


Criptomoedas Blockchain e IA - Trabalho Final

An abstract graphic consisting of numerous thin, light blue lines that flow and curve across the dark background, creating a sense of movement and depth. The lines originate from the bottom left and fan out towards the top right, with some lines crossing each other.



“A inteligência artificial pode ser utilizada em blockchain e criptomoedas para enfrentar desafios relacionados à segurança e à análise de dados”



Estrutura do Projeto

03

PREVISÃO DO PREÇO DO BITCOIN COM IA

Modelo para previsão, com base em dados

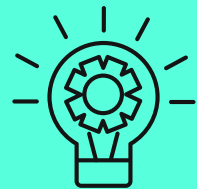
DESENVOLVIMENTO DE SMART CONTRACT

Armazena previsões e utilizado como ferramenta de monitoramento

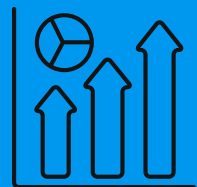
INTERAÇÃO IA - SMART CONTRACT

Previsões para a blockchain de forma segura

Justificativa da Aplicação



- Relevância do problema
 - previsão de preço do Bitcoin.



- Integração com Blockchain
 - segurança e transparência.



- Uso de IA para melhoria de precisão.

Por que aplicação com Blockchain?



- Segurança
 - fator crítico de finanças.



- Metodologia de imutabilidade.

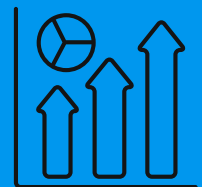


- Perdas: custos e escalabilidade

Justificativa da Implementação (Ethereum)



- Rede popular
 - utilizado durante as aulas.



- Contratos inteligentes.



- Integração com Chainlink.

Aplicações Similares



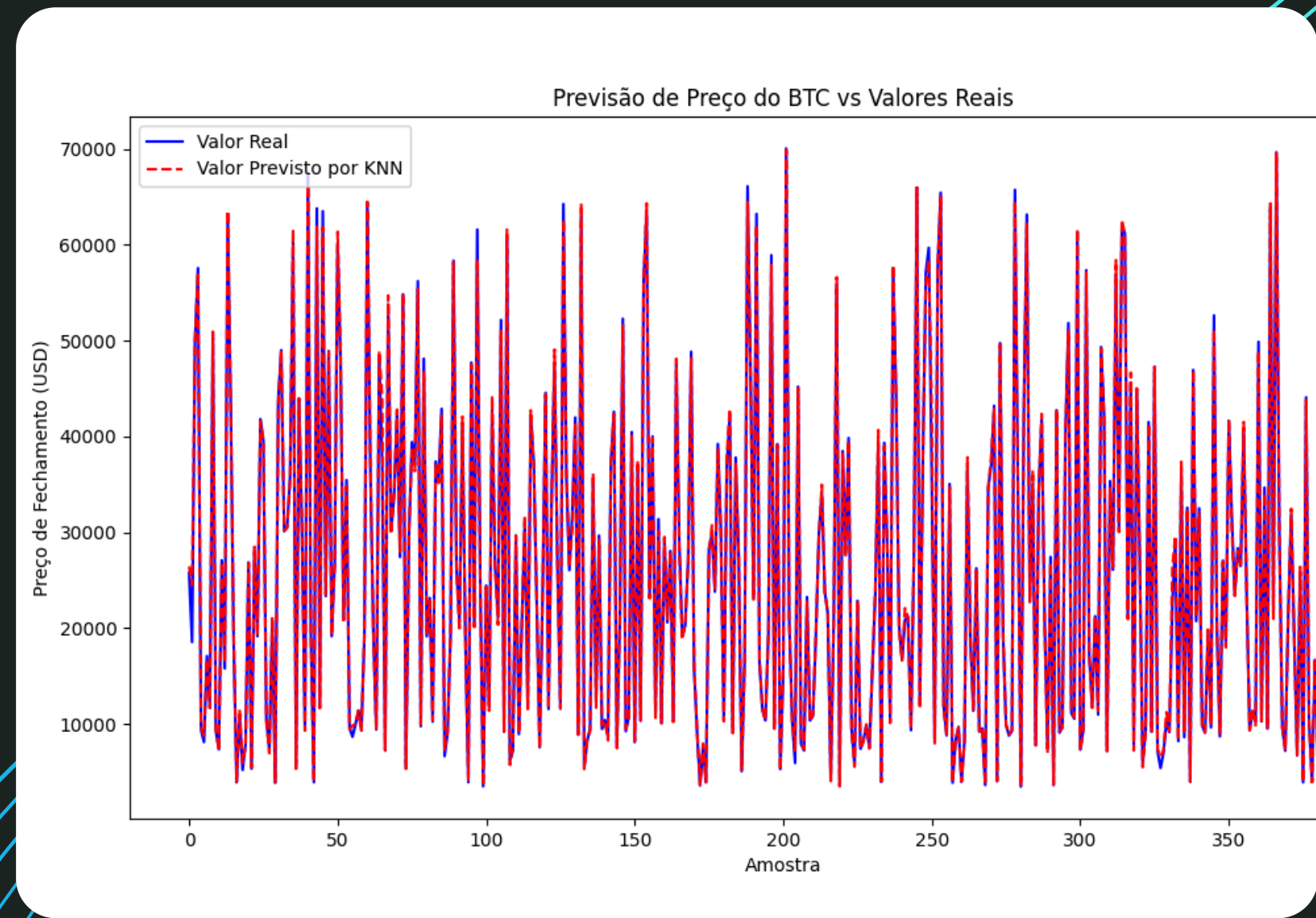
- Augut (Augur V2).



