**FGV - FUNDAÇÃO GETÚLIO VARGAS  
MBA Executivo em Business Analytics e Big Data**

**ALUNOS**

Arthur Salvador Rocha  
Guilherme Nakano Nalin  
Lucas Gabriel Mendes de Oliveira

Patricia Yoshie Kishi Bueno

Paulo Gustavo Maciel Lopes

**Criação de Modelos Preditivos para Operações com Criptomoedas (RAMster Trader - Grupo 2)**

Trabalho apresentado para obtenção de nota da disciplina Análise Preditiva Avançada  
Professor Bernardo Santos Aflalo

São Paulo/SP  
Unidade Paulista - Turma 9  
Outubro de 2021

**1. Introdução**

Em diversas áreas de investimentos, pessoas tentam ganhar dinheiro com estratégias de "day trade", que consiste na compra e venda de ativos no intervalo de um dia. No entanto, 95% dos investidores dessa estratégia perdem dinheiro.

Ao mesmo tempo, as tentativas de algoritmos para tentar prever tendências surgiram desde a década de 80, sendo a empresa "Bridgewater" do investidor Ray Dalio uma das pioneiras no uso dessas ferramentas com relativo grande sucesso, com grande entusiasmo do mercado até 2000. No entanto, a crise de 2008 levou ao maior ceticismo em relação ao uso de transações computadorizadas, sendo considerada um dos motivos da velocidade de propagação na crise.

Com o aumento da capacidade computacional e a complexidade da tomada de decisões por parte dos investidores, já que a quantidade de informação disponível que influencia determinado ativo é impossível para a compreensão do cérebro humano, além da capacidade muito maior do computador de fazer vários investimentos diferentes em um intervalo de tempo muito mais curto, o uso de algoritmos para auxílio ou até substituição do ser humano ainda é um desafio que tem estimulado várias pessoas pelo mundo, mas ainda com taxas de sucesso controversas, mas animadoras.

Nesse contexto, um dos mercados de ativo mais desafiadores é o de criptomoedas, devido a sua grande volatilidade e diversidade de ativos. Neste trabalho estamos criando um robô para operação de criptomoedas em uma plataforma simulada e descrevendo aqui o principal desempenho dos modelos escolhidos.

**2. Criação das Features**

Para a criação dos modelos, foram criadas as seguintes features:

Features Criadas:

* Return: Diferença percentual entre os valores de fechamento e abertura (***Algoritmo: round(df['close'] / df['open'] - 1, 3)***);
* Change: é a diferença entre o preço do ativo em dois pontos do tempo.
* Volatility: ou volatilidade é o desvio padrão dos retornos de determinado ativo. Após o seguinte cálculo : retorno= (preço atual- preço anterior)/preço anterior.
* Momentum: mede a variação de preço de um instrumento financeiro ao longo de um determinado intervalo de tempo.
* RSI (Relative Strength Index): de forma bem resumida, o RSI é feito pela mensuração do movimento de queda e alta de um ativo, mostrando uma tendência recente. A fórmula de cálculo é RSI= 100- 100/ (1 + U/D), sendo U a média dos movimentos de alta sobre o total de dias do período e D a média dos movimentos de baixa sobre o total de dias do período.
* MACD: ou média móvel de convergência e divergência. É um cálculo pela subtração das médias exponenciais rápida (padrão de 12 períodos) e lenta (padrão de 26 períodos) de determinado ativo, criando uma curva MACD.
* Upper e Lower Bands: as bandas de Bollinger são calculadas a partir da média móvel dos últimos dias e calculados dois desvios padrão acima e abaixo. O valor de um ativo raramente sai desse intervalo.

