

In [813...

```
pip install yfinance
```

```

Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: yfinance in /usr/local/lib/python3.7/dist-packages (0.1.74)
Requirement already satisfied: multitasking>=0.0.7 in /usr/local/lib/python3.7/dist-packages (from yfinance) (0.0.11)
Requirement already satisfied: lxml>=4.5.1 in /usr/local/lib/python3.7/dist-packages (from yfinance) (4.9.1)
Requirement already satisfied: numpy>=1.15 in /usr/local/lib/python3.7/dist-packages (from yfinance) (1.21.6)
Requirement already satisfied: requests>=2.26 in /usr/local/lib/python3.7/dist-packages (from yfinance) (2.28.1)
Requirement already satisfied: pandas>=0.24.0 in /usr/local/lib/python3.7/dist-packages (from yfinance) (1.3.5)
Requirement already satisfied: python-dateutil>=2.7.3 in /usr/local/lib/python3.7/dist-packages (from pandas>=0.24.0->yfinance) (2.8.2)
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.7/dist-packages (from pandas>=0.24.0->yfinance) (2022.1)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from python-dateutil>=2.7.3->pandas>=0.24.0->yfinance) (1.15.0)
Requirement already satisfied: charset-normalizer<3,>=2 in /usr/local/lib/python3.7/dist-packages (from requests>=2.26->yfinance) (2.1.0)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /usr/local/lib/python3.7/dist-packages (from requests>=2.26->yfinance) (1.24.3)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.7/dist-packages (from requests>=2.26->yfinance) (2022.6.15)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.7/dist-packages (from requests>=2.26->yfinance) (2.10)

```

In [814...

```

import yfinance as yf
import pandas as pd
import numpy as np
import random
import matplotlib.pyplot as plt
import seaborn as sns
import datetime as dt

```

In [815...

```

# Choose an ETF with a minimum of 100 assets, identify those assets
spy500_url = "https://www.ssga.com/us/en/intermediary/etfs/library-content/products/fund-data/etfs/us/holdings-tickers"
tickers = list(pd.read_excel(spy500_url, header=4).dropna().Ticker)

```

In [816...

```
print(tickers[:10], '\n', 'ticker amount:', len(tickers))
```

```
['AAPL', 'MSFT', 'AMZN', 'TSLA', 'GOOGL', 'GOOG', 'BRK.B', 'UNH', 'NVDA', 'JNJ']
ticker amount: 505
```

In [817...

```
# Retrieve historical data for your chosen ETF
data = yf.download(tickers, period="10y")[[ "Adj Close", "Volume" ]]
data_5y = yf.download(tickers, period="5y")[[ "Adj Close", "Volume" ]]

[*****100%*****] 505 of 505 completed

3 Failed downloads:
- CASH_USD: No data found, symbol may be delisted
- BF.B: No data found for this date range, symbol may be delisted
- BRK.B: No data found, symbol may be delisted
[*****100%*****] 505 of 505 completed

3 Failed downloads:
- CASH_USD: No data found, symbol may be delisted
- BF.B: No data found for this date range, symbol may be delisted
- BRK.B: No data found, symbol may be delisted
```

In [818...

```
tickers=list(data['Adj Close'].dropna(how='all', axis=1).columns)
len(tickers)
```

Out[818... 502

In [839...

```
# Calculate the price momentum factors for each asset in your ETF

class BasketSelector:

    def __init__(self, price, volume, tickers):
        self.price = price
        self.volume = volume
        self.factor_df= pd.DataFrame(index= tickers)

    def momentum_factors(self):
        # calculate m_of_52wk_trend
        fit=[]
        for i in tickers:
            fit= np.polyfit(np.arange(252), self.price[-252:], deg=1)
            self.factor_df['m_of_52wk_trend']= fit[0]
            # fit= np.polyfit(np.arange(252), self.price.rolling(252)[-20:], deg=1)
            # self.factor_df['m_of_52wk_trend']= fit[0]
```

```

# caculate pct above 260 days low
self.low_price= self.price.rolling(260).min()
self.factor_df['pct_abv_260_low'] = ((self.price-self.low_price)/self.low_price)[-20:].mean()

# calculate 4/52 price_oscillator
price_window_4wk = self.price.rolling(20)
price_window_42wk = self.price.rolling(260)
self.four_wk_high, self.four_wk_low = price_window_4wk.max(), price_window_4wk.min()
self.fifty2_wk_high, self.fifty2_wk_low = price_window_42wk.max(), price_window_42wk.min()
self.factor_df['4/52_price_oscillator'] = ((self.four_wk_high- self.four_wk_low)
                                           / (self.fifty2_wk_high-self.fifty2_wk_low))[-20:].mean()

# calculate 39 week return
self.factor_df['39_week_return']=self.price.pct_change(int(260*.75))[-20:].mean()

# calulate 51wk volume price trend
# VPT = Previous VPT + Volume x (Today's Closing Price - Previous Closing Price)
/ Previous Closing Price
self.factor_df['51wk_volume_price_tr']=(self.price.pct_change() * self.volume)
                                           .rolling(252).sum()[-20:].mean()

return self.factor_df

def output(self):
    self.factors=self.momentum_factors()
    # Using the price momentum factors, calculate the monthly z-factor score for each asset
    self.z_scored = ((self.factor_df - self.factor_df.mean())/self.factor_df.std()).sum(axis=1)
    # Identify long and short baskets (10 to 15 assets in each) using calculated z-factors
    return self.z_scored.nlargest(10).index,self.z_scored.nsmallest(10).index