- Gnosis Protocol.

Previsão de BTC com IA

- Modelo para previsão de preços da BTC
- Dataset: Kaggle
 - Dados entre 2019 e 2024
- Algoritmo KNN
 - Previsão do preço de fechamento
- RMSE de \$802
 - Variação de 1% até 8% do valor real



CÓDIGO .PY SOBRE PREVISÃO DE BTC
DISPONÍVEL AO FINAL DA APRESENTAÇÃO





- Contrato escrito em Solidity, em ambiente Remix



- Armazena a previsão do BTC recebido do KNN



- Condição de transação:
 - se previsão ultrapassar um valor, contrato enviará uma quantia para o endereço designado

Desenvolvimento do Smart Contract

```
38 function storePrediction(uint _prediction) public payable {
39     lastPrediction = _prediction;
40     emit PredictionStored(_prediction); // Loga a previsão armazenada
41
42     // Verifica se a previsão excede o limite
43     if (_prediction > predictionThreshold) {
44         uint amount = address(this).balance; // Define o valor a ser enviado (p
45         require(amount > 0, "Sem saldo suficiente para transferir");
46
47         recipient.transfer(amount);
48         emit ThresholdBreachd(_prediction, recipient, amount);
49     }
50 }
```

logs: 0 hash: 0x48b...0e5e5
transact to BitcoinPredictionContract.deposit pending ...

[vm] from: 0xAb8...35cb2 to: BitcoinPredictionContract.deposit() 0x148...04927 value: 0 wei
data: 0xd0e...30db0 logs: 0 hash: 0x4d0...91a65
transact to BitcoinPredictionContract.storePrediction pending ...

[vm] from: 0xAb8...35cb2 to: BitcoinPredictionContract.storePrediction(uint256) 0x148...04927 value:
data: 0xc5e...0000a logs: 1 hash: 0x903...d682b

CÓDIGO .SOL SOBRE CONTRATO DE
MONITORAMENTO DE BTC DISPONÍVEL AO
FINAL DA APRESENTAÇÃO





- Interação, via Ganache, para rodar o contrato no arquivo .py



- Carrega o contrato e modelo KNN, para preparar a transação



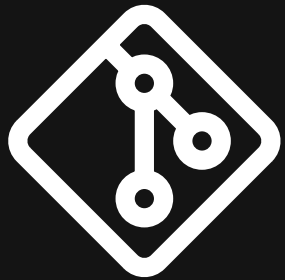
- Transação ocorre de uma conta origem para conta destino, quando supera um valor de previsão

Interação IA - Smart Contract

```
28
29 # Carregar o contrato
30 contract = w3.eth.contract(address=contract_address, abi=contract_abi)
31
32 # Carregar o modelo KNN
33 knn = joblib.load('knn_model.pkl')
34
35 # Fazer uma previsão
36 predicted_price = knn.predict([[10000]])
37
38 # Preparar a transação para armazenar a previsão no contrato
39 tx = contract.functions.storePrediction(predicted_price[0]).buildTransacti
40     'from': w3.eth.accounts[0], # Endereço da conta
41     'gas': 2000000,
42     'gasPrice': w3.toWei('20', 'gwei'),
43 }
```

CÓDIGO .PY SOBRE INTERAÇÃO DE
CONTRATO E MODELO KNN DE BTC
DISPONÍVEL AO FINAL DA APRESENTAÇÃO

Diretório de Implementação



[Link do GitHub \(clique aqui\)](#)

14

- BTC-USD.csv: dataset com dados da BTC;
- dadosBTC.py: código Python para previsão de dados;
 - knn_model.pkl: arquivo exportado para fazer interação com o contrato;
 - scaler.pkl: arquivo exportado para fazer interação com o contrato;
- BitcoinPredictionContract.sol: estrutura inicial de um contrato para armazenar previsões;
- modeloKNN-Contract.py: envia as previsões para blockchain através do contrato, para fazer transação;
- BitCoindPredictionContract-1.sol: dada uma previsão maior que o limite, envia-se fundos ao destinatário.

Obrigado!

CARLOS HENRIQUE HANNAS DE CARVALHO
Nº USP: 11965988



E-MAIL

carloshannas@usp.br



LINK DO VÍDEO

[clique aqui](#)



AUTOAVALIAÇÃO

Nota: 8,5

Autoavaliação

CARLOS HENRIQUE HANNAS DE CARVALHO

NOTA:

8,5

DESCRIÇÃO:

Aplicação prática da IA em análise preditiva de criptomoedas e monitoramento com contratos inteligentes. Isso demonstra a eficácia da ferramenta no contexto de criptomoedas e blockchain.