo com a estratégia proposta	21
Figura 9 – Simulação de Monte Carlo com 50 Passeios Aleatórios.	22
Figura 10 – Fluxograma do algoritmo de estratégia de negociação de ativos com utilização de simulações de Monte Carlo.	23
Figura 11 – Exemplo da implementação proposta para Simulações de Monte Carlo.	25

LISTA DE TABELAS

Tabela 1 – Empresas selecionadas para implementação dos filtros de médias móveis e aplicação de métodos estocásticos	16
---	----

LISTA DE ABREVIATURAS E SIGLAS

AAL	American Air Lines
AAPL	Apple Inc.
ADBE	Adobe Systems Inc.
ADSK	Autodesk Inc.
AMD	Advanced Micro Devices Inc.
AMZN	Amazon.com Inc.
ATVI	Activision Blizzard
AXP	American Express Co
BA	Boeing Company
BAC	Bank of America Corp
BBY	Best Buy Co Inc.
BLK	BlackRock
C	Citigroup Inc.
CMCSA	Comcast Corp.
COST	Costco Wholesale Corp.
CSCO	Cisco Systems
CVX	Chevron Corp.
DAL	Delta Air Lines Inc.
DIS	The Walt Disney Company
EA	Electronic Arts
EBAY	eBay Inc.
EWMA	Exponential Weighted Moving Average
F	Ford Motor
FB	Facebook, Inc.
FDX	FedEx Corporation
GE	General Electric

GM	General Motors
GOOG	Alphabet Inc Class C
GPS	Gap Inc.
GRMN	Garmin Ltd.
GS	Goldman Sachs Group
GT	Goodyear Tire & Rubber
HAS	Hasbro Inc.
HD	Home Depot
HFT	High Frequency Trading
HIG	Hartford Financial Svc.Gp.
HOG	Harley-Davidson
INTC	Inter Corp.
JNJ	Johnson & Johnson
K	Kellog Co.
KO	Coca-Cola Company
KORS	Michael Kors Holding
LMT	Lockheed Martin Corp.
M	Macy's Inc
MA	Mastercard Inc.
MAV	Moving Average
MAR	Marriott Int'l
MAT	Mattel Inc.
MDLZ	Mondelez International
MGM	MGM Resorts International
MIT	Massachusetts Institute of Technology
MMM	3M Company
MS	Morgan Stanley
MSFT	Microsoft Corp.

MSI	Motorola Solutions Inc.
NFLX	Netflix Inc.
PEP	PepsiCo Inc.
PG	Procter & Gamble
QCOM	QUALCOMM Inc.
RCL	Royal Caribbean Cruises Ltd
RL	Polo Ralph Lauren Corp.
ROST	Ross Stores
RTN	Raytheon Co.
SBUX	Starbucks Corp.
STX	Seagate Technology
SYMC	Symantec Corp.
T	AT&T Inc.
TTWO	Take-Two Interactive
TWTR	Twitter, Inc.
TXN	Texas Instruments
V	Visa Inc.
VIAB	Viacom Inc.
VRSN	Verisign Inc.
VZ	Verizon Communications
WFC	Wells Fargo
WMT	Wal-Mart Stores
WU	Western Union Co
XOM	Exxon Mobil Corp.
XX	Xerox Corp.

LISTA DE SÍMBOLOS

α	Grau de Ponderação para o Filtro EWMA
μ	Média do Retorno Diário
σ	Desvio padrão do Retorno Diário
σ^2	Variância do Retorno Diário
W_t	Processo de Wiener

SUMÁRIO

1 – INTRODUÇÃO	1
1.1 CONTEXTO	1
1.1.1 Motivação	1
1.2 OBJETIVO GERAL	1
1.2.1 Objetivos Específicos	2
1.3 JUSTIFICATIVA	2
1.4 ORGANIZAÇÃO DO TRABALHO	2
2 – ESTADO DA ARTE	4
3 – FUNDAMENTAÇÃO TEÓRICA	6
3.1 ANÁLISE FINANCEIRA BÁSICA	6
3.1.1 Retorno Diário	6
3.1.2 Retorno Diário Acumulado	6
3.1.3 Volatilidade	7
3.2 FILTROS DE MÉDIAS MÓVEIS	8
3.2.1 Filtro de Média Móvel Simples	8
3.2.2 Filtro de Média Móvel Exponencialmente Ponderada	9
3.3 MÉTODO DE MONTE CARLO	9
3.4 MOVIMENTO BROWNIANO GEOMÉTRICO	10
4 – MATERIAIS E MÉTODOS	12
4.1 FERRAMENTAS COMPUTACIONAIS	12
4.1.1 Python	12
4.1.2 NumPy	12
4.1.3 SciPy	13
4.1.4 Pandas	13
4.1.5 Matplotlib	13
4.1.6 Jupyter Notebook	13
4.1.7 Spyder	13
4.1.8 Distribuição Anaconda	14
4.2 VISUALIZAÇÃO DOS DADOS OBTIDOS	14
4.3 IMPORTAÇÃO DE BIBLIOTECAS E OBTENÇÃO DOS DADOS FI- NANCEIROS	14
4.3.1 Empresas escolhidas para o cenário de Simulação	15
4.4 UTILIZAÇÃO DOS FILTROS DE MÉDIAS MÓVEIS	15

4.4.1	Implementação em Python	15
4.4.2	Estratégia para Tomada de Decisão	18
4.4.3	Parâmetros Utilizados	20
4.5	SIMULAÇÃO PREDITIVA PELO MÉTODO DE MONTE CARLO . . .	21
4.5.1	Implementação em Python	21
4.5.2	Estratégia para Tomada de Decisão	23
4.5.3	Parâmetros Utilizados	24
5	– ANÁLISE E DISCUSSÃO DOS RESULTADOS	26
5.1	DISCUSSÃO DOS RESULTADOS OBTIDOS COM O USO DE FILTROS DE MÉDIAS MÓVEIS	26
5.2	DISCUSSÃO DOS RESULTADOS OBTIDOS COM O USO DE SIMU- LAÇÕES DE MONTE CARLO	27
6	– CONCLUSÃO	29
6.1	CONSIDERAÇÕES FINAIS	29
6.2	TRABALHOS FUTUROS	29
	Referências	31
	 Apêndices	 33
	APÊNDICE A–CÓDIGO EM PYTHON PARA IMPLEMENTAÇÃO DE ESTRATÉGIA UTILIZANDO FILTROS DE MÉ- DIAS MÓVEIS	34
	APÊNDICE B–CÓDIGO EM PYTHON PARA IMPLEMENTAÇÃO DE NEGOCIAÇÃO UTILIZANDO SIMULAÇÕES DE MONTE CARLO	40
	APÊNDICE C–RESULTADOS OBTIDOS A PARTIR DE COMBINA- ÇÕES DE FILTROS DE MÉDIAS MÓVEIS SIMPLES	45
	APÊNDICE D–RESULTADOS OBTIDOS A PARTIR DE COMBINA- ÇÕES DE FILTROS DE MÉDIAS MÓVEIS EXPO- NENCIALMENTE PONDERADAS	49
	APÊNDICE E–RESULTADOS OBTIDOS A PARTIR DA APLICA- ÇÃO DE MÉTODOS ESTOCÁSTICOS E SIMULA- ÇÕES DE MONTE CARLO	53

1 INTRODUÇÃO

1.1 CONTEXTO

Na última década, nota-se um expressivo crescimento em aplicações de processamento de sinais e técnicas estatísticas em diferentes segmentos do mercado e da indústria. O mercado financeiro é um dos pioneiros na adoção e aplicação de modelos sofisticados para analisar dados relevantes e tomar decisões inteligentes automatizadas.

Estima-se que atualmente mais da metade de todas as operações em bolsas de valores são realizadas por algoritmos especialmente projetados para extrair o *alpha* do mercado (AKANSU; KULKARNI; MALIOUTOV, 2016), ou seja, para extrair lucro maior que o crescimento do próprio mercado, seja por técnicas de arbitragem, negociação em pares, reversão à média, entre outras.

Assim sendo, a validação de qualquer algoritmo com este fim passa por rigorosos testes com dados passados. O comportamento dos papéis é estudado em função de sua volatilidade e diferentes modelos são propostos, testados, e caso desempenhem positivamente, aplicados para operar em dados reais. O processo de *backtesting* é então, parte crucial da validação de um modelo matemático adequado para operar em mercados reais.

1.1.1 Motivação

A abundância e disponibilidade de dados financeiros reais de longas séries históricas, somadas às ferramentas disponíveis atualmente para tratamento, organização e visualização destes dados tornaram uma tarefa que até alguns anos atrás ficava restrita a computadores projetados somente para este fim em algo factível para acadêmicos de qualquer nível.

A multidisciplinaridade do tema e a devida aplicação de ferramentas de engenharia ajudam a tornar algo que era visto pelo autor como mera curiosidade em objeto de estudo para, se possível, alicerçar conhecimentos e desenvolver uma carreira promissora, seja com tratamento de dados diversos ou em áreas de pesquisa e desenvolvimento de engenharia financeira.

1.2 OBJETIVO GERAL

Desenvolver uma estratégia para tomada de decisões de compra e venda de ativos no mercado financeiro, realizando a análise de séries de dados históricos para a criação de um modelo de estimação futura que, a partir de outras técnicas já tradicionais, auxilie a otimizar operações de compra e venda de ativos, avaliados em função do retorno obtido.

1.2.1 Objetivos Específicos

- Estudo e embasamento teórico em tópicos de econometria e processamento digital de sinais;
- Seleção e estudos das ferramentas open source apropriadas para a obtenção de dados, processamento, análise, criação e simulação de modelos matemáticos;
- Implementação de filtros digitais nos dados coletados, que auxiliem a parametrização de modelos preditivos;
- Criação e treinamento de modelos estocásticos para a previsão de valores futuros utilizando Simulações de Monte Carlo;

1.3 JUSTIFICATIVA

Aliando a implementação de técnicas estudadas em disciplinas como Controle Avançado, Instrumentação Eletrônica e Processamento Digital de Sinais à base teórica de disciplinas como Processos Estocásticos e Técnicas Analíticas para a Engenharia Elétrica, busca-se fundamentar a criação de modelos matemáticos complexos a partir de uma série de dados. Portanto, o estudo da tecnologia atual e estado da arte em técnicas de processamento de sinais, filtros dinâmicos e do funcionamento do mercado financeiro em si expandem possibilidades de estudo e trabalho futuros nestas áreas, além de prover oportunidades de ganho no mercado financeiro com a aplicação das técnicas e criação das estratégias estudadas, caso mostrem-se efetivas.

1.4 ORGANIZAÇÃO DO TRABALHO

No [Capítulo 2](#) será apresentado o histórico dos estudos e aplicações que mudaram a forma da academia em encarar o mercado financeiro, que passou de uma mera roleta especulativa para um tópico sério e promissor de estudos, trazendo grandes nomes da matemática e engenharia para a área de finanças.

O [Capítulo 3](#) deste trabalho provê um breve estudo sobre as diferentes técnicas de processamento digital de sinais utilizadas no mercado financeiro, entre eles, a utilização de Médias Móveis Simples e Médias Móveis Exponencialmente Ponderadas e a criação de um modelo estocástico preditivo, baseado no movimento Browniano Geométrico.

Em seguida, no [Capítulo 4](#) são apresentadas as ferramentas computacionais selecionadas para obtenção de dados, análise e criação dos modelos propostos, bem como as técnicas escolhidas e o banco de testes utilizado para validar estas técnicas. O banco de testes inclui quais ações serão avaliadas, e quais parâmetros serão variados na aplicação de cada técnica.

O [Capítulo 5](#) discute os resultados obtidos com a implementação e simulação das estratégias propostas, em seções distintas quanto ao uso de filtros de médias móveis e com o método estocástico preditivo.

E finalmente, no [Capítulo 6](#), as considerações finais acerca do trabalho realizado e as sugestões do autor para trabalhos futuros.

2 ESTADO DA ARTE

Louis Bachelier, em 1900, publicou sua tese "A Teoria da Especulação". Nela, argumentava que os preços de ações são fundamentalmente imprevisíveis. Praticamente ao mesmo tempo, Albert Einstein buscava uma explicação para o movimento Browniano, observado pelo botânico Robert Brown quase um século antes. (FEYNMAN, 1964) O tratamento matemático dado por Einstein ao tema foi publicado em 1905, e embora similar ao trabalho já desenvolvido por Bachelier, era menos avançado que aquele derivado por Bachelier para o mercado de ações.

Em 1948, um estatístico britânico chamado Maurice Kendall deu uma palestra na Sociedade Estatística Real em Londres, tratando do preço semanal do trigo em mercados de *commodities* de Chicago. Kendall desejava saber quão bem alguém poderia prever os preços futuros do trigo, dados os preços passados. Sua conclusão inesperada foi de que não é possível de maneira alguma prever os preços. Disse na ocasião que os valores vagavam sem direção, "quase como se o Demônio da Sorte escolhesse um número aleatório por semana... e soma-se ao preço corrente para determinar o preço da semana seguinte"(POUNDSTONE, 2005). Kendall sugeriu que o mesmo princípio poderia ser aplicado ao mercado de ações, e que as pessoas que imaginavam poder prever os valores estavam apenas se iludindo. Esta ideia sugeria que o mercado acionário nada mais é que um grande cassino, pois se os movimentos dos preços são tão imprevisíveis quanto números da loteria, talvez as pessoas que façam fortuna tenham apenas sorte, e nada mais.

No início da década de 1970, os primeiros sistemas computadorizados começaram transportar ordens de compra, com a introdução de um sistema eletrônico que roteava ordens entre postos de vendas, que então eram executadas manualmente na Bolsa de Valores de Nova Iorque. Ao final da década de 1980, os mercados financeiros já possuíam execução eletrônica total, encorajando negociações a partir de algoritmos que aumentaram a liquidez do mercado.

Em 1992, o economista matemático William A. Brock publicou um artigo pioneiro que testava estratégias de negociação utilizando Médias Móveis Simples e também simulando o comportamento do mercado com Passeios Aleatórios (BROCK; LAKONISHOK; LEBARON, 1992). Concluiu que estudos anteriores que desacreditavam do método utilizado por ele, conhecido como análise técnica, estavam precipitados em fazer esta afirmação. Ao todo, 20 regras de médias móveis foram avaliadas, em longas séries de dados do Índice Dow Jones, de 1897 a 1986.

A adoção de estratégias com algoritmos ganhou maior encorajamento em 2001, quando um time de pesquisadores da IBM publicou um artigo na Conferência Internacional de Inteligência Artificial em que mostravam versões de experimentos de leilões eletrônicos usados em mercados financeiros com duas estratégias algorítmicas que eram

consistentemente melhores que corretores humanos (DAS et al., 2001). No artigo, o time de pesquisadores escreveu que o impacto financeiro da performance de algoritmos sobre corretores humanos poderia ser medida em bilhões de dólares anualmente.

Enquanto mais mercados eletrônicos apareciam, novas estratégias de negociação foram introduzidas. Estas estratégias são facilmente implementadas por computadores, uma vez que podem reagir rapidamente a qualquer condição favorável em curtíssimos espaços de tempo.

A atual demanda para negociações eletrônicas demanda baixa latência e maior proximidade às bolsas em si. Com a tecnologia atual, a latência é determinada pela velocidade da luz, que corresponde a cerca de 3.3 milisegundos para cada 1000 km de fibras ópticas - e qualquer sinal regenerado ou roteado introduz latências maiores que essa. Negociações eletrônicas dessa natureza são conhecidas como HFT (*High Frequency Trading*) ou Negociações de Alta Frequência (DUHIGG, 2009)

3 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão apresentados os conceitos básicos de análise financeira utilizados no trabalho, assim como a definição dos filtros de médias móveis e das técnicas estocásticas utilizadas para modelar o comportamento de ações do mercado financeiro.

3.1 ANÁLISE FINANCEIRA BÁSICA

Nesta seção, discutiremos os termos que serão utilizados para avaliar os resultados finais do trabalho, tais como o retorno diário e o retorno diário acumulado, bem como a definição de volatilidade de um papel.

3.1.1 Retorno Diário

O retorno diário é definido por

$$r[n] = \frac{p[n]}{p[n-1]} - 1 \quad (1)$$

ou seja, define o retorno como o valor de um ativo em um tempo n dividido pelo preço em $n-1$ (neste caso, o dia anterior). Basicamente informa o percentual de ganho ou perda para uma ação comprada em um dia e vendida no dia seguinte, e é o fator utilizado para se analisar a volatilidade de uma ação (MILLS; MARKELLOS, 2008).

Na [Figura 1](#), estão dispostos os histogramas de mercados financeiros diversos em função do desvio padrão de cada mercado. Na imagem, as linhas pontilhadas representam a distribuição de uma variável aleatória normal que utiliza a média e desvio padrão dos retornos observados.

