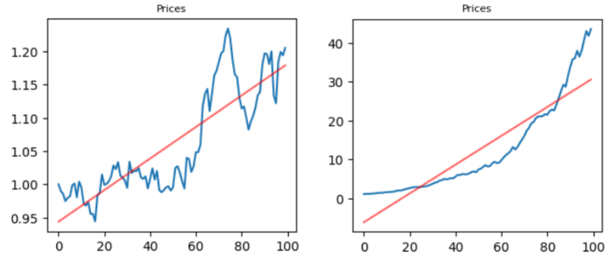
Filter

Problem

One problem we noticed in the samples generated by our model after the application of the filter is that some are really good and realistic, while others are have no sense, for exemple let's take the samples shwon in the image below:

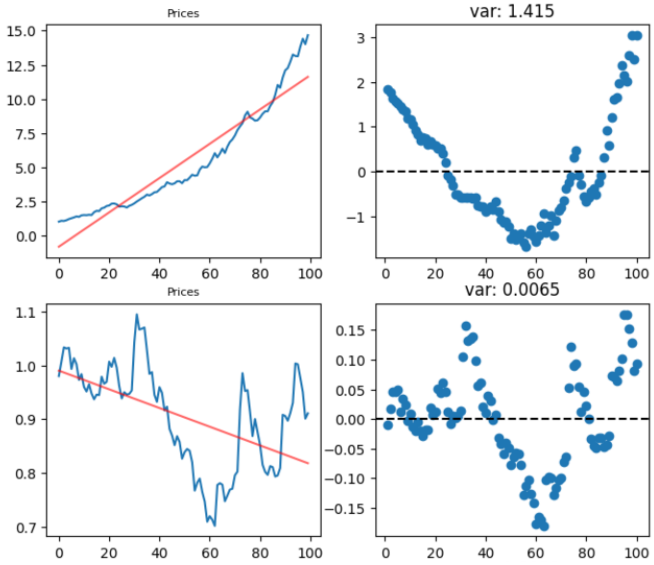


as we can see the first serie is very realistic while the other is completly out of scale and present a trend that is not typical of fiancial time series. To solve this problem we created a filter that chose from the samples generated the realistic series without modifyng the model output, so we end up with a realistic series dataset.

How It works

To select the realistic series we came up with bunch of metrics, the value of these metrics can be tuned in the filter paramethers:

Distance between residuals We noticed that once we apply a linear regression to a serie the distance between consecutive residuals is smaller when the serie has a not relistic behavior, while when we have a realistic serie the discance between consecutive points residuals is bigger.

