

自动化项目工程

基本介绍

该项目支持多个基金产品净值数据批量导出报告，每只基金会生成一个 word 报告和一个 excel 绘图底稿。

下面是一些批量导出报告的约束：

- 这些批量计算的基金要么都是指增基金，要么都是非指增基金
- 批量计算的基金对标的指数必须一致。
- 指增基金只能指定一个指数数据，于是，如果希望批量导出指增基金，那么它们对标的指数必须一致
- 下面是一些示例：

基金	指数	计算类型	是否允许	解释
图灵1000, 裕锦1000	中证1000	指数增强	允许	指增计算，且对标的指数唯一且一致
图灵500, 裕锦1000	中证1000	指数增强	允许	允许计算，但是图灵500的超额会按照中证1000来计算，会导致错误
图灵500, 裕锦1000	中证500, 中证1000	指数增强	不允许	指增计算，但是对标的指数不唯一且不一致
图灵500, 裕锦1000	中证500, 中证1000	指数增强	不允许	指增计算，但是对标的指数不唯一且不一致
图灵1000, 裕锦1000	中证1000, 上证指数	指数增强	不允许	指增计算只能对标一个指数，就是指增跟踪的指数，不能再指定其它任何指数
九章幻方, 明沅1号	中证300, 上证指数, 沪深300	非指数增强	允许	非指增计算，可以对标多个指数，并且可以将这些指数全部绘制在净值走势图里，也就是，九章幻方的净值走势图会包括这三个指数，明沅1号的净值走势图也会包括这三个指数
图灵1000, 裕锦1000	中证1000, 上证指数	非指数增强	允许	可以将指数增强基金当作非指增基金来计算，此时可以指定多个指数，但是不会计算任何超额相关数据

如何使用该项目代码？

第一步：准备数据

准备两个 xlsx 文件放入合适位置：1.某只基金的净值数据 2. 指数数据。该项目通过数据表列名指定基金和指数。

基金净值数据说明：

- 基金净值列的列名需要和基金名称相同，净值开始值可以不是1.00；
- 最左侧必须是日期列，日期格式支持 2017-01-20 或者 2017/01/20 这两种，其它格式会直接报错，第二列及后面的列必须是净值列。
- 不要包含其它任何多余列。也就是，净值数据表的格式应该是：首列是日期，第二列 --> 第N列 是不同基金的净值数据，并且列名必须是基金名称。

下方是一个可以计算三个基金的净值数据表示例。它可以同时计算三只基金(九章幻方，降本1号，稳中向好2号)的数据并导出基金报告。注意到以下几个特点：

- 稳中向好2号的净值开始值不是1.00，稳中向好2号的净值有空值，这都不会报错。因为代码会将净值数据自动归一化，并且会使用线性插值 填充空值。
- 允许基金净值数据前若干行是空值，也允许包含纯空行，比如下面的2017-01-14。下表的每个基金计算区间分别是：九章幻方(2017-01-20 --> 2017-03-17)，降本1号(2017-02-03 --> 2017-03-17)，稳中向好2号(2017-02-17 --> 2017-03-17)

	九章幻方	降本1号	稳中向好2号
2017-01-14			
2017-01-20	1.00		
2017-01-26	1.00		
2017-02-03	1.00	1.00	
2017-02-10	1.00	1.07	
2017-02-17	1.00	1.08	1.08
2017-02-24	1.00	1.09	
2017-03-03	1.00	1.10	1.07
2017-03-10	1.01	1.11	1.06
2017-03-17	1.02	1.02	1.05

指数数据文件说明：

- 最左侧必须是日期列，日期格式支持 2017-01-20 或者 2017/01/20 这两种，其它格式会直接报错。
- 第二列及后面的列必须是每个指数的走势，这个走势，可以是 收盘价、净值，无需对指数数据进行标准化或者归一化。
- 日期列可以和“净值数据表”的日期列一致，也可以不一致，例如：如果净值数据表的第一个日期是2017-01-20，指数数据表的开始日期可以是 2016-01-20，但是需要保证，“指数数据表”的日期必须包含“净值数据表”的所有日期。
- 指数走势列的列名必须是指数名称，不要写其它名称。
- 不要包含多余的，没有用处的列。
- 于非指增计算，第2至N列可以包含若干指数数据，这些指数数据都会被绘制在每个几几年的净值走势图中；但是对于指增基金，只能包含一列指数数据，即这些指增基金跟踪的指数数据(因此要求指增基金跟踪的指数必须一致)。

下面是一个合理的指数数据文件(注意：没有和本文件的“净值数据表”的日期对应)，指数数值如果不想填写净值，那么填写收盘价即可。

	中证1000	中证500
2022-09-02	6776.8268	12345
2022-09-09	6913.579	67891
2022-09-16	6481.2442	51515
2022-09-23	6363.6835	17194
2022-09-30	6124.8639	12548
2022-10-14	6406.4573	12846
2022-10-21	6432.3255	18460
2022-10-28	6239.9316	89131
2022-11-04	6709.6446	19841

第二步：设置参数

- 打开 `main.py` 文件，你会看到一个 `def multi_fund_report_interface():` 函数，你需要在这个函数内部设置参数
- 分别设置第一步设定的两个 `xlsx` 文件数据的路径。下面是一个示例：

```
netval_path = "data/中证1000基金-净值数据.xlsx" # 包含两个基金：裕锦中证1000指数增强，图灵进取中证1000指数增强
index_path = "data/裕锦中证1000指数增强-指数数据.xlsx" # 包含一个指数：中证1000指数
```

- 指定是否是指增基金，如下面的代码所示。注意：指增基金如果该参数指定为 `False` 则当成普通基金计算，不会计算超额。

```
enhanced_fund = True # 需要计算超额就指定为 True，不需要就 False
```

- 为每只基金指定开始计算的日期，按照净值数据表的基金净值列顺序从前往后按顺序指定起始计算日期；可以是空。也就是 `start_dates = []`。假设 `start_dates` 包含的元素叫做 `start_dates`，表示每只基金的开始计算日期。
`start_date` 以本文件提到的“降本1号”净值数据进行说明：①如果不指定开始日期，那么会默认从第一个有净值数据的日期开始计算，也就是 2017-02-03；②如果指定了开始日期，比如 2017-02-10，那么会从 2017-02-10 基金净值 1.07 开始计算，标准化净值数据时，1.07 会被标准化为 1.00。③如果指定了开始日期，比如 2017-1-20，由于该开始日期对应的降本1号净值数据是空，那么依然会默认从第一个有净值数据的日期开始计算，也就是 2017-02-03。

```
start_dates = [None, date(2023, 5, 26)] # 如果有两只基金，它代表第一个不指定开始日期，第二个指定开始日期
# 如果有三只基金，它代表第一个不指定开始日期，第二个指定开始日期，第三个不指定开始日期
start_dates = [date(2022, 8, 19), date(2022, 9, 19)] # 如果有三只基金，仅指定前两个基金的日期，可以这么写。
start_dates = [date(2022, 8, 19), date(2022, 9, 19), None] # 如果有三只基金，仅指定前两个基金的日期，还可以这么写。
start_dates = [date(2022, 8, 19), None, date(2022, 9, 19)] # 如果有三只基金，仅指定第一、第三个基金的开始计算日期，可以这么写。
```

- 为每只基金指定公司名称，用法与上面的`start_dates`类似，如果某只基金没有指定公司名称，将采用默认名称“私募管理人”。该参数设定示例如下面代码所示：

```
corp_names: list = ["裕锦私募", "图灵量化"]
```

下面是一个完整的参数设定示例(详见[main.py](#))，前文没有提到的参数建议不要修改。

```
from report_generate import *

def multi_fund_report_interface():
    """ 在基金对的标指数数据一致的情况下，允许同时导出多个基金的基金报告。 """
    netval_path = "data/中证1000基金-净值数据.xlsx"
    index_path = "data/裕锦中证1000指数增强-指数数据.xlsx"
    enhanced_fund = True
    start_dates = [] # 如果需要指定日期，可能需要导入包 from datetime import date 然后使用 date(2018, 5, 12)
    这种形式指定日期
    corp_names = ["裕锦私募", "图灵量化"]
    add_indicators_tables: bool = True # 添加 “关键指标汇总”，“滚动收益率分布”，“收益概率统计” 这三张表
    multi_fund_report(netval_path, index_path, enhanced_fund, corp_names = corp_names,
                      start_dates = start_dates, add_indicators_tables = add_indicators_tables)

def main():
    single_fund_report_interface()

if __name__ == "__main__":
    main()
```

第三步：启动代码

使用合适的IDE运行main.py即可，导出的word/excel文件会被写入当前目录的output。

可能遇到的问题

- **Python环境问题。** 本工程Python版本是3.8.8，所有包的版本在这个文件中[requirements.txt](#)。你需要设置合理的Python运行环境，如果提示没有安装某个包需要您自行安装。下面是一些主要包的版本。

```
numpy==1.24.4
pandas==2.0.3
openpyxl==3.1.2
pywin32==306
ipykernel==6.28.0
tqdm==4.66.1
matplotlib==3.7.4
```

- **Word/Excel/Windows 版本问题。** 由于电脑上word/excel版本不一致，可能出现代码出现错误或者图表显示格式不及预期的情况。由于word/excel版本或者winows系统环境导致的问题本人爱莫能助。因为我不并不知道您使用设备的运行环境，我只能告知我的运行环境是WINDOWS11 22H2(23H2)，Office 版本 Microsoft® Excel® 2019MSO (版本 2312 Build 16.0.17126.20132) 64 位。此时需要您自己查看报错的位置，并修改代码，当然，这可能有些困难。
- **日期设置。** Python 中有多种日期格式，为了保证统一处理，在代码中手动输入的日期格式必须是 datetime.date 格式，下面是一个示例。不要输入其它格式的日期，比如 datetime.datetime 格式是不允许的。

```
from datetime import date
start_date: date = date(2023, 8, 19) # 注意：日期需要这样按照 年-月-日 来创建
```

其它功能

程序是十分灵活的，在编写时预设了很多可对外调用的函数和方法。如果不希望导出WORD，只希望根据净值数据计算一些指标，比如夏普、年化等，可以参考 [interactive_code.ipynb](#)。

更新情况

- 2024-01-26 添加新功能，除了导出基金报告的WORD外，还可以导出关键指标和滚动收益分析的WORD。
- 2024-01-26 支持有限度的基金报告批量导出。
- 2024-01-29 优化 [main.py](#)，大幅简化参数配置，细化了 [README.md](#) 文档。