

Universidade Federal do Paraná
Setor de Ciências Exatas
Departamento de Estatística
Programa de Especialização em *Data Science* e *Big Data*

Jhonattan Lino

Aplicação de High Performance Computing em Finanças

**Curitiba
2024**

Jhonattan Lino

Aplicação de High Performance Computing em Finanças

Monografia apresentada ao Programa de Especialização em *Data Science* e *Big Data* da Universidade Federal do Paraná como requisito parcial para a obtenção do grau de especialista.

Orientador: Prof. Dr. Marco A. Zanata Alves

Curitiba
2024

Aplicação de High Performance Computing em Finanças

Application of High Performance Computing in Finance

Lino, Jhonattan¹, Zanata, Marco²

¹Aluno do programa de Especialização em Data Science & Big Data, jhonattanlino@outlook.com

²Professor do Departamento de Informática, mazalves@inf.ufpr.br

Este estudo analisa a eficácia de algoritmos de *High Frequency Trading* (HFT) aplicando técnicas de *Machine Learning*, com foco nos modelos de Regressão Logística e *Random Forest*. Os dados utilizados foram coletados da Binance Exchange, considerando transações de criptomoeda Solana. A análise incluiu a consideração do tempo de treinamento, latência e tratamento dos dados. Os resultados mostraram que, sem considerar a latência, o modelo *Random Forest* alcançou uma rentabilidade de 138,625%, enquanto a Regressão Logística atingiu 66,774%. No entanto, ao incluir o tempo de execução, a rentabilidade final do *Random Forest* caiu para 6,920%, enquanto a Regressão Logística apresentou um desempenho negativo de -0,58%. Esses resultados destacam a importância de considerar a latência em estratégias de HFT e indicam que o *Random Forest* é mais robusto em termos de precisão e rentabilidade, embora a latência permaneça um desafio significativo.

Palavras-chave: *High Frequency Trade*. Computação de alta performance. Aprendizado de Máquina. Criptomoedas. Finanças Quantitativas.

This study examines the effectiveness of High Frequency Trading (HFT) algorithms by applying Machine Learning techniques, focusing on Logistic Regression and Random Forest models. Data was collected from the Binance Exchange, considering Solana cryptocurrency transactions. The analysis included training time, latency, and data processing considerations. Results showed that, without considering latency, the Random Forest model achieved a profitability of 138,625%, while Logistic Regression reached 66,774%. However, when execution time was included, the final profitability of Random Forest dropped to 6,920%, and Logistic Regression showed a negative performance of -0.58%. These findings highlight the importance of considering latency in HFT strategies and indicate that Random Forest is more robust in terms of accuracy and profitability, although latency remains a significant challenge.

Keywords: High Frequency Trade. High Performance Computing. Machine Learning. Cryptocurrency. Quantitative Finance.

1. Introdução

Nas últimas décadas, o mercado financeiro global se automatizou por meio de tecnologias e infraestrutura cada vez mais complexas. As execuções de ordens no mercado acionário passaram de 25 ms em 2015 para menos de 0.017 ms em 2017 (Angel, 2018).

Esse avanço na velocidade foi resultado do investimento em infraestrutura de telecomunicação, tais como a instalação de uma fibra óptica de Chicago à Nova York em linha reta, apenas para executar ordens (Lewis, 2015); e a otimização da infraestrutura computacional, por meio de processamento em paralelo através de GPUs (Aldridge, 2015).

Com o avanço de técnicas de *Machine Learning* e *high performance computing*, operações cada vez mais

rápidas no mercado financeiro dominaram a estrutura de negociação.

Os *High Frequency Trades* (HFT) - como são chamados - são responsáveis pela formação de mercado¹. Entretanto, existem outras classes de HFTs que presumidamente buscam lucro.

Os HFTs, diferentemente dos investidores comuns, não operam com base em dados econômicos, fundamentalistas ou idiossincráticos, mas sim com microestrutura de mercado, order flow² e dinâmica do book de ofertas (Benos e Sagade, 2016).