```

```

In [841... BS= BasketSelector(price=data['Adj Close'][tickers], volume= data['Volume'][tickers], tickers=tickers)
Long, Short = BS.output()

```

```

In [842... Long

```

```

Out[842... Index(['OXY', 'DVN', 'ENPH', 'MRO', 'DLTR', 'CF', 'APA', 'ON', 'CTRA', 'AZO'], dtype='object')

```

```

In [843... Short

```

```

Out[843... Index(['NFLX', 'PYPL', 'META', 'NVR', 'ALGN', 'CCL', 'ADBE', 'ILMN', 'CHTR',
      'AMZN'],

```

```
dtype='object')
```

In [834...

```
# factor_df = BS.momentum_factors(). # will have to make changes in the class to have the following table
# factor_df['z_score']=z_score
# factor_df
```

Out[834...

	m_of_52wk_trend	pct_abv_260_low	4/52_price_oscillator	39_week_return	51wk_volume_price_tr	z_score
A	-0.220534	0.120062	0.216077	-0.180252	-4.248036e+05	-1.499844
AAL	-0.025045	0.178783	0.225148	-0.284185	4.927216e+06	-0.381727
AAP	-0.116891	0.149033	0.251062	-0.111210	-4.327748e+05	-0.392660
AAPL	0.003304	0.197862	0.377237	0.080084	-1.537707e+05	2.477168
ABBV	0.227003	0.435081	0.166895	0.392766	-2.290504e+06	3.088144
...
YUM	-0.059405	0.102318	0.268906	-0.028349	-1.160477e+05	0.120404
ZBH	-0.137934	0.068124	0.165832	-0.235138	-8.562623e+05	-2.449923
ZBRA	-1.337125	0.119601	0.138312	-0.371479	-2.578047e+05	-5.881269
ZION	-0.023765	0.113965	0.201004	-0.156231	-1.027990e+05	-1.077739
ZTS	-0.195437	0.132026	0.180886	-0.119330	-8.345355e+05	-1.541280

502 rows × 6 columns

In [844...

```
# create empty list to store long, short, long_return and short_return
lol=[]
los=[]
lr=[]
sr=[]
```

In [847...

```
# Create a backtest to validate performance of your algorithm based on monthly restructuring over the
# previous 5 years
monthly_date= pd.date_range(data_5y['Adj Close'].index[0], end=dt.datetime.today(), freq='M')
df=pd.DataFrame(index=monthly_date)
for date in df.index:
    BS= BasketSelector(price=data['Adj Close'][tickers][data['Adj Close'][tickers].index <= date],
```

```

                                Project_3
                                volume= data['Volume'][tickers], tickers=tickers)

Long, Short = BS.output()
lol.append(Long)
los.append(Short)

for next_date in df.index.shift(1):
    lr.append(data['Adj Close'][data['Adj Close'].index <= next_date]
               [Long].pct_change(20)[-1:].mean(axis=1)[0])
    sr.append(data['Adj Close'][data['Adj Close'].index <= next_date]
               [Short].pct_change(20)[-1:].mean(axis=1)[0])

```

In [848...

```

# Put appended stuff into the dataframe
df['Long']=[i for i in lol]
df['Short']=[i for i in los]
df['Long_return']=[i for i in lr]
df['Short_return']=[i for i in sr]
df['Avg_port_return']=df[['Long_return', 'Short_return']].mean(axis=1)

```

In [849...

```
df.head(10)
```

Out [849...

	Long	Short	Long_return	Short_return	Avg_port_return
2017-08-31	[[NVR, NVDA, BKNG, NRG, ANET, SEDG, TTWO, ALGN...	[[AZO, GWW, BBWI, ORLY, KR, DVN, AAP, QCOM, AK...	0.133873	-0.000967	0.066453
2017-09-30	[[NVR, NVDA, TTWO, NRG, ALGN, VRTX, SEDG, MU, ...	[[AZO, AAP, ORLY, BBWI, AKAM, CMG, GWW, KR, MO...	0.015469	0.098115	0.056792
2017-10-31	[[NVR, SEDG, NVDA, ALGN, MU, TTWO, NRG, LRCX, ...	[[AZO, AAP, KR, CMG, ORLY, MOS, BBWI, SJM, KIM...	0.046078	0.049232	0.047655
2017-11-30	[[NVR, ALGN, SEDG, ENPH, ANET, TTWO, NVDA, MU,...	[[AZO, AAP, CMG, ULTA, KR, ORLY, GE, KIM, EFX,...	0.056388	-0.027735	0.014327
2017-12-31	[[NVR, ENPH, SEDG, ALGN, ANET, ETSY, TTWO, PEN...	[[GE, CMG, INCY, NWL, AAP, ULTA, BKR, DISH, HS...	0.011358	0.116099	0.063729
2018-01-31	[[NVR, ALGN, SEDG, ANET, NVDA, ENPH, BA, ABMD,...	[[GE, CMG, NWL, INCY, DISH, CPB, ULTA, KHC, KI...	-0.131605	-0.020929	-0.076267
2018-02-28	[[NVR, ENPH, SEDG, ALGN, ANET, ABMD, NVDA, BA,...	[[GE, CMG, INCY, ULTA, NWL, DISH, EIX, PEAK, A...	0.005082	-0.001710	0.001686
2018-03-31	[[ENPH, NVR, SEDG, ETSY, MTCH, ABMD, NFLX, ALG...	[[GE, REGN, CMG, NWL, ULTA, INCY, ALK, WELL, D...	0.105089	0.087646	0.096367

