FBA | QUANTITATIVE FINANCE RESEARCH GROUP

Financial AI

Homework 2

Due at 06:00 pm (Korea Standard Time) on Saturday

Problem 1. KRX Database Data

You should create a simple python package to get the data from http://data.krx.co.kr/ easily. The package should be named as krx and have a class named 'WebSource'. Then we can import it as 'from krx import WebSource'.

The complete library should have followings.

(a) A method to set frequency of multiple requests. (If you don't have some sleep between requests, the host will block your ip address.)

(b) A method to get the index constituents data. The method should have parameters; 'index _name', 'start', 'end'. (Getting all indices is quite annoying, so it is okay to implement only 'KOSPI', 'KOSPI 200', 'KOSDAQ', and 'KOSDAQ 150'. Choose the best data structure to return.

(c) A method to get the historical daily OHLCV data. The method should have parameters; 'stock_code'(A000000), 'start', 'end'. Please return pandas.DataFrame.

(d) A method to get the OHLCV data for a given universe. The method should have parameters; 'index_name', 'start', 'end'. (This method returns OHLCV data of all stocks listed in a given index_name during the period.). Choose the best data structure to return.

(e) (Additional) IP blocking is so annoying! How about using proxy servers? Add a method to set proxy servers. If current ip is blocked, it automatically switches to another one.

Problem 2. Binance Public Data

You should create a simple python package to get the data from https://data.binance.vision/ easily. The package should be named as binance and have a class named 'PublicData. Then we can import it as 'from binance import PublicData.

(a) A method to set frequency of multiple requests. (If you don't have some sleep between requests, the host will block your ip address.)

(b) A method to get list of symbols of USDT futures daily kline(OHLCV) data from (https://data.binance.vision/?prefix=data/futures/um/daily/klines/).

(c) A method to get the kline data of USDT futures. The method should have parameters; 'symbol', 'start', 'end'. Please return the result as pandas.DataFrame.

(d) (Additional) It seems we must send lots of requests. Please use threading module in Python to send multiple requests concurrently. Allocating proxy to each thread will be great.