5.2 Challenge: Simulating the normal distribution

Wednesday, August 10, 2022   8:23 PM

Myroslava Sánchez Andrade A01730712 | 10/08/2022

We know that the column "r" of BTC cotains the historical cc returns of Bitcoin. We use the function random.normal to simulate random returns given the mean, standard deviation and size.

```
1  gen_values = np.random.normal(BTC["r"].mean(), BTC["r"].std(), BTC["r"].count())
2  print(gen_values)
```

```
[-0.03228258 -0.04825458  0.02707137 ... -0.00301276  0.02008491
 -0.02381526]
```

```
1  #Showing the real distribution of historical cc returns and simulated normal distribution
2  sim_bitcoin = pd.DataFrame(gen_values)
3  matplotlib.pyplot.hist(x= r_bitcoin, bins=90,alpha=0.5,range=(-0.4, 0.4),label="Real")
4  matplotlib.pyplot.hist(x=sim_bitcoin,bins=90,alpha=0.5,range=(-0.4, 0.4),label="Simulated")
5  matplotlib.pyplot.legend(loc='upper right')
6  matplotlib.pyplot.show()
```
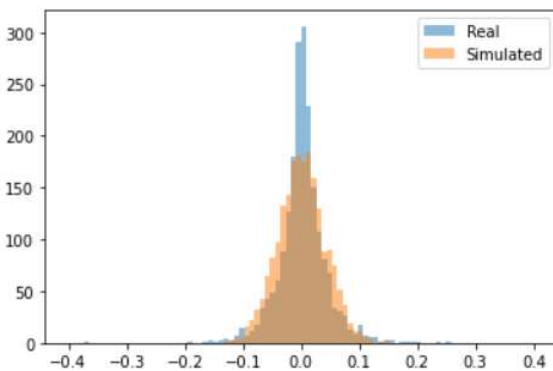

THERE IS A DIFFERENCE IN FREQUENCY WHEN X≈0. THE FREQUENCY USING THE REAL DATA WHEN THE RETURNING≈0 IS GREATER THAN IN THE SIMULATED ONE, BUT WE CAN STILL CLEARLY OBSERVE THAT BOTH DISTRIBUTIONS EXPOSE A SIMILAR NORMAL BEHAVIOUR.