	Long	Short	Long_return	Short_return	Avg_port_return
2018-04-30	[[ENPH, NVR, SEDG, NFLX, MTCH, ABMD, ETSY, ALG...	[[REGN, GE, NWL, DISH, CMG, EXPE, PPL, PEAK, K...	0.039903	0.097120	0.068512
2018-05-31	[[ENPH, NVR, ABMD, SEDG, NFLX, ALGN, SIVB, MTC...	[[REGN, GE, INCY, NWL, DISH, EXPE, KHC, CHTR, ...	0.033491	0.000228	0.016859

In [852...

```
# etf monthly return
etf_ret=[]
```

In [853...

```
for date in df.index:
    BS= BasketSelector(price=data['Adj Close'][tickers][data['Adj Close'][tickers].index <= date],
                        volume= data['Volume'][tickers], tickers=tickers)

    Long, Short = BS.output()
    for next_date in df.index.shift(1):
        etf_ret.append(data_5y['Adj Close'][tickers][data_5y['Adj Close'][tickers].index <= next_date]
                        .pct_change(20)[-1:].mean(axis=1)[0])
```

In [854...

```
df['etf_return']= [i for i in etf_ret]
```

In [859...

```
df.head(10)
```

Out [859...

	Long	Short	Long_return	Short_return	Avg_port_return	etf_return
2017-08-31	[[NVR, NVDA, BKNG, NRG, ANET, SEDG, TTWO, ALGN...	[[AZO, GWW, BBWI, ORLY, KR, DVN, AAP, QCOM, AK...	0.133873	-0.000967	0.066453	0.030148
2017-09-30	[[NVR, NVDA, TTWO, NRG, ALGN, VRTX, SEDG, MU, ...	[[AZO, AAP, ORLY, BBWI, AKAM, CMG, GWW, KR, MO...	0.015469	0.098115	0.056792	0.016920
2017-10-31	[[NVR, SEDG, NVDA, ALGN, MU, TTWO, NRG, LRCX, ...	[[AZO, AAP, KR, CMG, ORLY, MOS, BBWI, SJM, KIM...	0.046078	0.049232	0.047655	0.040900
2017-11-30	[[NVR, ALGN, SEDG, ENPH, ANET, TTWO, NVDA, MU,...	[[AZO, AAP, CMG, ULTA, KR, ORLY, GE, KIM, EFX,...	0.056388	-0.027735	0.014327	0.010032
2017-12-31	[[NVR, ENPH, SEDG, ALGN, ANET, ETSY, TTWO, PEN...	[[GE, CMG, INCY, NWL, AAP, ULTA, BKR, DISH, HS...	0.011358	0.116099	0.063729	0.041199

	Long	Short	Long_return	Short_return	Avg_port_return	etf_return
2018-01-31	[[NVR, ALGN, SEDG, ANET, NVDA, ENPH, BA, ABMD,...	[[GE, CMG, NWL, INCY, DISH, CPB, ULTA, KHC, KI...	-0.131605	-0.020929	-0.076267	-0.037045
2018-02-28	[[NVR, ENPH, SEDG, ALGN, ANET, ABMD, NVDA, BA,...	[[GE, CMG, INCY, ULTA, NWL, DISH, EIX, PEAK, A...	0.005082	-0.001710	0.001686	0.005771
2018-03-31	[[ENPH, NVR, SEDG, ETSY, MTCH, ABMD, NFLX, ALG...	[[GE, REGN, CMG, NWL, ULTA, INCY, ALK, WELL, D...	0.105089	0.087646	0.096367	0.024887
2018-04-30	[[ENPH, NVR, SEDG, NFLX, MTCH, ABMD, ETSY, ALG...	[[REGN, GE, NWL, DISH, CMG, EXPE, PPL, PEAK, K...	0.039903	0.097120	0.068512	0.028331
2018-05-31	[[ENPH, NVR, ABMD, SEDG, NFLX, ALGN, SIVB, MTC...	[[REGN, GE, INCY, NWL, DISH, EXPE, KHC, CHTR, ...	0.033491	0.000228	0.016859	0.002553

In [949...

```
# Continuation of the above table
df.iloc[:,2:-1].head()
```

Out [949...

	Long_return	Short_return	Avg_port_return
2017-08-31	0.133873	-0.000967	0.066453
2017-09-30	0.015469	0.098115	0.056792
2017-10-31	0.046078	0.049232	0.047655
2017-11-30	0.056388	-0.027735	0.014327
2017-12-31	0.011358	0.116099	0.063729

In [903...

```
# Monthly portfolio return bar chart (pos/neg coloring) vs ETF
table_1=df.iloc[:,4:6]
```

In [905...

```
table_1.head()
```

Out [905...