¹Formadores de mercados são responsáveis por manter a liquidez do mercado, oferecendo ofertas de compra e vendas regularmente e continuamente durante o pregão (CVM, 2022).

²Dinâmica de execuções de ordens de compra e venda no book de ofertas.

No entanto, o fluxo de dados complexos e de alta velocidade são muito desafiadores para a indústria, que já tem como desafio latência e regulamentação³. Uma alternativa de melhorar a dinâmica de produção desses algoritmos é justamente buscar mercados alternativos, como o de criptomoedas, já que exchanges e dex⁴ disponibilizam os dados de suas transações, para quem se dispõe em operar em alta frequência, apesar de inexistir uma regulação formal sobre esse tipo de transação.

Diferentemente das transações padrão em uma bolsa de valores centralizada, como B3 e NYSE, cada exchange possui seu próprio sistema de negociação para cada criptomoeda listada em sua plataforma, o que abre um grande espaço para HFTs (Makarov & Schoar, 2020).

2. Revisão de Literatura

Os algoritmos de negociação são amplamente estudados na literatura financeira. O início das ordens automatizadas expressou um grande crescimento a partir da década de 90 (MacKenzie, 2021). A indústria financeira investiu muito em infraestrutura, não só computacional, mas também em infraestrutura de telecomunicação (Lewis, 2015). Com isso, o mercado ganhou grandes nomes (i. e. Citadel) que o dominaram e hoje são responsáveis por cerca de 60 a 70 por cento das negociações no mercado acionário mundial (Aldridge, 2013).

Apesar de todo o investimento, as operações logísticas estão cada vez mais próximas do seu limite, principalmente com a liberação do *co-location* pelas bolsas de valores. A *co-location* é o que permite que as empresas de HFT fixem seus servidores fisicamente próximos aos de execução de ordem pela bolsa de valores, minimizando a latência e proporcionando uma vantagem competitiva significativa (Johnson, 2010). Atualmente as empresas de *high frequency* buscam melhorar o desempenho (velocidade de análise e execução) através de modelos matemáticos melhores e processamento mais rápido (MacKenzie, 2021).

Neste sentido, um grande responsável por uma evolução no processamento de algoritmos de alta frequência foi o mercado de games, no qual, com a popularização das placas gráficas e o barateamento desses componentes, surgiram oportunidades de melhorar o

processamento de dados devido a arquitetura desses componentes, como demonstra a Figura 1.1.

Portanto, uma melhor aplicação paralela⁵ agora não depende apenas do software ou linguagem de programação, mas sim do hardware envolvido na aplicação. (Williams, 2008).

Na Figura 1 é apresentado um panorama da evolução do processamento computacional em função do custo ao longo do tempo. Observa-se que, ao longo do século XX, o processamento evoluiu de dispositivos manuais como o ábaco e a régua de cálculo para computadores equipados com unidades centrais de processamento (CPUs) e, eventualmente, a unidades mais avançadas, como FPGAs e GPUs.

Cook et. All (2011) demonstram que algoritmos executados utilizando SEJITS⁶, Just-in-Time para compilação em CUDA, utilizando *Python*, podem alcançar desempenhos superiores às compilações tradicionais em *Python*, e, até mesmo, de linguagens de baixo nível como C++.

3. Metodologia

O procedimento do trabalho será exposto em três seções. Na primeira parte, será abordada a descrição e seleção de variáveis do modelo. Na segunda seção, serão detalhados os modelos empregados e sua aplicação no conjunto de dados. Na terceira e última seção, será feita a comparação dos modelos em uma simulação de rentabilidade no mercado de criptomoedas.

3.1. Dados

Os dados utilizados para este trabalho foram obtidos através da *Crypto Lake*, uma empresa privada que coleta e comercializa dados referentes às *Exchange* com o maior volume de transações, e pares de criptomoedas com maior volume de dólares em transação.

