UNIVERSITY OF GLASGOW
ADAM SMITH BUSINESS SCHOOL

Data Science & Machine Learning in Finance (ACCFIN5246)

Assignment 1 – Spring 2024

Hormoz Ramian

A. Instruction

— This assignment counts towards 35% of the overall course grade. This is an individual assessment. Answer all questions. Submission to be made electronically via the course Moodle page. Each part specifies further instructions. The grading weights are described below:

Part	1	2	3	4
Weight	10%	30%	30%	30%

— Results should be reported in a clear format. Avoid reporting numbers in 'scientific format' e.g. `7.2031e-06`. All reported numbers should be rounded to two decimal points. For example, report `0.00` in place of `7.2031e-06`.

Report Organization The assignment requested results are described under Section (E.), Parts [1]-[4], clearly number each part as [1]-[4] in the report. The contents are to be structured as follows:

1. two numbers reported under Part [1], $\overline{xr}\%$ and $\overline{xrm}\%$ and a diagram (with captions, labels, legend, etc.) showing cumulative daily log-returns $\{r_t, r_{M,t}, r_{f,t}\}$ over the time horizon.
2. three diagrams (with captions, labels, legend, etc.) each including three series of estimation outputs described under Part [2],
3. optimal thresholds (τ_j^*)'s and FMSE(τ_j^*) values associated with (c_1)-(c_2) under Part [3], and
4. 500 words comments under Part [4].

The grading is carried out strictly based on the precision of results and clarity of visualisations. The final part is graded based on the relevance of finance analysis supported by the methodologies and empirical results.

B. Data Acquisition

Obtain data for the variables needed to construct and estimate the model in Section (D.). The data should cover the period *2000/01/03-2022/12/30*, on a daily basis. When acquiring the data, ensure relevant characteristics, such as *calendar dates* and *timestamps* are obtained as these additional characteristics are essential throughout the data cleaning and dataset arrangement.

(r_t) real log-returns to be constructed based on Microsoft stock price, acquired from WRDS[1]

[1] wharton.upenn.edu — *Get Data, CRSP, Annual Update, Stock / Security Files, Daily Stock File*

($r_{M,t}$) real market log-returns to be constructed based on the S&P500 composite market index, acquired from WRDS

($r_{f,t}$) real interest rates, associated with US 10-year maturity treasuries acquired from FRED[2]

(CPI) The US consumer price index may be used to transform nominal data to real terms[3]

All series must be researched thoroughly to ensure consistency with other variables, in terms economic interpretations, units, frequency, and other characteristics.

C. Data Preparation

- Construction of the financial dataset must, first takes into account the possibility of sporadic observations. When multiple series are used within the same model, variable timestamps must be aligned.
- The combined dataset including all variables alongside a common timeline then may amount to encountering missing value, NaNs and other irregularities, therefore the data cleaning retains datapoints when all observations are recorded and economically meaningful at each date.
- The definition of daily log-return provides a measurement for value changes between consecutive observations points which may or may not be consecutive days as a result of discarding unbalanced observations. In reality, when multiple days are omitted, for example as a result of an unbalanced dataset, then computed financial returns are split based on the time distance between the two neighbouring datapoints, to adjust for the corresponding performance over a fixed 24-hour window. For simplicity and consistency in this empirical exercise, a daily return is generated based on two adjacent post-cleaning datapoints, regarding the gap between as a calendar day.
- Assume a 'calendar year' comprises exactly 52 weeks, this may amount to minor discrepencies since, year $\approx$ 52.17 weeks – disregard this discrepancy and set each window (w) to contain exactly 52 consecutive weekly datapoints.
- When required, assume a trading year comprises 365 days, thus disregard any variations such as leap years or public holidays affecting the number of trading days.
- Weekly log-returns are defined as the percentage value change between a week's first trading day to next week's first trading day.

The cleaned dataset must be arranged in both daily and weekly frequencies in preparation for various results requested in Section (E.).

[2] fred.stlouisfed.org/ — key: `DGS10`
[3] fred.stlouisfed.org/ — key: `CPILFESL`

D. The Model

Consider the capital asset pricing model characterised by the following specification, used to interrelate the real excess log-return, on a given asset $r_t - r_{f,t}$ where $r_{f,t}$ is the risk-free rate, to the market real log-return denoted by $r_{m,t}$:

$$\underbrace{r_t - r_{f,t}}_{\text{xr}} \quad = \quad \alpha_w + \beta_w \underbrace{(r_{M,t} - r_{f,t})}_{\text{xrm}} + \ u_t \tag{1}$$

note that the object of interest is the time-varying feature of the coefficients $\widehat{\alpha}_w$ and $\widehat{\beta}_w$. In particular, $\widehat{\beta}_w$ summarizes the conditional relationship, given a rolling window incorporating a consecutive but limited span of data, between the market excess log-return $r_{M,t} - r_{f,t}$ and an individual investment excess log-return. The diagram below provides an illustration to describe overlapping windows (w), including a calendar year of data:

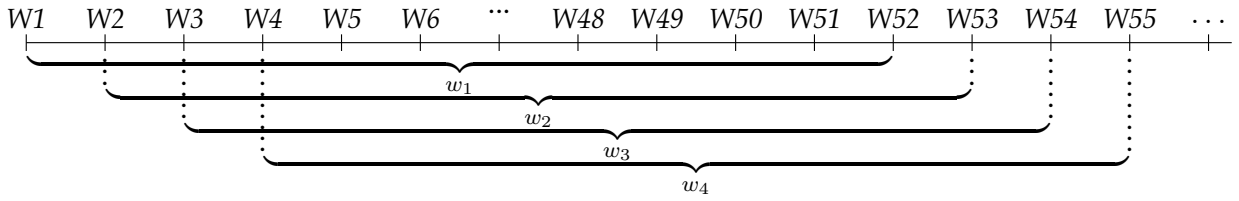

Figure 1: The timeline illustrates a rolling window set-up, where each iteration includes a consecutive 52 weekly datapoints, where W1, W2, ... refer to week numbers throughout the entire sample and w_i refer to a rolling window indentifier.

