bootstrapindex

index
/home/jirong/Desktop/github/bootstrap-index/bootstrapindex/bootstrapindex.py

```
Created on Tue Dec 15 23:57:07 2020

@author: jirong
```

Modules

io pandas requests
numpy random

Classes

builtins.object

bootstrapindex

class **bootstrapindex**(builtins.object)

```
bootstrapindex(data, window, num_samples_per_period, min_sample_size, prop_block_bootstrap, days_block, starting_index=None)
```

Methods defined here:

__init__(self, data, window, num_samples_per_period, min_sample_size, prop_block_bootstrap, days_block, starting_index=None)

```
Constructor for bootstrap_index class

Attributes:
    data (pandas data frame or series) Data-frame
    window (string) expanding or sliding
    num_samples_per_period (int) Number of blocks of samples to be extracted
    min_sample_size (int) To define minimum data points to be extracted for each sample
    prop_block_bootstrap (float) number of trials
    days_block (int) Used as parameter in expanding or sliding window block.
    starting_index (int) Starting index to create window of training and testing indexes
    expanding_windows_w_bootstrap_info (dict) Dictionary of indexes used for boostrapping
```

create_dictionary_window_n_bootstrap_index(self)

```
Method for creating dictionary of window and block bootstrap indexes.

Returns:
    dict: Dictionary of in_sample index, out_sample index, bootstrap_index extracted from in_sample_index range

Examples
--------
    url="https://github.com/jironghuang/trend_following/raw/main/quantopian_data/futures_incl_2016.csv"
    s=requests.get(url).content
    data=pd.read_csv(io.StringIO(s.decode('utf-8')))
    data['Date'] = pd.to_datetime(data['Date'], format='%Y-%m-%d')
    data.set_index('Date', inplace=True)

    bootstrap = bootstrapindex(data, window='sliding',
                               num_samples_per_period=10,
                               min_sample_size=60,
                               prop_block_bootstrap=0.25,
                               days_block=252,
                               starting_index = 5
                               )
    bootstrap.create_dictionary_window_n_bootstrap_index()
    bootstrap.expanding_windows_w_bootstrap_info
    {1: {'in_sample_index': [5, 256],
      'out_sample_index': [257, 508],
      'bootstrap_index': {'start_index': array([103,  39,  19,  65,  65, 164, 151,  87,  63, 123]),
       'end_index': array([166, 102,  82, 128, 128, 227, 214, 150, 126, 186])}},
     2: {'in_sample_index': [257, 508],
      'out_sample_index': [509, 760],
      'bootstrap_index': {'start_index': array([355, 291, 271, 317, 317, 416, 403, 339, 315, 375]),
       'end_index': array([418, 354, 334, 380, 380, 479, 466, 402, 378, 438])}},
     3: {'in_sample_index': [509, 760],
      'out_sample_index': [761, 1012],
      'bootstrap_index': {'start_index': array([607, 543, 523, 569, 569, 668, 655, 591, 567, 627]),
       'end_index': array([670, 606, 586, 632, 632, 731, 718, 654, 630, 690])}},
     4: {'in_sample_index': [761, 1012],
      'out_sample_index': [1013, 1264],
      'bootstrap_index': {'start_index': array([859, 795, 775, 821, 821, 920, 907, 843, 819, 879]),
...
```

create_window_index(self, days_block=None)

```
Method for creating window index

        Args:
    days_block: testing block size which is also used to create multiple of training block size

Returns:
    list: list of training and testing indexes

Examples
--------
url="https://github.com/jironghuang/trend_following/raw/main/quantopian_data/futures_incl_2016.csv"
s=requests.get(url).content
data=pd.read_csv(io.StringIO(s.decode('utf-8')))
data['Date'] = pd.to_datetime(data['Date'], format='%Y-%m-%d')
data.set_index('Date', inplace=True)
```

```
    bootstrap = bootstrapindex(data, window='sliding',
                               num_samples_per_period=10,
                               min_sample_size=300,
                               prop_block_bootstrap=0.25,
                               days_block=252,
                               starting_index = 5
                               )
    bootstrap = bootstrap_index(data)
    bootstrap.create_window_index()
    Out[93]:
    [[[5, 256], [257, 508]],
     [[257, 508], [509, 760]],
     [[509, 760], [761, 1012]],
    ...
```

extract_block_bootstrap_periods(self, sample_size, start_sample_index=0, end_sample_index=None)

```
    Function for selecting period

    Args:
        start_sample_index: Start of sample index
        end_sample_index: End of sample index

    Returns:
        dictionary of start and end indexes

    Examples
    --------
    url="https://github.com/jironghuang/trend_following/raw/main/quantopian_data/futures_incl_2016.csv"
    s=requests.get(url).content
    data=pd.read_csv(io.StringIO(s.decode('utf-8')))
    data['Date'] = pd.to_datetime(data['Date'], format='%Y-%m-%d')
    data.set_index('Date', inplace=True)
    bootstrap = bootstrapindex(data, window='sliding',
                               num_samples_per_period=10,
                               min_sample_size=300,
                               prop_block_bootstrap=0.25,
                               days_block=252,
                               starting_index = 5
                               )
    bootstrap.extract_block_bootstrap_periods(sample_size = 100, start_sample_index = 50, end_sample_index = 500)
    Out[143]:
    {'start_index': array([247, 118,  78, 171, 170, 368, 343, 215, 166, 287]),
     'end_index': array([372, 243, 203, 296, 295, 493, 468, 340, 291, 412])}
```

Data descriptors defined here:

__dict__

```
    dictionary for instance variables (if defined)
```

__weakref__

```
    list of weak references to the object (if defined)
```