Para este trabalho, a plataforma de negociação de criptomoedas selecionada foi a *Binance Exchange*; e o ativo estudado é criptomoeda Solana, a quinta maior criptomoeda em valor de mercado, na data deste trabalho, a qual foi precificada em dólar, utilizando a sta-

⁵Processamento paralelo é a técnica de dividir uma tarefa em várias subtarefas que podem ser executadas simultaneamente em múltiplos processadores ou núcleos

⁶É uma técnica de otimização de software que permite a especialização dinâmica de partes do código em tempo de execução, traduzindo trechos críticos para código altamente eficiente, geralmente em linguagens de baixo nível, enquanto mantém a produtividade e a facilidade de uso de linguagens de alto nível como Python (Cantazaro et. al, 2009)

³Após os algoritmos de alta frequência dissimularem o mercado de ações em um evento em 2010, a SEC (U.S. Securities and Exchange Commission) passou a regulamentar mais intensivamente os algoritmos de HFT

⁴Plataforma de negociação de criptoativos

blecoin⁷USDT, que atualmente é a terceira maior criptomoeda em valor de mercado.

Os dados coletados correspondem ao período entre 01/01/2024 e 31/01/2024. Os dados são referentes as transações executadas (*Tick-by-Tick*) e o *book* de ofertas retrata a cada mudança de ordem em 20 níveis do *book* de ofertas entre valor de compra e venda, além da quantidade de ofertas.

Foram criadas com base nos dados obtidos novas features (Kearns; Nevmyvaka, 2013):

- Retorno em % do preço da ordem;
- Desvio padrão da ordem;
- *Spread* do primeiro nível do *book*;
- Desequilíbrio no volume de ordens;
- Pressão nas ordens de compra e venda.

Ao todo foram coletadas 6 milhões de observações originando uma matriz onde sendo a matriz com as variáveis preditoras em t o retorno esperado em t .

Por fim, para evitar a discrepância entre os dados dos fatores e mensurar a variação dos impactos do *book* de ofertas na variável resposta, foi aplicado o logaritmo dos valores na série. A normalização dos dados proporciona a redução da amplitude de variação e a minoração de valores extremos.

Além disso, a normalização por logaritmo ajuda a estabilizar a variância dos dados, melhorando a robustez do modelo e a precisão na seleção das características mais relevantes que impactam a variável resposta.

3.2. Machine Learning

Em geral, modelos de *Machine Learning* são designados para a detecção automatizada de padrões significativos em dados (Shalev-Shwartz e Bem-Davis, 2014).

A aplicação desses modelos no campo das finanças possui duas principais propriedades (Rasekhschaffe, 2019). Primeiramente, confere a possibilidade de descobrir padrões complexos e relações ocultas entre as dinâmicas de mercado, com a inclusão de não linearidade entre essas relações, as quais modelos lineares tracionais não poderiam detectar. Em segundo lugar, confere maior efetividade em comparação aos modelos lineares na presença de multicolinearidade.

3.2.1. Seleção de Variáveis

Goyal e Welch (2008) tratam o excesso de retorno⁸ de um ativo como a seguinte regressão:

⁷Criptomoeda lastreada em uma moeda fiduciária.

⁸Retorno acima da taxa livre de risco.

$$r_t = \alpha + \sum_{j=1}^J \beta_j x_{jt-1} + \epsilon_t$$

Sendo α e β coeficientes de uma MQO e x os vetores de controle, para cada $j = 1, \dots, J$. Porém, essa regressão possui um problema de dimensionalidade⁹, já que há dezenas de variáveis que justificam dos retornos futuros.

Portanto, a Regressão LASSO (Least Absolute Shrinkage and Selection Operator) descrita por Tibshiran (1996), foi utilizada como modelo para a seleção de variáveis na matriz resposta, cuja definição é:

$$\beta_{\text{lasso}} = \arg \min_{\beta} \sum_{i=1}^N \left(y_i - \sum_{j=1}^p x_{ij} \beta_j \right)^2,$$

sujeito a $\sum_{j=1}^p |\beta_j| \leq \lambda.$

Onde α é o parâmetro de penalidade do modelo e o termo regularizador é dado por $\sum_{j=1}^p |\beta_j|$, costumeiramente chamado de L_1 .

O fator de penalização (λ) foi definido para cada um dos índices testados através da otimização via *cross-validation*. Para cada um dos modelos foi definido um λ que apresentasse o maior R^1 .

