

Precificação de NFT's com Random Forest Regressor e Bagging Regressor: um estudo de caso

ISAAC MIRANDA CAMARGOS, LEANDRO CRUVINEL LEMES

Universidade Federal do Triângulo Mineiro, UFTM, Brasil

Isaacmirandacamargos@gmail.com, leandro.lemes@uftm.edu.br

Resumo

O presente estudo compara os algoritmos Random Forest Regressor, Bagging Regressor, quanto à estimativa de preço das cartas NFT's do jogo Gods Unchained. O trabalho foi dividido em três etapas: coleta, pré-processamento e modelagem dos dados. As métricas utilizadas foram o erro médio absoluto (EMA), erro médio quadrático (EMQ) e coeficiente de determinação (R^2). Os resultados obtidos da precificação em Ethereum e GODS foram tecnicamente iguais para os correspondentes modelos. Com relação aos modelos, o Random Forest Regressor obteve os melhores resultados nas duas precificações, sendo seus respectivos valores para ETH: EMA 0.1435 ± 0.0115 ; EMQ 0.0544 ± 0.0101 ; R^2 0.7994 ± 0.0322 . Ademais, os seguintes valores para GODS: EMA 0.1441 ± 0.0118 ; EMQ 0.0553 ± 0.0105 ; R^2 0.7996 ± 0.0322 .

Introdução

Com o advento das tecnologias *Blockchain* e *smart contracts* uma nova categoria de produtos digitais surgiu, os NFT's. Em prática, os NFT's podem ser qualquer tipo de bem digital, como imagens, músicas ou vídeos, além disso eles podem possuir altos e oscilantes valores de mercado. Neste contexto, considerando as possíveis manipulações de mercado e a variabilidade dos preços, o presente estudo objetiva precificar dos NFT's do jogo Gods Unchained utilizando algoritmos de aprendizagem de máquina, afim definir preços individuais levando em conta os atributos e valores das outras cartas.



Objetivos

O estudo tem como objetivo a precificação das cartas e geração de possíveis valores de aquisição e venda com base nos preços atuais de mercado. Além disso, os modelos desenvolvidos objetivam precificar novas cartas que ainda vão lançar ou cartas hipotéticas criadas pelos desenvolvedores.

Desenvolvimento e Metodologia

O estudo foi dividido em três etapas: coleta de dados, pré-processamento e modelagem dos dados. A linguagem de programação utilizada foi Python e as bibliotecas empregadas foram escritas nesta mesma linguagem.

Com relação a coleta de dados, foram utilizadas API's para se obter as informações de cada carta, sendo elas: nome, efeito, divindade, raridade, tribo, mana, ataque, vida, tipo e coleção. Ademais, se obteve os dados relacionados ao mercado do jogo: preços, número de vendas semanais, variação de preços semanais, tendência, porcentagem de baralhos que contem a carta e porcentagem de vitória de baralhos que contem a carta. Ao todo, foram obtidos dados sobre 1498 cartas, contudo foram descartas 271 cartas, sendo as cartas descartadas: cartas intransferíveis ou extremamente raras as quais não possuem preços de mercado.

Iniciado o pré-processamento, foram realizados os seguintes passos: valores extremos foram removidos utilizando-se boxplot; discretização dos dados categóricos (divindade, raridade, tribo, tipo, coleção); normalização dos dados numéricos (mana, ataque, vida, tendência, número de vendas semanais); e, para o tratamento dos dados textuais, foi removido os caracteres especiais, tags, stopwords (palavras de alta frequência e pouca significância), extraiu-se o radical das palavras e vetorizou as mesmas. Ao final do pré-processamento, obteve-se as informações tratadas de 1016 cartas e um banco de dados de 404 colunas.

Por fim, na etapa de modelagem dos dados se aplicou os modelos Random Forest Regressor e Bagging Regressor. As métricas utilizadas foram erro médio absoluto (EMA), erro médio quadrático (EMQ) e coeficiente de determinação (R^2).

Resultados

Os resultados obtidos na precificação das cartas em Ethereum e GODS, podem ser vistos na figura 1.