3.1.2 Retorno Diário Acumulado

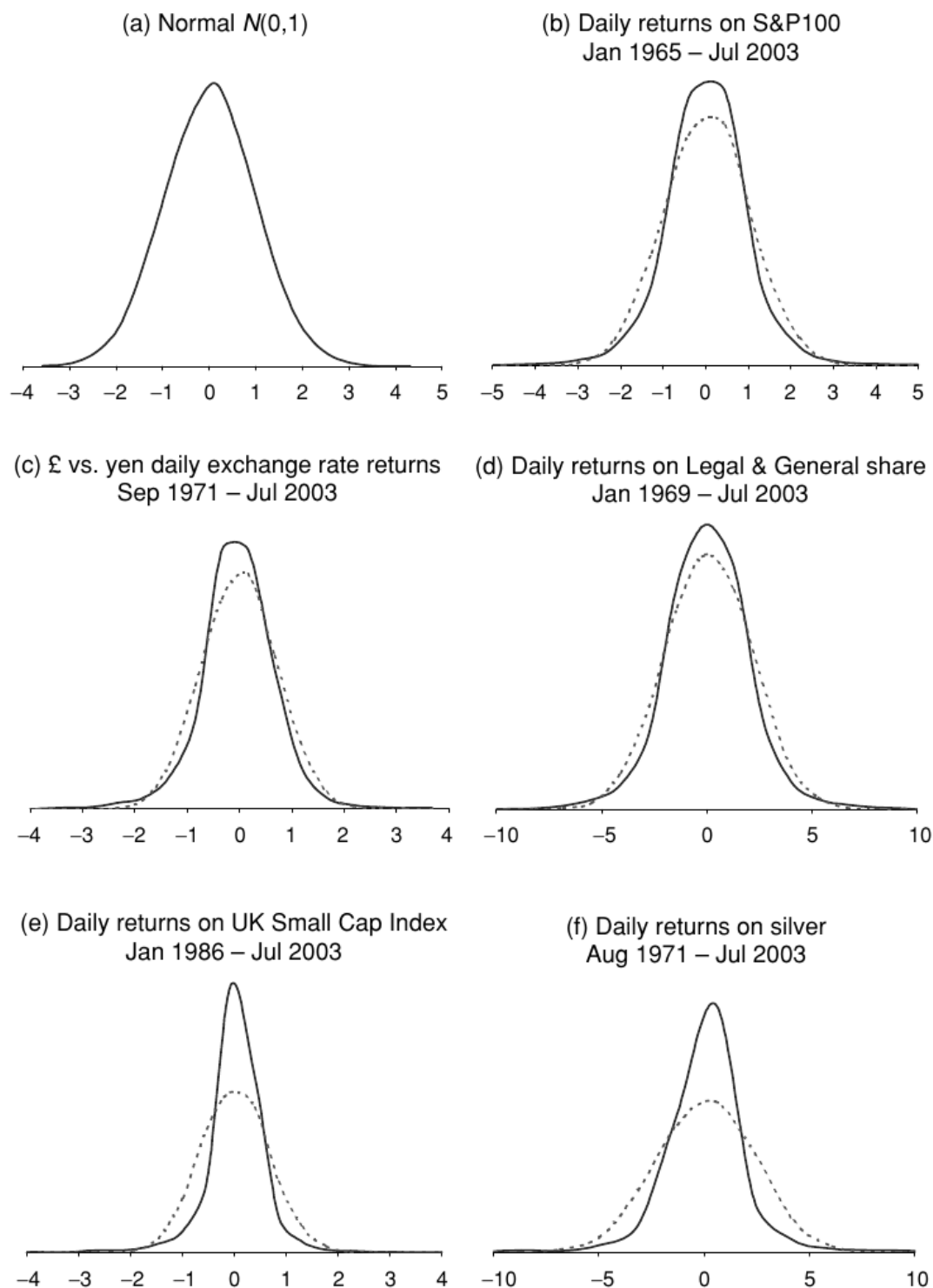
É o retorno computado do dia em que o investimento foi realizado ao dia em que sua venda é concretizada. É calculado por:

$$R[N] = \prod_{n=1}^N (r[N-n] + 1) - 1 \quad (2)$$

o retorno diário acumulado é o resultado do retorno diário acrescido de uma unidade e multiplicado pelo retorno obtido no período anterior, sequencialmente, de tal forma que ao final do tempo desejado tem-se a razão ganha ou perdida para cada unidade monetária investida no período inicial (MILLS; MARKELLOS, 2008).

Na [figura 2](#) observa-se o acúmulo de capital gerado para um investimento realizado no período inicial das observações. Ao final, tem-se a proporção ganha (ou perdida) para cada unidade monetária investida naquele período inicial.

Figura 1 – Distribuição do retorno diário de mercados financeiros diversos.

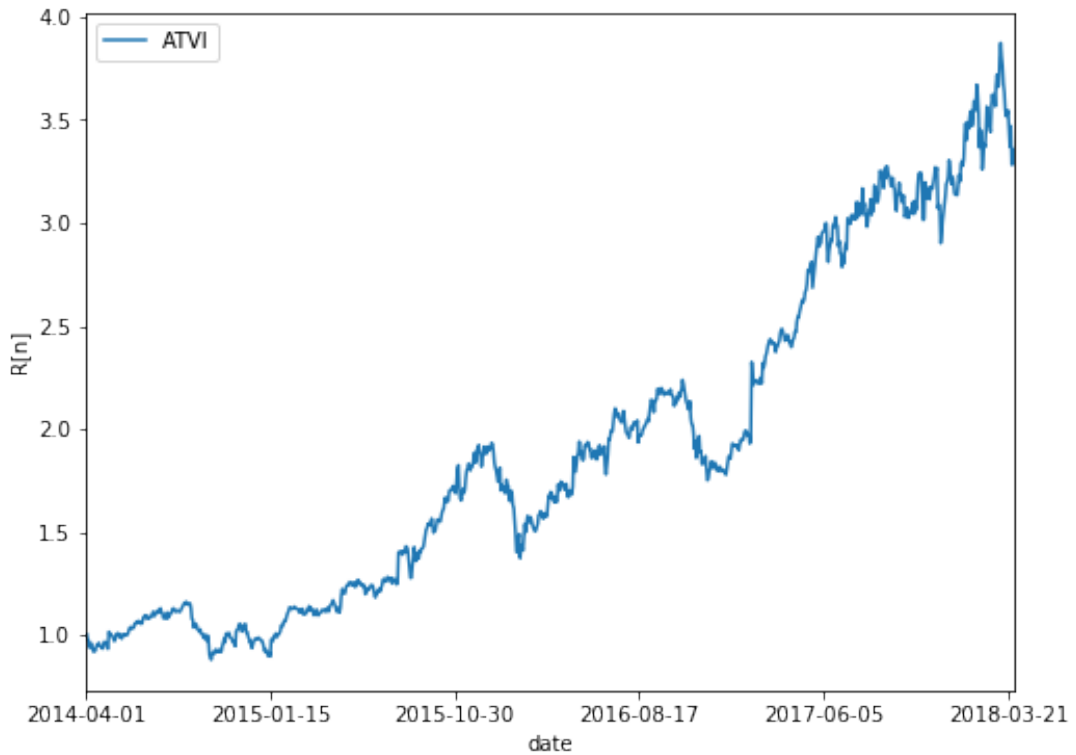


Fonte: (POON, 2005)

3.1.3 Volatilidade

Volatilidade refere-se à dispersão das variações de uma variável. Tipicamente em mercados financeiros, a preocupação dá-se com a dispersão dos retornos de um ativo

Figura 2 – Retorno acumulado de uma ação ao longo do tempo, em relação ao investimento inicial.



Fonte: O autor

(POON, 2005). Estatisticamente, volatilidade é frequentemente medida como um simples desvio padrão

$$\sigma[N] = \sqrt{\frac{1}{N-1} \sum_{n=1}^N (r[n] - \mu)^2} \quad (3)$$

onde $r[n]$ é o retorno no dia n e μ é a média do retorno no N -ésimo dia.

Às vezes, a variância σ^2 também é utilizada como medida de volatilidade. Sendo a variância o quadrado da volatilidade. Como o desvio padrão possui a mesma unidade de medida da média (se expressa em dólares ou reais, o desvio padrão também estará expresso na mesma unidade). Por esta razão, o desvio padrão é mais conveniente e intuitivo quando se discute volatilidade.

3.2 FILTROS DE MÉDIAS MÓVEIS

3.2.1 Filtro de Média Móvel Simples

O filtro de Média Móvel Simples ou filtro MA (*Moving Average*) é um filtro de resposta finita ao impulso. Em aplicações financeiras, é a média não ponderada de n dados

anteriores, ou seja, é dado por

$$\bar{x}[n] = \frac{1}{N} \sum_{i=0}^{N-1} x[n-i] \quad (4)$$

onde N é igual ao tamanho da janela desejado e $x[n-i]$ é o valor da ação no período $n-i$.

Em termos financeiros, os níveis de um filtro deste tipo podem ser interpretados como uma linha de suporte em um mercados em queda ou como uma linha de resistência em um mercado em alta.

3.2.2 Filtro de Média Móvel Exponencialmente Ponderada

O filtro de Média Móvel Exponencialmente Ponderada ou filtro EWMA (*Exponential Weighted Moving Average*) aplica ponderações não uniformes, de modo que dados mais recentes possuem maior influência no resultado, e é um tipo de filtro de resposta infinita ao impulso. É dado por

$$\bar{x}[n] = \alpha x[n-1] + (1-\alpha)x[n] \quad (5)$$

onde

$$\alpha = \frac{M}{M+1} \quad (6)$$

sendo M igual ao tamanho da janela escolhida.

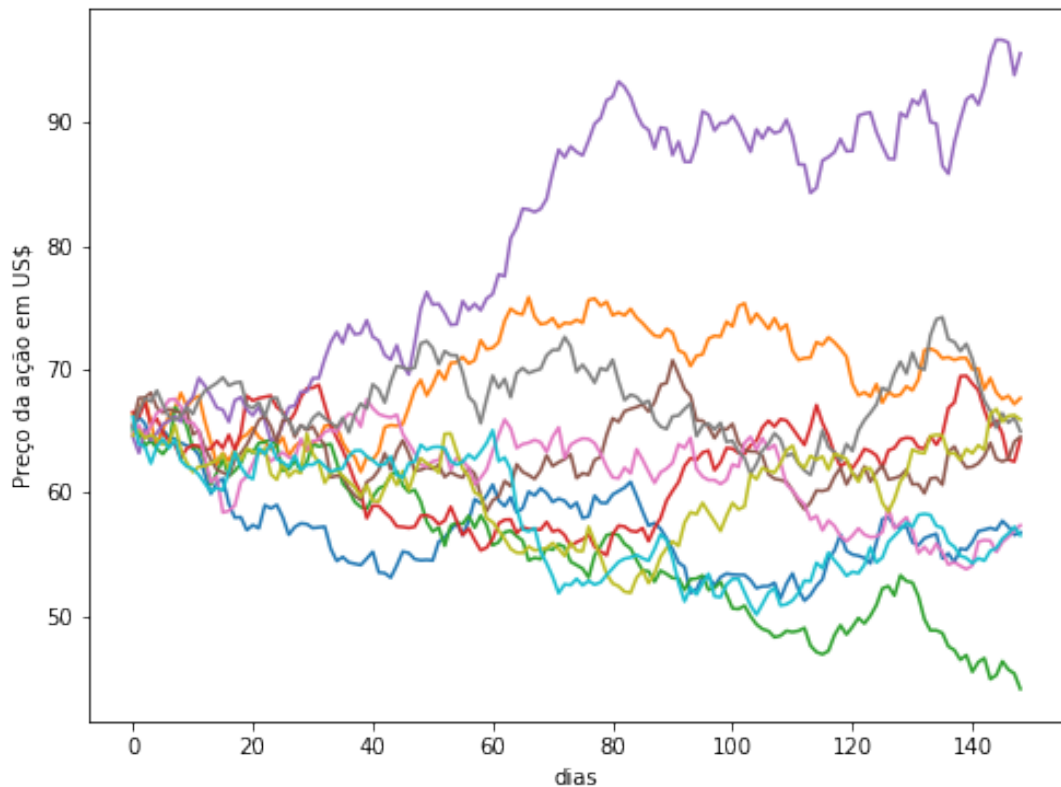
3.3 MÉTODO DE MONTE CARLO

Métodos de Monte Carlo são uma vasta classe de algoritmos computacionais que utilizam-se de amostragens aleatórias para que se obtenham resultados numéricos, sendo utilizados em três grandes classes de problemas: otimização, integração numérica e geração de extrações de uma distribuição de probabilidades ([KROESE et al., 2014](#)).

O nome refere-se ao cassino do Principado de Mônaco, conhecido como um ícone do jogo. O termo "Monte Carlo" foi introduzido pela primeira vez em 1947 por Nicholas Metropolis ([METROPOLIS, 1987](#)), um físico grego-americano.

A princípio, métodos de Monte Carlo podem ser usados para resolver qualquer problema que possua uma interpretação probabilística. Pela lei dos grandes números, integrais descritas pelo valor esperado de alguma variável aleatória podem ser aproximados tomando a média empírica (a média da amostra) de amostras independentes da variável ([HASTINGS, 1970](#)).

Figura 3 – Passeio aleatório com 150 passos e dez iterações.



Fonte: O autor

3.4 MOVIMENTO BROWNIANO GEOMÉTRICO

O movimento browniano é o movimento aleatório de partículas suspensas em um fluido resultante da colisão das moléculas no fluido (FEYNMAN, 1964), observado e descrito pelo botânico Robert Brown, em 1827.

Em matemática, o movimento Browniano é descrito como um processo de Wiener; um processo estocástico contínuo no tempo. O processo de Wiener W_t é caracterizado pelas seguintes propriedades:

- $W_0 = 0$;
- W possui incrementos independentes: para cada $t > 0$, os incrementos futuros serão independentes dos valores passados;
- W possui incrementos gaussianos;
- W possui caminhos contínuos.

O processo estocástico de Wiener que consiste de passos temporais discretos é conhecido como Passeio Aleatório Simples, e foi introduzido pela primeira vez por Karl Pearson em 1905 (PEARSON, 1905).

Em finanças, a hipótese do Passeio Aleatório é usada para modelar preços de ações. Como ações possuem a particularidade de não possuírem valores negativos, um tipo

especial de Passeio Aleatório é introduzido, chamado de Movimento Browniano Geométrico ou Movimento Browniano Exponencial (ROSS, 2014), descrito por

$$\hat{r}[n] = e^{\mu n - \frac{\sigma^2 n}{2} + \sigma W_t} \quad (7)$$

onde μ é a média do processo, σ o desvio padrão do processo e W_t é o processo de Wiener - um valor aleatório, de distribuição normal, média zero e desvio padrão σ .

4 MATERIAIS E MÉTODOS

Neste capítulo serão apresentadas as ferramentas computacionais selecionadas para a obtenção, análise e criação dos modelos matemáticos em que serão simuladas ações de compras e vendas de ativos, bem como a maneira que serão desenvolvidas as simulações e análises dos resultados obtidos.

Todas as ferramentas selecionadas são do tipo *open source*, ou seja, possuem desenvolvimento descentralizado e seus códigos-fonte e documentação estão disponíveis gratuitamente ao público.

4.1 FERRAMENTAS COMPUTACIONAIS

4.1.1 Python

As ferramentas utilizadas para análise de dados (ou seja, para a extração de valor a partir de dados brutos) comumente utilizadas são aquelas originadas nas seguintes linguagens de programação: SAS, R, Hadoop, Python e SQL.

Destas linguagens, Python é uma linguagem de programação de alto nível, interpretada, de script, imperativa, orientada a objetos, funcional, de tipagem dinâmica e forte (VENNERS, 2003). Foi lançada por Guido van Rossum em 1991 e atualmente é um dos componentes padrão de vários sistemas operacionais, entre eles estão a maioria das distribuições do Linux, AmigaOS 4, FreeBSD, NetBSD, OpenBSD e OS X.

Python é extremamente popular em ciência de dados, uma vez que existem diversas bibliotecas específicas para manipulação de dados e, sendo uma linguagem derivada de C, seu aprendizado é rápido e fácil quando comparado às demais linguagens utilizadas para este fim. A versão utilizada é a 3.6 e encontra-se disponível para download em sua página oficial (PYTHON, 2018).

4.1.2 NumPy

NumPy é o pacote fundamental para computação científica em Python. Contém, entre outras coisas:

- Vetor de objetos N-dimensional;
- Ferramentas de integração com códigos em C/C++ e Fortran;
- Álgebra linear, transformadas de Fourier, entre outras funções matemáticas sofisticadas.

Além do uso científico óbvio, os vetores multi-dimensionais podem armazenar tipos de dados arbitrários. Isto permite que o NumPy possa ser integrado em uma grande variedade de bases de dados. A versão utilizada é a 1.14.3 e as informações de instalação encontram-se em sua página oficial (NUMPY, 2018).

4.1.3 SciPy

SciPy é uma biblioteca para Python utilizada para computação técnica e científica. Possui módulos para otimização, álgebra linear, integração, interpolação, funções especiais, transformada rápida de Fourier, processamento de imagem e sinal, entre outras tarefas comuns para ciência e engenharia. Sua estrutura de dados é aquela provida pelo módulo NumPy. A versão utilizada é a 1.1.0 e as informações de instalação encontram-se em sua página oficial ([SCIPY, 2018](#))

4.1.4 Pandas

Biblioteca para Python que permite a manipulação de tabelas numéricas e séries temporais com indexação integrada. Permite a leitura de estruturas de dados em diferentes formatos: arquivos de texto e CSV, Microsoft Excel e bases de dados SQL e é utilizado em diversos domínios acadêmicos e comerciais, como financeiro, neurociências, economia e estatística. A versão utilizada é a 0.22.0 e as informações de instalação encontram-se em sua página oficial ([PANDAS, 2018](#))

4.1.5 Matplotlib

Matplotlib é uma biblioteca para plotagem de gráficos em linhas, barras, de dispersão, histogramas com interface semelhante àquela oferecida pelo MATLAB. A versão utilizada é a 2.2.2 e as informações de instalação encontram-se em sua página oficial ([MATPLOTLIB, 2018](#))

4.1.6 Jupyter Notebook

O Jupyter Notebook é uma aplicação web que permite a criação e compartilhamento de documentos com código, equações e gráficos. Seus usos incluem a limpeza e transformação de dados, simulação numérica, modelagem estatística, visualização de dados, entre outros, com suporte a mais de 40 linguagens de programação. A versão utilizada é incluída na distribuição Anaconda, que inclui a última versão de Python 3 ([JUPYTER, 2018](#)).