	Avg_port_return	etf_return
2017-08-31	0.066453	0.030148
2017-09-30	0.056792	0.016920

	Avg_port_return	etf_return
2017-10-31	0.047655	0.040900
2017-11-30	0.014327	0.010032
2017-12-31	0.063729	0.041199

In [875...

```
plt.figure(figsize=(15, 10))
plt.bar(df.index, height=df['Avg_port_return'], width=25)
plt.bar(df.index, height=-df['etf_return'], width=25)

plt.xlabel("Dates (Month)")
plt.ylabel("Returns")
plt.title("Monthly portfolio return bar chart (pos/neg coloring) vs ETF")
plt.show()
```




In [925...

```
# Monthly return for long picks vs short picks vs ETF
table_2= df.iloc[:,2:6].drop(columns=['Avg_port_return'])
table_2.head()
```

Out [925...

	Long_return	Short_return	etf_return
2017-08-31	0.133873	-0.000967	0.030148

	Long_return	Short_return	etf_return
2017-09-30	0.015469	0.098115	0.016920
2017-10-31	0.046078	0.049232	0.040900
2017-11-30	0.056388	-0.027735	0.010032
2017-12-31	0.011358	0.116099	0.041199

In [933...

```
plt.figure(figsize=(10, 8))
plt.plot(table_2['Long_return'], label='Long_return')
plt.plot(table_2['Short_return'], label='Short_return')
plt.plot(table_2['etf_return'], label='etf_return')

plt.legend()
plt.xlabel("Dates (Month)")
plt.ylabel("Return")
plt.title("Monthly return for long picks vs short picks vs ETF")
plt.show()
```



In [936...

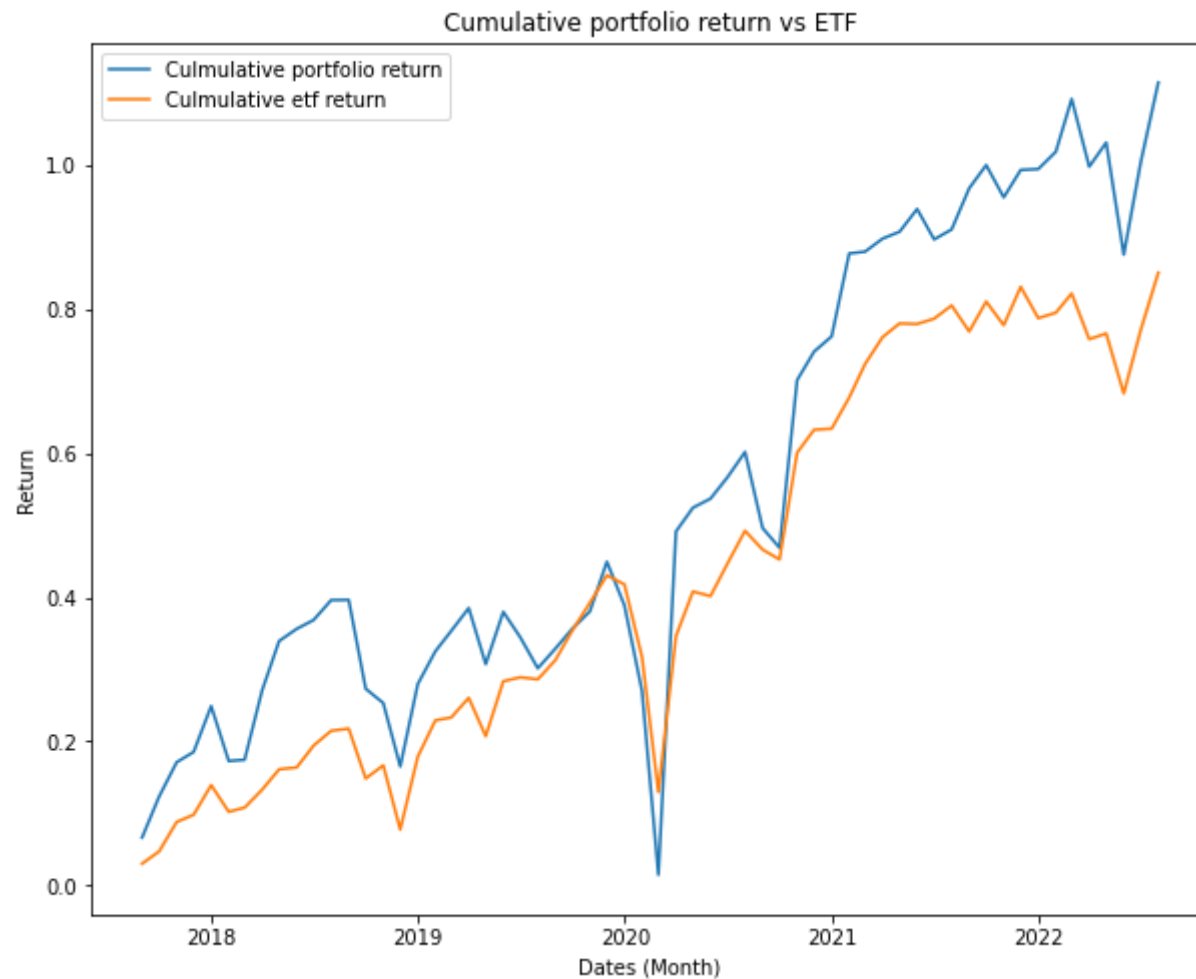
```
# Cumulative portfolio return vs ETF
cum_sum_port=df.Avg_port_return.cumsum()
cum_sum_etf=df.etf_return.cumsum()
```

In [938...

```
plt.figure(figsize=(10, 8))
plt.plot(cum_sum_port, label='Culmulative portfolio return')
plt.plot(cum_sum_etf, label='Culmulative etf return')

plt.legend()
plt.xlabel("Dates (Month)")
plt.ylabel("Return")
```

```
plt.title("Cumulative portfolio return vs ETF")
plt.show()
```



In [944...

```
# Extra add-in: annualized portfolio and etf return
annualize_return= pd.DataFrame(df[['Avg_port_return','etf_return']].mean()*12,columns=['returns'])
annualize_return
```

Out[944...

	returns
Avg_port_return	0.222942
etf_return	0.170157

In [947...

```
# same table as above, more entries
df
```

Out [947...