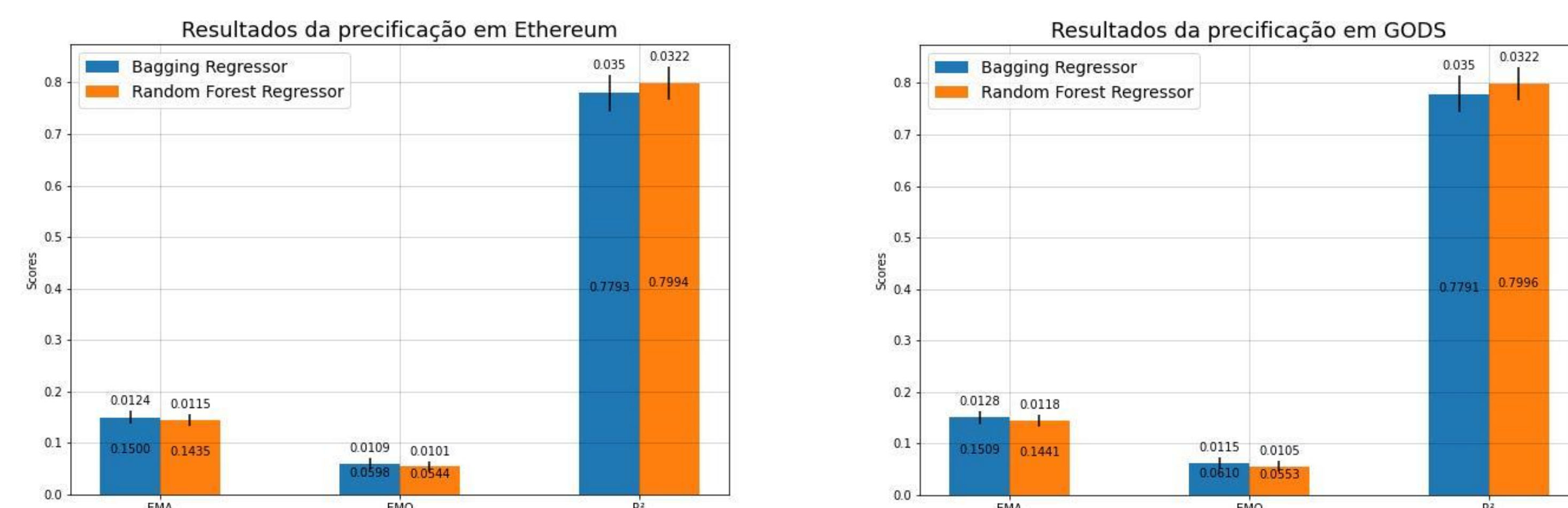


Figura 1: Resultados obtidos na precificação de cartas NFT's em ETH e GODS, sendo erro médio absoluto (EMA), erro médio quadrático (EMQ) e coeficiente de determinação (R^2). Fonte: Elaborada pelo autor.

Conclusões

Analisando os resultados obtidos, conclui-se que dentre os modelos implementados o que obteve menor EMA, menor EMQ e maior R^2 , foi o Random Forest Regressor em ambas as precificações. Desse modo, ele se mostrou mais assertivo. Ademais, analisando comparativamente os modelos, é possível inferir que a aprendizagem de máquina pode auxiliar jogadores de forma precisa na compra e venda de suas cartas NFT's.

Sugere-se para futuras pesquisas a utilização de outros algoritmos, inclusive redes neurais, incluir a etapa de hipertunagem de parâmetros e estimar valores futuros para os ativos de coleções a serem lançadas. Além disso, resultados preliminares sugerem que a inclusão de uma etapa de seleção de recursos do banco de dados utilizando algoritmos genéticos apresenta uma grande influência na assertividade dos modelos, como pode ser visto na figura 2.