4.1.7 Spyder

O Spyder é um acrônimo para Ambiente de Desenvolvimento Científico em Python. Suas características principais são:

- Opções de edição avançadas, testes interativos e *debugging*;
- Ambiente de computação numérica com suporte a IPython (interpretador de Python interativo) e bibliotecas populares como NumPy, SciPy e Matplotlib

O Spyder é particularmente útil para a verificação e edição dos valores de variáveis armazenados na memória quando algum código é executado. É uma excelente alternativa

ao Jupyter Notebook, principalmente para escrever o código e procurar por erros lógicos e de execução. A versão utilizada é a 3.2.6 e encontra-se disponível em sua página oficial ([SPYDER, 2018](#)) ou pode ser instalado automaticamente a partir da Distribuição Anaconda.

4.1.8 Distribuição Anaconda

A distribuição Anaconda permite a instalação automática da última versão de Python e de mais de 1000 pacotes de ciência de dados, em suas versões mais estáveis para cada sistema operacional, além de incluir ambientes de desenvolvimento integrados, como Jupyter, Spyder e RStudio. A instalação da versão 5.1 desta distribuição inclui todas as ferramentas citadas anteriormente, e está disponível para Windows, MacOS e Linux em sua página oficial ([ANACONDA, 2018](#)).

4.2 VISUALIZAÇÃO DOS DADOS OBTIDOS

Qualquer série de dados criada em um objeto pode ser visualizada com o auxílio da biblioteca Matplotlib. O método básico para a plotagem de uma série é o comando `.plot()`, onde os argumentos que podem ser passados modificam legendas, título, eixos, tamanho da figura, tipos e cores do traço.

Nesta mesma biblioteca, o método `.hist()` apresenta um histograma dos valores da série, que podem ser divididos em um número específico de colunas pelo argumento *bins*.

A visualização das séries obtidas e criadas é importantíssima como parte do processo de *debugging* do algoritmo que se queira criar, assim como para que ajude a desenvolver novos *insights* a respeito dos dados que estão sendo trabalhados ou avaliar a estratégia de negociação criada, com a marcação dos sinais indicativos de compra e venda de uma ação ao longo de sua série histórica.

4.3 IMPORTAÇÃO DE BIBLIOTECAS E OBTENÇÃO DOS DADOS FINANCEIROS

A utilização das bibliotecas instaladas a partir da distribuição Anaconda é realizada no ambiente de desenvolvimento instalado. O autor optou pela utilização do Jupyter Notebook, dada sua modularidade de funcionamento, onde o código criado pode ser executado em diferentes segmentos (ou células), à medida que sua utilização torna-se necessária ou que modificações sejam realizadas. Estas células também permitem que textos sejam escritos, sendo possível a alternância entre pedaços de código e observações sobre o que está sendo realizado, tornando o código mais didático e legível.

O primeiro passo então é importar as bibliotecas instaladas, para que suas funções (ou métodos) possam ser implementados no código. Em Python, é utilizado o comando `import` seguido do nome do pacote desejado, conforme vê-se nos [Apêndice A](#) e [Apêndice B](#).

Os dados das séries temporais utilizados são obtidos a partir de uma ferramenta da biblioteca Pandas, o `DataReader`. Esta função permite o download dos valores de abertura,

fechamento, máximo e mínimo de cada dia de ações de empresas listadas na S&P500 (a bolsa de valores que agrega as 500 maiores empresas em valor de mercado dos EUA). A base de dados escolhida para download é aquela da IEX (The Investors Exchange). Esta base permite que sejam baixados os dados diários das empresas listadas dos últimos 5 anos. A opção por essa base de dados é passada pelo argumento 'iex' no método `DataReader`, da biblioteca `Pandas`. Embora existam outras opções de bancos de dados disponíveis, esta mostrou-se a mais estável e confiável, sem que houvesse a necessidade de completar dados faltantes ou excluir valores aberrantes.

O autor optou pela utilização de dados de empresas estrangeiras devido à facilidade de obtenção e gratuidade. Infelizmente, a BOVESPA (Bolsa de Valores de São Paulo) não possui ferramenta similar e tampouco disponibiliza os dados das empresas brasileiras em ferramentas do tipo *open source*.

4.3.1 Empresas escolhidas para o cenário de Simulação

As empresas selecionadas compõem diversos ramos de atuação, tais como entretenimento, finanças, varejo, linhas aéreas e tecnologia. Desta forma, o comportamento médio das ações destas empresas não está diretamente correlacionado - ou seja, uma eventual crise no setor de entretenimento tem pouco, senão nenhum impacto em bancos ou fundos de investimentos, por exemplo - e, na média, apresentam o crescimento do mercado como um todo.

As empresas foram escolhidas à esmo, compondo uma amostra do total de empresas que compõe a bolsa de valores Standard & Poor's 500 (S&P500), formada por ações emitidas por 500 companhias de grande porte que, por sua vez, são responsáveis por 80% do mercado de acionário norte-americano (PRESS..., 2015). As empresas escolhidas encontram-se listadas na [Tabela 1](#).

4.4 UTILIZAÇÃO DOS FILTROS DE MÉDIAS MÓVEIS

4.4.1 Implementação em Python

A filtragem de dados em Médias Móveis Simples ou Médias Móveis Exponencialmente Ponderadas encontram-se disponíveis como métodos da biblioteca `Pandas`.

O filtro MA (*Moving Average*) é criado a partir do método `.rolling()` e deve ser utilizado em uma série numérica de um objeto `DataFrame`. Uma estrutura do tipo `DataFrame` nada mais é do que uma matriz, onde cada coluna pode aceitar elementos de tipos de dados diferentes (uma coluna com objetos numéricos do tipo *float* enquanto outra coluna pode possuir objetos do tipo *string*, por exemplo).

Portanto, como pode ser observado no [Apêndice A](#), o objeto do tipo `DataFrame` é criado a partir do *download* dos dados de cada ação selecionada pelo método `.DataReader()` e então é criada uma nova coluna neste objeto, cujos elementos são os valores médios

Tabela 1 – Empresas selecionadas para implementação dos filtros de médias móveis e aplicação de métodos estocásticos.

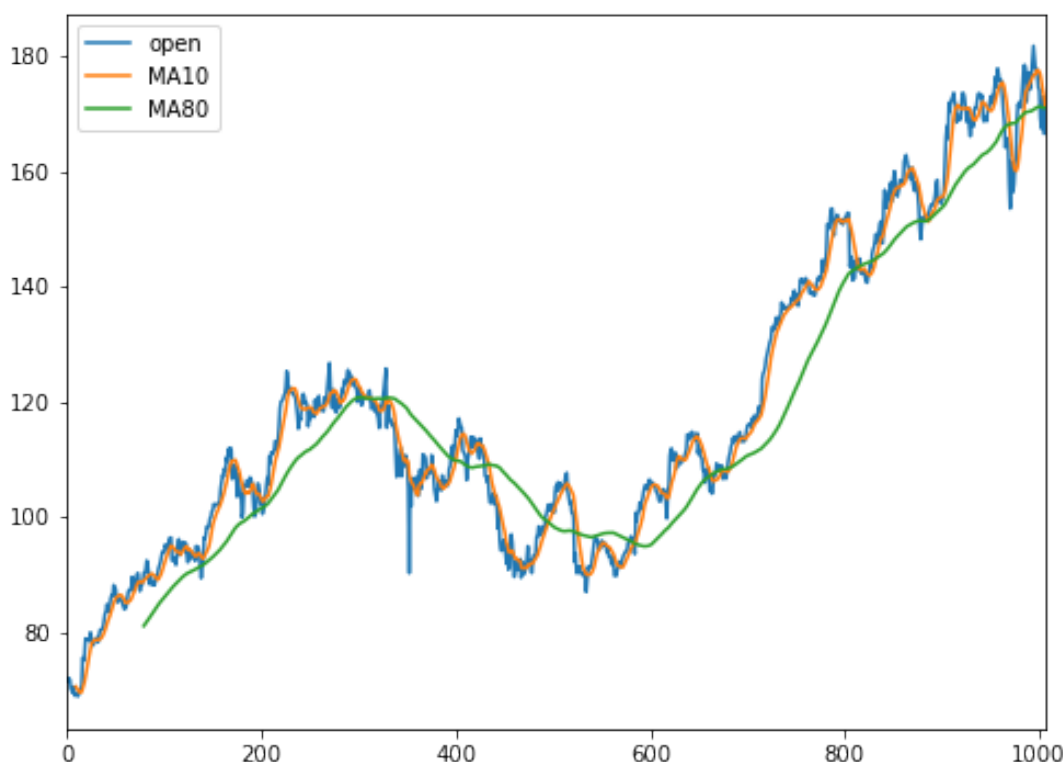
Sigla	Empresa	Sigla	Empresa
AAL	American Air Lines	AAPL	Apple Inc.
ADBE	Adobe Systems Inc.	ADSK	Autodesk Inc.
AMD	Advanced Micro Devices Inc.	AMZN	Amazon.com Inc.
ATVI	Activision Blizzard	AXP	American Express Co
BA	Boeing Company	BAC	Bank of America Corp
BBY	Best Buy Co Inc.	BLK	BlackRock
C	Citigroup Inc.	CMCSA	Comcast Corp.
COST	Costco Wholesale Corp.	CSCO	Cisco Systems
CVX	Chevron Corp.	DAL	Delta Air Lines Inc.
DIS	The Walt Disney Company	EA	Electronic Arts
EBAY	eBay Inc.	F	Ford Motor
FB	Facebook, Inc.	FDX	FedEx Corporation
GE	General Electric	GM	General Motors
GOOG	Alphabet Inc Class C	GPS	Gap Inc.
GRMN	Garmin Ltd.	GS	Goldman Sachs Group
GT	Goodyear Tire & Rubber	HAS	Hasbro Inc.
HOG	Harley-Davidson	INTC	Inter Corp.
JNJ	Johnson & Johnson	K	Kellogg Co.
KO	Coca-Cola Company	KORS	Michael Kors Holding
LMT	Lockheed Martin Corp.	M	Macy's Inc
MA	Mastercard Inc.	MAR	Marriott Int'l
MAT	Mattel Inc.	MDLZ	Mondelez International
MGM	MGM Resorts International	MMM	3M Company
MS	Morgan Stanley	MSFT	Microsoft Corp.
MSI	Motorola Solutions Inc.	NFLX	Netflix Inc.
PEP	PepsiCo Inc.	PG	Procter & Gamble
QCOM	QUALCOMM Inc.	RCL	Royal Caribbean Cruises Ltd
RL	Polo Ralph Lauren Corp.	ROST	Ross Stores
RTN	Raytheon Co.	SBUX	Starbucks Corp.
STX	Seagate Technology	SYMC	Symantec Corp.
T	AT&T Inc.	TTWO	Take-Two Interactive
TWTR	Twitter, Inc.	TXN	Texas Instruments
V	Visa Inc.	VIAB	Viacom Inc.
VRSN	Verisign Inc.	VZ	Verizon Communications
WFC	Wells Fargo	WMT	Wal-Mart Stores
WU	Western Union Co	XOM	Exxon Mobil Corp.
XRX	Xerox Corp.		

(método `.mean()`) dos elementos da janela de tamanho igual ao argumento passado ao método `.rolling()`, criando assim a Média Móvel Simples dos valores da série numérica selecionada. A sequência completa está exemplificada na [Subseção 4.4.2](#).

Por padrão, se apenas o argumento para o tamanho da janela for passado, não existirá *offset* desta janela em relação à posição do elemento de valor médio, e neste caso,

significa que os elementos da janela são todos anteriores ao valor médio calculado, desta forma, o filtro é causal e utiliza somente valores anteriores ao tempo presente como entrada de sinal. Na Figura 4, vê-se que quanto maior o tamanho da janela do filtro, mais o sinal é suavizado, e maior a inércia do filtro para alterar seu percurso ao acompanhar um sinal. Nota-se também que o primeiro valor de saída deste filtro está sempre na primeira posição ao fim da sua janela de entrada de dados.

Figura 4 – Implementação de dois filtros de médias móveis simples com sensibilidades à variação diferentes.



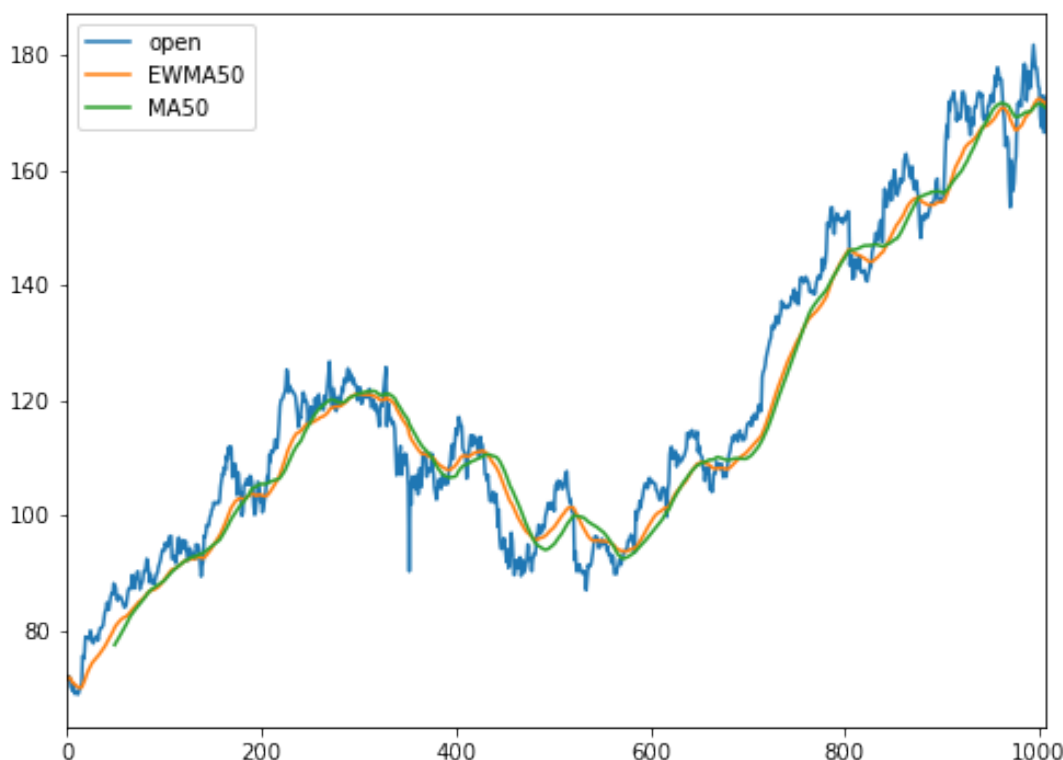
Fonte: O autor

O filtro EWMA (*Exponential Weighted Moving Average*), de maneira similar, é criado pelo método `.ewm()`, cujo argumento *span* especifica o decaimento dos termos em função de sua janela, a partir da equação (5) e novamente o método `.mean()` especifica que deseja-se o valor médio dos termos que encontram-se ao alcance do argumento *span*. Estes valores médios são também passados a uma nova coluna do objeto tipo `DataFrame`, criado para armazenar os valores da Média Móvel Exponencialmente Ponderada.

A principal diferença entre os filtros MA e EWMA é que para o filtro MA, o primeiro valor calculado é sempre aquele em posição igual à sua janela de entrada, enquanto que para o filtro EWMA, os valores médios já são calculados imediatamente em função dos dados anteriores, como pode-se observar na figura Figura 5.

Também nota-se na imagem a "velocidade de reação" do filtro EWMA, pois, uma vez que valores mais recentes possuem um peso maior para o cálculo do valor médio, o

Figura 5 – Comparação de desempenho entre os filtros do tipo MA e EWMA de mesma janela de entrada.



Fonte: O autor

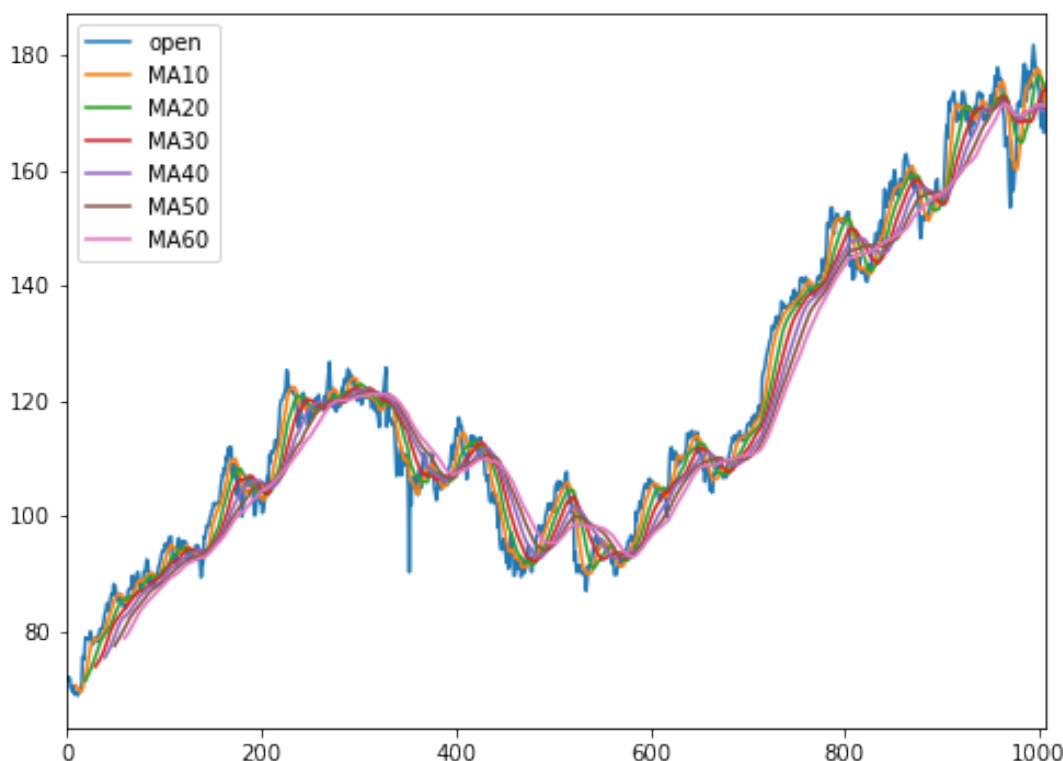
filtro responde mais rapidamente a mudanças bruscas na direção do sinal, apesar de ambos possuírem a mesma janela de entrada.

