

KriptoWatcher: Robô de investimentos em criptomoedas utilizando algoritmo de arbitragem triangular

Mauricio Sugimoto Polloni

RA:191025895

Orientador: Prof. Dr. Kleber Rocha de Oliveira

Introdução

- Definição de robô de investimentos
 - Programa que executa uma Estratégia de Investimento de forma automatizada.
- Vantagens
 - Baixo custo
 - Visão estruturada de seus investimentos
 - Execução contínua

Introdução

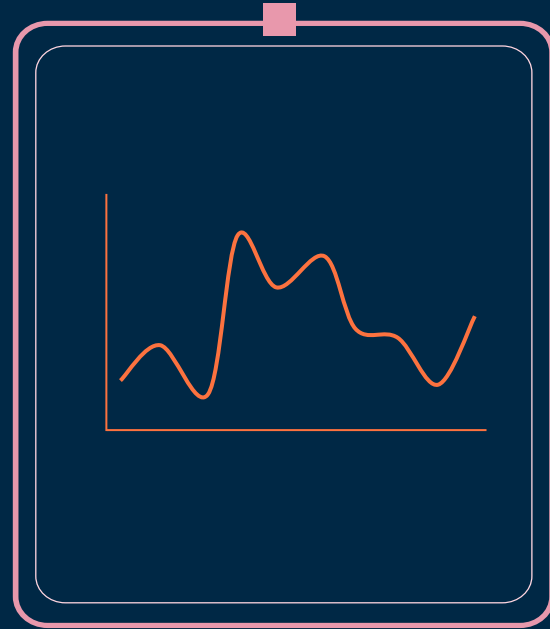
- API Restful
 - HTTP
- Corretoras que disponibilizam API
 - Binance
 - CoiNext

Fundamentação Teórica



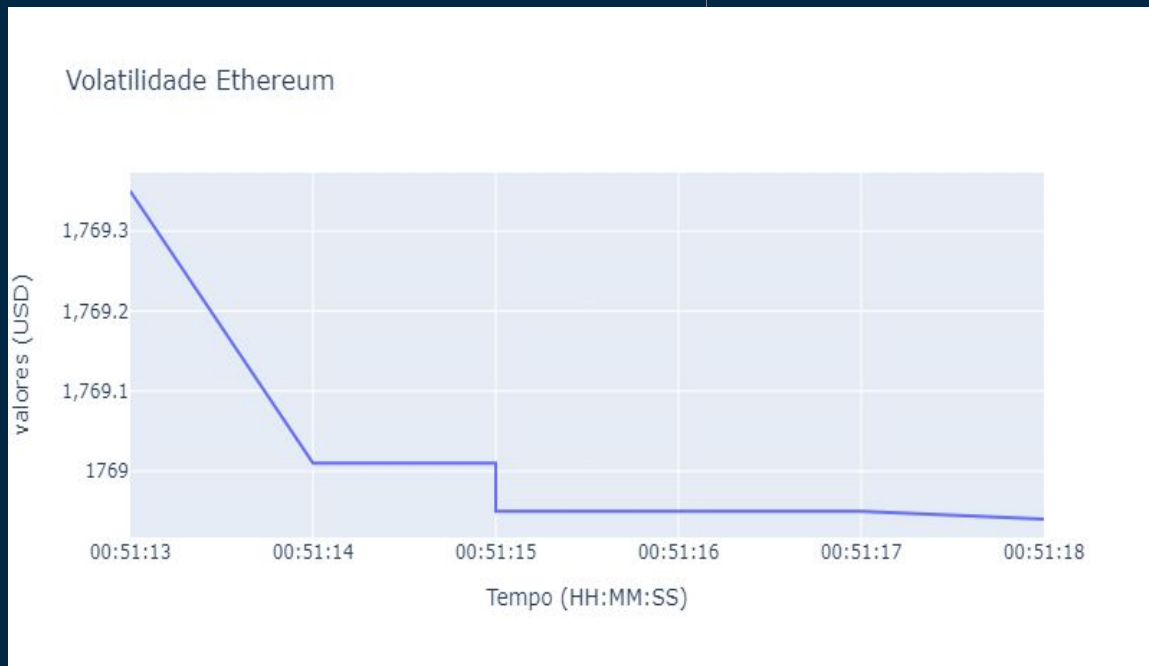
Volatilidade

- Variação do valor da criptomoeda
 - Frequência
 - Quantidade
- Bitcoin
 - Baixa volatilidade
- Dogecoin
 - Alta volatilidade



