ST326 Assessed Coursework
Deadline: 12pm, December 13, 2024

1 Questions

This project is on the analysis of a bundle of stocks that are constituents of S&P500. You have the freedom to choose 10 stocks from the top 100 constituents by weight. The weights change everyday, but as long as the 10 stocks you have chosen have been the top 100 constituents on a particular day and have been traded over the past 5 years, it is fine.

1. Download the daily closing prices of the 10 stocks and the S&P500 index price for the past 5 years. Do not include 2 or more stocks from the same company but only of different classes. Plot their log-prices on the same plot.

   You can deal with potential missing values using R codes similar to Chapter 3 of your lecture notes, or any other methods, but you need to justify them.

   If you are downloading data using the `quantmod` package, you may want to export the data to a text file first using for example

   ```
   library(quantmod)
   getSymbols('F')
   F = as.data.frame(F)
   F = cbind(as.numeric(as.Date(rownames(F))), F)
   write.table(F, "F.txt", row.names=FALSE)

   getSymbols('^GSPC')
   GSPC = as.data.frame(GSPC)
   GSPC = cbind(as.numeric(as.Date(rownames(GSPC))), GSPC)
   write.table(GSPC, "GSPC.txt", row.names=FALSE)
   ```

   Then the corresponding lines in `read.bossa.data` inside a for loop should be changed to


```
filename <- paste("project/", vec.names[i], ".txt", sep="")
### If you store your .txt files in a folder called "project"

tmp <- scan(filename, list(date=numeric(), NULL, NULL, NULL,
NULL, NULL, close=numeric()), skip=1, sep="")
```

Then you can read

```
ind = read.bossa.data(c("F", "GSPC"))
```

Are there any similar trends or not?

2. Our aim in this part is to predict the next day S&P500 return using q lags of S&P500 as well as the most up-to-date returns of the 10 stocks you have chosen.

   Split the data set into 50% training, 25% validation and 25% test sets.

   If you are following the steps in part 1., remember to change `shift.indices` in `pred.footsie.prepare` to appropriate values, and any other changes you need if you want to use the function in its entirety. (remember we are using the past 5 years of data only)

   For the 11 daily return series, write an R programme to use exponential smoothing to estimate their daily volatilities over the horizon of the training data. Individual λ for each series should be estimated by MLE. In doing so, you should write down the assumed model for each time series.

3. Write down a modified prediction algorithm similar to the one in Section 3.4 of your lecture notes (define all notations involved), so that:

   i. It takes in a warmup time t_0, a window length D, and the appropriately normalised (using the same λ's found in part 2) $10+q$ return series as input.

   ii. It uses ordinary least squares for linear regression over a rolling window of length D as a way to estimate the next day normalised return for the S&P500.

   iii. The investment strategy is to invest 1 unit of money into S&P500 if the next day return is predicted to be positive, and -1 unit of money (i.e., short-selling) if the next day return is predicted to be negative.

iv. The annualised Sharpe ratio is calculated in the end, using daily true return (i.e., true day-$(t+1)$ return for your investment at time t for S&P500), but ignoring all transaction costs.

4. Code the above algorithm in R, for training, validation and test sets. For the validation and test sets, the same λ found in part 2 can be used. The output should be Sharpe ratios for different values of window lengths.

   Run the algorithm with $q = 0$ and $q = 1$. In both cases, comment on the appropriateness of using ordinary least squares over the training, validation and test sets, with justifications (include corresponding graphs if possible) to your arguments.

5. As a way to improve upon ordinary least squares, the one-day-ahead S&P500 return is to be predicted using the factors from the 10 stocks you have chosen as covariates. Instead of determining the number of factors using a scree plot for each window, treat the number of factors as another tuning parameter, on top of the window length. To simplify your task, consider number of factors up to 2. (The technique is called principal component regression)

   Hence in each window, perform a multi-factor analysis, and use the estimated factor series as the covariates, still using a linear model for predicting the one-day-ahead S&500 return. The output Sharpe ratios for our trading strategy should then be dependent on window length as well as number of factors considered in each window (you can assume the number of factors used in each window is a constant).

   Is this method better than just using ordinary least squares? Describe your findings, with supporting arguments and outputs.

2 Submission

- Submit your work **anonymously** under your **candidate number** in LSE For You. (**NOT** your ID Number starting with 20XX). **Write your candidate number on a cover page as well within the pdf file.**

- Plagiarism will be checked, and students who found to plagiarise will not only be penalised, but also face potential disciplinary actions from the school.

- Upload a **single pdf file** to the corresponding course-work upload link on Moodle.
- The single pdf file should contain your presented answers including graphs and tables. All R codes used should be added in an appendix in the end.
- The upload link will stop working after the deadline indicated on the link. You can still submit then by sending the file directly to me.

  Late submission will result in penalties: 5 marks (out of maximum 100) will be deducted for every half-day (12 hours). This will result in a maximum penalty of 10 marks for the first 24 hours. A further 5 marks will be deducted per 24 hour period thereafter (including weekends.)
- Extensions to deadlines for coursework will only be given in fully documented serious extenuating circumstances.