4.4.2 Estratégia para Tomada de Decisão

Com o auxílio dos filtros implementados, deve-se então definir as regras para que as operações de compra e venda sejam realizadas. O conjunto de regras para a tomada de decisão pode ser definida como a estratégia de negociação.

Com o uso dos filtros MAV e EWMA, optou-se pela estratégia clássica de realizar as operações nos pontos de inversão de tendência. Esta estratégia utiliza filtros de sensibilidades diferentes (ou seja, com janelas de entrada de tamanhos diferentes) para auxiliar na tomada de decisão. Janelas com muitos dados de entrada (filtros de janelas mais longas) possuem menos sensibilidade à variação de novos valores, uma vez que estes têm pouca relevância frente a uma grande quantidade de valores passados para o cálculo da média, e filtros de janelas curtas são bastante sensíveis à variação dos valores entradas, acompanhando mais fielmente os movimentos do sinal de entrada (neste caso, ao valor de abertura da ação). A variação da sensibilidade para diferentes comprimentos de janelas de entrada pode ser melhor observada na [Figura 6](#), sendo a sensibilidade inversamente proporcional ao comprimento da janela do filtro.

Figura 6 – Filtros de Médias Móveis Simples com diferentes janelas de entrada de dados.



Fonte: O autor

Esta estratégia, descrita no fluxograma da [Figura 7](#), e funciona da seguinte maneira:

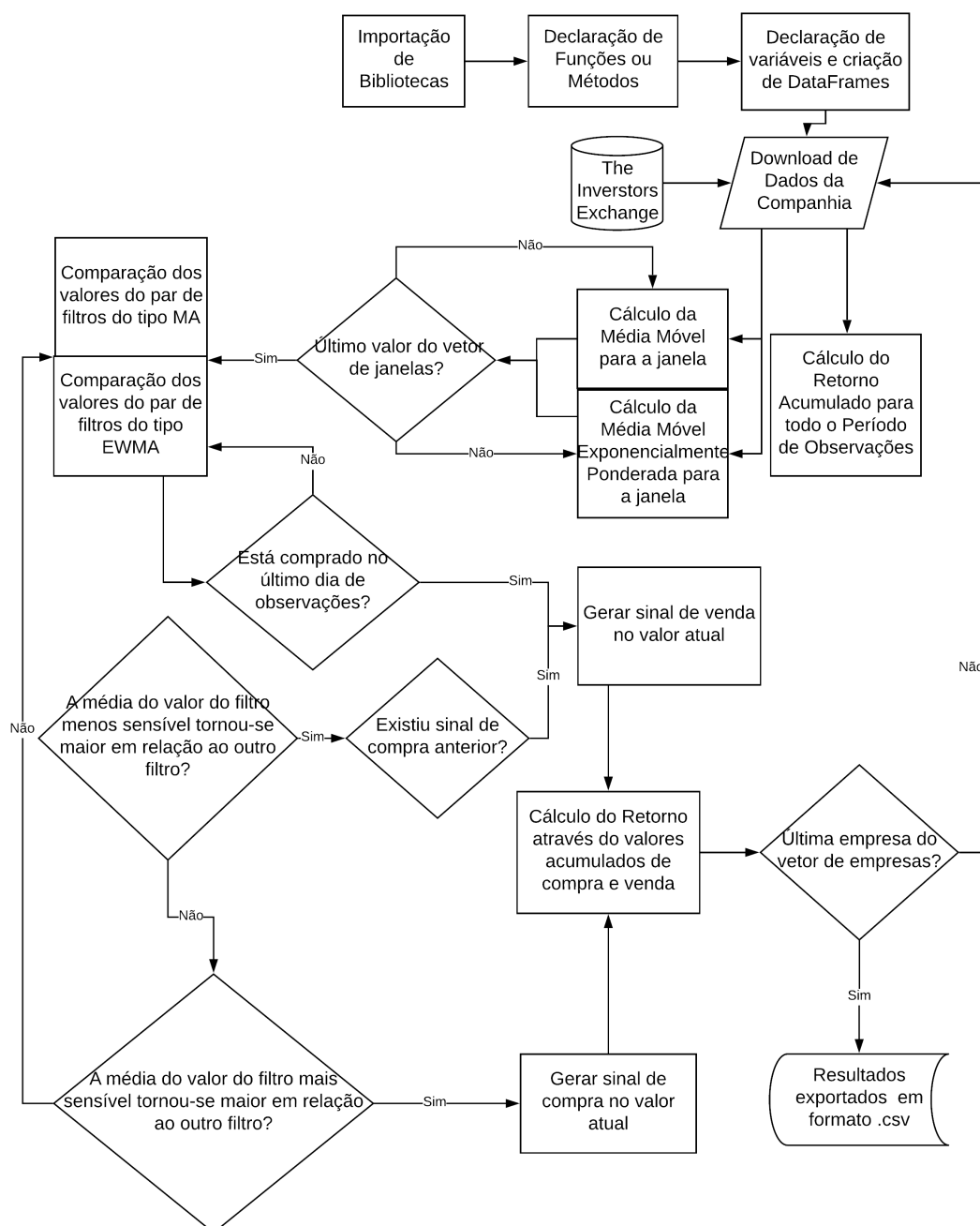
1. Observa-se a posição dos valores do par de filtros, sendo um de janela longa e outro de janela curta;
2. O cruzamento dos valores de cada um destes filtros sinaliza uma oportunidade de compra/venda deste ativo;
3. Caso o valor médio do filtro mais sensível (de menor janela) torne-se maior, então é uma oportunidade de COMPRA do ativo;
4. Caso contrário, o sinal é para que seja efetuada a VENDA do ativo;
5. Caso o ativo já esteja comprado, este será vendido no último período das observações, independente do cruzamento entre as médias móveis.

A interação entre os filtros, com as respectivas oportunidade de compra/venda do papel podem ser melhor visualizadas na [Figura 8](#).

Alguns cuidados adicionais foram tomados e implementados em código para que não houvessem situações indesejadas tais como:

- Permitir que se efetue a venda de um ativo antes de tê-lo comprado anteriormente;
- Permanecer com um ativo comprado ao último dia de observações, sem efetuar sua venda.

Figura 7 – Fluxograma do algoritmo de estratégia de negociação de ativos com utilização de médias móveis.



Fonte: O autor

4.4.3 Parâmetros Utilizados

A análise da estratégia citada na [Subseção 4.4.2](#) será realizada pela combinação de filtros MA entre si, e de filtros EWMA entre si, com diferentes tamanhos de janelas de entrada. As janelas utilizadas na simulação serão de 10, 20, 30, 40, 50 e 60 para cada filtro (tais como mostrados na [Figura 6](#), em todas as combinações de janelas possíveis entre eles (15 combinações dois a dois para os filtros do tipo MAV e 15 combinações dois a dois para

Figura 8 – Interação entre filtros de médias móveis sinalizando pontos para compra e venda do ativo, de acordo com a estratégia proposta



Fonte: O autor

os filtros do tipo EWMA), para todas as 75 empresas selecionadas em cada uma destas 30 combinações.

Os resultados do retorno obtido no período de observação serão comparados entre si, e ao retorno simples do valor da ação, em relação a seu valor no início das observações e ao final.

4.5 SIMULAÇÃO PREDITIVA PELO MÉTODO DE MONTE CARLO

4.5.1 Implementação em Python

Para a implementação do método de Monte Carlo, é necessária a criação de vetores para o armazenamento das variáveis utilizadas para o Passeio Aleatório (média e desvio padrão). Estes vetores são então convenientemente organizados como colunas de um único objeto do tipo DataFrame. A sequência de eventos para a simulação encontra-se na [Subseção 4.5.2](#).

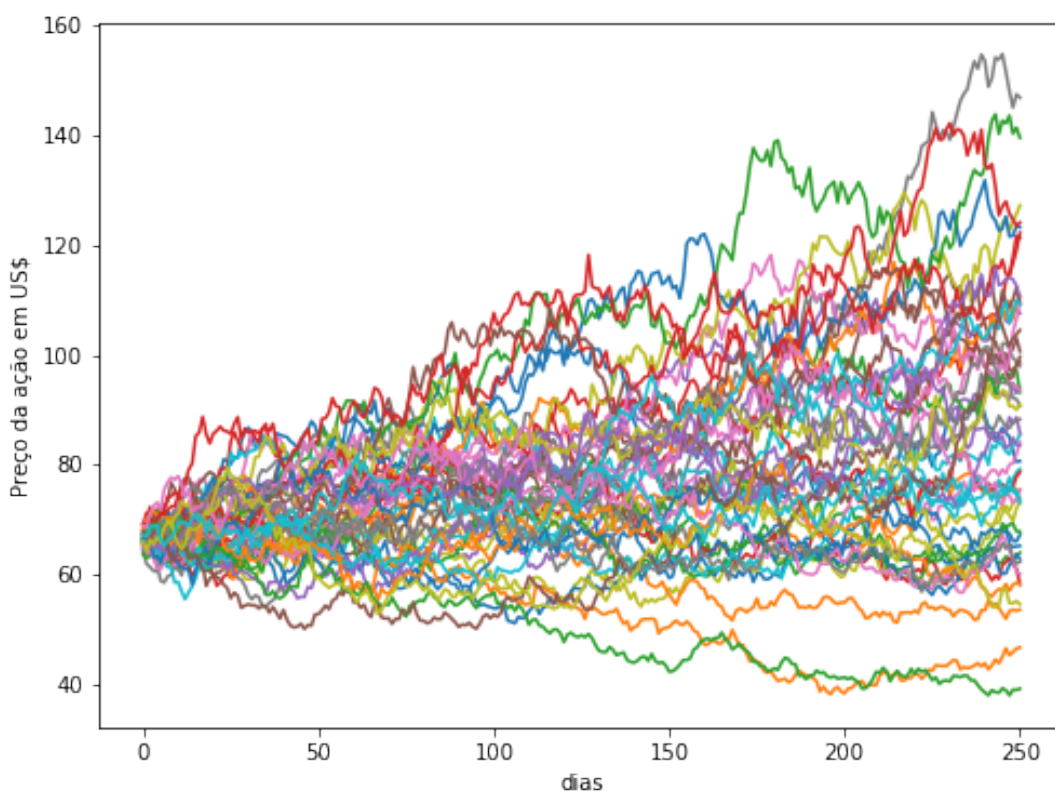
A componente aleatório é obtida a partir do método `.random()`, da biblioteca NumPy. Nas simulações efetuadas, este método é inicializado sempre com o mesmo valor de semente (`random.seed(3)`), de forma que os números aleatórios criados sejam sempre os mesmos para cada simulação. Esta pseudo-aleatoriedade é importante para fins de

debugging durante a criação do algoritmo, e assim, pode-se obter sempre os mesmos valores para diferentes compilações do código, auxiliando a busca de erros lógicos e garantindo a consistência do algoritmo.

O retorno diário é então simulado a partir da [Equação \(7\)](#) para o passeio aleatório, e as variáveis aleatórias utilizadas na multiplicação possuem distribuição normal e são criadas no formato desejado. No algoritmo, utilizou-se de uma matriz de tamanho igual ao número de períodos (ou passos) futuros pelo número de simulações a serem realizadas. Ou seja, para cada simulação do comportamento futuro da ação irá possuir um número de variáveis aleatórias normalmente distribuídas igual ao número de passos futuros a serem calculados. Esta implementação encontra-se disponível no [Apêndice B](#) deste trabalho.

Na [Figura 9](#), vê-se uma simulação de Monte Carlo com 50 passeios aleatórios criados a partir de parâmetros como média e desvio padrão quaisquer, mas constantes em todas as simulações. Nota-se que os valores para a simulação do preço de ações são sempre relativos e nunca absolutos. Desta forma, não obtém-se valores menores que 0 na simulação, o que não corresponderia à realidade do mercado acionário.

Figura 9 – Simulação de Monte Carlo com 50 Passeios Aleatórios.

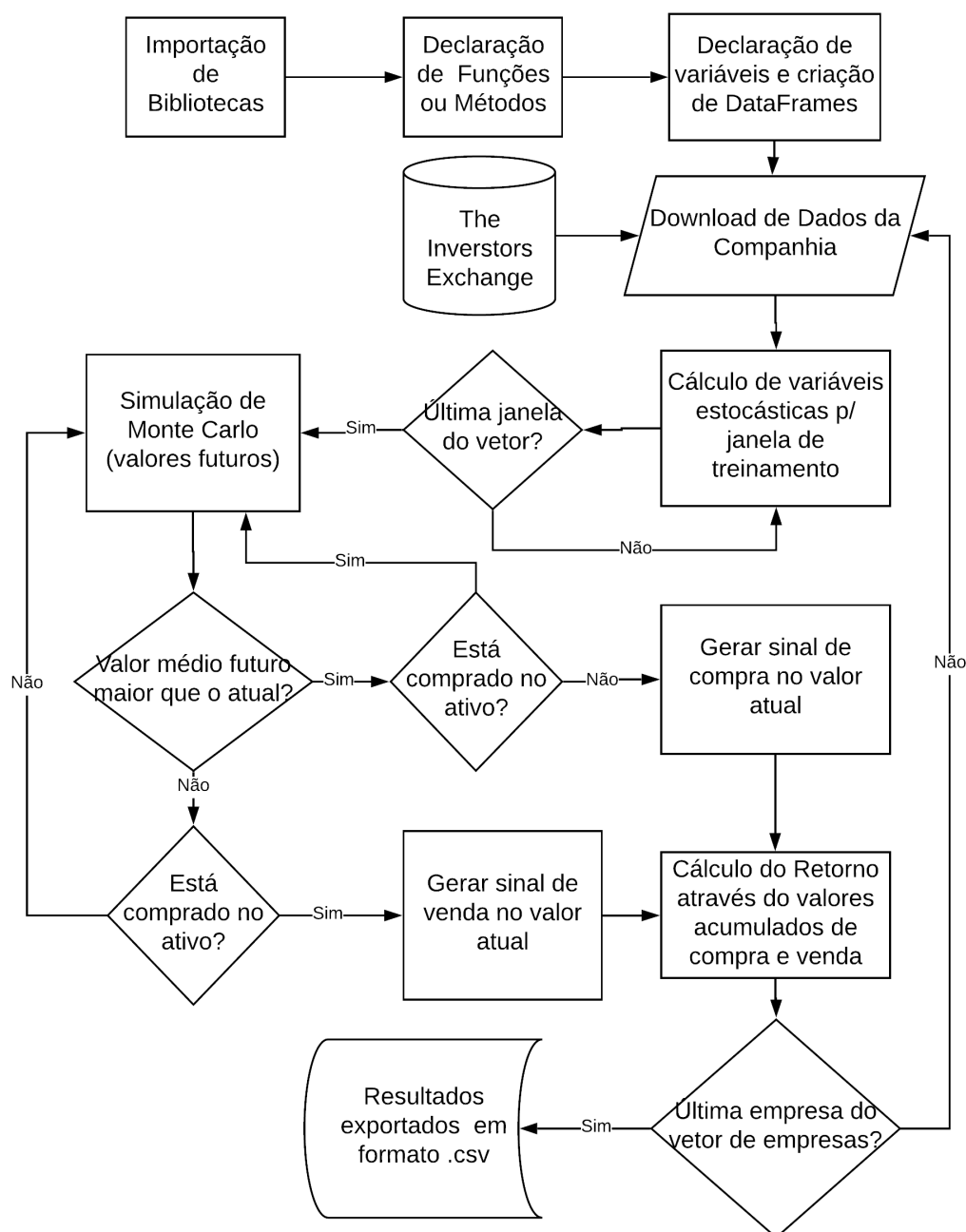


Fonte: O autor

4.5.2 Estratégia para Tomada de Decisão

A estratégia utilizada para sinalização de oportunidades de compra e venda com o auxílio de Simulações de Monte Carlo é, conforme o fluxograma da [Figura 10](#):

Figura 10 – Fluxograma do algoritmo de estratégia de negociação de ativos com utilização de simulações de Monte Carlo.



Fonte: O autor

1. Determinar os parâmetros para criar um Movimento Browniano Geométrico, a partir das observações imediatamente anteriores, conforme a janela estipulada;

2. Simular Movimentos Brownianos Geométricos a partir dos parâmetros observados (média e desvio padrão do retorno diário), tais como aqueles da [Figura 9](#);
3. Com os dados das simulações, calcular o valor médio obtido para cada passo futuro;
4. Observar o valor no N-ésimo passo futuro e compará-lo ao valor atual do ativo;
5. Caso o valor médio futuro seja maior que o valor atual, o sinal será de COMPRA, e caso já esteja comprado, assim permanecerá;
6. Caso o valor médio futuro seja menor que o valor atual, o sinal será de VENDA caso esteja comprado.
7. Caso o ativo esteja comprado no último dia de observações, o ativo será vendido no valor atual deste dia, independente de qualquer previsão futura.