As covariáveis que possuem coeficiente diferente de zero foram selecionadas para a composição do modelo de HFT, conforme a tabela 3.1

Tabela 1: Coeficientes

| Covariável | Coefficiente |
|---------------------------|--------------|
| price_return ¹ | -9,2879 |
| price_stddev ² | 1,3721 |
| price_spred ³ | -2,7238 |

Fonte: Elaboração própria

⁹A dimensionalidade é o tamanho da dimensão das características que compõem a observação, os dois principais problemas de dimensionalidade são o custo de processamento computacional em alta dimensionalidade e a melhora na precisão do modelo computacional, já que características de pequeno impacto poder enviesar o modelo (Jain et al., 1999).

3.3. Modelos de Estimação

Foram empregados dois modelos de algoritmos de *Machine Learning* para a estratégia de HFT. Regressão Logística e o modelo de *Random Forest*.

3.3.1. Regressão Logística

A Regressão Logística é um modelo linear de classificação amplamente utilizado para prever a probabilidade de um resultado binário, como sucesso/falha ou sim/não. Este modelo é especialmente útil em contextos em que a variável dependente é categórica. A Regressão Logística se baseia na função logística, que mapeia qualquer valor real em um intervalo entre 0 e 1, representando probabilidades. (Faraway, 2021).

A função logística é expressa pela seguinte fórmula:

$$g(p) = \ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1x_1 + \beta_2x_2 + \dots + \beta_nx_n$$

Onde $g(p)$ é a função de ligação *logit*, sendo o logaritmo da razão de chances, p é probabilidade de o evento ocorrer, β_0 é o intercepto, e $\beta_1, \beta_2, \beta_3\dots$ são os coeficientes das covariáveis $x_1, x_2, x_3\dots$

3.3.2. Random Forest

O algoritmo *Random Forest* (RF) é composto por um conjunto de modelos de árvores de decisão. Elas são construídas a partir de um vetor da matriz de características aleatória e independente para cada árvore, mas mantendo a mesma distribuição de dados para todas as árvores do modelo.

O resultado final da RF é justamente a média dos resultados individuais das árvores de decisão previamente treinadas com o conjunto da amostra.

Para o modelo de *Random Forest*, foi utilizado a API CUDA (Nvidia, 2024).

3.4. Backtest

Os modelos de *Machine Learning* foram testados para o par SOL-USDT entre o período de 01/01/23 à 31/01/23.

1. Não foi adicionado custo de transação;
2. O *delay* entre receber o dado, tratar e adicionar a latência do envio da ordem até os servidores da Binance foi de 350ms (cerca de 1 tick);
3. As ordens de compra e venda são executadas no melhor preço de venda e compra respectivamente;

4. Cada negociação é fixado o valor de 0.1 SOL a ser negociado.

3.4.1. Estratégia

Os dados foram rotulados utilizando uma janela móvel de 50 unidades de tempo. Se a diferença entre os preços iniciais dentro desta janela fosse positiva ou igual a zero, o rótulo atribuído era 1; caso contrário, o rótulo era 0.

Os algoritmos executam as ordens de compra no output 1 e executam as ordens de venda no output 0.

3.4.2. Ambiente

Os *backtests* dos modelos foram realizados em uma máquina Linux, utilizando o sistema operacional Ubuntu 22.04. Para o treinamento dos modelos, empregou-se a biblioteca cuML, versão 24.04.08, em *Python*, que oferece suporte acelerado por GPU para algoritmos de *Machine Learning*. A GPU utilizada foi uma Nvidia T4, equipada com 2560 núcleos CUDA e 16 gigabytes de memória DRAM.

3.5. Resultados

Nesta seção, apresentamos e discutimos os resultados obtidos com os modelos de *Machine Learning* utilizados na estratégia de *High Frequency Trading* (HFT). As análises foram conduzidas considerando diferentes aspectos de desempenho, incluindo a precisão dos modelos; a rentabilidade, sem considerar o tempo de execução; o tempo de resposta dos algoritmos; e a rentabilidade final, incluindo o tempo de execução.