Volatilidade

- Ethereum
 - Alto valor quantitativo
 - Baixo valor percentual



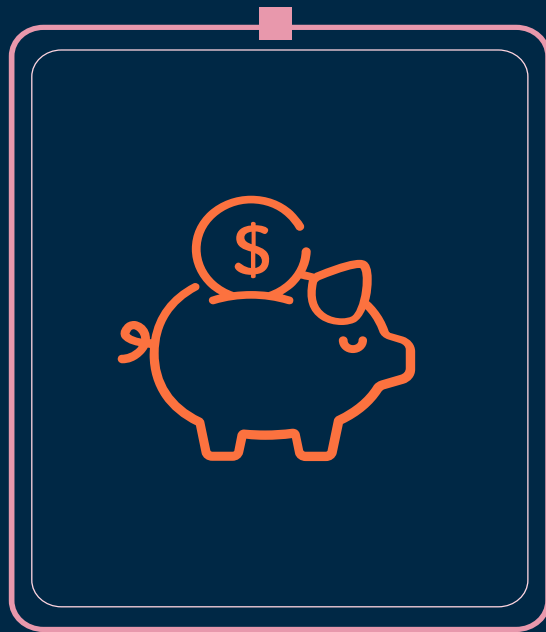
Volatilidade

- Apecoin
 - Baixo valor quantitativo
 - Alto valor percentual



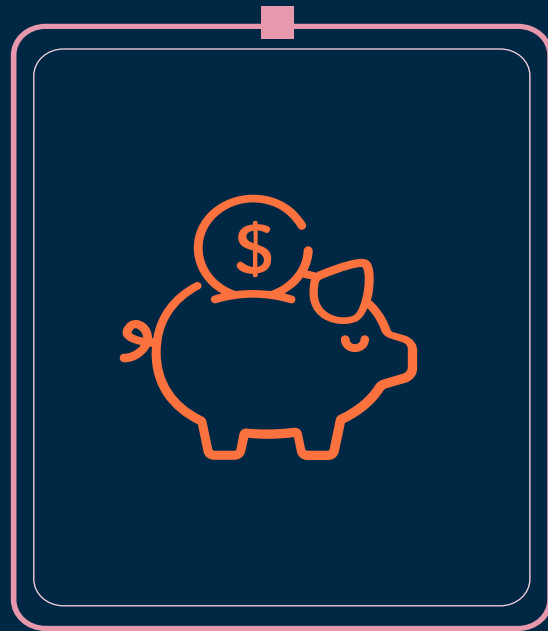
Criptomoedas

- Moeda digital
- Sem lastro
- Sem representação física



Stablecoins

- Associada a moeda fiduciária
 - Dolar
 - Euro
- Evitar altas volatilidades



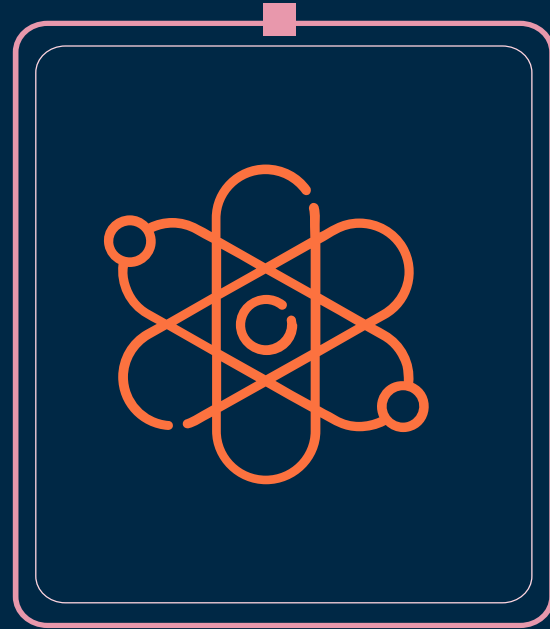
High frequency trade

- Negociações
 - Compra
 - Venda
- Alta frequencia
 - Processo automatizado



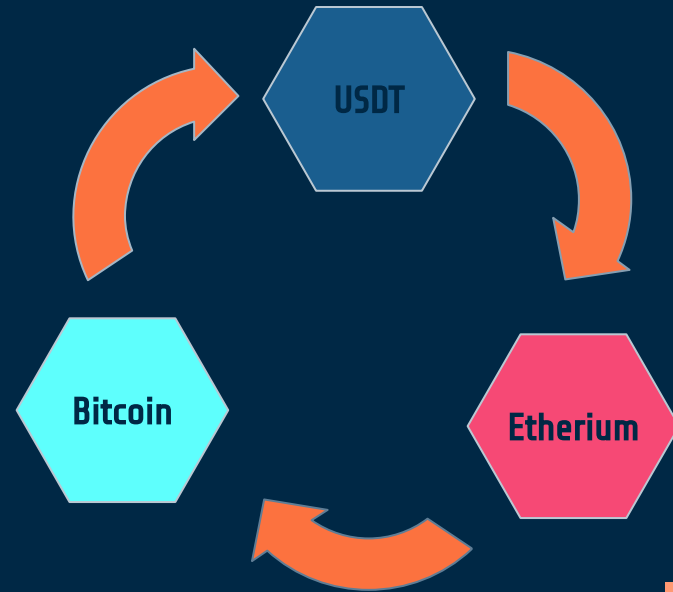
Algoritmo de Arbitragem

- Desequilíbrio de preços
 - Ineficiência nas comunicações
 - High frequency trade
 - **Oportunidade**
- Mercados diferentes
- Compra e venda simultânea
 - Compra mais barata
 - Venda mais cara

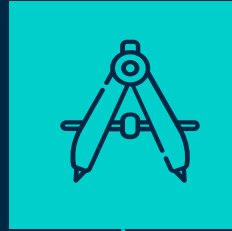


Algoritmo de Arbitragem Triangular

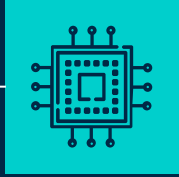
- Mesmo mercado
- Oportunidades com pouca duração
- 3 moedas