	Long	Short	Long_return	Short_return	Avg_port_return	etf_return
2017-08-31	[[NVR, NVDA, BKNG, NRG, ANET, SEDG, TTWO, ALGN...	[[AZO, GWW, BBWI, ORLY, KR, DVN, AAP, QCOM, AK...	0.133873	-0.000967	0.066453	0.030148
2017-09-30	[[NVR, NVDA, TTWO, NRG, ALGN, VRTX, SEDG, MU, ...	[[AZO, AAP, ORLY, BBWI, AKAM, CMG, GWW, KR, MO...	0.015469	0.098115	0.056792	0.016920
2017-10-31	[[NVR, SEDG, NVDA, ALGN, MU, TTWO, NRG, LRCX, ...	[[AZO, AAP, KR, CMG, ORLY, MOS, BBWI, SJM, KIM...	0.046078	0.049232	0.047655	0.040900
2017-11-30	[[NVR, ALGN, SEDG, ENPH, ANET, TTWO, NVDA, MU,...	[[AZO, AAP, CMG, ULTA, KR, ORLY, GE, KIM, EFX,...	0.056388	-0.027735	0.014327	0.010032
2017-12-31	[[NVR, ENPH, SEDG, ALGN, ANET, ETSY, TTWO, PEN...	[[GE, CMG, INCY, NWL, AAP, ULTA, BKR, DISH, HS...	0.011358	0.116099	0.063729	0.041199
2018-01-31	[[NVR, ALGN, SEDG, ANET, NVDA, ENPH, BA, ABMD,...	[[GE, CMG, NWL, INCY, DISH, CPB, ULTA, KHC, KI...	-0.131605	-0.020929	-0.076267	-0.037045
2018-02-28	[[NVR, ENPH, SEDG, ALGN, ANET, ABMD, NVDA, BA,...	[[GE, CMG, INCY, ULTA, NWL, DISH, EIX, PEAK, A...	0.005082	-0.001710	0.001686	0.005771
2018-03-31	[[ENPH, NVR, SEDG, ETSY, MTCH, ABMD, NFLX, ALG...	[[GE, REGN, CMG, NWL, ULTA, INCY, ALK, WELL, D...	0.105089	0.087646	0.096367	0.024887
2018-04-30	[[ENPH, NVR, SEDG, NFLX, MTCH, ABMD, ETSY, ALG...	[[REGN, GE, NWL, DISH, CMG, EXPE, PPL, PEAK, K...	0.039903	0.097120	0.068512	0.028331
2018-05-31	[[ENPH, NVR, ABMD, SEDG, NFLX, ALGN, SIVB, MTC...	[[REGN, GE, INCY, NWL, DISH, EXPE, KHC, CHTR, ...	0.033491	0.000228	0.016859	0.002553
2018-06-30	[[ENPH, ABMD, NFLX, TWTR, ALGN, NVR, ETSY, SED...	[[REGN, GE, INCY, CHTR, NWL, DISH, HWM, PM, MH...	0.035455	-0.010686	0.012385	0.030371
2018-07-31	[[ENPH, ABMD, NFLX, ETSY, ALGN, BKNG, GWW, TWT...	[[GE, REGN, INCY, CHTR, ALB, DISH, NWL, PM, NL...	0.006606	0.048592	0.027599	0.020620
2018-08-31	[[ENPH, ABMD, DXCM, ETSY, CMG, NFLX, GWW, PAYC...	[[NVR, GE, INCY, ALB, CHTR, BBWI, REGN, MHK, N...	0.015622	-0.015101	0.000260	0.002928
2018-09-30	[[AMD, ABMD, ETSY, DXCM, ENPH, NFLX, MTCH, CMG...	[[NVR, MHK, GE, XRAY, NWL, IVZ, WHR, INCY, DIS...	-0.122359	-0.124399	-0.123379	-0.069256
2018-10-31	[[ENPH, DXCM, ABMD, ETSY, AMD, NFLX, FTNT, MOH...	[[NVR, MHK, XRAY, INCY, WDC, BBWI, NWL, WHR, L...	-0.044272	0.004583	-0.019844	0.018285