A Tabela 2 apresenta os resultados de precisão dos modelos sem considerar o tempo de treinamento e a latência. Observa-se que o modelo *Random Forest* alcançou o maior nível de acurácia entre os dois modelos avaliados, com uma precisão de 98,491%, comparado aos 81,080% obtidos pela Regressão Logística.

Tabela 2: Precisão

| Modelo | Acurácia |
|---------------------|----------|
| Regressão Logística | 81,080% |
| Random Forest | 98,491% |

Fonte: Elaboração própria

A Tabela 3 mostra a rentabilidade dos modelos sem levar em conta o tempo de treinamento e a latência. Esses resultados fornecem uma visão inicial do desempenho financeiro dos modelos aplicados, destacando

a superioridade do modelo *Random Forest* também em termos de rentabilidade.

Tabela 3: Rentabilidade

| Modelo | Retorno |
|---------------------|----------|
| Regressão Logística | 66,774% |
| Random Forest | 138,625% |

Fonte: Elaboração própria

Fonte: Elaborado pelo autor.

O tempo de resposta dos algoritmos é apresentado na Tabela 4 Este aspecto é crucial para a aplicação em ambientes de alta frequência, onde a latência pode impactar significativamente o desempenho da estratégia.

O modelo *Random Forest*, embora tenha apresentado maior acurácia e rentabilidade, também exibiu um tempo de resposta significativamente maior em comparação à Regressão Logística.

Tabela 4: Tempo de resposta

| Modelo | Tempo (ms) |
|----------------------|------------|
| Regressão Logística | 3,58 |
| <i>Random Forest</i> | 127,22 |

Fonte: Elaboração própria

A Tabela 5 apresenta a rentabilidade final dos modelos, agora incluindo o tempo de execução. Esta análise permite avaliar a eficácia prática dos modelos quando implementados em um cenário realista de trading, no qual o tempo de resposta dos algoritmos influencia diretamente os resultados financeiros.

Isso ocorre porque, em ambientes de High-Frequency Trading (HFT), as ordens de compra e venda são emitidas em uma fração de segundo. Neste contexto, até mesmo pequenas diferenças na latência podem resultar em uma mudança no preço do ativo, podendo resultar em oportunidades perdidas ou execução em um preço desfavorável, comprometendo a rentabilidade esperada.

Se o tempo de resposta do algoritmo for alto, o preço-alvo da transação pode já ter se deslocado, reduzindo a eficácia da estratégia e impactando negativamente os resultados.

Portanto, a rapidez na execução é tão crítica quanto a precisão dos modelos, pois somente um algoritmo capaz de reagir rapidamente às mudanças de mercado pode capturar as oportunidades mais vantajosas e evitar perdas potenciais.

Tabela 5: Rentabilidade

| Modelo | Rentabilidade |
|----------------------|---------------|
| Regressão Logística | -0,580% |
| <i>Random Forest</i> | 6,920% |

Fonte: Elaboração própria

Em resumo, os resultados indicam que o modelo *Random Forest*, apesar de possuir um tempo de resposta mais elevado, apresenta superioridade em termos de precisão e rentabilidade final.

Sem considerar o *delay*, a rentabilidade do *Random Forest* foi significativamente alta, alcançando 138,625%. No entanto, ao incluir o tempo de execução, a rentabilidade final do modelo foi minorada à 6,920%. Esse declínio ressalta a importância de considerar a latência em estratégias de HFT, pois o tempo de resposta pode impactar drasticamente os resultados financeiros.

Já o modelo de Regressão Logística, que inicialmente apresentou uma rentabilidade de 66,774% sem *delay*, teve um desempenho final negativo de -0,580% quando o tempo de execução foi considerado. Esses resultados destacam a viabilidade do uso de *Random Forest* em estratégias de alta frequência, especialmente quando a precisão e a robustez do modelo são cruciais para o sucesso financeiro. No entanto, também sublinham a necessidade de otimização para reduzir a latência e melhorar a eficiência.