4.5.3 Parâmetros Utilizados

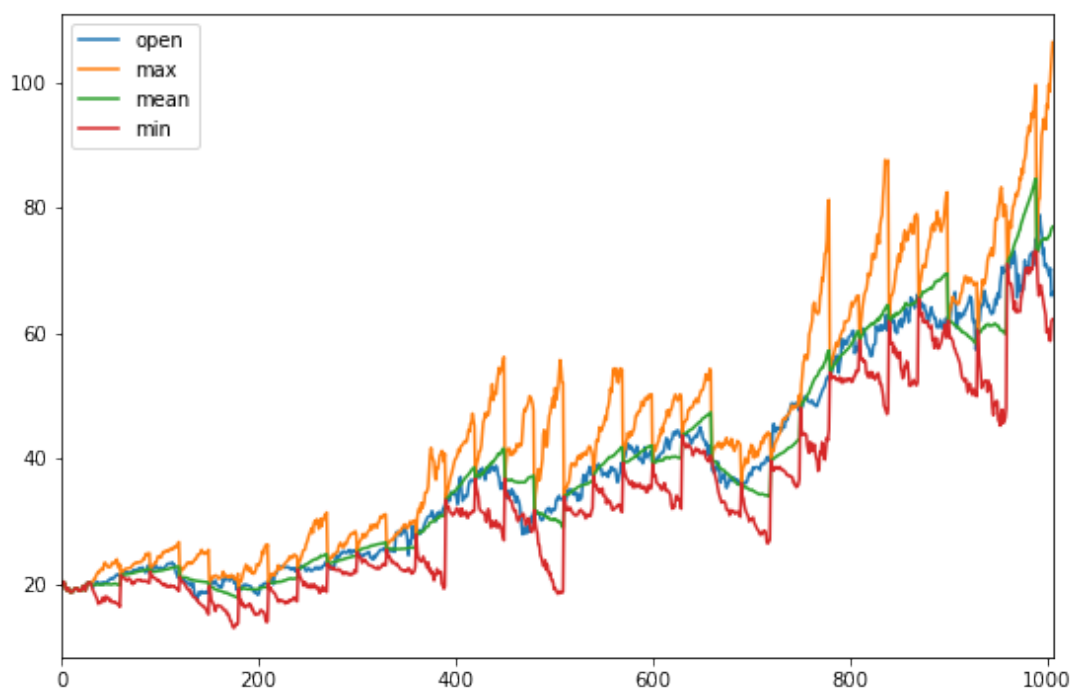
A análise da estratégia citada na [Subseção 4.5.2](#) será feita com diferentes janelas de treinamento para obtenção das variáveis estocásticas. Serão realizadas simulações com janelas de 30 períodos, tanto para treinamento como para simulação de valores futuros, e de 50 períodos, também para ambos.

O número de simulações futuras, em cada uma das janelas, será igual a 50 para ambos os casos. Como o período de observação proposto é de primeiro de abril de 2014 a 01 de abril de 2018, existem pouco mais de 1000 observações (referentes a quatro anos com 252 observações cada, dos dias úteis de abertura da bolsa de valores).

Portanto, o primeiro cenário possui 33 janelas com 30 períodos de observação, e em cada uma delas serão realizadas 50 simulações de valores futuros para a tomada de decisão de compra ou venda. O segundo cenário possui 20 janelas com 50 períodos de observação, onde novamente, em cada uma delas, serão realizadas 50 simulações de valores futuros.

Na [Figura 11](#), vê-se os instantes em que as simulações foram realizadas, formando um "cone" de valores máximos e mínimos futuros obtidos a partir destas simulações. Na janela seguinte, todos os parâmetros são recalculados com base nos valores reais obtidos na série anterior. O valor médio das simulações também está representado na imagem.

Figura 11 – Exemplo da implementação proposta para Simulações de Monte Carlo.



Fonte: O autor

5 ANÁLISE E DISCUSSÃO DOS RESULTADOS

Os resultados constantes dos [Apêndice C](#), [Apêndice D](#) e [Apêndice E](#) estão organizados de tal forma que apresentam o retorno acumulado em porcentagem (coluna Retorno) , ou seja, a valorização ou desvalorização do papel no período total para aquela ação, o retorno obtido utilizando a estratégia apresentada no [Subseção 4.4.2](#) para cada combinação de janelas de filtros de média móvel simples ([Apêndice C](#)), de média móvel exponencialmente ponderada ([Apêndice D](#)) e para diferentes janelas de treinamento e previsão estocástica ([Apêndice E](#)), conforme a estratégia discutida na [Subseção 4.5.2](#).

Independente do uso de filtros ou técnicas estocásticas, a média do retorno acumulado obtido pelo investidor que adquiriu ações de todas estas empresas no início do período, investindo igualmente em cada uma delas, e manteve-as consigo realizando o lucro apenas ao final do período observado (ou seja, 4 anos depois) foi de incríveis 78.7%. Este resultado é fruto de um mercado forte e em ascensão. Os melhores retornos acumulados no período foram obtidos com ações das empresas Amazon.com Inc (AMZN, 316%), Take-Two Interactive (TTWO, 345%) e Netflix Inc.(NFLX, 471%), enquanto os piores resultados foram das empresas Macy's Inc. (M, -42.4%), Viacom Inc. (VIAB, -60.6%) e Mattel Inc. (MAT, -60.8%).

A partir destes resultados do retorno acumulado, os resultados obtidos pela implementação dos filtros e das simulações preditivas serão discutidos.

5.1 DISCUSSÃO DOS RESULTADOS OBTIDOS COM O USO DE FILTROS DE MÉDIAS MÓVEIS

A melhor combinação de filtros obtida foi com o cruzamento de Médias de filtros EWMA com 50 e 60 janelas e auferiu, em média, 11% de ganhos neste período com sua, com ganhos de 28%, 164% e 82% para as empresas Amazon.com Inc. (AMZN), Take-Two Interactive (TTWO) e Netflix Inc.(NFLX), respectivamente.

Como os filtros suavizam os sinais de entrada, acabam limitando os possíveis ganhos por não ficarem tão expostos à volatilidade diária (ou melhor dizendo, ao ruído do sinal de entrada). Porém, a suavização do sinal funciona também em sentido contrário, de forma positiva, minimizando as perdas em papéis em queda. Isto pode ser observado nas ações das empresas que acumularam perdas no período. A utilização da mesma combinação de filtros EWMA reduziu estas perdas para -6.6% e -4.2% para papéis da Viacom Inc. (VIAB) e Macy's Inc. (M), e ainda auferiu lucro de 2.6% para a empresa Mattel Inc. (MAT), justamente a empresa com o pior desempenho de todas as empresas escolhidas para este estudo.

Outro detalhe interessante e facilmente observado na tabela de resultados é que o

desempenho médio dos filtros EWMA, independente da combinação de janelas utilizada, é sempre melhor que o obtido com o par de filtros MAV de mesma janela. Esta diferença pode ser atribuída à sensibilidade do filtro aos valores mais recentes. O filtro EWMA passa a indicar tendências de ascensão ou queda mais rapidamente, ao mesmo tempo que suaviza o sinal de entrada, eliminando o ruído gerado pela volatilidade diária do papel. Esta sensibilidade do filtro EWMA promove sinais de compra ou venda do ativo mais rapidamente, evitando perdas prolongadas decorrentes dos períodos de quedas e aproveitando melhor os períodos de valorização do papel.

Vale também ressaltar que outras combinações de filtros obtiveram resultados melhores pontualmente, mas que, na média, tiveram um desempenho pior em relação aos demais, mas isso também não significa que esta mesma combinação de filtros obterá desempenho superior ou inferior quando utilizado nesta mesma ação. Para a empresa Marriott Int'l (MAR), o par de filtros EWMA 40x60 obtiveram uma performance 4.2% superior ao do filtro EWMA 50x60 (a combinação com melhor retorno médio das 30 testadas).

A adoção desta estratégia de negociação com filtros de médias móveis parece ser mais recomendada em mercados que costumam "andar de lado", ou mesmo em períodos de crise financeira, já que minimizam prejuízos nestes períodos. Em mercados que não apresentam força o suficiente para crescer, ela também não limitaria os ganhos do investidor da mesma maneira que limitou os ganhos em papéis em forte ascensão mostrados neste trabalho.

5.2 DISCUSSÃO DOS RESULTADOS OBTIDOS COM O USO DE SIMULAÇÕES DE MONTE CARLO

Os resultados obtidos pelo uso de Simulações de Monte Carlo encontram-se no [Apêndice E](#) deste trabalho. Na média, estas implementações obtiveram um resultado bastante próximo àquele dos Filtros de Médias Móveis, com médias de ganhos de 10.4% para a simulação com 30 passos de treinamento do modelo e previsão futura e de 8.4% para o modelo com 50 janelas de treinamento e previsão.

Apesar da média final destes modelos ter sido próxima àquela obtida com Filtros de Médias Móveis, existem notáveis diferenças nos resultados individuais. A principal delas é a reversão de prejuízos com ações que tiveram performances horríveis no período. Papéis como os da empresa Gap Inc. (GPS), Twitter, Inc. (TWTR) e Michael Kors Holding(KORS) saíram de um prejuízo acumulado de 14.4%, 38.4% e 34.3%, respectivamente, para operações lucrativas em ambos os modelos, com ganhos de 19.9%, 20.3% e 19.2% para o modelo de 30 janelas, e 24.1%, 10.1% e 41.7% para o modelo de 50 janelas, respectivamente para estas mesmas empresas.

Entretanto, empresas com performances positivas acumularam perdas ou poucos ganhos com estes modelos. Empresas como a Raytheon Co. (RTN) e Amazon.com Inc.

(AMZN), que acumularam ganhos de 132% e 316%, respectivamente, sofreram perdas em ambos os modelos, de 7.2% e 0.86% para o primeiro modelo, e 25.6% e 7.1% para o segundo, para estas mesmas empresas.

6 CONCLUSÃO

6.1 CONSIDERAÇÕES FINAIS

O trabalho desenvolvido consiste em uma pequena amostra do que pode ser feito a partir de técnicas simples de tratamento de dados e processamento digital de sinais em uma área que, a princípio, parece distante do campo de atuação de um engenheiro eletricitista. Com foco, conhecimentos e pontos de vista totalmente distintos daqueles de um economista, a interação entre profissionais destas áreas tende a ser benéfica a todos. Não à toa, grandes bancos e fundos de investimento empregam engenheiros eletricitistas em seus quadros em funções chave de pesquisa, para estudar e implementar novas técnicas e métodos para tomadas de decisões otimizadas e consistentemente acertadas.

Logo, o estudo destas técnicas, e principalmente das ferramentas utilizadas para sua aplicação, são de grande importância para qualquer trabalho que exija manipulação de dados, seja para trabalhos de engenharia ou para a área financeira. A aquisição de dados a partir de dispositivos móveis ou remotos é crescente, com muitos produtos disponíveis para este fim, de sensores em redes elétricas inteligentes (redes *smart-grid* e aparelhos que compõem a "Internet das Coisas") a veículos autônomos que utilizam-se de câmeras e para detecção de obstáculos e prevenção de acidentes.

Como todos os dados, independente de sua fonte, podem ser manipulados da mesma maneira, a familiarização com as ferramentas aqui estudadas, bem como o aprendizado de sua respectiva linguagem de programação são a maior contribuição para a formação do autor no desenvolvimento deste estudo.

6.2 TRABALHOS FUTUROS

Este trabalho apresenta uma pequena amostra do que é possível realizar com a utilização de filtros e técnicas estocásticas no mercado financeiro. A partir do *framework* computacional criado para este trabalho, pode-se alterar 3 componentes principais utilizadas para a obtenção dos resultados. São eles:

- a combinação de filtros com diferentes janelas de entrada - por exemplo, com a obtenção de sinais de compra/venda provenientes de filtros híbridos EWMA e MA, ou de qualquer outro filtro que auxilie na remoção de ruído do sinal;
- com a técnica utilizada para obtenção do sinal para compra/venda do ativo (ou seja, com uma estratégia diferente baseada em maneiras diversas de obtenção dos sinais de tomada de decisão);
- e finalmente, aplicando estes filtros, técnicas e estratégias para diferentes ativos, papéis ou bolsas de valores, como a brasileira.

A utilização das técnicas e filtros pode também, por exemplo, levar em consideração

a utilização de valores dinâmicos para a janela de entrada ou para o tipo de filtro implementado, baseados no comportamento do sinal de entrada e na tomada de decisão em si (se para geração de um sinal de compra ou venda ou parametrização de variáveis do próprio filtro, ou utilização de outra técnica mais adequada para aquele momento).

As técnicas para tomada de decisão também não precisam limitar-se a analisar os ativos separadamente, um a um. Há a possibilidade da criação de modelos utilizando pares de papéis de comportamento comum (de empresas que atuam no mesmo ramo ou então de ativos da mesma empresa negociados em mercados futuros, por exemplo). Qualquer comportamento anômalo deste par poderia indicar um sinal para o investidor, que poderia apostar na reversão ao comportamento médio deste par, baseando sua estratégias em decisões tomadas sempre que este comportamento cruzar um *threshold* estabelecido em sua estratégia.

Outra variável passível de ser utilizada é o tempo de observação. Neste trabalho, optou-se por usar o valor diário de abertura de cada um dos papéis, mas o valor de entrada de sinal é arbitrário, podendo ser outros valores diários (fechamento, máximo, mínimo, médio), semanais ou *intraday* (hora em hora, ou inclusive, minuto a minuto, dependendo do volume de negociações). Vale lembrar que existem custos associados de corretagem a cada operação de compra e/ou venda, além da incidência de impostos sobre lucros auferidos, que também poderiam ser implementados nas simulações, aproximando-a de um resultado mais próximo da realidade.

Dentre as técnicas possíveis de serem aplicadas, existe uma gama de métodos de regressão para modelagem do sinal, além daqueles criados a partir de algoritmos de Aprendizado de Máquina, cadeias de Markov, inferências Bayesianas, entre outras. A utilização de todas as técnicas e estratégias citadas depende apenas da criatividade e disposição dos autores dos trabalhos futuros, pois as possibilidades de estudo e aplicação para o mercado financeiro são praticamente infinitas.

Referências

- AKANSU, A. N.; KULKARNI, S. R.; MALIOUTOV, D. **Financial Signal Processing and Machine Learning**. 1. ed. [S.l.]: John Wiley & Sons, Ltd, 2016. Citado na página 1.
- ANACONDA. 2018. Disponível em: <<https://www.anaconda.com/download>>. Citado na página 14.
- BROCK, W.; LAKONISHOK, J.; LEBARON, B. Simple technical trading rules and the stochastic properties of stock returns. **The Journal of Finance**, 1992. Citado na página 4.
- DAS, R. et al. Agent-human interactions in the continuous double auction. **IBM T.J. Watson Research Center**, 2001. Citado na página 5.
- DUHIGG, C. Stock traders find speed pays, in milliseconds. **The New York Times**, July 2009. Disponível em: <<https://www.nytimes.com/2009/07/24/business/24trading.html>>. Citado na página 5.
- FEYNMAN, R. The brownian motion. **The Feynman Lectures of Physics**, Addison-Wesley, v. 1, 1964. Citado 2 vezes nas páginas 4 e 10.
- HASTINGS, W. K. Monte carlo sampling methods using markov chains and their applications. **Biometrika**, v. 57, n. 1, p. 97–109, Abril 1970. Citado na página 9.
- JUPYTER. 2018. Disponível em: <<http://jupyter.org/install>>. Citado na página 13.
- KROESE, D. P. et al. Why the monte carlo method is so important today. **WIREs Computational Statistics**, Junho 2014. Citado na página 9.
- MATPLOTLIB. 2018. Disponível em: <<https://matplotlib.org/users/installing.html>>. Citado na página 13.
- METROPOLIS, N. The beginning of the monte carlo method. **Los Alamos Science**, p. 125, 1987. Citado na página 9.
- MILLS, T. C.; MARKELLOS, R. N. **The econometric modelling of financial time series**. [S.l.]: Cambridge University Press, 2008. Citado na página 6.
- NUMPY. 2018. Disponível em: <<https://www.scipy.org/install.html>>. Citado na página 12.
- PANDAS. 2018. Disponível em: <<https://pandas.pydata.org/getpandas.html>>. Citado na página 13.
- PEARSON, K. The problem of the random walk. **Nature**, 1905. Citado na página 10.
- POON, S.-H. **A Practical Guide to Forecasting Financial Market Volatility**. [S.l.]: John Wiley & Sons, Ltd, 2005. Citado 2 vezes nas páginas 7 e 8.
- POUNDSTONE, W. **Fortune's Formula**. [S.l.]: Will and Wang, 2005. Citado na página 4.

PRESS Release - S&P U.S. Indices Methodology Update. 2015. Disponível em: <https://www.spice-indices.com/idpfiles/spice-assets/resources/public/documents/134544_v5methodologychgmultisecondary2.pdf>. Citado na página 15.

PYTHON. 2018. Disponível em: <<https://www.python.org/downloads/>>. Citado na página 12.

ROSS, S. M. **Introduction to Probability Models**. [S.l.]: Amsterdam: Elsevier, 2014. 612 p. Citado na página 11.