	Long	Short	Long_return	Short_return	Avg_port_return	etf_return
2018-11-30	[[ETSY, ENPH, DXCM, ABMD, CMG, AMD, AAP, MOH, ...	[[NVR, MHK, XRAY, IVZ, BLK, WDC, WYNN, AMAT, I...	-0.093128	-0.082913	-0.088020	-0.089177
2018-12-31	[[ETSY, DXCM, ENPH, MOH, CMG, AMD, ILMN, AAP, ...	[[NVR, MHK, WDC, GE, BLK, XRAY, EA, URI, WYNN,...	0.090616	0.138236	0.114426	0.101260
2019-01-31	[[ETSY, DXCM, ENPH, CMG, AMD, MOH, AZO, ORLY, ...	[[NVR, BKNG, MHK, WDC, BLK, WYNN, IVZ, NVDA, A...	0.025714	0.065181	0.045448	0.050375
2019-02-28	[[ENPH, ETSY, AMD, CMG, DXCM, PAYC, AZO, MOH, ...	[[NVR, BKNG, MHK, BLK, FDX, PXD, ATVI, WRK, HA...	0.026591	0.029974	0.028282	0.003978
2019-03-31	[[ETSY, ENPH, AZO, CMG, AMD, DXCM, MOH, PAYC, ...	[[NVR, BKNG, MHK, FDX, BLK, STZ, WRK, ATVI, WY...	-0.000285	0.063799	0.031757	0.027331
2019-04-30	[[AZO, AMD, CMG, ENPH, PAYC, ETSY, ORLY, ZBRA,...	[[BKNG, MHK, KHC, NVR, NWL, ATVI, ABMD, LUMN, ...	-0.101007	-0.054617	-0.077812	-0.053308
2019-05-31	[[ENPH, AZO, CMG, PAYC, MTCH, ETSY, NOW, AMD, ...	[[BKNG, ABMD, BIIB, KHC, MHK, ATVI, WAB, FDX, ...	0.086083	0.058952	0.072518	0.076311
2019-06-30	[[ENPH, NVR, AZO, CMG, MKTX, MTD, PAYC, NOW, A...	[[ABMD, BIIB, FDX, KHC, VTRS, ATVI, HAL, STT, ...	-0.014658	-0.055880	-0.035269	0.005471
2019-07-31	[[ENPH, NVR, AZO, CMG, MKTX, PAYC, MTD, NLOK, ...	[[ABMD, BIIB, KHC, SIVB, PVH, HAL, ATVI, VTRS,...	-0.067587	-0.018553	-0.043070	-0.002762
2019-08-31	[[ENPH, NVR, CMG, SEDG, AZO, MKTX, MTCH, TDG, ...	[[BIIB, ABMD, REGN, KHC, WBA, FDX, APA, VTRS, ...	0.068166	-0.013537	0.027314	0.026608
2019-09-30	[[ENPH, NVR, CMG, SEDG, MKTX, PAYC, MTCH, AZO,...	[[ABMD, BIIB, REGN, DXC, ALGN, CNC, KHC, PTC, ...	-0.003684	0.059129	0.027723	0.041314
2019-10-31	[[ENPH, NVR, SEDG, CMG, AZO, MKTX, LRCX, TER, ...	[[ABMD, DXC, KHC, REGN, OXY, VTRS, HAL, BBWI, ...	-0.016798	0.064838	0.024020	0.038338
2019-11-30	[[NVR, ENPH, SEDG, CMG, AMD, LRCX, MKTX, QRVO,...	[[ABMD, VTRS, OXY, ULTA, MOS, DXC, BBWI, CTRA,...	0.092200	0.045609	0.068905	0.038105
2019-12-31	[[ENPH, NVR, SEDG, AMD, LRCX, PAYC, MKTX, CMG,...	[[ABMD, OXY, ANET, ETSY, DXC, VTRS, PTC, CTRA,...	-0.103217	-0.018100	-0.060659	-0.012597
2020-01-31	[[ENPH, NVR, SEDG, TSLA, AMD, TDG, PAYC, DXCM,...	[[ABMD, DXC, ULTA, CTRA, BEN, ANET, SPG, BA, D...	-0.153606	-0.083560	-0.118583	-0.100750
2020-02-29	[[ENPH, TSLA, NVR, SEDG, DXCM, PENN, AMD, GNRC...	[[ABMD, DXC, CTRA, ULTA, BEN, OXY, ETSY, ANET,...	-0.357850	-0.153101	-0.255476	-0.187664
2020-03-31	[[ENPH, TSLA, NVR, SEDG, DXCM, MRNA, LRCX, GNR...	[[ABMD, ULTA, DXC, PARA, DD, MRO, OXY, FANG, P...	0.615914	0.337508	0.476711	0.216599

