USE BANDIT FOR STOCK SELECTION

Weitao CHEN, Lingfang XU

HKUST MAFM 2021 Intake

1. PROBLEM DEFINITION

Bandit algorithm aims to learn the probability distributions of action values of each arm through exploration, and exploit the best arms to expect best rewards. Under the assumption that the probability distributions of stock returns are stable throughout time, it's possible that we use Bandit algorithms to do stock selection. There are mainly two disadvantages using stocks as bandit arms and stock returns as rewards. For one thing, according to many previous researchers, the probability distributions of stock returns not only varies between stocks, but also changes through time. We may fix the step size parameter to capture this characteristic to some extent. For another, there're too many stocks in the market. Even if we narrow the stock pool to CSI 300, there will be hundreds of arms for bandit to choose from which requires a long time span to make the algorithms work. Thus we propose to use self-constructed portfolios and factors as arms. More specifically, due to the lack of complete data, we narrow the stock pool to 190 stocks of CSI 300. For portfolio part, we construct three portfolios and let Bandit choose between them on a daily basis, whose results are established in Section 3. For factor part, Bandit chooses one factor over ten style factors on a daily basis and then constructs a portfolio to get exposure to the factor, for more details, please see Section 4.

To tune parameters, for each parameter set, we run the Bandit for 100 times and calculate the mean and standard deviation of cumulative returns of the models. And we choose the best parameters according to the ratio, mean over standard deviation. Results are too long to be stated here, you can refer to portfoliobandit.py and factorbandit.py to see details.

2. DATA SOURCE

In order to gain authoritative data, we applied for API from multiple quant platforms, such as JoinQaunt, RiceQuant, Wind, Tushare. Most data were download through RiceQuant and been processed into daily data. More information can be found through the official guide from RiceQuant (Ricequant SDK Dividion). As mentioned earlier, we choose 190 stocks from CSI300 as stock pool to keep the completeness of data. With the data which could not be provided daily, we manually broadcast it into corresponding time slot. All the data, code and details could be found on Github.

3. PORTFOLIOS SELECTION

3.1. Portfolios Construction

The portfolio selection problem is a fundamental issue in the financial field. To reveal the performance of portfolio selection of Bandit algorithm, we decide to test on the three basic traditional portfolio as shown below.

Equally-Weighted PortfolioAll stocks are equally weighted during the whole time period.

$W_t = \frac{1}{n}1$

Value-Weighted Portfolio: Stocks positions are re-balanced daily base on the current capital of each stock.

$W_t = \frac{C_(t-1)}{C_(t-1)^T 1}$

Mean-Variance Portfolio: Mean-variance portfolio is a strategy built under the Markovwitz' theory. We would like to provide the portfolio with the highest Sharpe ratio on the efficient frontier.

Fig. 1. The returs of portfolios and CSI 300

As the plot shows, these portfolios style differently and the best one is the Mean-Variance which performs stable and exceeds CSI300. In the following part, we would like to test whether bandit can have an apparent preference among these portfolios.

3.2. Use Bandit to Select Portfolio

With the method mentioned in definition,the best parameters are found as:epsilon = 0.01, step size = 0.4, UCB param = 2. To test whether the Bandit algorithm can choose the same portfolio each time and whether they would performance likelywe run it for ten times and plot its result.

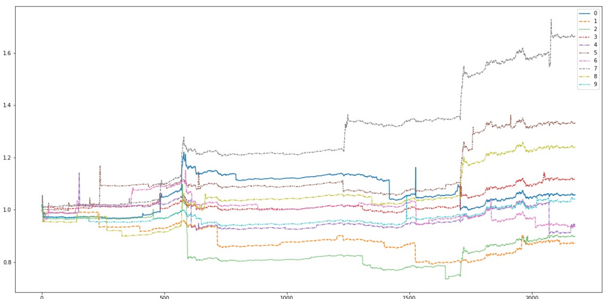

Fig. 2. The returns of ten models of the same parameters

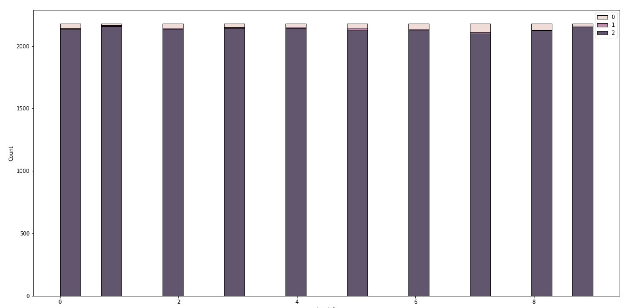

Fig. 3. The actions of ten models of the same parameters

From Fig.3 we can obviously observe that the Portfolio Bandit model select Mean-Variance Portfolio mostly. However, as Fig.2 shows, for some times the Bandit model doesn't choose the same arm for all actions, which caused large difference with the final result of returns.