4. Conclusão

Este estudo teve como objetivo principal analisar a eficácia dos modelos de *High Frequency Trading* (HFT) através da aplicação de algoritmos de *Machine Learning*, com foco específico no *Random Forest* e na Regressão Logística. A análise incluiu a consideração do tempo de treinamento, latência, e tratamento dos dados provenientes da *exchange*.

A escolha desses algoritmos baseou-se em sua capacidade de lidar com grandes volumes de dados e alta dimensionalidade, características típicas do ambiente de HFT. O *Random Forest*, com sua estrutura de árvores de decisão, e a Regressão Logística, conhecida por sua simplicidade e eficiência, foram selecionadas para testar a robustez e a precisão na predição de movimentos de preços.

Para a análise, foram utilizados dados de alta frequência, o que permitiu avaliar a performance dos modelos em cenários reais de mercado. Os resultados mostraram que, sem considerar a latência e o tempo de

execução, o modelo *Random Forest* alcançou uma precisão de 98,491% e uma rentabilidade de 138,625%, enquanto a Regressão Logística teve uma precisão de 81,080% com uma rentabilidade de 66,774%.

A inclusão do tempo de execução revelou um impacto significativo nos resultados, destacando a importância de considerar a latência em estratégias de HFT. Com a latência incluída, a rentabilidade final do *Random Forest* foi reduzida para 6,920%, e a Regressão Logística apresentou uma rentabilidade negativa de -0,580%. Esses resultados corroboram a teoria de que a latência é um fator crítico em HFT, influenciando diretamente os retornos financeiros, como discutido por Aldridge (2013) e Easley et al. (2012).

Os resultados obtidos indicam que, embora o *Random Forest* apresente superioridade em termos de precisão e rentabilidade, a latência permanece como um desafio significativo para a implementação prática em ambientes de HFT. A capacidade de criar portfólios baseados em previsões de modelos de *Machine Learning* oferece uma perspectiva promissora para futuras aplicações, alinhando-se às discussões de autores como Chan (2013) e Narang (2013) sobre a integração de técnicas avançadas de modelagem em estratégias de trading.

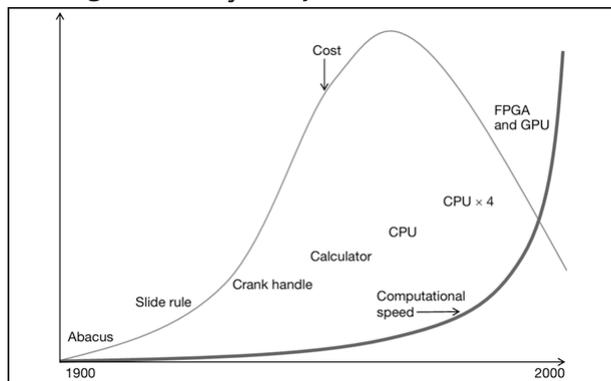
No contexto de velocidade versus precisão, foi possível notar que ambas as abordagens possuem importância significativa na área de *High Frequency Trading*. O modelo *Random Forest* demonstrou uma precisão superior, o que é crucial para a tomada de decisões acertadas e a maximização dos lucros. Entretanto, a Regressão Logística, embora seja menos precisa, apresentou uma velocidade maior no processamento, o que é vital para a rápida execução de trades em mercados de alta frequência.

No fim, a eficiência de estratégias de HFT depende de um equilíbrio entre precisão e velocidade. Um modelo robusto deve considerar a integração desses dois aspectos para otimizar a performance geral. Portanto, é imperativo que ambos, precisão e velocidade, estejam presentes no contexto de HFT, garantindo tanto a acurácia nas previsões quanto a rapidez na execução das operações.