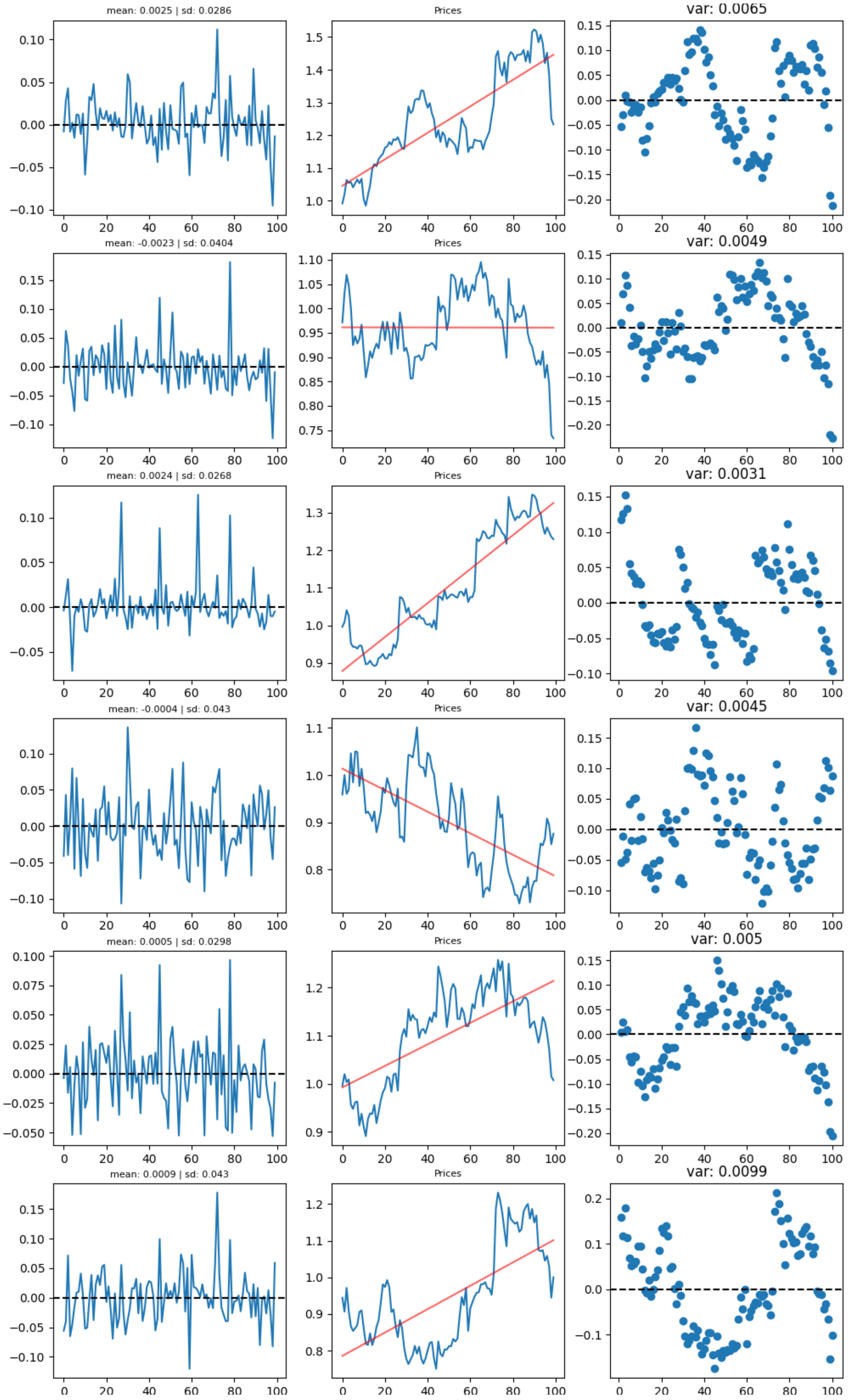


What we did is take the price of the generated serie, scaling it between 0 and 1 to make it comparable and apply a linear regression to it. Once we obtainid the linear regression we computed the summation of the distance between consecutive residuals. What the filter does at this point is just select the series which summation of consecutive residuals is within a given threshold.
DfGenerator parameter: **variance_th**.

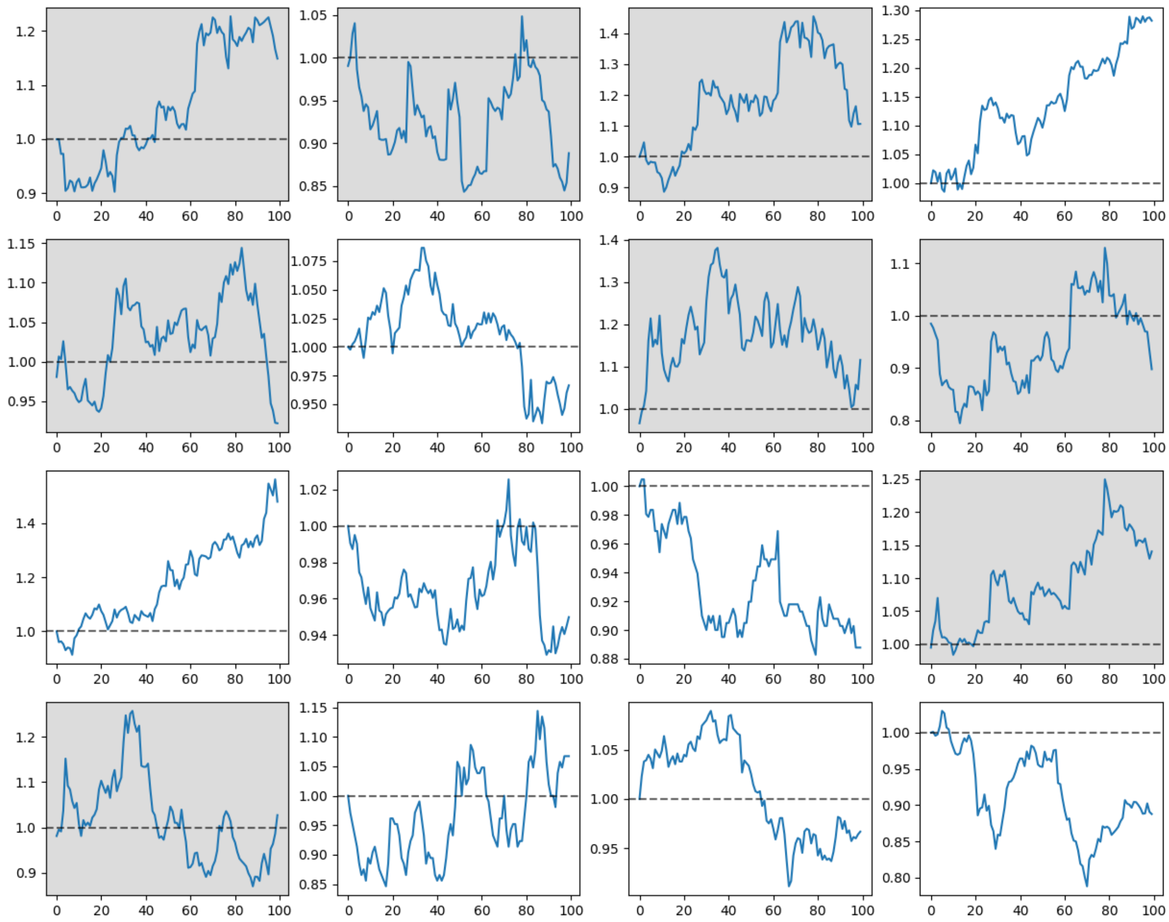
Maximum range of oscillation We decidet to introcuce in the filter a paramether to chose the maximum range of oscillation between the first point an the last one in percentage. This because we think it could be usefull for the porpouse to have the freedom to decide the type of series we want to generate to expand our dataset. Maybe we want to expand our dataset with very volatile series, in this case we can chose to use an hight max range of osscilation. One the other side oure need could be to expand our dataset with more stable series, maybe to decrease the volatility of the predictions made by a model, in that case we can tune the parameter chosing a lower max range.
DfGenerator parameter: **max_range**.

Final Results

the final result of the project is a program capable of generate realistic financial series such the followings:



the generated series compared with some real series look like that the generated series are shown with a grey background:



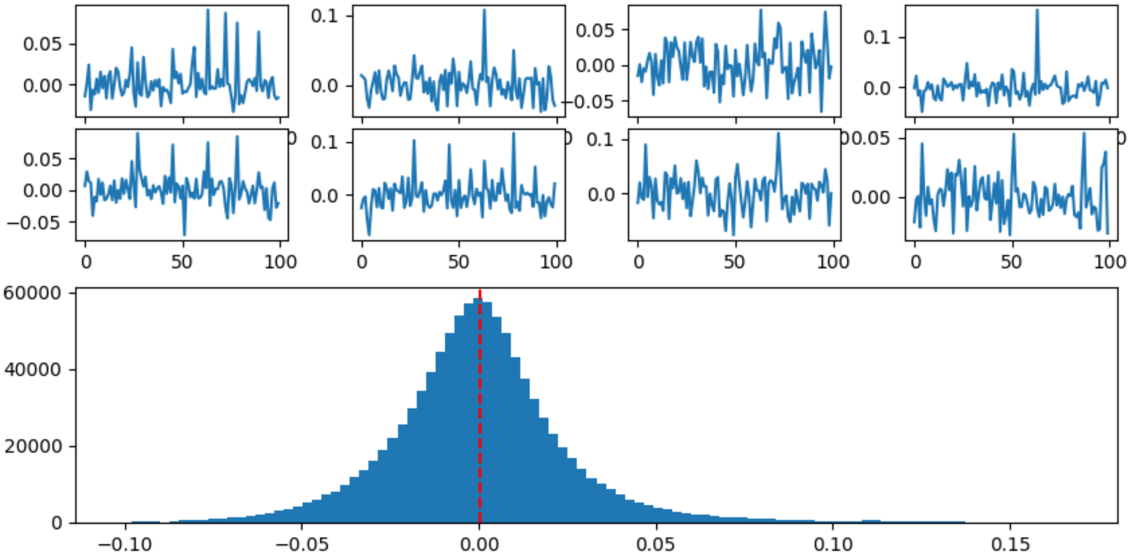
one of the main characteristic of retuns of financial series is the mean that is almost equal to 0. For that reason we tested if the mean of the returns generated by our model and the mean of the real series of the dataset is the same. To do the following hypothesis test:

$$\begin{cases} H_0 : & \overline{X} = 0 \\ H_1 : & \overline{X} \neq 0 \end{cases} . \tag{1}$$

$$T = \frac{\overline{X} - \mu_0}{S_n/\sqrt{n}} \qquad T|H_0 \sim t_{n-1}$$

to test this hypothesis we generated a dataset of 10 000 samples an computed the mean: 0.00026567.

The program takes approximatly 20 mintutes to generate a 10 000 samples dataset on a single cpu Intel Core I7. The distribution of the generated samples, looks like this:


0.05
0.00
0.1
0.0
0.05
0.00
−0.05
0.1
0.0
0.05
0.00
−0.05
0.1
0.0
0.1
0.0
0.05
0.00
0
50
100
0
50
100
0
50
100
0
50
100
60000
40000
20000
0
−0.10
−0.05
0.00
0.05
0.10
0.15