

```
In [1]: # Check path during tests
import sys
import os
module_path = os.path.abspath(os.path.join('.', ''))
if module_path not in sys.path:
    sys.path.append(module_path+"/")

# Plib imports
import Plib.PCA as pcl
import Plib.DataFarm.MStack as datafarm1

# Other libraries
import warnings
#warnings.filterwarnings("ignore")
import numpy as np
import pandas as pd
from dateutil import parser
from time import gmtime, strftime
from datetime import timedelta, date

%matplotlib inline
```

```
In [2]: df1=pcl.getExchangesByCountry(datafarm1.getExchanges(),'USA')
df1.head(20)
```

```
Out[2]:
```

	name	mic	country	city	currency.name	currency.code	timezone.timezone
0	NASDAQ Stock Exchange	XNAS	USA	New York	US Dollar	USD	America/New_York
1	New York Stock Exchange	XNYS	USA	New York	US Dollar	USD	America/New_York
2	NYSE ARCA	ARCX	USA	New York	US Dollar	USD	America/New_York
3	OTC Markets	OTCM	USA	New York	US Dollar	USD	America/New_York
59	American Stock Exchange	XASE	USA	New York	US Dollar	USD	America/New_York
60	Cboe BZX U.S. Equities Exchang	XCBO	USA	Chicago	US Dollar	USD	America/Chicago
61	US Mutual Funds	NMFQS	USA	New York	US Dollar	USD	America/New_York
62	OTC Bulletin Board	OTTC	USA	Washington	US Dollar	USD	America/New_York
63	OTC Grey Market	PSGM	USA	New York	US Dollar	USD	America/New_York
64	OTCQB Marketplace	OTCB	USA	New York	US Dollar	USD	America/New_York
65	OTCQX Marketplace	OTCQ	USA	New York	US Dollar	USD	America/New_York
66	OTC PINK current	PINC	USA	New York	US Dollar	USD	America/New_York
67	Investors Exchange	IEXG	USA	New York	US Dollar	USD	America/New_York

```
In [28]: tdf,SP_tickers=pcl.getTickers(edf.iloc[60]['mic'])
tdf.head(10)
```

Total stocks: 391

```
Out[28]:
```

	name	symbol	has_intraday	has_eod
0	ALPS Fund Services ALPS Clean Energy ETF	ACES	False	True
1	ETF Series Solutions Aptus Collared Income Opp...	ACIO	False	True
2	AMERICAN CUSTOMER SATISFACTION ETF	ACSI	False	True
3	ISHARES EDGE MSCI MIN VOL GLOBAL ETF	ACWV	False	True
4	Aptus Behavioral Momentum	ADME	False	True
5	Two Roads Shared Trust Anfield U.S. Equity Sec...	AESR	False	True
6	Two Roads Shared Trust Anfield Universal Fixed...	AFIF	False	True
7	ISHARES MSCI ARGENTINA AND GLOBAL EXPOSURE ETF	AGT	False	True
8	ALPHACLONE ALTERNATIVE ALPHA ETF	ALFA	False	True
9	PROSHARES MORNINGSTAR ALTERNATIVES SOLUTION ETF	ALTS	False	True

```
In [29]: #Set the dates interval for downloading data
today_date=strftime("%Y-%m-%d %H:%M:%S", gmtime())
dt_start = (parser.parse(today_date) + timedelta(days=-120)).strftime("%Y-%m-%d")
dt_end = (parser.parse(today_date) + timedelta(days=-3)).strftime("%Y-%m-%d")
dt_train=(parser.parse(today_date) + timedelta(days=-40)).strftime("%Y-%m-%d")
index_ticker='SPY'
```

```
In [30]: asset_prices = pd.read_pickle('PCA_data')
asset_prices['Date'] = pd.to_datetime(asset_prices.index)
asset_prices.set_index('Date',inplace=True)
#asset_prices=pcl.getAssetPrices(SP_tickers,index_ticker,dt_start, dt_end)
#asset_prices.to_pickle('PCA_data')
n_stocks_show = 12
print('Asset prices shape', asset_prices.shape)
asset_prices.iloc[:, :n_stocks_show].head()
```

Asset prices shape (84, 307)

```
Out[30]:
```

	ACIO	ACSI	ACWV	ADME	AESR	AFIF	AGT	ALFA	ALTS	ARCM	BAPR	BAUG
Date												
2020-07-06	25.44	34.39	89.13	32.78	10.20	9.82	21.94	58.52	34.90	100.40	27.93	26.62
2020-07-07	25.35	33.98	88.58	32.59	10.14	9.84	21.83	57.99	34.76	100.41	27.84	26.38
2020-07-08	25.43	34.24	88.97	32.87	10.23	9.83	22.05	58.72	34.76	100.42	27.86	26.56
2020-07-09	25.29	33.86	88.40	32.91	10.19	9.85	21.76	58.69	34.67	100.42	27.69	26.53
2020-07-10	25.40	34.48	88.78	33.01	10.29	9.82	22.09	58.94	34.74	100.41	27.94	26.73