E. Implementations and Results

Based on the dataset and instructions in Sections (B.)-(C.), complete the following parts.

Part [1] Construct daily real excess MSFT log-returns (xr) and daily real excess market log-returns (xrm). Report (i) the precise values for two averages: $\overline{xr}\%$ and $\overline{xrm}\%$ over the entire sample in daily net log-return averages (rounded to two decimal points) and (ii) a diagram depicting the cumulative daily log-returns $\{r_t, r_{M,t}, r_{f,t}\}$, overlaid within the same diagram[4] space, with the vertical axis showing the cumulative log-returns `cumsum(rets)` versus the horizontal axis, showing a representation of calendar time. **(Mark: 10%)**

Arrange the dataset, based on a *calendar variable*, such that all data are set to a weekly frequency. The construction must be based on the first trading days between two consecutive weeks. Proceed to parts **[2]-[4]** based on this data frequency. Each window within the rolling specification contains 52 adjacent weekly observations.

Part [2] Implement a restricted least squares model based on specification (1) in addition to the

[4]Overlaid diagrams are created via various approaches, e.g. use `plot(xseries,yseries); hold on`, followed by additional plots in the same format and ending the last plot with `hold off`.

following constraints, implemented individually across two cases c_j, $j = 1, 2$:

$$\alpha_w \geq \tau \tag{c_1}$$

$$\alpha_w < \tau \tag{c_2}$$

Assume $\tau = 0$. Store all values obtained for $\widehat{\alpha_w}(c_j)$ and $\widehat{\beta_w}(c_j)$ and the estimated Lagrange multiplier(s) $\widehat{\lambda_w}(c_j)$ for each of the cases described in expressions (c_1)-(c_2). Present the results in three diagrams for each of the estimated variables alongside the time horizon — for example, the first diagram depicts two $\widehat{\alpha_w}$ series in expressions (c_1)-(c_2) (overlaid in the same diagram on the vertical axis) versus time (horizontal axis, displayed as the year or a simplified date format), with the diagram legend identifying each series as the outcome for expressions (c_1)-(c_2).[5] The illustration of lines must clearly be identifiable either with colors or line patterns. Clearly label all axes with variable names and their units. **(Mark: 30%)**

Part [3] Assume τ is now variable and implement the following optimization. Consider a linear regression between the LHS of expression (1) versus the lagged value of $\widehat{\lambda_w}(c_j)$, i.e. two cases given $j = 1, 2$, $xr_t = \theta_0 + \theta_1 \widehat{\lambda}_{w,t-1}(c_j) + \nu_t$. Compute the forecast MSE of a 1-step ahead predictions (based on 52-consecutive observations to predict one week ahead, using weekly data) for both expressions (c_1)-(c_2) throughout the series horizon, computed separately once for all points in a grid for parameter $\tau \in [-1\% : 0.01\% : +1\%]$ i.e. the points are defined as -1%, -0.99%, ..., $+0.99\%$, 1.00%. Based on the FMSE's (lower FMSE is better) for each of the 201 cases within the grid search, report the best value τ_j^* for expressions (c_1)-(c_2) that generates the lowest FMSE, together with the FMSE(τ_j^*) values for both expressions (no additional comments). **(Mark: 30%)**

Part [4] Explain with comments, why predictions based on expressions (c_1)-(c_2) should result in the FMSE values (ranking) above. Comments should draw on finance analysis in connection with the empirical framework developed in Parts [1]-[3]. The research may refer to the two references (F[5], F[6]) **(Mark: 30%, 500 words)**.

F. Computational Notes

While calendar timeline may be handled via various approaches, using the following built-in libraries is recommended:

- Utilising `timetable` variable type facilitates timestamped operations. Upon inputting the variable type[6], (i) `week()` returns the week number in a given year, (ii) `weekday()` returns the day position in a given week.
- Inequality constrained within the constrained least-squared method are implemented via various approaches (G[3], G[4]). A built-in routine `lsqlin` is a suitable and efficient method to compute the problems (G[2]).

[5] Similarly, a diagram including $\widehat{\beta_w}$ series in expressions (c_1)-(c_2), and a separate diagram including $\widehat{\lambda_w}$ series in expressions (c_1)-(c_2).

[6] Timetables & Functions.

- Grid search can be constructed based on `linspace(lb,ub,201)` for the case described in Part [3] where the function generates evenly spaced values between `lb` to `ub` and for 201 points.

G. Background Reading

[G1] Lecture slides
[G2] Constrained least squares optimization with inequality constraints may be carried out using analytical or computational approaches such as `lsqlin` as described in The MathWorks Inc. Optimization Toolbox: Solve Constrained Linear Least-Squares Problems (lsqlin), 2022
[G3] Frank A Wolak. An exact test for multiple inequality and equality constraints in the linear regression model. *Journal of the American Statistical Association*, 82(399):782–793, 1987
[G4] Chong Kiew Liew. Inequality Constrained Least-Squares Estimation. *Journal of the American Statistical Association*, 71(355):746–751, 1976
[G5] S&P U.S. Indices Methodology
[G6] Microsoft Annual Report (2023)