Mid term Exam for Financial Econometrics with Python

PRAT Paul; GAVINI Charles; FOURNIER Justin; BLANC Mathieu

November 11, 2024

Contents

1 Introduction

For the midterm assignment, we composed a group of 4 with Gavini Charles; Fournier Justin; Prat Paul and Blanc Mathieu. This document contains all the results of our assignment, including tables, figures, and calculations. It is composed by 1 parts, first, importing the good python libraries, then initialising variables to separate the differents datas (daily, monthly, ..., returns, logreturns,...)

2 Preliminary

2.1 AMAZON

The chosen stock is Amazon, because it is higly related in the current actuality. We are very interested in a major company such as Amazon, which has grown significantly over time. The ticker from yahoo finance is "***AMZN*** " on the Nasdaq stock exchange AMAZON on Yahoo Finance First, importing the Amazon stock with yfinance, then display the pandas table. We will import 25 years, 8 months and 25 days of data (from 1999-01-21 to 2024-10-16).

2.2 Data Table

The data printed here, is the preview of the Amazon stock extraction from yahoo fiannce:

2.3 Checking the 25 Years range condition

We have to check that the data is correctly displayed over a 25years range, hopefully, the introduction from Amazon is from january 1999, so it we should be able to find a 25 year range of data of the Amazon stock. To check that we can compute a python code to count the gaps and visualize the dates of gaps in order to see for any huge gap that would be problematic for analyzing data.

Date	Open	High	Low	Close	Adj Close	Volume
1999-01-21	2.612500	2.759375	2.314063	2.650000	2.650000	940964000
1999-01-22	2.487500	3.146875	2.468750	3.075000	3.075000	875316000
1999-01-25	3.037500	3.084375	2.750000	2.809375	2.809375	546476000
1999-01-26	2.815625	3.031250	2.765625	2.877344	2.877344	490696000
1999-01-27	3.353125	3.493750	3.000000	3.140625	3.140625	700452000

Table 1: Preview of Amazon Stock Data from Yahoo Finance

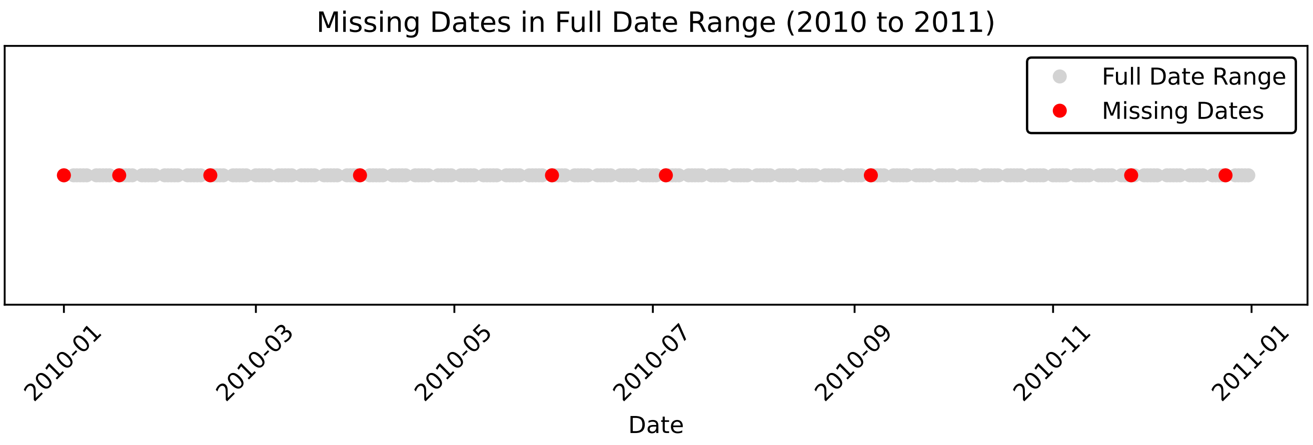

Figure 1: Missing Dates in Full Date Range (2010 to 2011)

We count less than 10 days of data gaps ponctually for one year over the 25-years range. Thus, the full data is still relevant to use for ou analysis of stilyzed facts.

3 First Results

3.1 Prices Evolutions

Then, ploting the prices evolution:

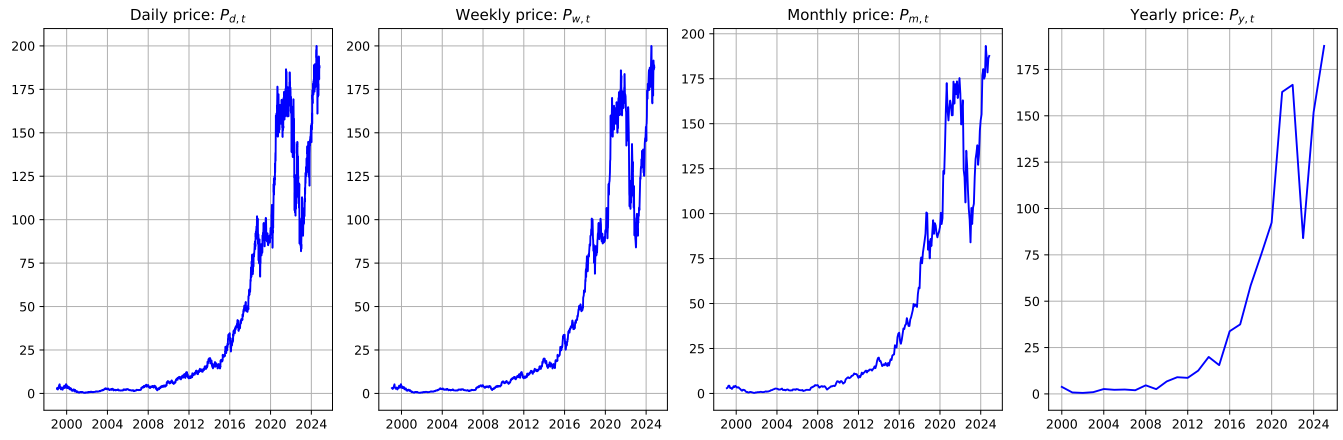

Figure 2: Prices over time by frequency

3.2 Calculating Returns

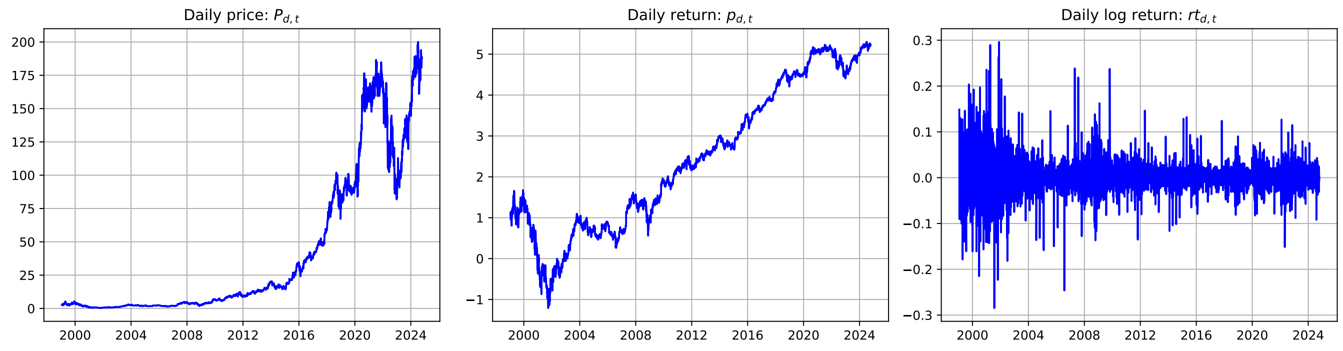

Figure 3: Prices, returns and log returns

3.3 Squared Returns

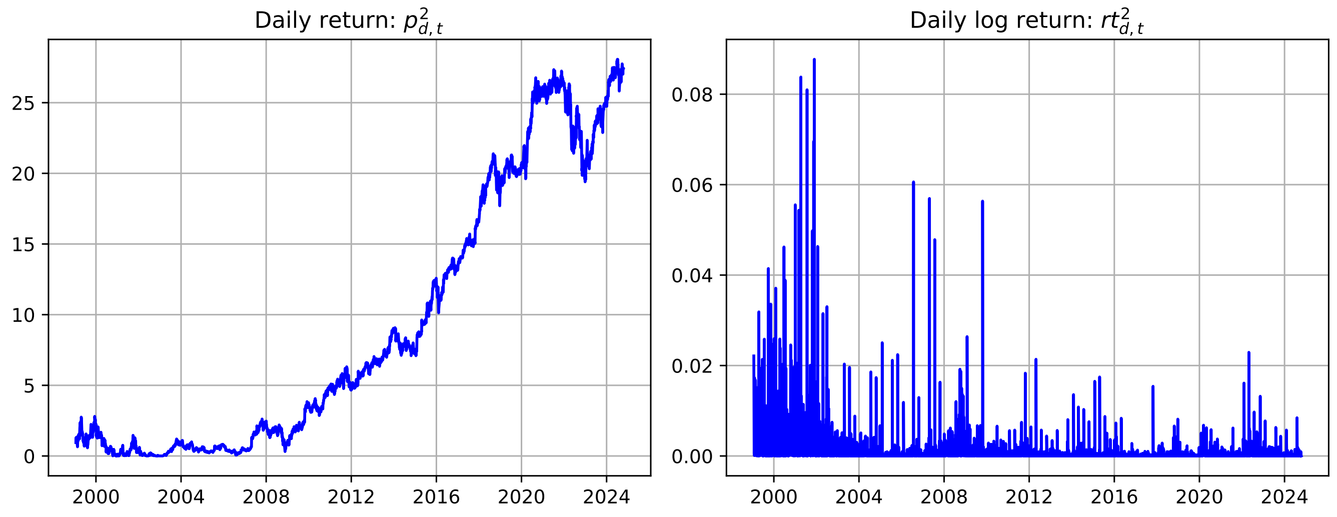

Figure 4: Squared daily returns and daily log returns

4 Amazon and the 8 Stylized Facts

4.1 Prices are non-stationary

The first feature that will highlight non-stationarity of the prices is the comparison of p_t vs p_{t-1}.