 - 1 Stablecoin
 - 2 criptomoedas



Ferramentas



AWS



Lambda

- Executa codigos
- Serverless
- Escalabilidade automática
- Aplicações curtas (15 minutos)



S3

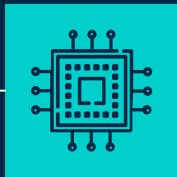
- Armazenamento de objetos
- Buckets
- Versionamento



CloudWatch e
EventBridge

- Monitoramento
- Regras
 - **Executar funções em períodos programados**

Codigo



Python

- Linguagem de programação
- Fácil execução de *Request*
- Aceito pelo Lambda



Boto3

- SDK em Python
- Acessar recursos AWS

Desenvolvimento

Utilizando Binance API e
ferramentas AWS



Requisições - Preços

Tipo	Get
URL	https://api1.binance.com/api/v3/ticker/price
Exemplo de resposta	<code>{"symbol": "LTCBTC", "price": "4.00000200"}</code>

Requisições - Compra/venda

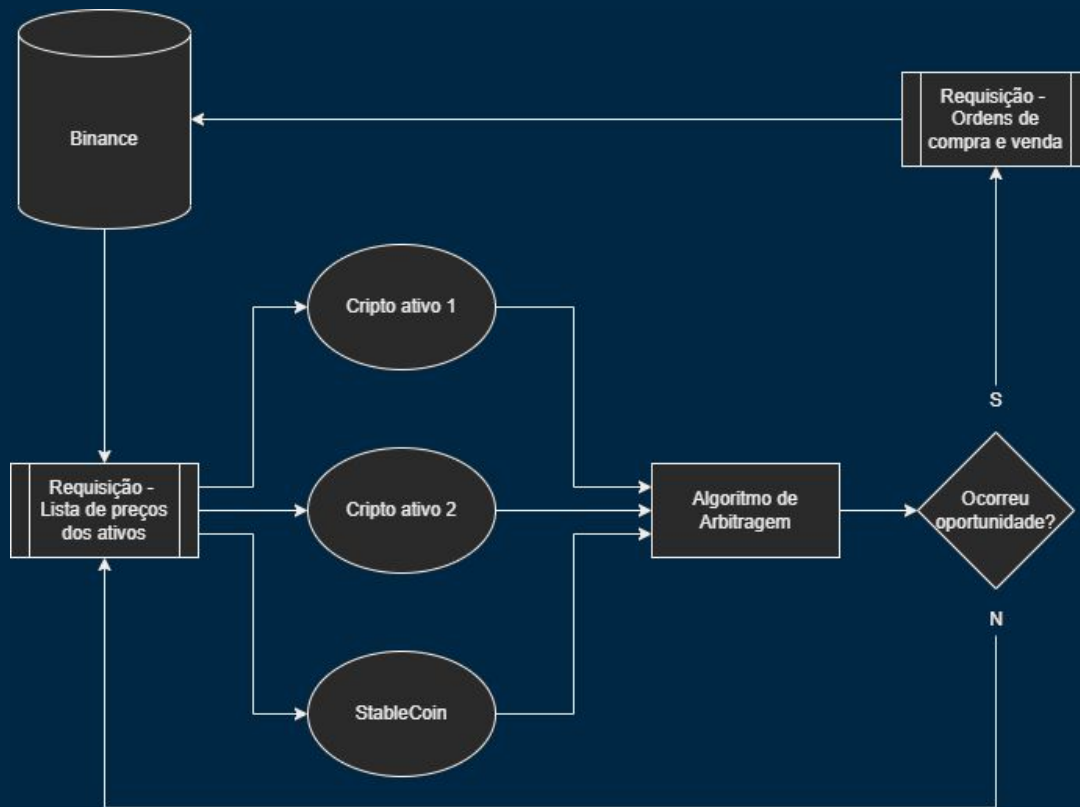
Tipo	Post
URL	https://api1.binance.com/api/v3/ticker/price
Corpo	<ul style="list-style-type: none">• Timestamp - tempo em milisegundos• Symbol - ex: BTCUSDT• Side - SELL ou BUY• Type - MARKET (padrão)