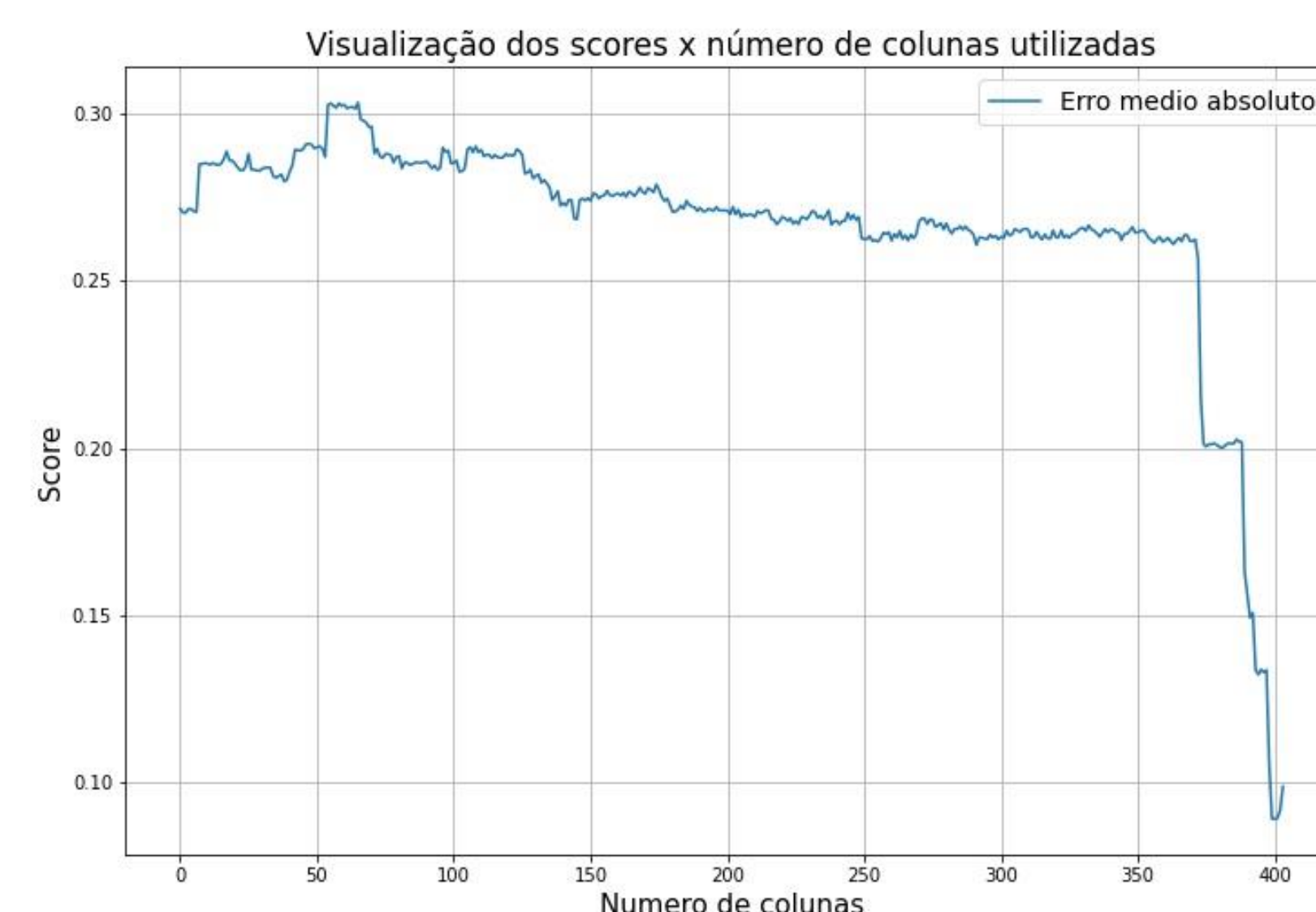


Figura 2: Análise preliminar da influência dos recursos do banco de dados no erro médio absoluto das precificações. Fonte: Elaborada pelo autor.

Referencias

- [1] CoinGecko. Site oficial do CoinGecko. Online. Acessado em 14/03/2022, <https://www.coingecko.com/en/coins/bitcoin>.
- [2] V. Fabian e B. Vitalik. Erc-20 token standard. Online. Acessado em 15/03/2022, <https://eips.ethereum.org/EIPS/eip-20>.
- [3] Immutable. GUDecks. Online. Acessado em 27/03/2022, <https://gudecks.com/meta/card-rankings>.
- [4] Satoshi Nakamoto. "Bitcoin: A peer-to-peer electronic cash system". Em: Decentralized Business Review (2008), p. 21260.
- [5] Nofer, M., Gomber, P., Hinz, O., and Schiereck, D. "Blockchain". Em: Business and Information Systems Engineering 59(3) (2017), 183–187. doi: {10.1007/s12599-017-0467- 3}.
- [6] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". Em: Journal of Machine Learning Research 12 (2011), pp. 2825–2830.
- [7] E. William et al. Erc-721 non-fungible token standard. Online. Acessado em 15/03/2022, <https://eips.ethereum.org/EIPS/eip-721>.