Alternative and Quantitative Investment Strategies

Mnacho Echenim, Sonia Jimenez

Grenoble INP-Ensimag

2020-2021

Notes

Quick recap on the Momentum strategy

- Strategy designed to exploit market inefficiency
  - Exploit the fact that not all financial actors have the same information at the same time, and that there is a time lapse before prices incorporate investors' private information
  - Principle: invest in underlying assets with the highest returns, in order to gain from continuing elevated returns
- Related questions
  - Over what period should the returns be computed?
  - After what period of time should the assets be reallocated?

Notes

Goal of the project

- About cryptocurrencies
  - Recent market with a high volatility
  - It seems plausible that the market is not efficient
  - But very few academic results on this topic
- Goal of the project
  - Investigate the efficiency of the cryptocurrency market
  - By applying the Momentum strategy on a set of 50 cryptocurrencies

Notes

Constructing portfolios $P_1, \dots, P_5$

Warning

The algorithms presented below are meant to make sure there is no misunderstanding, they are not necessarily the versions that should actually be implemented

Input: Value i for portfolio P_{i+1} ($i = 0, \dots, 4$)
Input: Market data for 50 cryptocurrencies
Input: List of rebalancing dates $(t_1, \dots, t_n)$, returns horizon h
foreach *rebalancing date t* **do**
  $R_t := (r_t^1, \dots, r_t^{50})$; // *compute returns* $\frac{S_t - S_{t-h}}{S_{t-h}}$
  $C_t := (c_t^1, \dots, c_t^{50})$; // *order cryptocurrencies, decreasing order of returns*
  $\kappa_t := (c_t^{10i+1}, \dots c_t^{10(i+1)})$; // *extract cryptocurrencies for portfolio* $i+1$
end
return $(\kappa_{t_1}, \dots, \kappa_{t_n})$;

Algorithm 1: GetPortfolioCompo

Notes

Constructing portfolios $P_1, \dots, P_5$ (2)

Computation of the values of P_i ($i = 1, \dots, 5$)

```
Input: Initial value V of portfolio, sequence of compositions
Input: Market data for 50 cryptocurrencies, list of rebalancing dates
foreach date t do
    if t is a rebalancing date then
        //nb: this condition should be true for first date of backtest
        UpdateCompo(κ_t, V); self-financing, equally weighted portfolios
    end
    V_t := UpdateValue(t);
end
return (V_{t_1}, ..., V_{t_N});
```

Algorithm 2: BacktestPortfolio

Notes

Outcome of the project

- **Goal:** produce a financial document analyzing the characteristics and performance of the Momentum strategy on cryptocurrencies
- Core version
  - Portfolios are rebalanced monthly, returns are computed over a period of 2 weeks
  - The analyses rely on the statistical indicators described in the upcoming slides
- Next version
  - Carry out same analyses using the CRIX index instead of a standard market index
  - Analyze the effect of changing the rebalancing period and over what horizon returns are computed
- If there is still time
  - Come up with your own strategy and compare it with this one

Notes

Strategy data generation

Portfolio values: generate the following portfolio values

- Portfolios $P_1, \dots, P_5$
- Portfolio P_{1-5}: strategy that consists in buying P_1 and selling P_5

Strategy returns: compute the following daily returns

- Daily return series R_i for portfolio P_i $(i = 1, \dots, 5)$
- Daily return series R_{1-5} for portfolio P_{1-5}
- Daily return difference $R_5 - R_1$
  - Nb: this difference is the one analyzed in the foundational article on the Momentum strategy: Jedadeesh, N., and Titman, S. (1993) *Returns to Buying Winners and Selling Losers : Implications for Stock Market Efficiency*

Get rid of extreme values for each of these series (winsorization)

Notes

Statistical descriptions

Compute the following indicators for each returns series:

- Mean
- Stddev
- Median
- Min and max return
- Skewness
- Kurtosis

Notes

Sharpe, Treynor, Jensen

Given the series R_f of risk-free rates and the series R_M of market returns, compute the following indicators for each returns series R with stddev σ:

- Sharpe ratio: $\frac{R-R_f}{\sigma}$
- Treynor ratio and Jensen's alpha: perform the regression

$$(R - R_f) = \alpha_J + \beta_M(R_M - R_f) + \varepsilon_P$$

  - $\frac{R-R_f}{\beta_M}$ is the Treynor ratio for the series
  - α_J is Jensen's alpha for the series

Notes

Fama & French and Carhart alphas

Given the following series:

- SMB: “Small minus Big”
- HML: “High minus Low”
- UMD: “Up minus Down”

Perform the following regressions

$$\begin{aligned} R - R_f &= \alpha_F + \beta_M(R_M - R_f) + \beta_S \mathrm{SMB} + \beta_H \mathrm{HML} + \varepsilon_F \\ R - R_f &= \alpha_C + \beta_M(R_M - R_f) + \beta_S \mathrm{SMB} + \beta_H \mathrm{HML} + \beta_U \mathrm{UMD} + \varepsilon_F \end{aligned}$$

- α_F is the Fama & French alpha
- α_C is the Carhart alpha

Nb

The P-Value and R-squared indicators must be computed to evaluate the statistical significance level of all the results

Notes

Quick intro to Pandas

- Main library for data analysis in Python
- Built on top of numpy for efficiency
- Main features
  - Data import from many different formats
  - Data cleaning
  - Data manipulation
  - Data visualization
- Main data structures
  - Series: one-dimensional arrays with a fixed size, homogeneous data, accessible via an index
  - **Dataframes**: 2-dimensional array with a mutable size, possibly different data types in each column
- Some operations on dataframes: data inspection, filtering, slicing, arithmetic, statistics...

Notes

Data and code

- What is available
  - Cryptocurrency data (*crypto_prices.csv*, clean data)
  - Series for alpha computations (*F-F_Research_Data_Factors_daily.csv* and *F-F_Momentum_Factor_daily.csv*, data requires some cleaning)
  - Hands-on Jupyter notebook
    - Basic tutorial on using Pandas with focus on financial data analysis
  - Solution to hands-on
- What to produce
  - A Jupyter notebook with analysis and code
  - **Nb:** the final notebook must be easy to read

Notes