SCIPY. 2018. Disponível em: <<https://www.scipy.org/install.html>>. Citado na página 13.

SPYDER. 2018. Disponível em: <<https://pythonhosted.org/spyder/installation.html>>. Citado na página 14.

VENNERS, B. **The Making of Python**. 2003. Disponível em: <<https://www.artima.com/intv/pythonP.html>>. Citado na página 12.

Apêndices

APÊNDICE A – CÓDIGO EM PYTHON PARA IMPLEMENTAÇÃO DE ESTRATÉGIA UTILIZANDO FILTROS DE MÉDIAS MÓVEIS

```
"""
```

```
Created on Tue May 12 10:35:09 2018
```

```
@author: Adriano Paul
```

```
"""
```

```
import numpy as np
```

```
import pandas as pd
```

```
import pandas_datareader.data as web
```

```
import datetime
```

```
def zero2Nan(values):
```

```
    return [float('nan') if x==0 else x for x in values]
```

```
def windowWait(series, window):
```

```
    for x in range(0, window-1):
```

```
        if series[x] != float('nan'):
```

```
            series[x] = float('nan')
```

```
    return series
```

```
def buyFirst(buy, sell):
```

```
    x = buy.first_valid_index()
```

```
    y = sell.first_valid_index()
```

```
    if x > y:
```

```
        sell[y] = float('nan')
```

```
    return sell
```

```
def sellLast(buy, sell, opening):
```

```
    x = buy.last_valid_index()
```

```
    y = sell.last_valid_index()
```

```
    if x > y:
```

```
        sell[len(sell)-1] = opening[len(sell)-1]
```

```
    return sell
```

```
def returns(buy, sell, opening):
```

```
    return ((sell.sum() - buy.sum()) / opening.first_valid_index())
```

```
inicio = datetime.datetime(2014, 4, 1)
```

```
fim = datetime.datetime(2018, 4, 1)
```

```
acao = []

companies = ["AAL", "AAPL", "ADBE", "ADSK", "AMD", "AMZN", "ATVI", "AXP", "BA",
             ", "BAC", "BBY", "BLK", "C", "CMCSA", "COST", "CSCO", "CVX", "DAL", "DIS", "EA",
             ", "EBAY", "F", "FB", "FDX", "GE", "GM", "GOOG", "GPS", "GRMN", "GS", "GT",
             ", "HAS", "HD", "HIG", "HOG", "INTC", "JNJ", "K", "KO", "KORS", "LMT", "M", "MA",
             ", "MAR", "MAT", "MDLZ", "MGM", "MMM", "MS", "MSFT", "MSI", "NFLX", "PEP", "PG",
             ", "QCOM", "RCL", "RL", "ROST", "RTN", "SBUX", "STX", "SYMC", "T", "TIWO", "TWIR",
             ", "TXN", "V", "VIAB", "VRSN", "VZ", "WFC", "WMI", "WU", "XOM", "XRX"]

windowMA = [10, 20, 30, 40, 50, 60]
windowEWMA = [10, 20, 30, 40, 50, 60]

retAcum = pd.DataFrame(index=companies)
retAcum['RetAcumulado'] = pd.Series(index=retAcum.index)

for i in range(len(windowMA)-1):
    for j in range(len(windowMA)):
        if j>i:
            retAcum['MM'+str(windowMA[i])+ 'x'+str(windowMA[j])] = pd.
                Series(index=retAcum.index)

for i in range(len(windowEWMA)-1):
    for j in range(len(windowEWMA)):
        if j>i:
            retAcum['EWMA'+str(windowEWMA[i])+ 'x'+str(windowEWMA[j])] =
                pd.Series(index=retAcum.index)

for x in range(len(companies)):

    acao.append(web.DataReader(companies[x], 'iex', inicio, fim))

    acao[x]['total'] = acao[x]['open']*acao[x]['volume']
    a = acao[x]['open'][0]

    retAcum['RetAcumulado'][x] = (acao[x]['open'][-1] - acao[x]['open']
        '[0]) / acao[x]['open'][0] * 100

    for y in windowMA:
        acao[x]['MM' + str(y)] = acao[x]['open'].rolling(y).mean()

    for i in range(len(windowMA)-1):
```

```

for j in range(len(windowMA)):
    if j>i:
        val = acao[x][ 'MM'+str(windowMA[i]) ].values > acao[x]
            [ 'MM'+str(windowMA[j]) ].values

        prox_val = val[1:]
        prox_val = np.append(prox_val, [val[-1]])

        acao[x][ 'sinalCompraMM'+str(windowMA[i])+ 'x'+str(
            windowMA[j]) ] = (val != prox_val) & (acao[x][ 'MM'+
            str(windowMA[j]) ].values > acao[x][ 'MM'+str(
            windowMA[i]) ].values)
        acao[x][ 'sinalVendaMM'+str(windowMA[i])+ 'x'+str(
            windowMA[j]) ] = (val != prox_val) & (acao[x][ 'MM'+
            str(windowMA[j]) ].values < acao[x][ 'MM'+str(
            windowMA[i]) ].values)

        acao[x][ 'sinalCompraMM'+str(windowMA[i])+ 'x'+str(
            windowMA[j]) ] = pd.Series(acao[x][ 'sinalCompraMM'+
            str(windowMA[i])+ 'x'+str(windowMA[j]) ]).shift(1)
        acao[x][ 'sinalVendaMM'+str(windowMA[i])+ 'x'+str(
            windowMA[j]) ] = pd.Series(acao[x][ 'sinalVendaMM'+
            str(windowMA[i])+ 'x'+str(windowMA[j]) ]).shift(1)

        acao[x][ 'compraMM'+str(windowMA[i])+ 'x'+str(windowMA[
            j]) ] = acao[x][ 'sinalCompraMM'+str(windowMA[i])+ 'x'+
            str(windowMA[j]) ] * acao[x][ 'open' ]
        acao[x][ 'vendaMM'+str(windowMA[i])+ 'x'+str(windowMA[j]
            ) ] = acao[x][ 'sinalVendaMM'+str(windowMA[i])+ 'x'+
            str(windowMA[j]) ] * acao[x][ 'open' ]

        acao[x][ 'compraMM'+str(windowMA[i])+ 'x'+str(windowMA[
            j]) ] = zero2Nan(acao[x][ 'compraMM'+str(windowMA[i]
            )+ 'x'+str(windowMA[j]) ])
        acao[x][ 'vendaMM'+str(windowMA[i])+ 'x'+str(windowMA[j]
            ) ] = zero2Nan(acao[x][ 'vendaMM'+str(windowMA[i])+
            'x'+str(windowMA[j]) ])

        acao[x][ 'compraMM'+str(windowMA[i])+ 'x'+str(windowMA[
            j]) ] = windowWait(acao[x][ 'compraMM'+str(windowMA[
            i])+ 'x'+str(windowMA[j]) ], j)
        acao[x][ 'vendaMM'+str(windowMA[i])+ 'x'+str(windowMA[j]

```

```

    ))] = windowWait(acao[x]['vendaMM'+str(windowMA[i]
    )]+x'+str(windowMA[j])], j)

acao[x]['vendaMM'+str(windowMA[i])+x'+str(windowMA[j]
    )] = buyFirst(acao[x]['compraMM'+str(windowMA[i])
    +x'+str(windowMA[j])], acao[x]['vendaMM'+str(
    windowMA[i])+x'+str(windowMA[j])])

acao[x]['vendaMM'+str(windowMA[i])+x'+str(j)] =
    sellLast(acao[x]['compraMM'+str(windowMA[i])+x'+
    str(windowMA[j])], acao[x]['vendaMM'+str(windowMA[i]
    )]+x'+str(windowMA[j])], acao[x]['open'])

retAcum['MM'+str(windowMA[i])+x'+str(windowMA[j])][x
    ] = ((acao[x]['vendaMM'+str(windowMA[i])+x'+str(
    windowMA[j])].sum() - acao[x]['compraMM'+str(
    windowMA[i])+x'+str(windowMA[j])].sum()) / acao[x
    ]['compraMM'+str(windowMA[i])+x'+str(windowMA[j])
    ].sum()) * 100

for y in windowEWMA:
    acao[x]['EWMA' + str(y)] = acao[x]['open'].ewm(span=y).mean()

for i in range(len(windowEWMA)-1):
    for j in range(len(windowEWMA)):
        if j>i:
            val = acao[x]['EWMA'+str(windowMA[i])].values > acao[
            x]['EWMA'+str(windowMA[j])].values

            prox_val = val[1:]
            prox_val = np.append(prox_val, [val[-1]])

            acao[x]['sinalCompraEWMA'+str(windowMA[i])+x'+str(
            windowMA[j])] = (val != prox_val) & (acao[x]['EWMA
            '+str(windowMA[j])].values > acao[x]['EWMA'+str(
            windowMA[i])].values)

            acao[x]['sinalVendaEWMA'+str(windowMA[i])+x'+str(
            windowMA[j])] = (val != prox_val) & (acao[x]['
            EWMA'+str(windowMA[j])].values < acao[x]['EWMA'+
            str(windowMA[i])].values)

acao[x]['sinalCompraEWMA'+str(windowMA[i])+x'+str(
    windowMA[j])] = pd.Series(acao[x]['sinalCompraEWMA

```

```

        '+str(windowMA[i])+ 'x'+str(windowMA[j]))).shift(1)
acao[x]['sinalVendaMM'+str(windowMA[i])+ 'x'+str(
    windowMA[j])] = pd.Series(acao[x]['sinalVendaEWMA'+
    str(windowMA[i])+ 'x'+str(windowMA[j]))).shift(1)

acao[x]['compraEWMA'+str(windowMA[i])+ 'x'+str(
    windowMA[j])] = acao[x]['sinalCompraEWMA'+str(
    windowMA[i])+ 'x'+str(windowMA[j])] * acao[x]['open']
acao[x]['vendaEWMA'+str(windowMA[i])+ 'x'+str(windowMA
    [j])] = acao[x]['sinalVendaEWMA'+str(windowMA[i])
    + 'x'+str(windowMA[j])] * acao[x]['open']

acao[x]['compraEWMA'+str(windowMA[i])+ 'x'+str(
    windowMA[j])] = zero2Nan(acao[x]['compraEWMA'+str(
    windowMA[i])+ 'x'+str(windowMA[j])])
acao[x]['vendaEWMA'+str(windowMA[i])+ 'x'+str(windowMA
    [j])] = zero2Nan(acao[x]['vendaEWMA'+str(windowMA[
    i])+ 'x'+str(windowMA[j])])

acao[x]['compraEWMA'+str(windowMA[i])+ 'x'+str(
    windowMA[j])] = windowWait(acao[x]['compraEWMA'+
    str(windowMA[i])+ 'x'+str(windowMA[j])], j)
acao[x]['vendaEWMA'+str(windowMA[i])+ 'x'+str(windowMA
    [j])] = windowWait(acao[x]['vendaEWMA'+str(
    windowMA[i])+ 'x'+str(windowMA[j])], j)

acao[x]['vendaEWMA'+str(windowMA[i])+ 'x'+str(windowMA
    [j])] = buyFirst(acao[x]['compraEWMA'+str(windowMA
    [i])+ 'x'+str(windowMA[j])], acao[x]['vendaEWMA'+str
    (windowMA[i])+ 'x'+str(windowMA[j])])
acao[x]['vendaEWMA'+str(windowMA[i])+ 'x'+str(j)] =
    sellLast(acao[x]['compraEWMA'+str(windowMA[i])+ 'x'+
    str(windowMA[j])], acao[x]['vendaEWMA'+str(
    windowMA[i])+ 'x'+str(windowMA[j])], acao[x]['open'
    ])

retAcum['EWMA'+str(windowMA[i])+ 'x'+str(windowMA[j])
    ][x] = (acao[x]['vendaEWMA'+str(windowMA[i])+ 'x'+
    str(windowMA[j])].sum() - acao[x]['compraEWMA'+str
    (windowMA[i])+ 'x'+str(windowMA[j])].sum()) / acao[
    x]['compraEWMA'+str(windowMA[i])+ 'x'+str(windowMA[
    j])].sum() * 100

```

```
result = retAcum  
result.loc['Media'] = result[result.columns].mean()  
  
result.to_csv('ResultadoFinalMM.csv')
```

APÊNDICE B – CÓDIGO EM PYTHON PARA IMPLEMENTAÇÃO DE NEGOCIAÇÃO UTILIZANDO SIMULAÇÕES DE MONTE CARLO

```
"""
```

```
Created on Tue May 15 16:58:59 2018
```

```
@author: Adriano Paul
```

```
"""
```

```
import numpy as np
import pandas as pd
import pandas_datareader.data as web
import datetime
```

```
from scipy.stats import norm
```

```
iterations = 50
window = 30
np.random.seed(3)
```

```
inicio = datetime.datetime(2014, 4, 1)
fim = datetime.datetime(2018, 4, 1)
```

```
acao = []
lista = []
dailyreturn = []
price_list = []
retorno = []
P0 = []
```

```
companies = ["AAL", "AAPL", "ADBE", "ADSK", "AMD", "AMZN", "ATVI", "AXP", "BA",
             ", "BAC", "BBY", "BLK", "C", "CMCSA", "COST", "CSCO", "CVX", "DAL", "DIS", "EA",
             ", "EBAY", "F", "FB", "FDX", "GE", "GM", "GOOG", "GPS", "GRMN", "GS", "GT",
             ", "HAS", "HD", "HIG", "HOG", "INTC", "JNJ", "K", "KO", "KORS", "LMT", "M", "MA",
             ", "MAR", "MAT", "MDLZ", "MGM", "MMM", "MS", "MSFT", "MSI", "NFLX", "PEP", "PG",
             ", "QCOM", "RCL", "RL", "ROST", "RTN", "SBUX", "STX", "SYMC", "T", "TWO", "TWIR",
             ", "TXN", "V", "VIAB", "VRSN", "VZ", "WFC", "WMT", "WU", "XOM", "XRX"]
```

```
cols = ['mu', 'var', 'drift', 'stdev', 'log_return']
rows = ['0-29', '30-59', '60-89', '90-119', '120-149', '150-179', '180-209',
        '210-239', '240-269', '270-299', '300-329', '330-359', '360-389',
```

```
390-419', '420-449', '450-479', '480-509', '510-539', '540-569', '  
570-599', '600-629', '630-659', '660-689', '690-719', '720-749', '  
750-779', '780-809', '810-839', '840-869', '870-899', '900-929', '  
930-959', '960-989']
```

```
for x in range(len(companies)):
```

```
    acao.append(web.DataReader(companies[x], 'iex', inicio, fim))
```

```
    lista.append(pd.DataFrame(columns=cols, index=rows))
```

```
    dailyreturn.append(pd.DataFrame(index=rows, columns=[1]))
```

```
    price_list.append(pd.DataFrame(columns=[1], index=rows))
```

```
    P0.append(pd.DataFrame(columns=[1], index=rows))
```

```
    acao[x].reset_index(inplace = True)
```

```
    lista[x].reset_index(inplace = True)
```

```
    dailyreturn[x].reset_index(inplace = True)
```

```
    price_list[x].reset_index(inplace = True)
```

```
    P0[x].reset_index(inplace = True)
```

```
    acao[x]['max'] = 0
```

```
    acao[x]['min'] = 0
```

```
    acao[x]['mean'] = 0
```

```
    acao[x]['buyflag'] = False
```

```
    acao[x]['buy'] = 1
```

```
    acao[x]['sellflag'] = False
```

```
    acao[x]['sell'] = 1
```

```
    acao[x]['retacumulado'] = 0
```

```
    acao[x]['dailyreturn'] = acao[x]['open'].pct_change(1).fillna(  
        value=0)
```

```
    acao[x]['total_sim'] = (1 + acao[x]['dailyreturn']).cumprod().  
        fillna(value=0)
```

```
    acao[x]['log_return'] = np.log(1+acao[x]['open'].pct_change(1)).  
        fillna(value=0)
```

```
    acao[x]['mu'] = 0
```

```
    acao[x]['var'] = 0
```

```
    acao[x]['drift'] = 0
```

```
    acao[x]['stdev'] = 0
```

```

flag = 0

aux = []
maxx = []
minn = []
mean = []

for y in range(1, len(acao[x][ 'open' ])):
    acao[x].loc[y, 'mu'] = acao[x][ 'log_return' ][:y].mean()
    acao[x].loc[y, 'var'] = acao[x][ 'log_return' ][:y].var()
    acao[x].loc[y, 'drift'] = acao[x][ 'mu' ][y] - (0.5*acao[x][ 'var'
        ' ][y])
    acao[x].loc[y, 'stdev'] = acao[x][ 'log_return' ][:y].std()
    acao[x].loc[y, 'retacumulado'] = (acao[0][ 'open' ][y] - acao[x][
        'open' ][0]) / acao[x][ 'open' ][0] * 100

for z in range(len(rows)):
    if z != len(rows):
        aux = np.log(1+acao[x].loc[window*z:window*(z+1), 'open'].
            pct_change(window))
        lista[x].loc[z, 'log_return'] = aux[window*(z+1)]
        lista[x].loc[z, 'mu'] = acao[x].loc[window*z:window*(z+1),
            'log_return'].mean()
        lista[x].loc[z, 'var'] = acao[x].loc[window*z:window*(z+1)
            , 'log_return'].var()
        lista[x].loc[z, 'drift'] = lista[x].loc[z, 'mu'] - (0.5*
            lista[x].loc[z, 'var'])
        lista[x].loc[z, 'stdev'] = acao[x].loc[window*z:window*(z
            +1), 'log_return'].std()

    else:
        aux = np.log(1+acao[x].loc[window*(z+1):, 'open'].
            pct_change(len(acao[x])%window))
        lista[x].loc[z, 'log_return'] = aux[len(acao[x])%window]
        lista[x].loc[z, 'mu'] = acao[x].loc[window*(z+1):, '
            log_return'].mean()
        lista[x].loc[z, 'var'] = acao[x].loc[window*(z+1):, '
            log_return'].var()
        lista[x].loc[z, 'drift'] = lista[x].loc[z, 'mu'] - (0.5*
            lista[x].loc[z, 'var'])
        lista[x].loc[z, 'stdev'] = acao[x].loc[window*(z+1):, '
            log_return'].std()

