**ESCOLHA DE ATIVOS PARA INVESTIMENTOS**

****

Busca outras opções e maneiras de escolher ativos? Essa postagem é para você. Nossa proposta é apresentar mais uma forma que podes utilizar para julgar quais ativos escolher na hora de trabalhar sua carteira. Obviamente que este não é o único critério, mas com certeza é de fundamental importância.

Minhas decisões seguem um processo sistemático que pode ser replicado, pois eu mesmo aprendi com diversos investidores, livros, técnicas e tudo o mais que sigo estudando; inclusive conforme vamos evoluindo os métodos vão acompanhando as mudanças universais, o que nos garante sucesso nas estratégias com a minimização de riscos e maiores rendimentos. Faz-se necessário e importante a utilização da análise fundamentalista para dar mais força às escolhas.

Este documento tem caráter educacional. Quaisquer decisões tomadas a partir do que está sendo ensinado para posicionamento em investimentos, é de total responsabilidade do leitor.

PRIMEIRO: Quando decidimos investir, devemos responder algumas perguntas:

* Qual o objetivo do meu investimento?
* Qual o montante que almejo alcançar?
* Em quanto tempo?
* Quanto tempo tenho disponível para estudar as opções?
* Onde devo investir visando meu objetivo?

SEGUNDO: Depois que responder essas perguntas, podemos entender dois tipos básicos de rendas:

* Renda Fixa: Sabemos quanto nosso dinheiro vai render e, por causa dessa certeza, os riscos são menores e com isso, os rendimentos também tendem a ser menores. No entanto, dependendo do tempo dos aportes, rendas fixas podem apresentar melhor resultados do que rendas variáveis. Inclusive com a alta da Taxa Selic.
* Renda Variável: Compra e venda de ações, renda não constante, não se sabe o quanto vai valorizar no decorrer do tempo. Os riscos são maiores, mas com bom trabalho de escolha, quantidade e tempo, pode apresentar rendimentos maiores.

Para saber em qual tipo de investimento alocar o dinheiro, é importante saber qual nosso perfil de investidor. Assim, saberemos se tem mais aceitação ao risco para aproveitar investimentos mais voláteis.

**Fundos Imobiliários** apresentam volatilidades menores do que as ações. Fundos imobiliários podem ser considerados como pedaços de imóveis dos quais você recebe valores de aluguéis mensais, e esses pedaços conhecidos como cotas, podem valorizar ou desvalorizar com o passar do tempo.

Agora, realizaremos um processo algorítmico utilizando a linguagem de programação Python.

Pipeline para definição dos ativos:

* Coleta de Dados
* Limpeza e Tratamento dos Dados
* Definição dos Critérios de Escolha
* Visualização dos Dados
* Balanceamento com Algoritmo Genético
* Conclusão

**Coleta de Dados**

O nosso maior desafio para desenvolver estratégias quantitativas é encontrar uma boa base de informações. No Brasil, a B3 é a principal responsável por entregar dados financeiros. No entanto, existem alguns sites paralelos que oferecem ótimos dados gratuitos. Como vamos trabalhar com dados de fundos imobiliários, coletaremos os dados do site [Funds Explorer](https://www.fundsexplorer.com.br/).

Utilizaremos o [Google Colab](https://colab.research.google.com/) para realizar nossa análise.



**Importando as bibliotecas necessárias**

Instale:

!pip install yfinance

As demais bibliotecas são nativas, apenas a chamaremos em nosso ambiente:

from datetime import datetime #para trabalhar com datas

import pandas as pd #para trabalhar com tabelas

import yfinance as yf #para coletar dados financeiros

import requests #para enviar e receber solicitações da internet

from bs4 import BeautifulSoup #coletar dados de páginas web

import matplotlib.pyplot as plt #para trabalhar com gráficos

Agora faremos o webscraping:

url = '<https://www.fundsexplorer.com.br/ranking>'

agent = {"User-Agent":"Mozilla/5.0"}

response = requests.get(url, headers = agent)

soup = BeautifulSoup(response.text, 'lxml')

table = soup.find\_all('table')[0]

df = pd.read\_html(str(table), decimal=',', thousands='.')[0]

stocks\_df = df

print(df)

Agora tratamos os dados para coletar suas informações dos pregões diários.

**Limpeza e Tratamento dos Dados**

Para coletar os dados dos Fundos Imobiliários pela mesma biblioteca, precisamos incluir .SA na variável código. Criamos uma base de dados que recebe somente os valores dos códigos dos Fiis:

symbols = stocks\_df['Códigodo fundo'] + '.SA'

symbols

Temos agora um vetor com todos os códigos editados para coleta. Criamos uma função que coleta os dados financeiros para cada código. Para isso, definimos uma data de início de 2020-03-30 a 2022-03-30 (data de hoje). Após, criamos um dicionário para armazenar os dados de cada ativo.

start = datetime(2020, 3, 30)

end = datetime.today() #hoje

def get\_stocks(name):

stock = yf.download(name, start=start, end=end)

return stock

tickers = [tick for tick in symbols]

stock\_dict = {}

for tick in tickers:

try:

stock\_dict[tick] = get\_stocks(tick)

except:

print('Exception occured for - ', tick)

Para exemplificar o que fizemos no código acima, podemos pensar o seguinte:

data = yf.download('FIVN11.SA', start = '2020-3-30', end = datetime.today())

data

Dessa forma, temos as variáveis (Open, High, Low, Close, Adj Close e Volume).

**Critérios de Escolha**

Agora precisamos estabelecer uma estratégia. Minhas escolhas são sustentadas pelo valor atual do ativo com relação ao terceiro quartil. Estabelecemos um preço de referência que representa o valor do ativo em 75% das vezes comparado com o valor atual, nos dando um valor de desconto ou valorização do ativo. A variável escolhida é a de fechamento (Close).

**Descontos**

Criamos uma função para calcular os descontos dos ativos:

discount\_dict = {}

def calculate\_discount(df):

max\_price = df['Adj Close'].describe()['75%']

min\_price = df['Adj Close'].min()

return (min\_price - max\_price)/max\_price

for tick in stock\_dict.keys():

discount\_dict[tick] = calculate\_discount(stock\_dict[tick])

**Retornos**

Calculamos os retornos de cada ativo desde a data inicial:

returns = pd.DataFrame()

for key in stock\_dict.keys():

returns[key] = stock\_dict[key]['Adj Close'].pct\_change()

returns.head()

**Risco**

Entendemos como risco o desvio padrão dos retornos:

def get\_stockname(ticker):

return [x for x in stocks\_df[stocks\_df['Códigodo fundo'] == ticker]['Códigodo fundo']][0]

std\_list = []

for tick in stock\_info\_df['ticker']:

std\_list.append(returns[tick].std()) # risk metric

stock\_info\_df['risk'] = std\_list

stock\_info\_df

**Visualização**

Agora com os descontos e retornos, geramos a visualização dos dados para nos auxiliar na tomada de decisão. Os dados oferecidos pela página [Funds Explorer](https://www.fundsexplorer.com.br/) oferecem também o setor de cada ativo. Com isso, criamos uma função para relacionar os setores de cada um e assim, temos um novo dataframe com as seguintes variáveis:

* Ticker = código do ativo
* Discount = desconto com relação ao terceiro quartil
* Sector = setor do ativo
* Risk = desvio padrão dos retornos

def get\_sector(stock):

return [x for x in stocks\_df[symbols == stock]['Setor']]

stock\_info\_df = pd.DataFrame.from\_dict(discount\_dict, orient='index').reset\_index()

stock\_info\_df.columns = ['ticker', 'discount']

stock\_info\_df['sector'] = [get\_sector(tick)[0] for tick in stock\_info\_df['ticker']]

stock\_info\_df

Agora criamos a visualização:

import plotly.express as px

stock\_info\_df = stock\_info\_df.dropna()

fig = px.scatter(stock\_info\_df, x = "risk", y = "discount", size = "current\_price", color = "sector", hover\_name = "ticker", log\_x = True)

fig.show()

No gráfico gerado, temos todos os FIIS distribuídos em dois eixos. No eixo horizontal, os riscos e no eixo vertical, os descontos. Sobre cada bolha, obtemos os valores relacionados ao ativo.

O gráfico nos ajuda a escolher um ativo pelo seu risco e desconto. Quanto mais a direita estiver no gráfico, maior o risco. E quanto mais abaixo estiver, maior o desconto. Assim podemos escolher ativos mais baratos em relação àqueles que estiverem na mesma linha de risco.