Pesos e Limites

- Limite de peso
 - Por IP
 - 1200 de peso por minuto
- Limite de ordens (compra ou venda)
 - Por conta
 - Até 50 ordens
 - Até 10 segundos



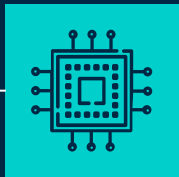
Esquema do projeto



Tratativa de Erros

Codigo do Status	Descrição
403	WAF Limit (Web Application Firewall)
429	Breaking a request rate limit
418	P has been auto-banned for continuing to send requests after receiving

Implantação AWS



Lambda

- Local de execução do código
- Até 30 segundos por execução
- Efetuou trade - Parar execução



S3

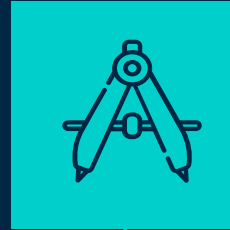
- Armazenamento de relatórios
- 1 Pasta por dia
- Arquivos JSON



CloudWatch e
EventBridge

- Monitoramento
- Regras
 - **Executar Lambda a cada 1 minuto**

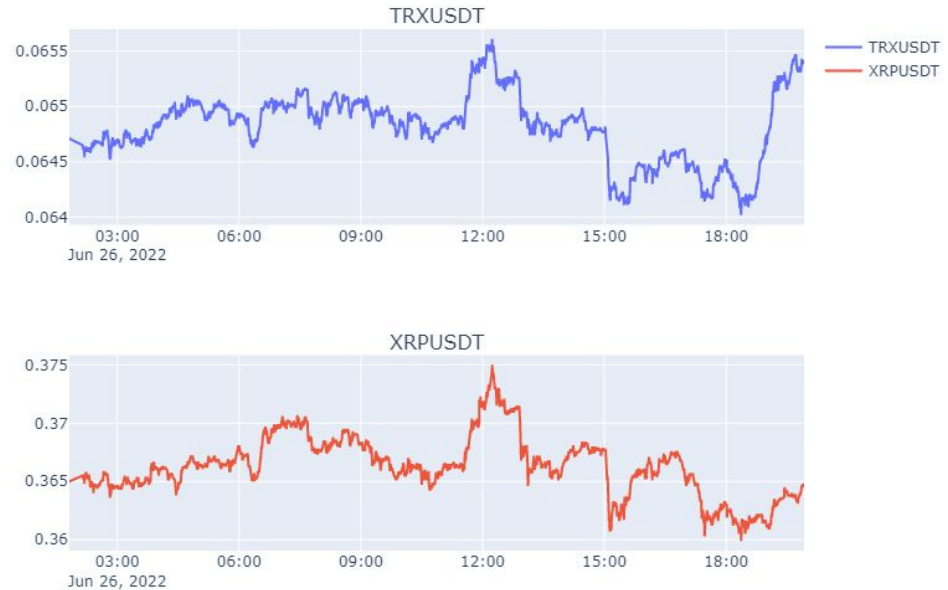
Conclusão



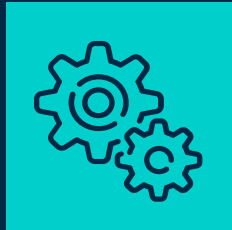
Analise dos dados

- Aproximadamente 18 horas
- Stablecoin USDT
- 2 criptomoedas
 - TRX
 - XRP

Variação do preço das criptomoedas TRX e XRP em relação a stablecoin USDT



Parâmetros e resultado



- Dados extraídos anteriormente
- Valor inicial fictício de 100 USDT
- Taxa de 0,1% por negociação

- ≈ 30 Oportunidades
- $\approx 0,12\%$ de ganho por execução



Projetos Futuros

- Melhorias no algoritmo
- Análise de melhores horários
- Melhores combinações