```

```

dailyreturn[x].loc[z,1] = np.exp(lista[x]['drift'][z] + lista
[x]['stdev'][z] * norm.ppf(np.random.rand(window,
iterations)))

P0[x].loc[z,1] = acao[x].loc[(z+1)*window, 'open']
price_list[x].loc[z,1] = np.zeros_like(dailyreturn[x].loc[z
,1])
price_list[x].loc[z,1][0] = P0[x].loc[z,1]

dailyreturn[x].loc[z,1][0] = 1

for t in range(1,window):
    price_list[x].loc[z,1][t] = price_list[x].loc[z,1][t-1]*
    dailyreturn[x].loc[z,1][t]

for t in range(0,window):
    maxx.append(price_list[x].loc[z,1][t].max())
    minn.append(price_list[x].loc[z,1][t].min())
    mean.append(price_list[x].loc[z,1][t].mean())

acao[x].loc[:window, 'max'] = acao[x].loc[:window, 'open']
acao[x].loc[:window, 'mean'] = acao[x].loc[:window, 'open']
acao[x].loc[:window, 'min'] = acao[x].loc[:window, 'open']

acao[x].loc[window:, 'max'] = maxx[:len(maxx)-(window-len(acao[x]
['open']))%window]
acao[x].loc[window:, 'mean'] = mean[:len(maxx)-(window-len(acao[x]
['open']))%window]
acao[x].loc[window:, 'min'] = minn[:len(maxx)-(window-len(acao[x]
['open']))%window]

for z in range(1,len(rows)):
    if (acao[x].loc[(z+1)*window, 'mean'] > acao[x].loc[z*window, '
open']):
        if flag == 0:
            flag = 1
            acao[x].loc[z*window, 'buyflag'] = True

    elif (acao[x].loc[(z+1)*window, 'mean'] < acao[x].loc[z*window
, 'open']):
        if flag == 1:

```

```
flag = 0
acao[x].loc[z>window, 'sellflag'] = True

elif z == len(rows):
    if flag == 1:
        acao[x].loc[-1, 'sellflag'] = True

acao[x]['buy'] = acao[x]['buyflag']*acao[x]['open']
acao[x]['buy'][acao[x]['buy'] == 0] = float('nan')

acao[x]['sell'] = acao[x]['sellflag']*acao[x]['open']

retorno.append(((acao[x]['sell'].sum()-acao[x]['buy'].sum()) /
                acao[x]['buy'].sum()) * 100)

result = pd.DataFrame(retorno)
result.to_csv('ResultadoFinalMntCarlo30.csv')
```

**APÊNDICE C – RESULTADOS OBTIDOS A PARTIR DE
COMBINAÇÕES DE FILTROS DE MÉDIAS MÓVEIS SIMPLES**

Ações	Retorno	10x20	10x30	10x40	10x50	10x60	20x30	20x40	20x50
		20x60	30x40	30x50	30x60	40x50	40x60	50x60	
AAL	43.81	-0.22	0.2	0.0	-1.71	-3.18	0.1	0.59	1.4
		-0.63	1.31	0.22	-1.15	0.07	0.61	-2.52	
AAPL	134.04	1.85	1.52	2.06	2.51	3.14	0.59	0.53	0.11
		-0.25	-0.45	0.2	-1.05	1.32	0.23	-1.51	
ADBE	223.75	3.14	4.16	7.57	10.12	9.99	5.28	7.92	7.01
		10.05	3.73	10.39	11.65	7.69	16.76	14.89	
ADSK	156.24	0.44	2.76	2.6	2.54	4.85	1.09	0.47	5.13
		6.15	4.22	8.37	9.2	7.99	4.76	2.9	
AMD	146.04	2.07	0.31	3.52	7.12	4.48	3.92	6.97	6.1
		9.14	8.59	5.3	9.5	3.76	5.6	-3.27	
AMZN	315.87	5.19	9.17	11.62	23.24	28.49	11.26	17.2	16.6
		24.47	10.75	18.97	35.35	13.55	17.85	12.97	
ATVI	234.29	1.87	5.69	9.21	10.84	10.48	4.75	9.57	14.51
		22.63	8.1	11.48	19.34	10.36	17.22	12.84	
AXP	8.61	0.91	0.68	1.36	1.51	1.6	-0.3	0.76	3.54
		2.28	1.01	1.45	3.0	1.32	3.45	2.76	
BA	185.45	5.3	13.55	12.2	15.84	14.71	11.63	15.51	20.23
		16.23	10.18	14.54	17.94	9.42	18.7	11.98	
BAC	80.58	1.53	1.7	3.66	6.52	8.32	2.05	4.76	4.88
		7.23	3.06	4.46	3.8	2.86	4.75	3.07	
BBY	197.84	2.01	1.27	2.28	5.78	4.96	3.96	5.05	7.93
		12.58	3.45	5.74	10.56	4.61	7.55	5.09	
BLK	85.47	1.75	1.69	2.52	2.25	2.43	1.9	1.76	2.95
		1.3	2.5	2.96	2.5	2.97	1.95	1.87	
C	48.14	1.52	0.56	0.4	1.37	2.22	0.99	3.03	2.78
		6.08	2.25	1.97	2.46	3.38	3.34	2.08	
CMCSA	41.53	1.52	1.23	0.71	-0.26	0.83	1.29	0.37	-0.17
		0.65	-0.81	-0.76	-0.4	-0.41	0.01	1.03	
COST	85.11	1.73	3.54	3.22	2.45	2.54	2.62	3.12	2.89
		2.51	2.46	1.44	0.5	0.52	-0.52	-1.02	
CSCO	112.21	0.1	-0.12	1.32	1.31	1.79	1.65	2.04	2.22
		3.82	2.21	4.09	8.37	3.34	6.59	5.64	
CVX	10.92	0.7	-0.12	-0.3	0.42	1.56	0.85	-0.05	0.28
		1.94	1.61	0.21	1.45	1.96	0.4	0.95	

Ações	Retorno	10x20 20x60	10x30 30x40	10x40 30x50	10x50 30x60	10x60 40x50	20x30 40x60	20x40 50x60	20x50
DAL	64.01	-0.66 -0.7	0.23 -1.02	-1.1 -0.9	-0.86 -1.05	-1.29 -0.78	0.21 -1.19	-0.18 0.44	-0.6
DIS	30.15	0.94 1.88	1.44 2.94	3.56 1.49	5.29 0.87	3.33 2.09	1.27 0.33	4.78 0.78	2.78
EA	313.22	0.37 18.36	3.22 11.97	4.02 11.94	9.38 12.86	11.44 10.9	4.7 13.5	7.65 10.28	10.25
EBAY	72.85	1.96 0.78	1.36 3.13	0.96 2.78	1.45 2.06	1.11 1.78	0.58 1.0	2.17 -0.0	3.03
F	-14.0	-1.81 -4.86	-1.1 -0.61	-1.88 -3.48	-1.77 -5.16	-1.84 -3.65	-0.33 -3.37	-1.67 -1.78	-4.7
FB	156.62	0.44 7.56	0.9 3.19	0.24 8.03	2.19 8.61	2.05 4.75	1.58 3.98	9.43 3.03	7.87
FDX	83.48	1.1 3.19	1.03 2.76	1.96 4.62	1.22 1.86	0.59 1.94	0.52 1.89	2.1 1.74	3.06
GE	-40.0	-0.5 -0.82	0.13 0.54	-1.2 -1.79	-1.24 -2.44	-0.07 -1.69	-0.87 -2.77	-1.59 -1.81	0.76
GM	22.58	-0.41 2.46	-0.55 1.22	-0.56 4.56	2.09 3.6	1.78 1.85	1.15 2.72	3.04 2.48	2.93
GOOG	81.07	-0.56 2.09	1.18 0.36	0.94 3.7	1.26 5.09	1.34 2.36	0.91 3.06	1.03 0.68	1.63
GPS	-14.42	1.37 -3.09	-0.53 -2.88	-0.5 -6.43	-0.8 -5.79	-1.6 -4.14	-0.67 -7.57	-0.69 -6.79	-2.21
GRMN	25.79	-0.53 1.99	0.07 0.08	-0.7 2.69	-0.73 1.61	-1.6 1.97	0.23 0.5	0.56 -0.03	1.12
GS	60.59	0.53 7.51	1.03 1.76	2.07 6.13	2.13 7.17	4.79 4.49	1.67 4.3	2.42 1.93	4.8
GT	5.33	0.52 -3.46	-0.19 -4.72	-1.06 -4.89	-2.19 -5.21	-3.15 -2.81	-0.57 -5.35	-1.39 -2.95	-3.2
HAS	66.65	0.75 1.04	1.02 -1.01	1.76 0.72	2.38 0.89	3.43 0.12	1.52 -0.1	0.81 -0.86	1.47
HIG	57.7	0.78 3.35	0.59 0.43	0.08 3.16	1.14 4.09	1.64 2.01	0.23 3.4	0.38 2.58	0.96
HOG	-28.35	-2.07 -2.3	-3.2 -1.16	-2.87 -3.3	-1.66 -4.76	-2.45 -3.23	-2.07 -4.76	-0.93 -4.79	-2.7
INTC	118.08	1.19 2.96	3.1 2.85	2.76 2.1	3.44 1.42	2.01 0.71	3.77 1.01	7.57 1.88	4.23
JNJ	45.2	0.46 0.43	0.33 -0.2	0.17 0.12	0.34 -0.0	-0.0 0.94	0.37 0.21	-0.2 0.49	-0.07
K	15.51	-0.11	0.21	-0.18	-0.82	-1.17	-0.72	-1.21	-0.97

Ações	Retorno	10x20 20x60	10x30 30x40	10x40 30x50	10x50 30x60	10x60 40x50	20x30 40x60	20x40 50x60	20x50
		-0.31	-0.52	-0.81	-1.75	-0.52	-0.47	0.01	
KO	28.47	-0.82	-1.33	-1.27	-0.96	-1.21	-0.68	-0.31	-0.09
		0.1	-0.24	0.24	0.5	0.29	-0.61	-0.8	
KORS	-34.31	-1.01	-0.02	-1.51	0.31	-0.09	1.11	2.08	5.83
		2.05	1.37	1.21	1.63	1.71	-2.72	-2.45	
LMT	130.06	1.08	2.69	4.8	6.23	12.58	3.27	6.95	7.58
		14.34	2.65	9.11	10.79	7.71	7.08	5.86	
M	-42.44	0.09	-0.31	-0.72	-1.88	-0.14	-1.13	-1.87	-2.62
		-3.56	-3.47	-3.17	-5.79	-1.96	-5.94	-4.94	
MA	134.83	2.8	5.15	4.59	8.35	13.43	5.67	10.19	8.81
		12.02	8.42	8.34	14.08	8.38	9.77	8.01	
MAR	151.95	2.03	3.01	2.33	5.85	6.43	4.4	5.47	6.18
		7.25	5.2	6.79	6.55	5.7	5.14	5.32	
MAT	-60.81	-3.26	-4.45	-3.99	-1.42	-2.62	-4.52	-4.49	-2.17
		-2.62	-2.45	-1.62	-2.42	-2.74	-2.18	-4.88	
MDLZ	26.52	-1.37	-1.4	-1.35	-0.44	-1.08	-0.66	-0.56	0.31
		-0.69	-0.22	0.33	-0.24	1.57	0.19	-0.07	
MGM	30.27	0.27	-1.0	-1.69	-0.83	-1.61	1.56	0.59	-0.11
		-0.77	-1.27	0.03	-0.21	3.14	1.64	-0.44	
MMM	76.95	0.62	1.45	3.36	5.01	4.61	1.99	5.09	7.08
		6.99	3.6	5.37	6.68	4.88	6.71	2.42	
MS	82.49	1.24	1.04	1.74	3.64	5.39	1.52	2.2	6.33
		6.96	2.94	9.08	9.46	7.1	9.12	1.75	
MSFT	142.26	1.95	4.18	7.01	6.59	13.05	2.41	5.92	9.06
		10.69	4.41	9.9	8.1	9.15	12.98	12.12	
MSI	77.07	0.77	1.7	1.12	1.07	1.17	2.18	3.32	3.82
		3.95	3.23	2.57	2.94	1.04	1.04	0.05	
NFLX	471.15	5.3	6.4	8.49	8.54	9.76	7.24	9.5	13.92
		12.23	9.79	9.01	12.45	5.81	15.16	7.77	
PEP	48.54	-0.25	-0.52	0.08	-0.49	-0.69	1.37	-0.11	0.18
		-0.28	-0.09	-0.91	-0.69	-0.49	-0.97	-0.21	
PG	11.83	0.17	1.3	1.17	1.07	-0.17	1.05	2.08	2.84
		2.57	1.44	2.81	-0.77	0.14	-0.03	-0.97	
QCOM	-21.1	-0.3	0.52	-0.25	-1.09	-0.8	-0.3	-2.44	-2.62
		-1.67	-1.42	-1.96	-1.48	-2.25	0.89	1.4	
RCL	129.95	0.79	0.88	0.8	1.52	3.46	0.46	1.96	3.12
		5.57	3.27	6.11	8.44	6.12	5.2	7.24	
RL	-25.8	-1.27	-1.66	-2.27	-1.53	-2.47	0.71	0.64	-0.56
		-0.97	-0.26	-0.66	-1.38	-1.7	-4.74	-5.14	

Ações	Retorno	10x20 20x60	10x30 30x40	10x40 30x50	10x50 30x60	10x60 40x50	20x30 40x60	20x40 50x60	20x50
ROST	125.69	1.59	3.57	4.8	8.74	7.56	2.76	5.07	7.0
		6.07	4.7	10.59	11.41	5.16	7.71	6.78	
RTN	131.74	1.58	3.93	3.17	2.59	4.67	3.31	4.85	4.1
		9.21	4.57	6.39	9.26	5.57	8.75	6.36	
SBUX	65.56	0.63	0.77	-0.34	2.12	1.54	1.78	3.54	3.1
		2.66	2.51	3.96	4.9	1.37	1.62	1.1	
STX	26.99	2.92	4.6	4.36	4.48	5.54	1.79	3.1	2.45
		8.37	2.77	7.21	8.52	3.7	1.75	2.83	
SYMC	73.88	1.32	1.36	2.43	0.04	0.78	1.09	0.93	-1.15
		-1.69	-0.09	-1.23	0.75	0.95	1.86	1.35	
T	23.76	0.07	-0.07	-0.29	0.3	0.99	0.43	-0.24	-1.72
		-2.24	-0.62	-0.37	-0.16	1.37	0.46	0.51	
TTWO	345.46	9.76	4.06	8.47	17.62	23.27	5.78	9.6	13.63
		21.34	9.08	15.03	24.89	16.39	28.91	25.9	
TWTR	-38.43	-0.43	-2.61	1.14	1.95	-1.3	-0.85	2.86	1.07
		0.82	4.13	-1.31	0.68	0.15	-0.85	-1.67	
TXN	140.76	1.9	2.25	3.39	5.44	9.46	1.75	5.35	6.29
		9.77	6.17	9.21	13.72	9.17	8.79	7.11	
V	122.94	1.81	2.23	5.33	5.15	6.81	2.93	4.11	6.87
		8.1	3.71	5.05	6.93	4.79	6.38	8.2	
VIAB	-60.59	-2.74	-2.67	-4.16	-6.26	-6.85	-3.17	-5.0	-5.62
		-6.66	-2.75	-3.45	-6.21	-5.4	-8.13	-4.7	
VRSN	116.22	1.4	2.5	2.17	2.56	3.24	3.29	3.91	4.79
		3.11	3.25	5.35	2.95	3.96	2.98	1.75	
VZ	21.79	-0.05	-0.86	-1.11	-1.01	-1.41	-0.77	-1.45	-1.96
		-3.19	-1.72	-1.78	-2.92	-0.31	-1.83	-2.59	
WFC	16.85	0.16	0.15	-0.49	-1.48	-2.24	-0.66	0.22	-1.49
		-1.98	-0.7	-1.05	-2.94	-2.14	-1.99	-1.53	
WMT	27.97	1.26	2.41	3.5	5.8	4.54	1.55	3.6	3.39
		6.22	2.47	3.58	9.48	3.67	9.27	7.76	
WU	34.16	-0.24	0.46	-0.49	-0.91	-0.51	-0.35	-0.71	-0.13
		-1.78	0.92	-1.94	-2.29	-1.92	-2.64	-2.34	
XOM	-13.1	-0.16	-0.86	-1.47	-2.04	-1.56	-1.2	-1.17	-1.86
		-2.1	-0.48	-1.58	-0.58	-1.51	-1.0	-2.75	
XRX	8.96	-1.52	-2.12	-2.18	-1.79	-2.69	-1.05	-2.43	-2.6
		-4.98	-1.98	-2.64	-4.99	-2.1	-3.39	-1.71	
Media	78.69	0.88	1.33	1.73	2.69	3.15	1.57	2.76	3.24
		3.95	2.18	3.2	4.06	2.6	3.22	2.2	