Referências

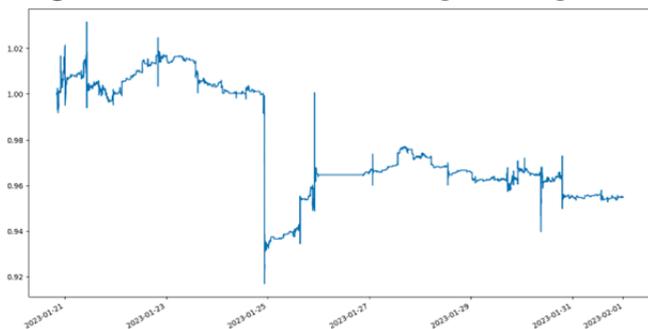
- [1] ALDRIDGE, I., «High-Frequency Trading» New Jersey, John Wiley, 2013.
- [2] AQUILINA, M., BUDISH, E., O'NEIL, P. «Quantifying the high-frequency trading “arms race”», Bank for International Settlements, n. 955, 2021.
- [3] BAREZ, F., BILOKON, P., GERVAIS, A., LISITSYN, N., «Exploring the Advantages of Transformers for High-Frequency Trading», Arxiv, 2023.
- [4] BREIMAN, L., «Random Forest» *Machine Learning*. v. 45, p. 5-32, 2001.
- [5] CANTAZARO, B., «Getting productivity and performance with selective embedded JIT specialization» *Programming Models for Emerging Architectures*, v. 1, n.1 p. 1-9, 2009
- [6] COOK, H et al., «CUDA-level Performance with Python-level Productivity for Gaussian Mixture Model Applications» In: *USENIX Workshop on Hot Topics in Parallelism*, 2011, Berkeley.
- [7] EASLEY, D., PRADO, M. L., O'HARA, M., «High-Frequency Trading» London, Risk Books, 2013.
- [8] FARAWAY, J. J., «Linear Models with Python» Florida, CRC Press, 2021
- [9] FARAWAY, J. J., «Linear Models with Python» Florida, CRC Press, 2021
- [10] FERROUHI, E. M., BOUABDALLAOUI, I. «A comparative study of ensemble learning algorithms for high-frequency trading» *Scientific African* v.24, p. e02161, 2024.
- [11] R. L. Viana, *Introdução à Eletrodinâmica Clássica*. Universidade Federal do Paraná, (2012),(Apostila), pag. 209.
- [12] HASTIE, T., TIBSHIRANI, R.; FRIEDMAN, J. «The Elements of Statistical Learning». New York, Springer, 2001.
- [13] LEWIS, M., «Flash boys: A Wall Street Revolt» New York: W. W. Norton & Company, 2015.
- [14] MACKENZIE, D., «Trading at the Speed of Light» New Jersey, Princeton University Press, 2021.
- [15] MAKAROV, I., & SCHOAR, A., «Trading and Arbitrage in Cryptocurrency Markets » *Journal of Financial Economics*, v. 135, ed. 2, p 293-319.
- [16] NVIDIA CORPORATION., «CUDA Toolkit Documentation» Version 12.5, 2024 CUDA Toolkit Documentation 12.5. Disponível em: (nvidia.com).
- [17] TIBSHIRANI, R., «Regression Shrinkage and Selection via the Lasso». *Journal of the Royal Statistical Society*, v. 58, n. 1, p. 267-288, 1996.
- [18] WELCH, I. & GOYAL, A., «A Comprehensive Look at the Empirical Performance of Equity Premium Prediction» *Oxford University Press* v. 21, n. 4, p. 1455-1508, 2008.

Figura 1: Evolução de processamento e custo



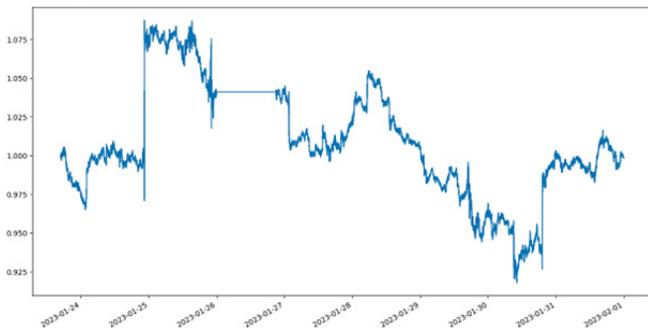
Fonte: Aldridge, 2013.

Figura 2: Rentabilidade do modelo Regressão Logística



Fonte: Elaboração própria

Figura 3: Rentabilidade do modelo Random Forest



Fonte: Elaboração própria