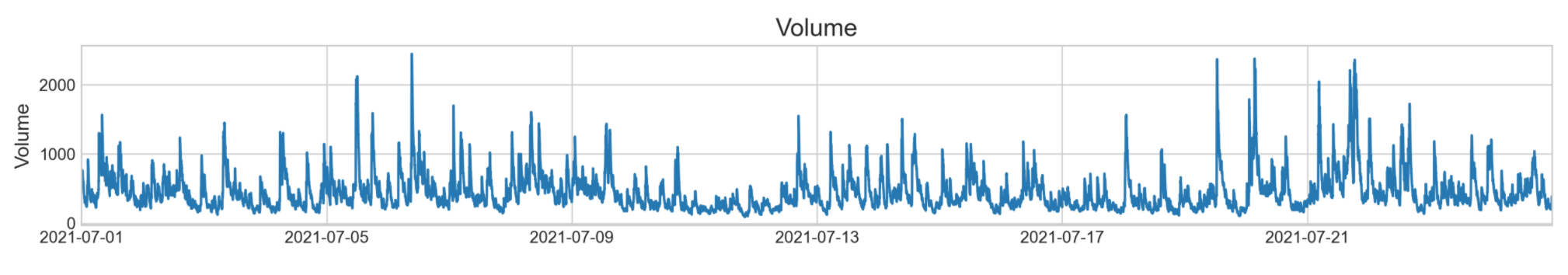
**3. Análise Exploratória**

Foram utilizados para análise dos modelos apenas dados no intervalo de janeiro a julho de 2021 da moeda Ethereum, com um total de 294536 observações.

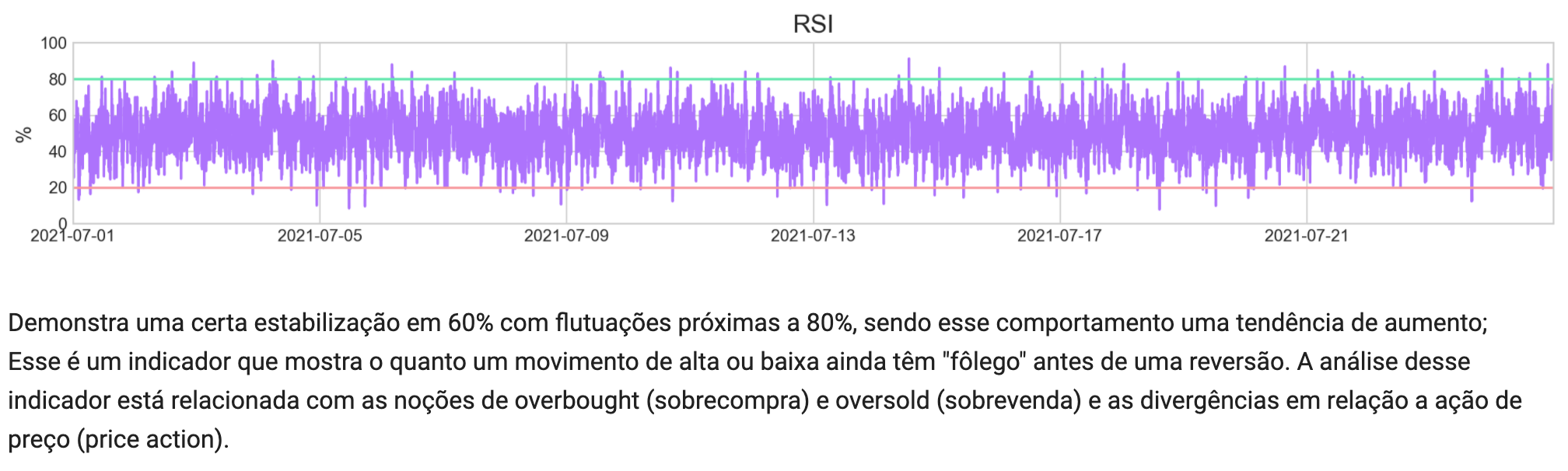
Na análise da correlação das variáveis independentes a maior correlação entre a variável target foi com a variável volume.

Fizemos as análises das seguintes variáveis independentes:

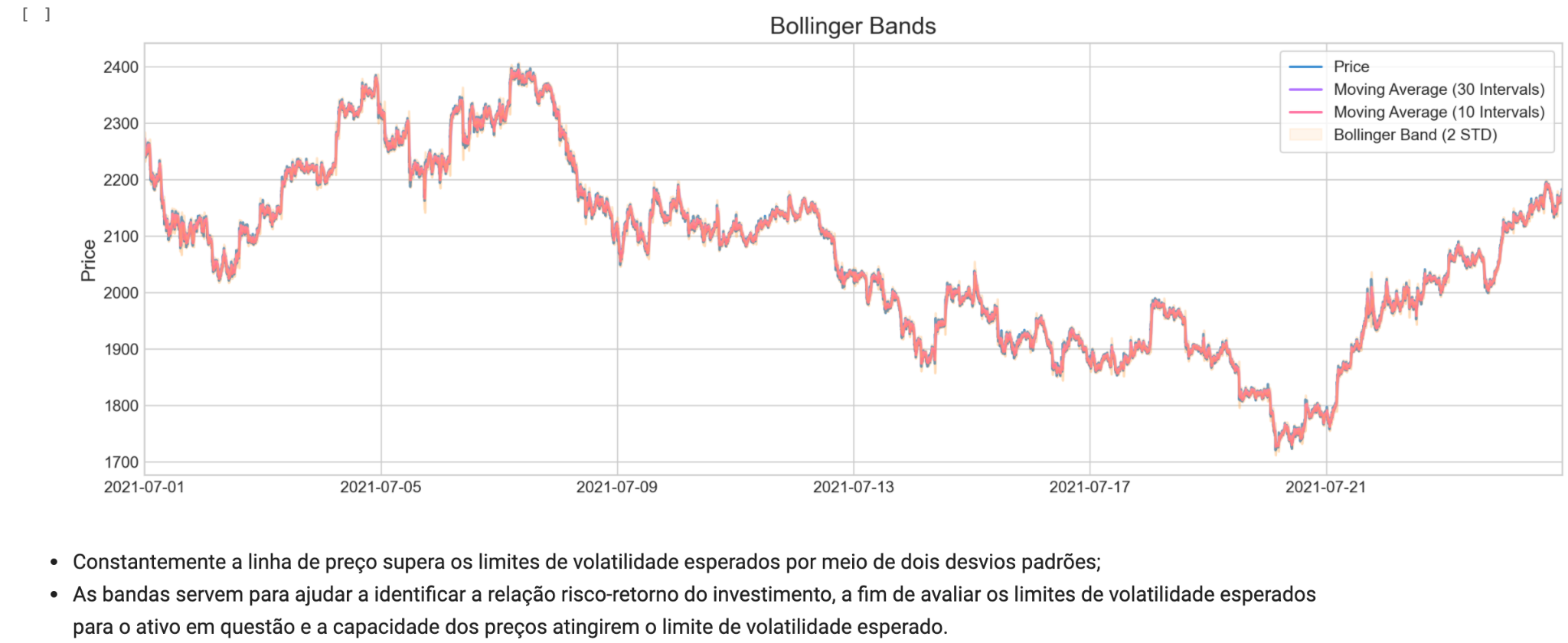
* Volume ao longo do tempo:



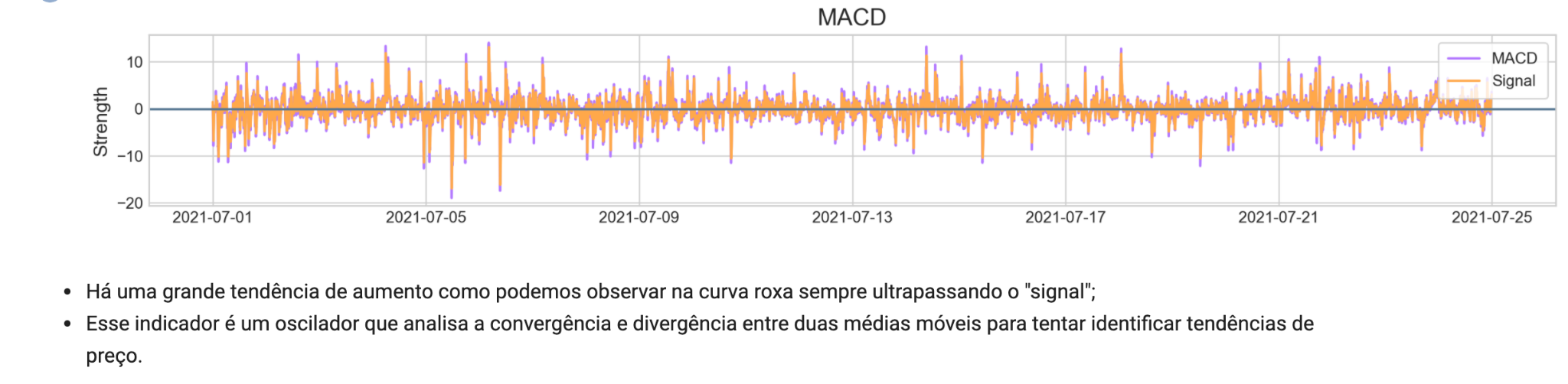
* RSI ao longo do tempo.



