STATS 601 Final Project

Due by 11:59pm on Apr 16th, 2022

In this project, you will apply what we have learned in STATS 601 to build a predictive model based on a real financial dataset. The goal is to predict *the 30-minutes forward returns* of 10 assets based on the historical price and volume series. You will be given minutely log-prices and trading volumes of these 10 assets from July 1 to Dec 31 in 2021 as the training and validation data to build your predictive model.

Format. You will write a Python function `get_r_hat(A, B)` that takes two Pandas dataframes `A` and `B` of past one-day minutely log-prices and volumes of the 10 assets (i.e., two 1440-by-10 numeric dataframes) as input, and that returns the prediction of the *30-minutes forward log-returns of the 10 assets* at the last minute of the input dataframe as output (i.e., a 10-dimensional numpy float array). You may also upload a model file that stores the fitted model for `get_r_hat()` to load. You should use `A` and `B` to construct the features and then feed them to the fitted model to generate predictions.

Online Judge (OJ) policy. After you submit your code and data file, our OJ will report the predictive performance on a testing dataset that is withheld from you. The size of your submission must be less than 20MB. Everyone within a team can submit code to the OJ for the entire team. The OJ allows *three submissions per day per team*, and only the highest score will be counted towards the grade. Therefore, please start early and exploit every opportunity to hit a higher score! During the final project, you will be updated with your team's current ranking based on your state-of-the-art result every 24 hours. The ranking of a team depends on the highest out-of-sample correlation that is achieved.

Grading. The specific grading policy will be finalized after the project is over. The full mark is 100 points. You will receive 30 points as long as your team makes a valid submission with positive correlation. We will consider giving extra bonus points to top teams depending on their performance. By default, everyone within the team receives the same scores for the final project, unless there are free riders.

Tips

- Note that there should not be any model fitting inside the function `get_r_hat`. Rather, you should train your model based on the given data outside this function, and extract the fitted model to build `get_r_hat`.
- Please go through `stats_601_project.ipynb` to learn the format of `get_r_hat` and how the OJ evaluates your model.
- The OJ will call `get_r_hat` nearly 10,000 times, and the time limit for the entire evaluation is 10 minutes. Please ensure that your function is computationally efficient before you submit it to the OJ.