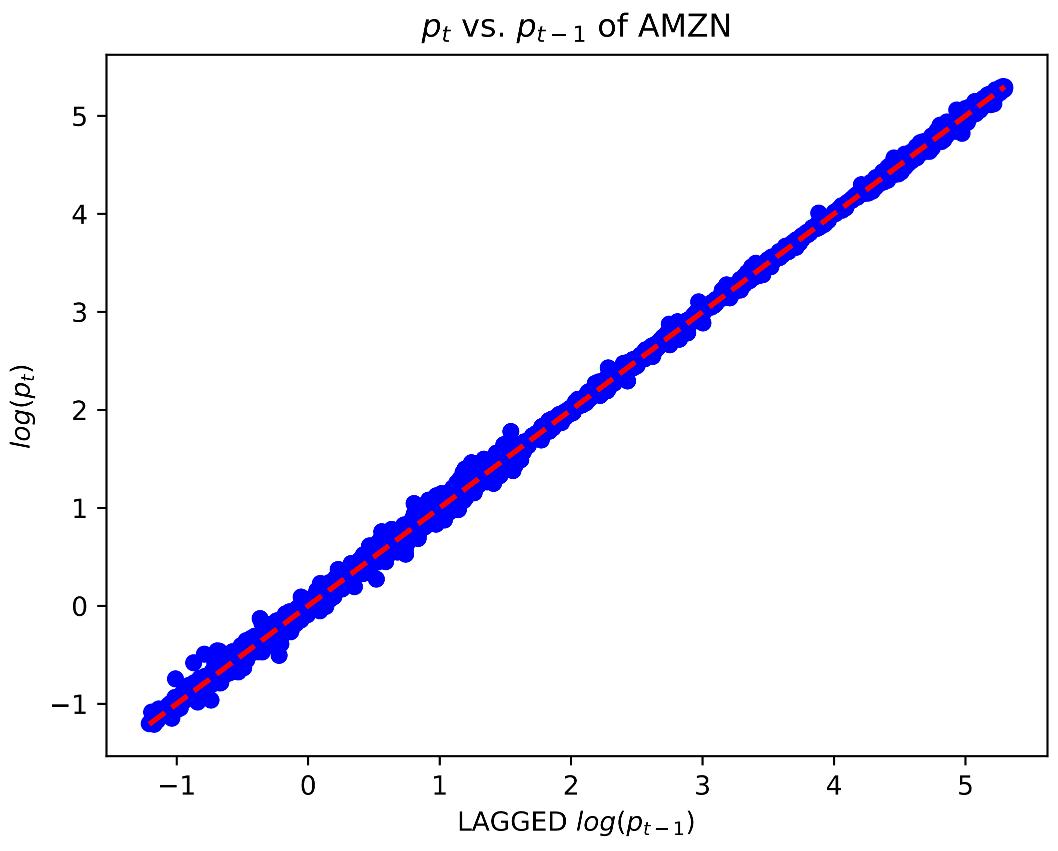

Figure 5: Comparison of $\log(p_t)$ vs $\log(p_{t-1})$

The graph in Figure 5 demonstrates this strong linear relationship, indicating that Amazon's prices at time t are highly dependent on those at $t-1$ and lack mean reversion, supporting the idea of non-stationarity.
Additionally, the empirical autocorrelation function (ACF) of Amazon's daily prices shows a slow decay, further suggesting non-stationarity, as shown in the next figure.

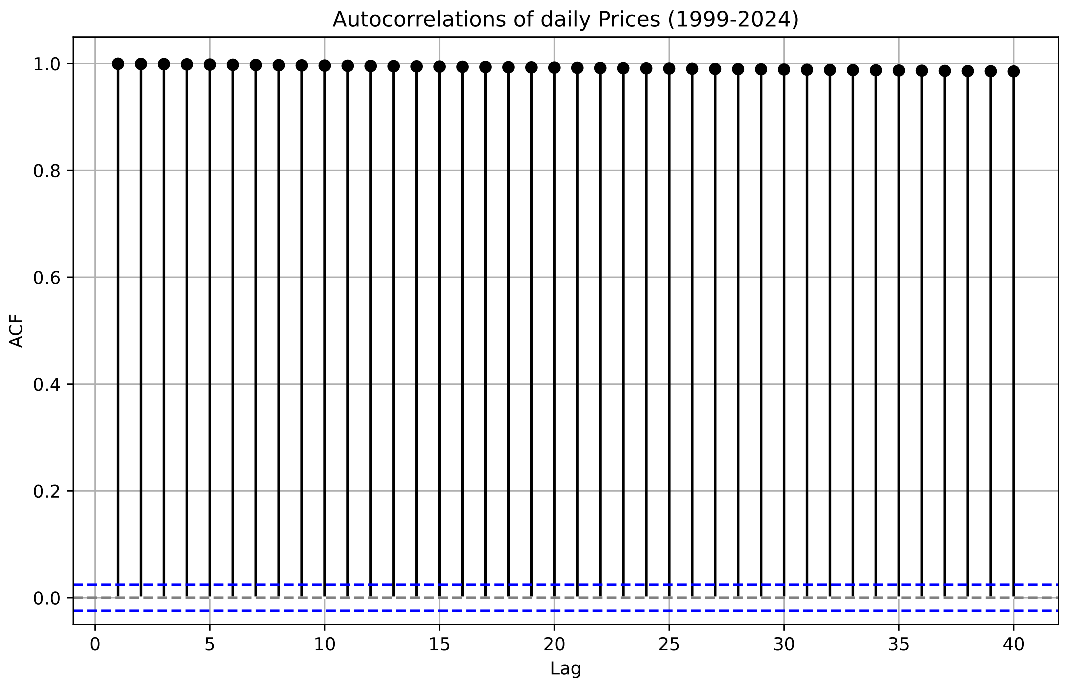

Figure 6: Autocorrelations of daily Prices (1999-2024)

4.2 Returns are stationary

4.3 Asymmetry

4.4 Heavy tails

4.5 Gaussianity

4.5.1 High frequency non-Gaussianity

4.5.2 Aggregational Gaussianity

A Appendix: Python Code

Below is the Python code used in this analysis.

```python
# Python code example
import numpy as np
import pandas as pd

def analyze_data(data):
    mean = np.mean(data)
    std_dev = np.std(data)
    return mean, std_dev

data = [1, 2, 3, 4, 5]
mean, std_dev = analyze_data(data)
print(f"Mean: {mean}, Standard Deviation: {std_dev}")
```

Listing 1: Python Code for Analysis