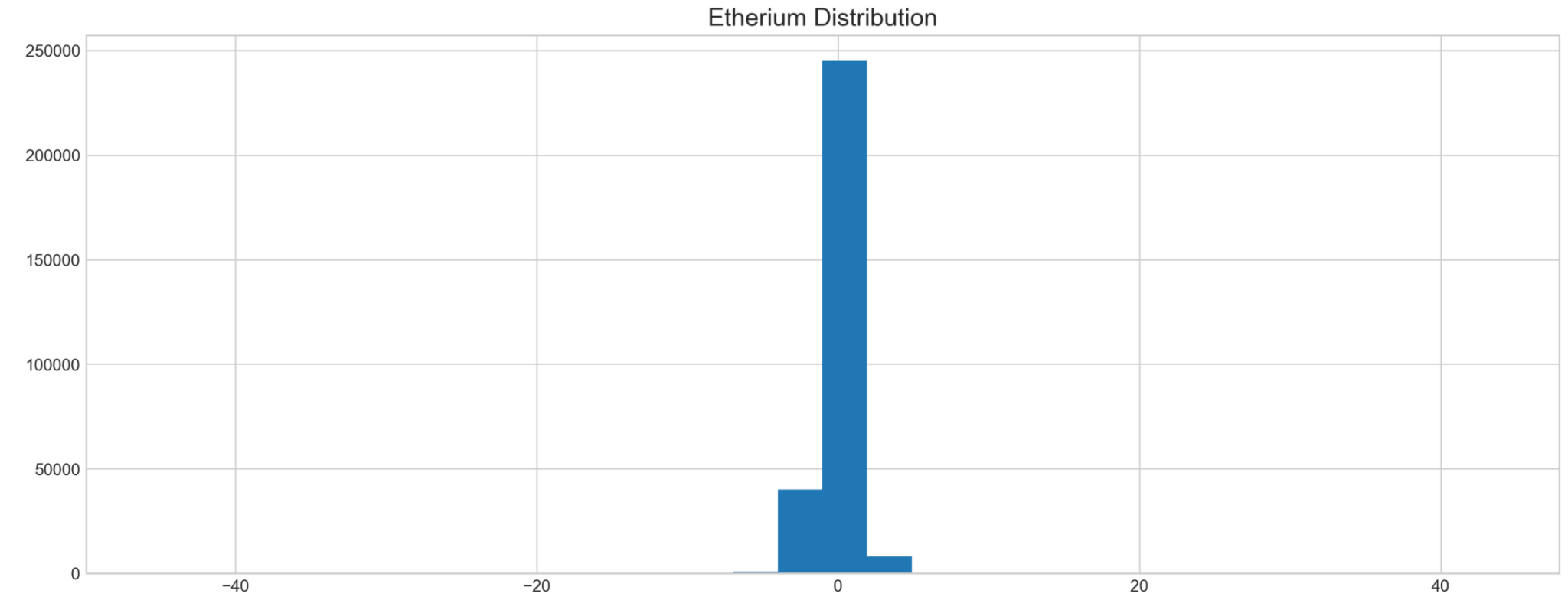
* Bollinger Bands ao longo do tempo

****

* MACD ao longo do tempo

****

* Distribuição da volatilidade



**4. Modelos Preditivos**

Para avaliação dos modelos foram separados amostra de treino e amostra de teste.

Foram testados os seguintes modelos com as seguintes métricas obtidas:

4.1. - CNN (Convolutional Neural Network)

É uma rede neural artificial do tipo *feed-forward.* Apresentou acurácia: 64% com erro médio quadrado de 0,013.

4.2. - LSTM

É um tipo de rede neural recorrente. Apresentou acurácia: **76%** e erro médio quadrado de 0,007.

4.3. - QLearning

Q-learning é um algoritmo de aprendizagem baseado em valores e concentra-se na optimização da função de valor de acordo com o ambiente ou problema utilizando um mecanismo de recompensa..

