

POLITECNICO MILANO 1863

FINANCIAL ENGINEERING 2024/25

Report Assignment 0 RM

LEONARDO PASCOTTO, ALBERTO PELLEGRINI, TOMMASO RESTA, DALILA ROSATI

Contents

Introduction	2
Bootstrap in Python vs Bootstrap in Matlab:	2

Introduction

The goal of this assignment is to replicate the Bootstrap implemented in Matlab in order to address the case study related to Assignment 2, using Python.

Bootstrap in Python vs Bootstrap in Matlab

To perform the bootstrapping, we used the same approach as in Matlab. Therefore, we utilized the first four interbank deposit rates, the first seven quoted interest rate futures, and an incomplete set of swaps with maturities up to 50 years, which was completed for the bootstrapping process. All the steps were implemented using a customized function:

```
bootstrapped_dates, bootstrapped_rates = bootstrap(datesSet, ratesSet)
```

Furthermore, we created a file containing the utility functions needed to complete the task. Specifically, the following functions were implemented in `utilities.py`:

- `yearfrac(start_dates, end_dates, flag)`
- `zeroRates(dates, discounts)`
- `read_mat_file_dates(file_path)`
- `read_mat_file_rates(file_path)`
- `adjust_to_next_business_day(date)`
- `convert_date_to_string(dateArray)`
- `plot_bootstrap_results(dates, discounts)`

In Table 1, we report the latest discount factor obtained using both Python and Matlab. It can be noticed that, as expected, they coincide.

Date	Python	Matlab
2073-02-02	0.3982522449122427	0.398252244912243

Table 1: Comparison of the discount values obtained with Python and Matlab

In addition, we created a plot of the discounts and the zero rates to further demonstrate that using the two software tools (Matlab and Python), we obtain coherently the same results.

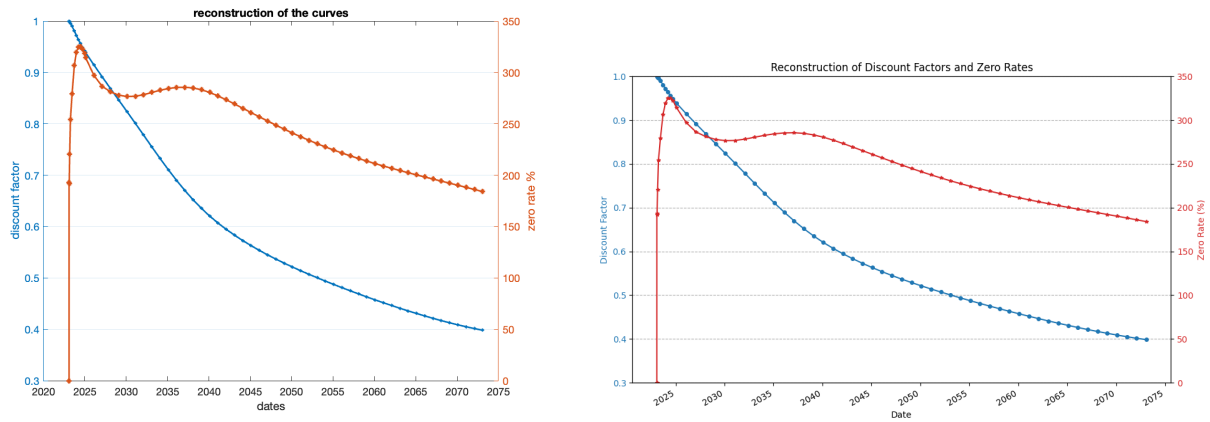


Figure 1: On the left, Matlab bootstrap. On the right, Python bootstrap