

Analise Cambial com Python

João Victor Ferreira de Andrade

October 15, 2023

Abstract

O artigo a seguir foi feito para o Joãozinho estudar python e Latex, apenas.

1 Introdução

Vamos "brincar" com algumas séries históricas que encontrei, e vamos criar funções dinâmicas para executar a palhaçada

2 Consultando as séries históricas

Vamos começar consultando as séries históricas de compra e venda de dólar americano e real brasileiro. A fonte é API pública do Banco Central do Brasil.

VENDA = Link do endpoint

COMPRA = Link do endpoint

```
import requests
import json

def dadosVenda():
    url_venda = {VENDA}
    response_venda = requests.request("GET", url_venda)
    return json.loads(response_venda.text)

def dadosCompra():
    url_compra = {COMPRA}
    response_compra = requests.request("GET", url_compra)
    return json.loads(response_compra.text)

for item in dadosCompra():
    print(item)

for item in dadosVenda():
    print(item)
```

No código python foram usadas até o momento as bibliotecas: requests e json, para requisição e tratamento da resposta da API.¹

Com estes dados em "mãos", podemos começar o desenvolvimento dos cálculos necessários para fazer a análise desse tipo de negociação.

¹Os laços "for" são apenas para visualização dos dados, e devem ser retirados para o uso mais funcional

3 Calculando a variação média

Esta função vai calcular a média de variação entre registros. O seu uso é dinâmico, sendo necessário apenas respeitar o padrão de retorno

```
[
  {
    'data': '26/12/2014',
    'valor': '2.6806'
  },
  {
    'data': '29/12/2014',
    'valor': '2.6777'
  },
  ...
]
```

```
def variacaoMedia(dados):
    total = 0

    key = 0
    for item in dados:
        atual = item['valor']
        if key > 0:
            posterior = dados[key - 1]['valor']

            variacao = float(posterior) - float(atual)
            total += variacao

        key += 1
    return (total / key)
```



4 Calculando a valorização

$$V = \frac{(final - inicial)}{inicial} X 100 \quad (1)$$

$$V = final - inicial \quad (2)$$

Usando as equações acima, vamos calcular as valorizações real[2] e percentual[1].

```
def valorizacaoReal(dados):  
    atual = float(dados[0]['valor'])  
    max = float(dados[-1]['valor'])  
  
    return (atual - max)  
  
def valorizacaoPct(dados):  
    atual = float(dados[0]['valor'])  
    min = float(dados[-1]['valor'])  
  
    return (((atual - min) / min) * 100)
```

Com estas funções, é possível calcular a valorização de uma série ou/e a variação total