*Acurácia*: **Perda de 74% do valor investido baseado na métrica de MACD e perda de 2% no modelo baseado no indicador de RSI (modelos aplicados em um dia de cotações).**

4.4. - Arima

É um modelo em análise de séries temporais, que em inglês significa modelo auto-regressivo de médias móveis.

Apresentou como resultado erro médio absoluto de 12,78, erro médio quadrado de 216,9 e R2 score de -4405, apresentando o pior resultado entre todos os modelos.

4.5. - XGBoost

O XGBoost é um modelo de aprendizado de máquina baseado em árvores de decisão.

Apresentou como resultado na amostra de treino erro médio absoluto de 0,31, erro médio quadrado de 0,45 e R2 score de -0,37. Na amostra de teste apresentou erro médio absoluto de 0,17, erro médio quadrado de 0,23 e R2 score de -0,14.

**5. Escolha do Modelo e Aplicação nos Dados Atuais**

Considerando os resultados dos modelos, optamos pelo LSTM devido ao seu melhor resultado, com menor erro médio quadrado. Realizamos a API e colocamos o nosso robô (RAMster) para operar na plataforma simulada. Até o fechamento deste trabalho, o nosso robô tinha apresentado uma perda de -0,05%.

**6. Conclusões**

Apesar da tentativa há muitos anos por muitas empresas de análises dos mercados e de ativos mais tradicionais, como ações, ou mais novos, como as criptomoedas, percebemos que a análise destes ativos apenas com os seus dados históricos é muito difícil. Algoritmos de aprendizado de máquina acabam interpretando esses dados como aleatórios.

Com a melhoria desses algoritmos e da capacidade computacional, o uso de outras informações, como análise fundamentalista das empresas (no caso do mercado de ações) ou de tendências de sentimentos em redes sociais, bem como análises micro e macroeconômicas locais e globais podem melhorar o resultado.

Essas possíveis melhorias e o consequente aumento do número desses robôs e do capital por eles geridos leva a uma outra questão que é o impacto do mercado que isso pode causar. Não só podendo aumentar tendências e talvez o impacto de crises, como ocorreu em 2008, como esse tipo de comportamento dos ativos ditados pelos próprios algoritmos, levando a alterações de tendências atuais, pode ser o próximo desafio para o desempenho desses algoritmos.

Os códigos e o maior detalhamento dos cálculos realizados estão no seguinte link: https://github.com/pykishi/RedesNeurais.

**7. Referências**

# **MACD: Aprenda o que é e saiba como usar no day trade.** Disponível em: <<https://smarttbot.com/trader/macd-aprenda-o-que-e-e-saiba-como-usar-no-day-trade/>>

# **RSI: Índice de Força Relativa - Conheça este índice.** Disponível em: <<https://coinext.com.br/blog/rsi-indice-de-forca-relativa>>

# **Análise Técnica: como utilizar com maior eficiência as Bandas de Bollinger.** Disponível em: <<https://www.infomoney.com.br/mercados/analise-tecnica-como-utilizar-com-maior-eficiencia-as-bandas-de-bollinger-2/>>

## **What is an Epoch?** Disponível em: <<https://deepai.org/machine-learning-glossary-and-terms/epoch>>

# **Stock Prediction in Python.** Disponível em: <<https://towardsdatascience.com/stock-prediction-in-python-b66555171a2>>

# **Predicting the Stock Market is Hard: Creating a Machine-Learning Model (Probably) Won’t Help.** Disponível em: <<https://towardsdatascience.com/predicting-the-stock-market-is-hard-creating-a-machine-learning-model-probably-wont-help-e449039c9fe3>>

# **Machine Learning for Day Trading.** Disponível em: <<https://towardsdatascience.com/machine-learning-for-day-trading-27c08274df54>>

# **A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way.** Disponível em: <<https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>>

**tf.keras.Model Documentation.** Disponível em: <<https://www.tensorflow.org/api_docs/python/tf/keras/Model#fit>>

# **Stacked Long Short-Term Memory Networks.** Disponível em: <<https://machinelearningmastery.com/stacked-long-short-term-memory-networks/>>

# **LSTM layer Documentation.** Disponível em: <<https://keras.io/api/layers/recurrent_layers/lstm/>>

# **Save and Load Machine Learning Models in Python with scikit-learn.** Disponível em: <<https://machinelearningmastery.com/save-load-machine-learning-models-python-scikit-learn/>>