Quantitative Equity Investing

MGMT 675: AI-Assisted Financial Analysis

RICE | BUSINESS
Jones Graduate School of Business

Outline

Motivation: Can we profitably trade on quantitative signals?

1. Example dataset
2. Returns of portfolios formed by sorting on characteristics
3. Regressing returns on characteristics at each date
4. Training a model on past data and sorting on its predictions

1. Example Data: stocks.csv

- Weekly data on stock characteristics, prices, and returns from 2021 to present
- Roughly top half of Russell 2000
  - Sort on marketcap each week.
  - Keep stocks 1,001 through 2,000.
- All items are as of the end-of-week market close except `ret`
- `ret` is the return from close of the date shown through close of the following week
- Idea is that we trade at each Friday close, holding portfolio until the following Friday close
- Original daily data comes from Nasdaq Data Link, specifically Sharadar Equity Bundle

Variables

- `open`, `high`, `low` are for the week
- `volume` is average daily volume for the week
- `closeunadj` is split but not dividend-adjusted close for the week
- `closeadj` is split and dividend-adjusted close for the week
- `pb`, `pe`, `ps` are price to book, earnings, and sales
- `evebit`, `evebitda` are enterprise value to EBIT and EBITDA
- `lag1` is the return over the week ending on the date shown
- `lag4` is the return over the prior 4 weeks including the week ending on the date shown, etc.
- `rsi` is the Relative Strength Index

2. Sorting

Method

- Sort stocks on some characteristic into quintiles (for example) at each date.
- Compute the average value of `ret` in each quintile at each date. This is the return of an equally weighted portfolio of the stocks in that group (weights $= 1/n$).
- Compare the returns over time of the quintile portfolios: mean, standard deviation, Sharpe ratio, (and alphas).
- Also look at the $5 - 1$ or $1 - 5$ portfolio: long one extreme quintile and short the other extreme.
- Annualize means, std devs, Sharpe ratios by multiplying by 52, $\sqrt{52}$, and $\sqrt{52}$ respectively.

Sample Questions to Answer

- Do stocks with higher returns last week (or last month or . . .) tend to have higher returns in the future, or should you be a contrarian? I.e., is there momentum or reversal on average?
- Is there a value effect in the data?

3. Regressions

Regression Example

- At a given date, run a regression over the 1,000 stock observations with $y =$ `ret` and $x_1, \ldots, x_n =$ some characteristics.
- Example: April 4, 2025. Characteristics = `pb`, `lag52`, `lag4`, `rsi`.

	coef	**std err**	**t**	**P$>$ $\|$t$\|$**	**[0.025**	**0.975]**
const	-3.001	1.390	-2.159	0.031	-5.729	-0.273
pb	0.042	0.016	2.631	0.009	0.011	0.074
lag52	0.019	0.004	4.912	0.000	0.012	0.027
lag4	-0.124	0.028	-4.406	0.000	-0.179	-0.069
rsi	0.077	0.034	2.266	0.024	0.010	0.144

- Interpretation example: if stocks A and B have past-4-week returns of 8% and 9% respectively, and have the same values for the other variables, then we would expect the return of stock B in the next week to be 12.4 basis points lower than the return of stock A.

Regression at Every Date

- To determine whether stocks with higher `lag4` usually have lower returns, we can run the regression at every date.
- Collect the regression coefficients at all dates.
- Is the coefficient on `lag4` negative on average?
- Is it statistically significant? Instead of checking significance at a single date, consider the series of coefficients as a sample and run a t test.
- Called Fama-MacBeth regressions.

Sample Julius Prompt

Ask Julius to run a regression of `ret` on `pb`, `lag52`, `lag4`, and `rsi` at each date. Collect the slope coefficients across dates and run a t-test on each one.

Or try this (it will probably work): Ask Julius to run Fama-MacBeth regressions of of `ret` on `pb`, `lag52`, `lag4`, and `rsi` and test for statistical significance.

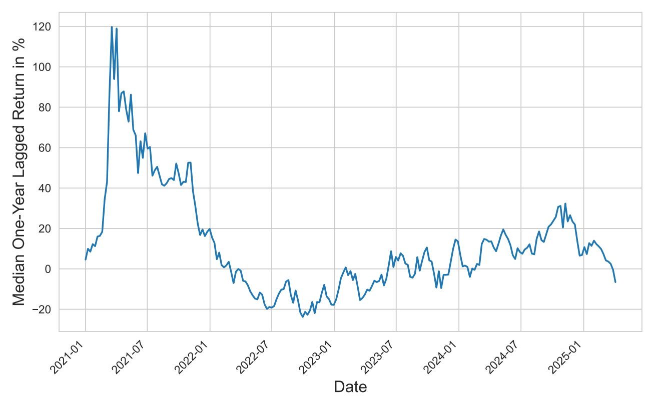
4. Train a Model and Trade on It

Overview

- At some date, use the panel of past data to estimate (= fit = train) a model with $y = \texttt{ret}$ and some x variables (characteristics).
  - Panel = all stocks at all past dates
  - A panel is a two-dimensional array of data, one dimension being `ticker` and the other dimension being `date`.
- Use the trained model and current characteristics to predict future returns.
- Form portfolios based on the predictions – for example, sort into quintiles.
- Can retrain next week with another week of data and use that model to predict for the following week, etc.

Need to Use Relative Data

- Below is a plot of the median value of `lag52` over time.
- A value of `lag52` of, e.g., 40% did not mean the same thing in the spring of 2022 as it did in the spring of 2021.



Standardize at Each Date

- At each date, standardize each feature to be used in the model by subtracting its mean value at that date and dividing by its standardization at that date.
  - This results in each feature at each date having a mean of 0 and a std dev of 1.
  - Avoids issue of noncomparability across dates and also makes models easier to train.
- At each date, standardize `ret` the same way.
  - Hard to forecast the market.
  - So, easier to forecast performance relative to the market (which stocks will beat the market and which won't) than to forecast absolute returns.

Summary of Method

1. Standardize all variables at each date (subtract mean and divide by std dev) – in the code, you might see StandardScaler used for this. Important: define the standardized return to be a new variable, for example stdret.
2. At a given date (e.g., 2024-01-01), use all data prior to that date to train a model to predict `stdret` from standardized features.
3. Use the model and the same standardized features at 2024-01-01 and all subsequent dates to predict `stdret`.
4. Sort into quintiles at 2024-01-01 and all subsequent dates based on the predictions and compute the mean `ret` in each quintile at each date. Important: compute the mean return not the mean standardized return.
5. Analyze the quintile returns over time and the long-short return $5 - 1$: mean and Sharpe ratio. Annualize for easier interpretability.

Question

If you use a multi-layer perceptron and `pb`, `lag52`, `lag4`, and `rsi` as the features, can you use the predictions from a trained model to trade successfully?

Julius may default to a network structure that is too simple, for example, 2 hidden layers with 16 and 4 neurons respectively. You may want to ask for a more complex network, for example, three hidden layers with 64, 64, and 32 neurons respectively.

Caution: Others are already using more sophisticated versions of this, so the market should be mostly efficient. E.g., Gu-Kelly-Xiu (2020)

Also, we are ignoring the fact that you buy at the ask and sell at the bid, which causes round-trip transactions to be costly.