Executive Summary

Problem

The objective of this thesis is to create a simulation of financial and macroeconomic data that can be deployed for a portfolio management game used by the Department of Banking and Finance at the University of Zurich. The existing portfolio management game is based solely on historical data from a specific time period in the past. Thus, it is possible that students find out which historical period is used for the game and can, therefore, outperform the game. For educational reasons, it should not be possible that students can specify their asset allocation based on the information about the historical period. As a consequence, the time series used for the game have to be simulated. Ideally, the students get predictions on the upcoming period in advance that are equal to the parameters set for the simulation of the period and therefore have a certain true content. Hence, students cannot invest based on their knowledge of historical time series but have to define their asset allocations solely on the market forecast.

Method

A simulation of financial and macroeconomic time series was programmed using Python. As a prerequisite, a database containing financial and macroeconomic data had to be built. Thus, modules to fetch the historical time series from various online databases were created and used in a first phase. Due to inconsistencies between the financial and macroeconomic series, the data points have to be harmonized in a next phase by interpolating macroeconomic series. Furthermore, portfolios are built based on the sector affiliations of the stocks in order to replicate sector indices for every regional market. In order to be able to classify the historical time series based on their mean and volatility, all series are standardized.

In the second phase, a simulation of new data was run based on the historical series. A lot of different methods to predict the future outcome of stock prices can be found in the literature. However, most of these approaches are only used to predict the stock price for short periods and single stocks. Therefore, a new way of simulating time series was used. The approach is based on a statistical technique, namely the bootstrap method, where new series are generated by randomly drawing historical points and stringing them together. More specifically, blocks of historical series were used based on specified characteristics. The user of the simulation can specify these characteristics on a web application. For every block that fits the characteristics, all available time series are appended to the existing ones. Thus, the correlation between the financial and macroeconomic series is preserved for every block.

Results

All historical time series are stored on an Influx database which is a database specially designed for time series. Unfortunately, not all series are available for the same historical period. As most European and American stocks are available from 1973 on, this year was selected as the start year. However, financial data is hardly available for emerging markets for the equivalent time range. Thus, the number of financial series grows every year as new stock, and market data is available. Furthermore, unlike financial data, macroeconomic data is only available on a monthly, quarterly or biannual frequency. As a consequence, the missing daily values of these series have to be interpolated. For the sake of simplicity, this is done with linear interpolation.

In order to build the portfolios for every sector, several methods were applied. First, portfolios were created by weighting according to the price of every stock. The problem of this approach is the high impact of new stocks with a high price in relation to the other constituents as it results in a significant jump in the price of the portfolio. As a consequence, portfolios were created according to the market capitalization of the constituents. However, this had not the desired effect as the jumps were still available when adding a new constituent to the portfolio. Finally, portfolios were created by weighting the return of the constituents according to the market capitalization. The resulting portfolio price curves were much smoother as jumps could be eliminated. Additionally, constituents that had a high variation compared to the other constituents in the past were eliminated as their impact on the price of the portfolio was too large.

With growing numbers of time series, a weakness of the Influx database was revealed. The database stores a lot of data in the working memory in order to be able to perform fast queries. Hence, with a growing number of series, the required working memory also increased. As a consequence, the parameters of the database had to be changed in order to prevent the database from storing the data in the working memory. This change required a reset of the database and with that a deletion of all time series stored on it.

The result of the simulation can vary a lot depending on the choice of the user. There can be no outcome if there do not exist enough historical data points that fit the parameters. In this case, the simulation cannot be performed and requires an adaptation of the characteristics. The user has almost an infinite number of possible combinations of the characteristics that can be set. One of the intentions of this thesis was to provide some use cases and possible scenarios. Due to a shut down of the server on which the Influx database runs, this part of the thesis could not be completed.

Evaluation

This thesis provides a simple approach for simulating new time series based on historical ones. Unfortunately, the results cannot be shown in this thesis. However, the simulation generated useful new time series although the tested combinations of characteristics only reflect a small part of all possible combinations. It can, therefore, be that a combination of characteristics returns useless series. This indeed calls for further testing of new combinations of characteristics. Furthermore, the length of the blocks should be optimized. In this thesis, the block size was set to one business week. However, it is possible that the simulation would return better results when using longer blocks, but on the other hand, longer blocks result in a lower number of blocks available for the simulation and therefore less possible combinations.

On the technical side, it is not clear at the moment if the new database structure will have a better behavior than the previous one. At the moment, less working memory is needed with the new, but it will come to light if the database is able to handle large data queries as needed for this simulation.