	Long	Short	Long_return	Short_return	Avg_port_return	etf_return
2020-04-30	[[TSLA, MRNA, ENPH, REGN, DXCM, EQIX, HUM, SED...	[[BKNG, BA, ABMD, ULTA, SPG, UAL, PARA, OXY, C...	-0.023245	0.089298	0.033027	0.061961
2020-05-31	[[MRNA, TSLA, ENPH, DXCM, PENN, REGN, SEDG, NO...	[[BKNG, NVR, BA, UAL, SPG, AAL, OXY, ULTA, DAL...	-0.014349	0.039553	0.012602	-0.006392
2020-06-30	[[TSLA, MRNA, PENN, REGN, DXCM, CZR, ENPH, SED...	[[NVR, BKNG, BA, DXC, CNP, AZO, UHS, CIN, ULT...	-0.000340	0.059754	0.029707	0.045205
2020-07-31	[[TSLA, MRNA, PENN, REGN, ENPH, NFLX, SEDG, DX...	[[BKNG, NVR, BA, SPG, UAL, DAL, TDG, WFC, ESS,...	-0.022631	0.092740	0.035054	0.045537
2020-08-31	[[TSLA, PENN, MRNA, ENPH, ETSY, SEDG, REGN, AM...	[[BKNG, BA, SPG, ESS, WFC, UAL, AVB, OXY, PSX,...	-0.142881	-0.068186	-0.105534	-0.026148
2020-09-30	[[TSLA, PENN, CMG, MRNA, ENPH, ETSY, REGN, SED...	[[BA, BKNG, SPG, WFC, UAL, OXY, ESS, OKE, HII,...	-0.048880	-0.006093	-0.027487	-0.013742
2020-10-31	[[TSLA, PENN, ENPH, SEDG, ETSY, MRNA, CMG, NVR...	[[BA, OXY, HII, SPG, UAL, FANG, MRO, CCL, EOG,...	0.353619	0.111978	0.232798	0.148296
2020-11-30	[[TSLA, PENN, MRNA, ENPH, CMG, SEDG, ETSY, NVR...	[[HII, OXY, BA, AAL, WFC, UAL, FE, OKE, SPG, C...	0.032920	0.046387	0.039653	0.031835
2020-12-31	[[PENN, TSLA, MRNA, CZR, ENPH, ETSY, NVR, CMG,...	[[HII, NOC, GILD, BIIB, O, EVRG, ESS, JKHY, SR...	0.036175	0.005950	0.021062	0.001428
2021-01-31	[[PENN, TSLA, ENPH, NVR, CZR, MRNA, ETSY, SEDG...	[[HII, FE, EVRG, NOC, VRTX, AVB, BXP, PNW, AMT...	0.191057	0.039267	0.115162	0.043246
2021-02-28	[[PENN, TSLA, NVR, CZR, MRNA, ENPH, BKNG, BBWI...	[[PPL, EVRG, ETR, FE, PNW, D, T, ED, SRE, BXP]]	-0.008100	0.013112	0.002506	0.046509
2021-03-31	[[PENN, TSLA, NVR, PARA, CZR, ENPH, WBD, BBWI,...	[[VRTX, JKHY, REGN, CTXS, BIIB, FE, INCY, LMT,...	-0.012617	0.048834	0.018109	0.037457
2021-04-30	[[NVR, TSLA, PENN, BBWI, CZR, MRNA, FCX, BKNG,...	[[VRTX, REGN, INCY, GILD, CLX, FE, MKC, BIIB, ...	0.031295	-0.012182	0.009556	0.018973
2021-05-31	[[NVR, TSLA, FCX, BBWI, CZR, SIVB, BKNG, TPR, ...	[[VRTX, REGN, INCY, MKC, EQIX, CLX, BIIB, PPL,...	-0.006443	0.069578	0.031567	-0.000759
2021-06-30	[[MRNA, NVR, DVN, SIVB, GNRC, FANG, MRO, BBWI,...	[[CLX, REGN, VRTX, CTXS, KMB, MKC, INCY, BDX, ...	-0.103731	0.019311	-0.042210	0.007335
2021-07-31	[[MRNA, NVR, FANG, BBWI, GNRC, SBNY, SIVB, MRO...	[[VRTX, CTXS, T, DLTR, MKTX, MKC, CLX, MAS, CR...	-0.011469	0.039197	0.013864	0.018399
2021-08-31	[[MRNA, NVR, SBNY, NUE, DVN, ALB, FANG, SIVB, ...	[[VRTX, MKTX, T, INCY, XEL, SRE, VZ, CHD, WBD,...	0.174123	-0.059775	0.057174	-0.036186

	Long	Short	Long_return	Short_return	Avg_port_return	etf_return
2021-09-30	[[MRNA, NVR, SBNY, DVN, GNRC, SIVB, AZO, CTRA,...	[[VRTX, CLX, WBD, TTWO, T, CTXS, MKTX, LW, INC...	0.070207	-0.005917	0.032145	0.041605
2021-10-31	[[MRNA, DVN, NVR, FANG, MRO, SBNY, AZO, SIVB, ...	[[MKTX, WBD, VRTX, CLX, LVS, VTRS, GPN, PENN, ...	-0.026217	-0.063410	-0.044814	-0.032925
2021-11-30	[[DVN, NVR, MRNA, NVDA, AZO, EPAM, SBNY, TSLA,...	[[MKTX, WBD, PARA, LW, PENN, CTXS, LVS, INCY, ...	0.048940	0.027274	0.038107	0.053320
2021-12-31	[[NVR, AZO, NVDA, MRNA, EPAM, DVN, FTNT, INTU,...	[[WBD, PARA, PENN, GPN, LVS, MKTX, CTXS, ATVI,...	0.122599	-0.120252	0.001174	-0.043433
2022-01-31	[[DVN, AZO, NVR, TSLA, NVDA, F, MRO, APA, FANG...	[[PENN, MKTX, PYPL, TWTR, TFX, WYNN, ZBH, WBD,...	0.091075	-0.042990	0.024043	0.007472
2022-02-28	[[DVN, AZO, NVR, MRO, TSLA, NVDA, COP, NUE, FA...	[[PENN, PYPL, BIIB, MKTX, TWTR, GPN, ZBH, TFX,...	0.105658	0.040972	0.073315	0.026789
2022-03-31	[[AZO, DVN, OXY, TSLA, CF, MOS, APA, MRO, NVDA...	[[PYPL, BIIB, MKTX, PENN, META, CHTR, ILMN, GP...	0.004815	-0.192919	-0.094052	-0.063500
2022-04-30	[[AZO, TSLA, DVN, MOS, CF, APA, OXY, MRO, NUE,...	[[PYPL, BIIB, CHTR, ALGN, NFLX, EPAM, PENN, MR...	0.138404	-0.071510	0.033447	0.007874
2022-05-31	[[DVN, OXY, AZO, APA, MRO, CF, CTRA, VLO, PXD,...	[[PYPL, NFLX, ALGN, NVR, MRNA, MKTX, ADBE, BIL...	-0.164893	-0.146167	-0.155530	-0.083279
2022-06-30	[[DVN, OXY, APA, MRO, AZO, CTRA, FANG, VLO, EO...	[[NVR, ALGN, NFLX, CMG, PYPL, MRNA, IDXX, CHTR...	0.095879	0.160629	0.128254	0.088149
2022-07-31	[[OXY, DVN, MRO, APA, DLTR, CTRA, AZO, CF, VLO...	[[NFLX, PYPL, META, NVR, ALGN, AMZN, CCL, CMG,...	0.032443	0.188875	0.110659	0.079685