4. FACTORS SELECTION

4.1. Factors Construction

On a daily basis, we define and construct ten style factors at the first place according to the Barra risk model. We change the construction of some factors due to the lack of data. Below are factor definitions.

More details of factors calculation can be found in generatefactors.py. We now have ten factor values for each stock on each trading day. To construct factor portfolios, for each factor, we long 19 stocks of highest factor values and short 19 stocks of lowest factor values, weights are evenly distributed. In this way, we can obtain 10 portfolios on a daily basis.

Factors	Definition
ALPHA	Unsystematic risk
BETA	Systematic risk to CSI 300
MOMENTUM	Exponential weighted return
SIZE	ln(market capital)
EARNYILD	0.45*EP+0.55*OCF/Market Capital
RESVOL	0.74*VOL+0.16*Spread+0.1*$\sqrt{SSE}$
GROWTH	0.61*ES+0.39*Adjusted ES
BP	Book-to-price
LEVERAGE	0.54*Debt-to-asset+0.46*Book-to-market
LIQUIDTY	Weighted Turnover Rate

Table 1. Factors Definition

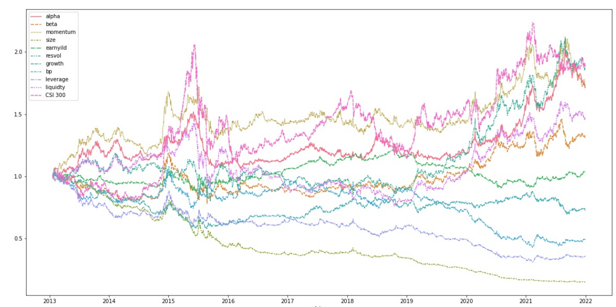

Fig. 4. The returs of factor portfolios and CSI 300

From the plots, we can tell that constructed in this way, these factors weren't informative about these stock returns. We use Bandit in the hope that the algorithm can learn to tell which factors are predictive in different time periods.

4.2. Use Bandit to Select Factors

As the definition part says, we choose the best parameters for the Factors Bandit model. Here we state the best parameters: epsilon = 0.02, step size = 0.3, UCB param = 2. And we used a trick here to improve the results (use 100 times of the portfolio return as the backward rewards). For the best parameters set, we train ten models and the results are represented below. Fig.5 shows the cumulative returns of each model and Fig.6 shows the actual actions of ten models where the label 0 to 10 represents the choice of factors.

From the plot, we can tell that the returns and also the actions of the same model are very inconsistent.

4.3. Potential Improvements

To draw a conclusion, standard Bandit algorithm is of no use to select factors while using simple returns as rewards. However, it does shows a preference for selecting portfolios, which results aren't consistent as well. Maybe using other indicators

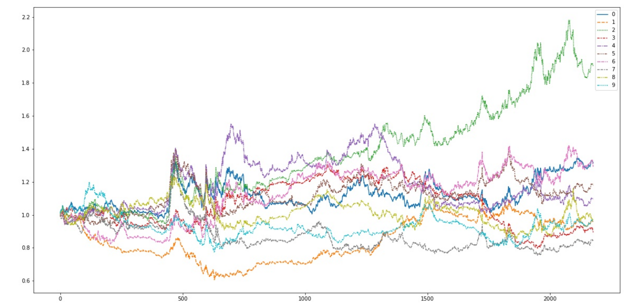

Fig. 5. The returns of ten models of the same parameters

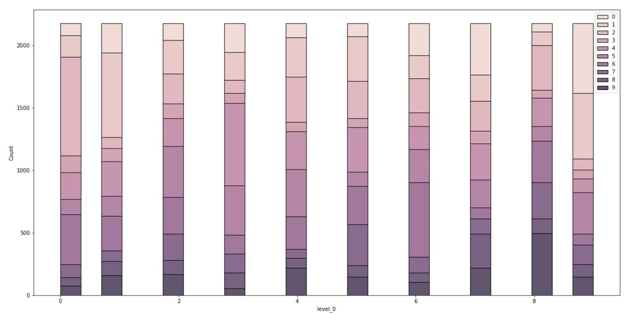

Fig. 6. The actions of ten models of the same parameters

such as Sharpe ratio and Sortino ratio could receive a better results. The construction of factor-based portfolio is another issue. Portfolios that optimize the exposure on each factors could be used to improve the results.

5. GROUP DIVISION

Weitao CHEN: Coding for factor selection part, Section 1, 4 and 5 of report and idea generation.

Lingfang XU: Coding for portfolio selection part, Section2 and 3 of report and data resourcing.

6. CODES

After download the repo, please follow the instructions of README.md. Github address: Github Repo.