**APÊNDICE D – RESULTADOS OBTIDOS A PARTIR DE
COMBINAÇÕES DE FILTROS DE MÉDIAS MÓVEIS
EXPONENCIALMENTE PONDERADAS**

Ações	Retorno	10x20 20x60	10x30 30x40	10x40 30x50	10x50 30x60	10x60 40x50	20x30 40x60	20x40 50x60	20x50
AAL	43.81	0.9 -0.29	0.83 -0.91	0.78 0.12	-0.34 -0.31	-0.24 -0.45	-0.33 -0.23	-1.72 -0.8	-0.29
AAPL	134.04	3.36 7.31	5.44 5.27	6.28 7.63	5.56 9.8	6.85 9.04	4.5 9.89	5.89 20.83	6.61
ADBE	223.75	4.96 16.88	5.8 20.72	8.06 24.39	9.14 20.26	8.38 21.26	10.14 34.44	10.72 43.57	15.48
ADSK	156.24	3.7 6.82	3.27 7.25	4.91 6.23	6.5 6.59	6.37 8.52	5.91 6.57	5.06 6.99	5.97
AMD	146.04	5.5 2.46	4.54 5.28	3.26 1.18	3.31 1.59	5.27 -0.55	5.65 0.11	5.53 3.09	4.11
AMZN	315.87	11.92 19.13	14.26 24.41	18.35 22.64	20.9 18.35	20.41 19.77	27.17 25.59	26.04 27.69	24.61
ATVI	234.29	6.96 19.58	10.16 12.86	16.07 18.76	18.02 22.6	15.62 31.04	19.58 29.46	16.11 25.79	13.14
AXP	8.61	0.98 0.99	1.44 0.1	0.41 -0.0	1.84 4.81	1.79 3.59	1.74 2.71	0.73 2.82	0.5
BA	185.45	9.88 13.62	15.13 14.81	12.74 13.01	14.1 15.94	11.21 21.73	13.89 25.15	15.82 24.45	14.85
BAC	80.58	4.05 4.62	4.66 4.49	7.11 4.79	8.05 4.03	6.42 4.7	6.44 8.74	5.36 10.6	4.92
BBY	197.84	2.76 11.19	3.35 18.12	3.84 15.03	8.92 15.68	6.06 15.62	4.42 16.44	8.23 30.06	11.95
BLK	85.47	1.98 7.95	2.42 6.27	2.71 7.22	4.02 10.39	4.56 10.18	3.55 15.45	3.75 13.92	3.96
C	48.14	2.1 4.76	1.04 5.94	2.1 5.61	2.46 5.93	3.51 5.7	2.01 8.19	5.4 5.12	7.02
CMCSA	41.53	2.51 3.69	1.16 2.7	1.96 3.45	1.89 3.06	1.78 3.06	1.77 0.65	0.92 -0.16	2.5
COST	85.11	3.34 4.4	4.71 5.75	5.36 5.73	4.32 4.11	6.77 4.17	4.8 3.54	5.29 2.97	4.94
CSCO	112.21	2.48 5.03	2.45 5.1	2.09 6.04	2.39 7.15	2.79 7.23	1.93 4.9	2.72 5.01	2.97
CVX	10.92	1.43	1.42	1.25	1.99	1.56	2.17	1.82	3.08

Ações	Retorno	10x20 20x60	10x30 30x40	10x40 30x50	10x50 30x60	10x60 40x50	20x30 40x60	20x40 50x60	20x50
		4.21	2.82	3.9	2.97	2.85	2.83	2.01	
DAL	64.01	1.77	0.57	0.05	-0.28	0.04	-0.46	0.44	0.31
		-0.57	0.11	-0.19	0.62	-0.18	-1.56	-1.79	
DIS	30.15	4.84	4.36	4.98	4.28	4.27	4.64	2.47	2.11
		2.63	2.5	2.37	2.54	2.84	1.52	2.9	
EA	313.22	6.5	5.39	8.03	6.98	6.63	10.94	10.73	11.63
		7.54	11.92	9.84	10.54	10.72	21.37	19.77	
EBAY	72.85	2.8	3.1	2.09	1.91	0.93	1.56	2.45	2.01
		3.09	2.25	4.48	5.26	3.85	3.6	4.58	
F	-14.0	-0.55	-0.77	-0.49	-0.72	-0.91	-0.64	-1.63	-2.26
		-2.32	-2.67	-3.08	-3.74	-4.14	-4.53	-4.4	
FB	156.62	2.98	2.64	3.61	5.1	4.31	3.68	6.66	5.31
		5.4	5.25	7.18	15.5	14.5	22.03	33.18	
FDX	83.48	2.05	1.47	2.02	2.6	3.48	1.9	5.05	9.71
		6.4	10.99	5.62	8.01	7.77	9.63	14.47	
GE	-40.0	0.8	0.34	0.07	0.09	0.82	0.99	0.91	0.26
		-0.0	-0.56	-1.33	-1.08	-1.35	-1.14	-0.58	
GM	22.58	1.07	0.42	0.95	1.64	1.05	2.05	1.37	0.76
		0.65	0.5	0.76	0.46	1.87	0.84	0.05	
GOOG	81.07	0.96	0.65	1.79	1.57	3.19	2.61	3.89	3.93
		4.36	3.04	2.41	1.88	2.42	5.79	7.67	
GPS	-14.42	1.51	1.27	0.45	-0.4	-0.78	-0.45	-2.5	-0.95
		-1.09	0.19	-1.47	-1.98	-4.39	-5.64	-6.49	
GRMN	25.79	1.06	0.97	1.02	0.73	1.17	-0.14	-0.48	0.78
		0.77	0.44	1.76	4.41	4.11	3.92	4.85	
GS	60.59	2.96	3.52	2.98	4.95	7.88	4.43	6.59	5.25
		5.19	5.65	4.68	5.82	6.26	8.48	10.54	
GT	5.33	2.84	0.89	-1.21	-1.17	-2.0	-1.28	-2.47	-2.87
		-3.6	-3.95	-4.92	-5.04	-4.71	-4.89	-5.13	
HAS	66.65	2.37	2.9	3.68	4.11	4.44	5.26	4.07	4.86
		3.29	4.43	2.37	2.26	1.01	2.12	2.0	
HIG	57.7	1.73	2.47	2.01	2.22	2.1	1.78	1.75	2.42
		3.83	3.55	5.56	9.37	8.78	6.72	6.28	
HOG	-28.35	0.1	-1.28	-0.45	-1.7	-1.35	-0.53	-1.26	-1.6
		-0.85	-1.69	-0.6	-1.67	-0.88	-1.37	-3.19	
INTC	118.08	3.84	4.23	4.22	4.99	5.49	4.9	6.07	6.58
		5.55	6.15	4.84	5.5	5.37	5.92	9.1	
JNJ	45.2	1.29	1.87	1.76	2.53	2.63	2.3	2.48	1.58
		1.94	1.8	2.02	1.69	1.76	2.32	5.02	

Ações	Retorno	10x20 20x60	10x30 30x40	10x40 30x50	10x50 30x60	10x60 40x50	20x30 40x60	20x40 50x60	20x50
K	15.51	0.34	0.85	1.12	0.45	-0.1	0.09	-0.41	0.44
		0.93	0.45	1.87	1.57	1.38	1.39	2.28	
KO	28.47	0.35	0.07	-0.17	0.09	0.53	-0.25	-0.02	0.24
		1.4	0.76	1.77	1.51	2.34	2.15	1.93	
KORS	-34.31	0.69	0.4	1.15	1.84	1.64	2.51	2.42	3.41
		1.4	3.42	1.67	0.31	0.26	-0.1	-0.32	
LMT	130.06	3.36	3.53	8.68	8.53	6.86	9.83	8.91	17.57
		16.65	16.91	16.25	14.51	15.4	15.71	15.58	
M	-42.44	0.96	0.44	-0.71	-0.55	0.71	0.02	0.86	-0.57
		0.2	-0.49	0.46	-0.45	-1.23	-3.33	-4.24	
MA	134.83	4.51	6.42	7.65	6.45	10.27	6.92	10.3	11.03
		13.34	11.83	11.87	14.86	12.76	15.52	14.51	
MAR	151.95	4.55	5.16	7.54	7.01	7.39	7.56	7.74	7.16
		10.69	10.41	10.69	16.78	16.31	21.4	17.19	
MAT	-60.81	-1.04	-1.67	-0.65	-1.76	-1.79	-0.96	-2.3	-3.95
		-2.97	-3.8	-2.7	-1.43	-1.87	-1.48	2.63	
MDLZ	26.52	0.4	-0.05	-0.36	-0.16	-0.94	-0.02	-0.72	-0.81
		-0.83	-0.25	-0.92	-0.51	-0.7	-1.43	-0.94	
MGM	30.27	0.08	-0.44	0.25	-0.67	0.03	-0.16	-0.38	-0.49
		-0.67	-0.49	0.2	-0.29	-1.7	-0.66	-0.24	
MMM	76.95	3.62	3.49	3.27	3.35	2.96	5.3	7.28	4.58
		6.02	5.17	7.75	9.36	10.56	8.78	8.67	
MS	82.49	3.71	3.28	2.95	5.05	4.08	4.74	4.29	5.73
		7.04	7.93	7.87	5.7	8.76	9.03	16.98	
MSFT	142.26	5.15	6.81	7.12	8.06	7.36	10.55	9.09	9.02
		9.71	8.29	12.57	17.22	17.92	22.03	22.88	
MSI	77.07	3.34	3.35	3.04	2.09	2.49	3.21	2.74	2.43
		2.01	1.92	2.52	3.71	3.72	4.11	4.62	
NFLX	471.15	9.03	10.35	13.42	14.82	14.81	11.38	13.0	16.57
		23.1	15.55	35.82	46.16	41.93	56.37	82.04	
PEP	48.54	0.83	0.91	0.62	0.27	1.19	0.56	1.01	1.25
		2.04	1.72	1.83	1.31	0.94	0.91	0.86	
PG	11.83	1.58	1.17	2.16	1.18	0.93	1.12	1.16	1.17
		0.83	0.53	0.94	0.75	0.85	0.92	0.7	
QCOM	-21.1	1.12	0.2	0.36	0.24	-0.23	-0.01	-1.22	-1.14
		-0.14	-0.53	0.78	1.41	1.35	-0.98	-2.72	
RCL	129.95	2.32	3.49	4.37	4.21	8.55	4.47	6.5	9.36
		15.78	9.23	19.02	20.42	27.24	24.44	35.11	
RL	-25.8	-0.26	0.45	0.73	0.68	0.35	-0.02	-0.91	-0.52

Ações	Retorno	10x20 20x60	10x30 30x40	10x40 30x50	10x50 30x60	10x60 40x50	20x30 40x60	20x40 50x60	20x50
		0.78	-0.04	-0.77	-0.55	-0.68	0.05	1.21	
ROST	125.69	5.0	5.61	7.76	6.58	8.95	8.98	9.29	11.64
		11.23	11.22	12.24	11.04	11.2	10.42	16.07	
RTN	131.74	2.25	2.65	3.31	4.82	7.09	3.66	5.21	8.16
		18.77	11.28	18.86	20.82	21.11	40.13	39.83	
SBUX	65.56	1.49	2.02	2.32	1.68	1.93	1.2	2.55	1.85
		1.84	1.29	1.49	2.11	1.5	2.04	2.53	
STX	26.99	5.26	6.38	11.6	10.36	10.62	7.77	7.0	12.38
		8.35	11.58	12.91	12.48	12.76	14.97	19.35	
SYMC	73.88	2.51	2.47	2.92	4.32	4.59	2.47	2.71	3.65
		5.15	3.33	2.66	2.51	3.41	3.68	6.34	
T	23.76	1.14	0.8	0.82	0.67	0.27	0.27	0.61	0.25
		0.36	-0.04	-0.18	-0.7	-0.67	-1.19	-2.29	
TTWO	345.46	6.17	9.46	9.02	11.35	16.68	13.88	23.05	27.58
		22.91	22.31	19.79	110.14	111.1	165.63	163.77	
TWTR	-38.43	1.26	3.57	3.78	2.17	-0.26	3.91	-2.02	-0.95
		-1.14	-2.07	-2.58	-2.59	-2.74	-4.6	-4.81	
TXN	140.76	3.7	5.08	7.59	11.38	11.59	11.87	9.72	14.64
		13.93	14.3	15.44	15.54	18.67	18.43	23.03	
V	122.94	2.55	4.05	5.73	5.71	7.1	4.84	7.19	7.23
		8.48	7.67	10.15	11.16	10.98	11.46	18.5	
VIAB	-60.59	-0.38	-1.11	-1.81	-2.12	-2.18	-2.01	-1.91	-2.12
		-4.15	-3.95	-4.77	-4.02	-4.17	-5.23	-6.65	
VRSN	116.22	3.47	2.83	4.12	5.85	4.73	5.26	5.94	5.44
		6.56	6.48	4.97	8.44	8.9	6.8	7.48	
VZ	21.79	0.77	0.29	0.2	0.08	-0.1	0.43	0.02	-0.43
		-0.94	-1.22	-1.74	-1.18	-1.15	-1.8	-1.59	
WFC	16.85	1.42	0.38	0.02	0.22	0.46	0.79	-0.42	-0.25
		-0.79	-0.39	-1.26	-2.37	-2.31	-2.44	-1.97	
WMT	27.97	3.46	3.34	3.33	3.96	4.14	5.29	6.36	6.02
		14.21	8.56	12.29	9.79	10.76	11.32	10.22	
WU	34.16	0.58	-0.31	-0.17	-0.34	-0.26	-0.56	-0.3	-0.3
		-0.72	-0.96	-1.29	-1.8	-1.26	-0.49	2.61	
XOM	-13.1	0.94	0.83	-0.02	0.01	-0.02	-0.67	-0.72	-0.53
		-0.24	-0.63	-1.34	-0.59	0.15	-1.25	-0.45	
XRX	8.96	1.11	0.03	0.22	-0.12	-0.86	-0.96	-1.2	-1.67
		-1.71	-2.36	-2.27	-3.87	-3.92	-2.16	-2.49	
Media	78.69	2.68	2.92	3.51	3.84	4.02	4.01	4.21	4.78
		5.2	4.96	5.6	7.47	7.75	9.59	11.26	

**APÊNDICE E – RESULTADOS OBTIDOS A PARTIR DA APLICAÇÃO
DE MÉTODOS ESTOCÁSTICOS E SIMULAÇÕES DE MONTE CARLO**

Index	Retorno (%)	30 Janelas (%)	50 Janelas (%)
AAL	43.81	16.45	-2.63
AAPL	134.04	18.1	-8.48
ADBE	223.75	-5.52	-47.59
ADSK	156.24	0.14	-8.98
AMD	146.04	41.11	94.75
AMZN	315.87	-0.86	-7.11
ATVI	234.29	0.37	-4.35
AXP	8.61	12.99	16.54
BA	185.45	-4.76	-25.28
BAC	80.58	21.03	-11.32
BBY	197.84	23.34	-8.72
BLK	85.47	18.86	-11.95
C	48.14	17.78	25.47
CMCSA	41.53	12.17	23.15
COST	85.11	20.41	-7.21
CSCO	112.21	-6.5	-8.63
CVX	10.92	17.48	16.52
DAL	64.01	13.7	15.79
DIS	30.15	12.48	14.78
EA	313.22	1.16	-5.41
EBAY	72.85	-1.93	-9.43
F	-14.0	9.65	9.29
FB	156.62	17.93	36.33
FDX	83.48	14.0	42.05
GE	-40.0	7.72	10.4
GM	22.58	13.69	24.5
GOOG	81.07	15.47	22.08
GPS	-14.42	19.88	24.12
GRMN	25.79	14.25	-19.0
GS	60.59	-2.54	-10.08
GT	5.33	16.8	21.09
HAS	66.65	-1.48	31.16
HD	141.43	23.51	23.53
HIG	57.7	10.98	13.87
HOG	-28.35	10.16	13.63

INTC	118.08	-1.66	-6.76
JNJ	45.2	8.75	15.72
K	15.51	-5.16	11.15
KO	28.47	7.36	10.0
KORS	-34.31	19.16	41.72
LMT	130.06	-9.75	-23.34
M	-42.44	4.13	1.93
MA	134.83	-8.61	-25.3
MAR	151.95	24.07	-4.66
MAT	-60.81	1.12	18.97
MDLZ	26.52	8.6	10.89
MGM	30.27	14.6	-8.08
MMM	76.95	12.71	26.46
MS	82.49	18.18	-12.1
MSFT	142.26	-10.85	-24.57
MSI	77.07	-3.63	-10.69
NFLX	471.15	3.78	-0.87
PEP	48.54	8.89	11.74
PG	11.83	9.88	9.07
QCOM	-21.1	13.35	18.62
RCL	129.95	37.06	-5.42
RL	-25.8	12.62	-3.17
ROST	125.69	21.6	31.9
RTN	131.74	-7.22	-25.59
SBUX	65.56	12.55	16.36
STX	26.99	6.25	0.68
SYMC	73.88	1.94	19.15
T	23.76	8.91	10.92
TTWO	345.46	34.7	217.96
TWTR	-38.43	20.28	10.08
TXN	140.76	29.4	-22.56
V	122.94	26.28	-23.05
VIAB	-60.59	1.43	-3.95
VRSN	116.22	-2.19	-8.84
VZ	21.79	10.8	8.7
WFC	16.85	9.68	12.99
WMT	27.97	14.3	31.65
WU	34.16	8.4	17.89
XOM	-13.1	7.57	8.57
XRX	8.96	12.84	-7.34
Media	78.69	10.4	8.4