In [952...

```
df.iloc[:,2:-1]
```

Out [952...

	Long_return	Short_return	Avg_port_return
2017-08-31	0.133873	-0.000967	0.066453
2017-09-30	0.015469	0.098115	0.056792
2017-10-31	0.046078	0.049232	0.047655
2017-11-30	0.056388	-0.027735	0.014327
2017-12-31	0.011358	0.116099	0.063729

	Long_return	Short_return	Avg_port_return
2018-01-31	-0.131605	-0.020929	-0.076267
2018-02-28	0.005082	-0.001710	0.001686
2018-03-31	0.105089	0.087646	0.096367
2018-04-30	0.039903	0.097120	0.068512
2018-05-31	0.033491	0.000228	0.016859
2018-06-30	0.035455	-0.010686	0.012385
2018-07-31	0.006606	0.048592	0.027599
2018-08-31	0.015622	-0.015101	0.000260
2018-09-30	-0.122359	-0.124399	-0.123379
2018-10-31	-0.044272	0.004583	-0.019844
2018-11-30	-0.093128	-0.082913	-0.088020
2018-12-31	0.090616	0.138236	0.114426
2019-01-31	0.025714	0.065181	0.045448
2019-02-28	0.026591	0.029974	0.028282
2019-03-31	-0.000285	0.063799	0.031757
2019-04-30	-0.101007	-0.054617	-0.077812
2019-05-31	0.086083	0.058952	0.072518
2019-06-30	-0.014658	-0.055880	-0.035269
2019-07-31	-0.067587	-0.018553	-0.043070
2019-08-31	0.068166	-0.013537	0.027314
2019-09-30	-0.003684	0.059129	0.027723
2019-10-31	-0.016798	0.064838	0.024020
2019-11-30	0.092200	0.045609	0.068905
2019-12-31	-0.103217	-0.018100	-0.060659
2020-01-31	-0.153606	-0.083560	-0.118583
2020-02-29	-0.357850	-0.153101	-0.255476

	Long_return	Short_return	Avg_port_return
2020-03-31	0.615914	0.337508	0.476711
2020-04-30	-0.023245	0.089298	0.033027
2020-05-31	-0.014349	0.039553	0.012602
2020-06-30	-0.000340	0.059754	0.029707
2020-07-31	-0.022631	0.092740	0.035054
2020-08-31	-0.142881	-0.068186	-0.105534
2020-09-30	-0.048880	-0.006093	-0.027487
2020-10-31	0.353619	0.111978	0.232798
2020-11-30	0.032920	0.046387	0.039653
2020-12-31	0.036175	0.005950	0.021062
2021-01-31	0.191057	0.039267	0.115162
2021-02-28	-0.008100	0.013112	0.002506
2021-03-31	-0.012617	0.048834	0.018109
2021-04-30	0.031295	-0.012182	0.009556
2021-05-31	-0.006443	0.069578	0.031567
2021-06-30	-0.103731	0.019311	-0.042210
2021-07-31	-0.011469	0.039197	0.013864
2021-08-31	0.174123	-0.059775	0.057174
2021-09-30	0.070207	-0.005917	0.032145
2021-10-31	-0.026217	-0.063410	-0.044814
2021-11-30	0.048940	0.027274	0.038107
2021-12-31	0.122599	-0.120252	0.001174
2022-01-31	0.091075	-0.042990	0.024043
2022-02-28	0.105658	0.040972	0.073315
2022-03-31	0.004815	-0.192919	-0.094052
2022-04-30	0.138404	-0.071510	0.033447

	Long_return	Short_return	Avg_port_return
2022-05-31	-0.164893	-0.146167	-0.155530
2022-06-30	0.095879	0.160629	0.128254
2022-07-31	0.032443	0.188875	0.110659

Code or Reference Citation

I want acknowledge the cited works from professor John Droescher

<https://docs.python.org/3/>

Python for Finance: Mastering Data-Driven Finance, Author: Hilpisch, Yves, Publisher: O'Reilly Media, Incorporated, Edition: 2, Year Published: 2018

<https://corporatefinanceinstitute.com/resources/knowledge/trading-investing/volume-price-trend-indicator-vpt/>

In []: