Fama's Friends

Big Data Analysis Project

Robert LYU, Evan HU, Trista FANG, Veronica PENG, Antoine SCHULER

2020 / 1 / 13

1. Introduction

Considering that the investment opportunity in the financial market is written in the water and all the market information posses 3V features, our group believe that quantitative investment is the most common and effective use of the big data.

Traditional multi-factor model, which is mainly based on value investing, has been widely used in the A-share quantitative investment field, and the model has also obtained relatively stable excess returns in the past few years. Since 2017, however, dramatic market change impacted the stability of the strategy using traditional multi-factor model. According to a report of Guotai Junan Securities, alpha factors based on trading behavior, instead of companies' fundamentals like market value, have greater estimation ability and stronger return stability. The reason is that trading behavior has almost a decisive effect on the stock price in the short term and we can take full use of the advantages of quantitative investment to do arbitrage.

Therefore, our group decided to construct a multi-factor model using industry factors, style factors and 31 alpha factors, which is chosen from the Guotai Junan Securities report 'Multi-factor stock selection system based on short period price and volume characteristics'. Then, we will estimate the future return in the next day and establish our own portfolio by longing the 10% or 20% stocks with highest estimated return and shorting the bottom 10% or 20% stocks.

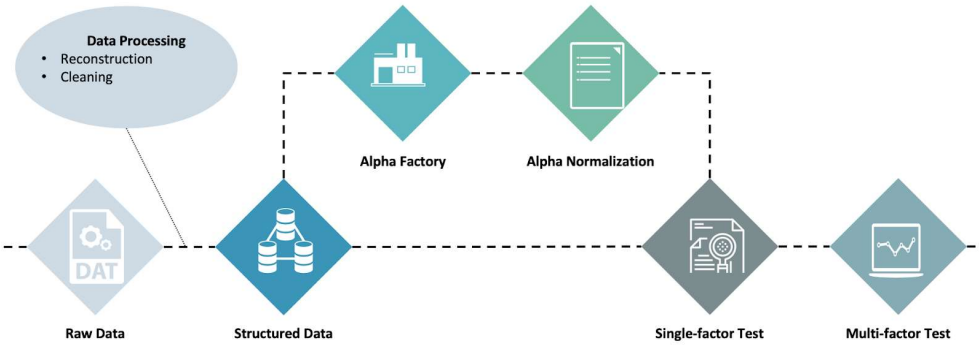
2. Project Workflow

The workflow of our strategy is as below. There are basically 5 steps: Data Processing, Alpha Calculation, Alpha normalization, Single-factor Test and Multi-factor test. The details of these five steps will be introduced in the following part. In this part, we will first introduce some facts about our raw data.

As for our raw data, we adopt the data from Wind containing the names, tickers, ohlc prices (daily open, high, low, close prices), total market capital, volumes, amounts, P/E ratios (trailing twelve month), whether they could be traded, whether they were in special treatment (ST), which industry they belong to, etc. of **3842 stocks** in A share for **2166** trading days **from 4 January, 2011 to 28 November, 2019**.

Specifically, Prices, total market capital and amounts are in US dollars. Whether they could be traded and whether they were in special treatment (ST) are Boolean values, with 1 as can be traded or in ST and 0 as the opposite. The industry factor exposure is Boolean which equals 1 (in

this industry) or 0 (not in this industry) for each stock. There are **34 industries** according to the Level 1 classification of SHENWAN HONGYUAN Securities. The style factors are divided into **9 categories**. They are β, momentum, size, earnings yield, volatility, growth, value, leverage and liquidity respectively. All of the information mentioned is stored in the Matlab as a struct.



2.1 Data Cleaning

In this module, two modes are provided, i.e. incremental mode(add 1 new observation) and historic mode(recalculate all available data). Then the data cleaning module will basically run with the following few steps:

Stock Screen→Data Integrity →Fill Data Plug→ Simplify Result

In stock screen step, we select stocks according to ST table and tradeable table(also, can add more tables if needed), where 3 rules are provided, allowing us to select stocks according to cumulative non-actionable days(here, ST or non-tradeable) over a given time interval, consecutive non-actionable days over a given time interval, not tolerating non-actionable days over a given time interval.

Then, the module will examine the data integrity, and fill unexpected missing data with 'previous' method(by default). In the end, the module will summarize the execution procedure and print the output. Also, some utils are provided for users' convenience.

2.2 Alpha Calculation

This is a module calculating all the factor exposures in batches. Specifically, this module will call the corresponding function to calculate alpha factors respectively using the config file and cleaned data imported. After that, the alpha factors will be combined into a cube called "alphaLoadings". Lastly, the data calculated will be saved.

To make our module easy to expand, all the definition of how to calculate each alphas are not defined in the module. The functions that doing the calculation are defined in a certain format so that our module can call them automatically. The module will call all the functions that has the function names registered in the config file one by one. As a result, we can simply add any alpha calculations by just appending it to the config.

We choose to make the framework suitable for all circumstances, so it should be able to process the two situations - receiving data of a single date and calculating the factor exposures of that day or receiving data of a certain period of time and calculating the exposures and returns throughout the period.

All the input the factor needs will be stored in the struct alphaPara, which is constructed by another method according to the config file. So that when following procedures call the factor, the function can access the input with simple key-value pairs. It will first load the corresponding information and catch errors if the information is not found. To separate the mode of whether just to calculate one day .There is a specific indicator in alphaPara which is the updateFlag, that will be put into every alphaPara. It is a boolean value that leads the function to either call the sub-function to perform the daily calculation or the calculation of a period like we discussed above. The module implement a convenient way to add alphas and to deal with daily demands, that makes the whole process easier.

2.3 Alpha Normalization

Through this module we want to process the extreme alpha factors and normalize them with the Winsor method to help build the model better. After normalizing the alpha, alpha are comparable in the same magnitude.

Input of this module is the 3-dimensional alpha loadings from the previous phase. The factor loadings can be really big and away from the median, so as shown below, we use Windsor method to compress the values whose absolute values are bigger than a setting value to certain ranges.

$$\widetilde{x_i} = \begin{cases} x_M + n \times D_{MAD}, & if\ x_i > x_M + n \times D_{MAD} \\ x_M - n \times D_{MAD}, & if\ x_i < x_M - n \times D_{MAD} \\ x_i, & else \end{cases}$$

x_i is the loading on stock i of a factor on a single date; is x_M the median of cross-sectional loading over stocks of the factor; D_{MAD} is the median of the sequence $(x_i - x_M)$.

Then, we use z-score to normalize the processed factor loadings. Record the mean, median, skewness and kurtosis of the distribution for further examination. We will also use the normalized factor loadings to do regression with the industry factors to get the residuals and

returns the orthogonalized factor loadings in this module.
Additionally, we compared the cumulative return of alpha factors orthogonal and not orthogonal to style factors and we found that return of alpha factors not orthogonal to style factors are larger. The possible explanation might be the style factor explains most of the stock returns. So we did the orthogonalization using only industry factors.

2.4 Single Factor Testing

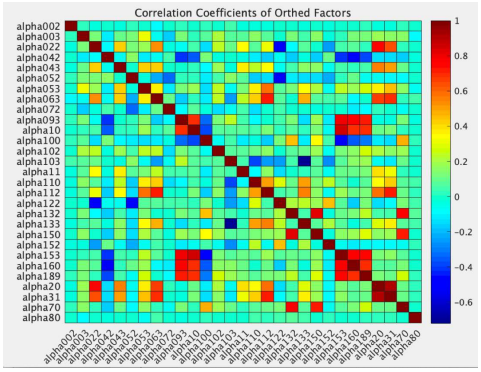
This module uses the normalized factor exposures to run regression to acquire each factor return and test whether they are significantly effective and stable.
The first step of this module is run a regression on a single alpha factor, which has been calculated and normalized in the former modules. The traditional Barra model included the style factors, but we omit this term considering the problem mentioned when discussing the last module. So the regression we used is as follow:

$$r_k^{(t+1)} = f_{industry}^{(t)} X_{industry}^{(t)} + f_{factorReturn}^{(t)} \epsilon_k^{(t)}, k = 1, 2, 3, .., N$$

In the above regression, the $X_{industry}^{(t)}, \epsilon_k^t$ are the respectively industry factors and alpha factors in time t. $f_{industry}^{(t)}, f_{factorReturn}^{(t)}$ are respectively their exposures in time t. For a given period of time, we can get time series of $f_{factorReturn}^{(t)}$. We believe the factor must provide explanatory power to portfolio returns and have delivered a persistent, robust, investable, intuitive and pervasive premium. To test the validity of factors, each alpha factor needs to pass a series of statistical tests, including the significance and stationarity of t-statistics of the regression, $f_{factorReturn}$ and IC(information coefficient).
Specifically, the significance of t-statistics is calculated using the hypothesis $H_0 : mean(|t_{f_k}| = 0)$ and the stationarity of t-statistics is calculated using $H_0 : |t| > 2$ or ADF test. The significance and stationarity of $f_{factorReturn}$ and IC are calculated in a similar way. At last, we plot the IC of all the factors and plot the compared factor return to see the contribution of a certain factor in this period of time.

2.5. Multi-Factor Modelling

Before enter the multi-factor modelling, we firstly need to eliminate the possibility of collinearity, i.e. executing feature selection. One way to do this is via 'variance threshold' method, as an example, we plotted the correlation heatmap of single factors passing the last test as below:



From the heatmap, we check the alpha factors with darker color, calculated the VIFs of these factors and delete the alpha factors with VIF larger than 5. Also, other feature selection methods are equipped, including Ridge and LASSO model.

Then, we come to the multi-factor modelling part, in this part, we do 4 things:

1. Obviously, different stocks worth different factor structure. The first thing to do is to group those stocks.
2. Decide what structure of factors to be constructed to a given group of stocks;
3. Based on the given factor structure and the given group of stocks, give a universal method to estimate stock return at T+1 time stamp when we are at T time stamp;
4. Provide methods to evaluate the performance of our multi-factor model.

As to the first thing, temporarily, we deploy the 'SW level one' sector classification(34 sectors included); As to the second thing, temporarily, this work is completely depending on feature selection(i.e. the factors are selected automatically from fator warehouse); As to the third thing, once a factor structure and a sector is fixed, we estimate the T+1 time stamp stock return using: stock screen at T time stamp; corresponding factor exposure at T time stamp; corresponding factor return from T-1 to T time stamp. While the T+1 time stamp stock return is simply estimated via linear model of sector base return(intercept) and the linear combination of factor return(scalar) and factor exposure(vector); As to the last thing, 2 major methods are proposed. The first one is IC value(i.e. spearman/pearson correlation coefficient) between estimated stock return and realized stock return; while the second one is long/short portfolio cumulative return, which means we long top x% stocks and short x% stocks(which forms a portfolio with no sector exposure in ideal case) on T time stamp through our estimation and record its realized return on T+1 time stamp.

3. Appendix: A Few Thoughts

Conducting a complete framework for multi-factor models seems to be a basic work. Considering to make it applicable in as many circumstances as possible, however, makes it a systematic engineering to require developers to structure and restructure again and again to meet the demands.

In this project, we take into consideration these aspects:

- Able to read either single day's data streams or data of a period;
- Able to process either single day's data streams or data of a period using vectorized methods;
- Able to read data of different structures or different hierarchy designs because of decomposed input interfaces;
- Able to store either single day's factor values or values for a period;
- Able to add new alpha factors into the warehouse without changing the whole framework because the flexible configuration settings;
- Able to split the framework and apply single part of it because of excellent encapsulation, detailed annotation and demo for application of each module;
- Able for both personalized application and easy basic application due to many of the parameters that can be self-defined and default settings without omission.

Nevertheless, due to limited time, there are still aspects yet to explore, which can be considered as a future perspective of this project. Below are some of the interesting things we can look into.

- Rule of synthesis and reserve: What is the rule to solve two high correlated factors? And why?
- How to create new alpha factors: The BARRA factors are all quantitative factors hard to find economic explanations. So, how to create reasonable factors? And how to explain their effectiveness?
- Back test of grouping layers: Though, unlike Fama-French model, it is unnecessary for BARRA models to implement this, it may still be helpful in determining the effectiveness of alpha factors on different group of stocks. Then, how many groups should we divide the stocks into? How to divide? What should we do to abnormal stocks?
- How to combine with machine learning: There are many modules of this project that can consider implementing machine learning. The grouping in the back test can be connected with machine learning in both supervised and unsupervised classification and clustering. The grouping function has taken into consideration many circumstances, but how to set interfaces for machine learning method is still yet to develop.

4. Member Contributions

Robert LYU: functions that calculate 10 alpha factors, prepare sector factor, the clean data module, the multi-factor modeling module in Matlab, debug, make slides draft.
Evan HU: functions that calculate 10 alpha factors and 2 style factors, the alpha factory module, the multi factor test module in Python, debug,make slides draft.
Trista FANG:Calculate 10 alphas which ending number is 2 and Calculate style Factors : BETA, HSIGMA, RSTR. Single Factor Test: Do the statistical testing for the whole alpha, calculate each factor return, IC series. Plot cumulative factor return and Plot long short portfolio cumulative product of the factor.Make presentation ppt draft slides.
Veronica PNEG: Participate in the discussion, prepare slides, writing the final report.
Antoine SCHULER: Participate in the discussion, prepare slides, combine all drafts to make a final version slides.

5. Acknowledge

We are very grateful to Mr. Zheng for his patient guidance in the whole project. He has taught us a lot of valuable and fancy information about factor investing. In this final project, we know deeply about how to apply this. We will continue to study factor models in the future.
We feel very lucky because we have a group of teammates who are good at thinking and solving problems. In each week, we discuss our project together and share valuable information from multiple perspectives. It is very happy to work with these friends.