# Data Science of Complex Systems

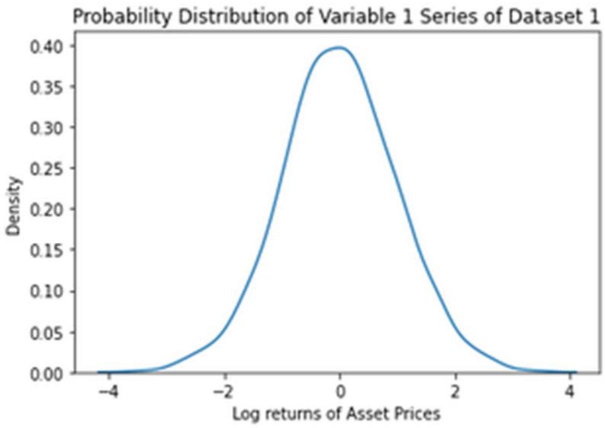
## Assignment 2 Report

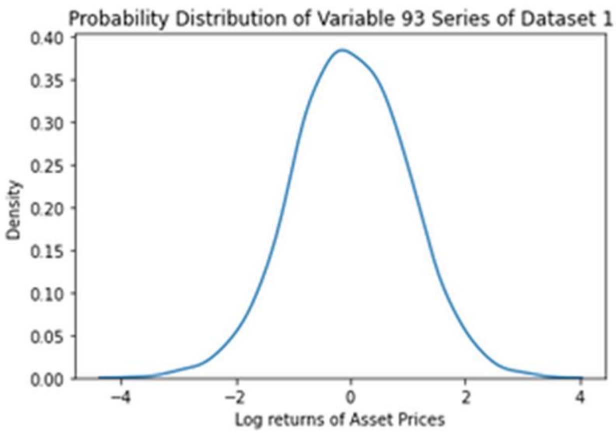
Name: Gudivada Harsha Vardhan
Roll No: cs21m021

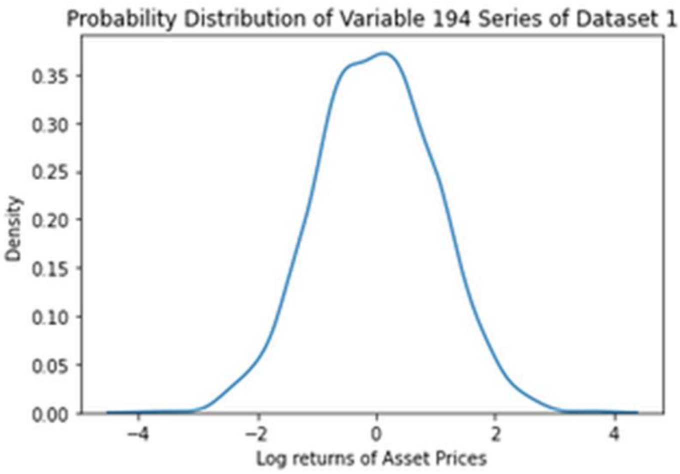
1.

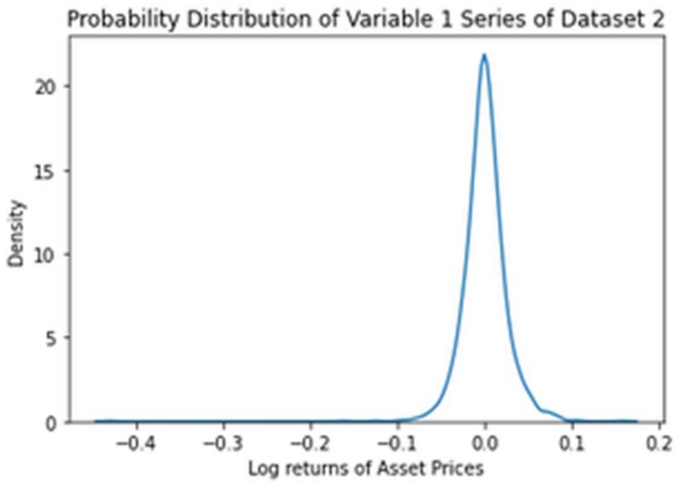
a. Probability distribution of the values r(t), where r is a sample time series from the dataset you are considering.

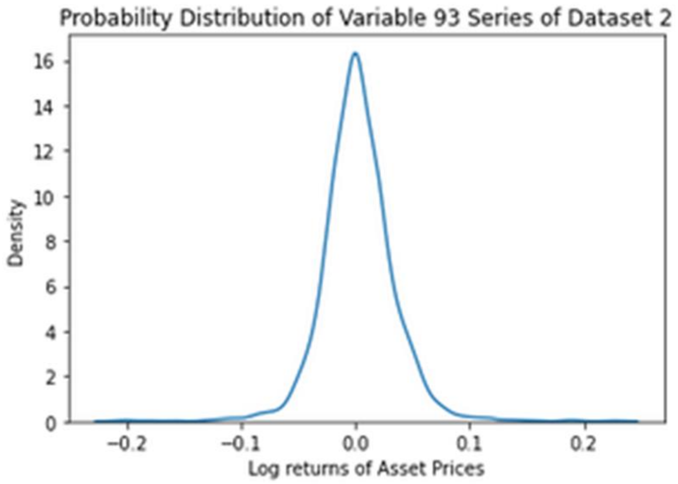
i.

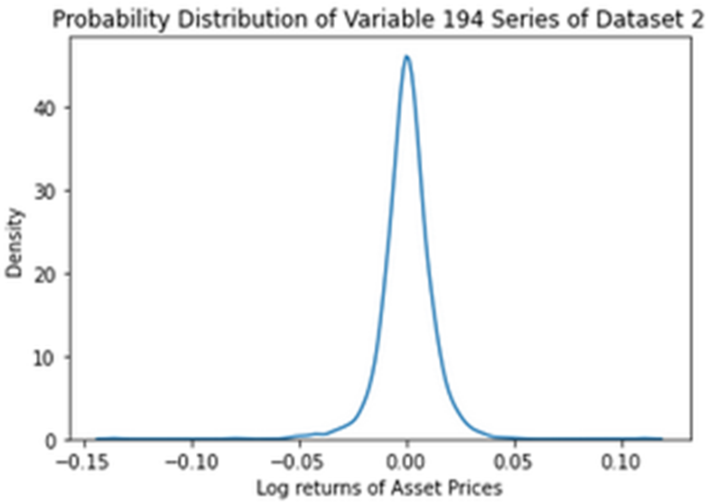


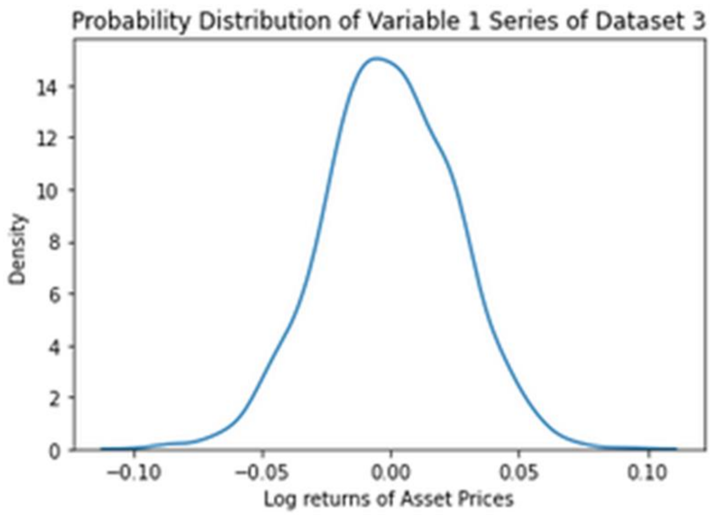



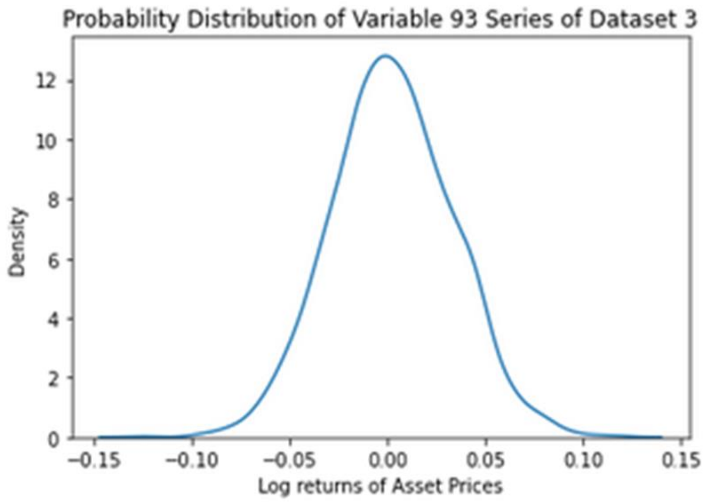



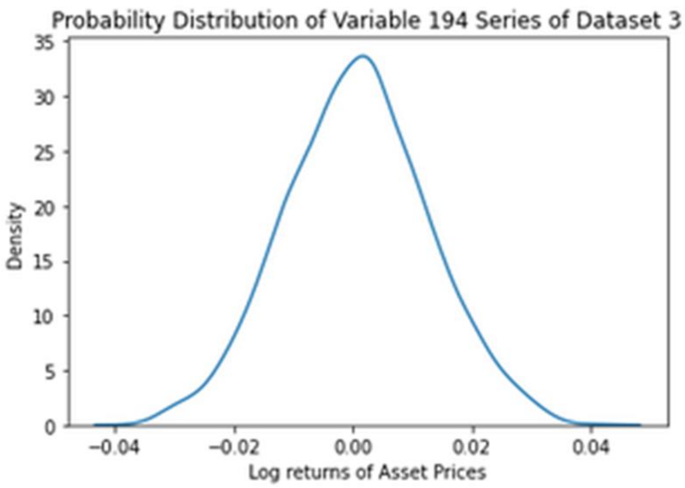

Probability Distribution of Variable 1 Series of Dataset 2
Density
Log returns of Asset Prices


Probability Distribution of Variable 93 Series of Dataset 2
Density
Log returns of Asset Prices


Probability Distribution of Variable 194 Series of Dataset 2
Density
Log returns of Asset Prices


Probability Distribution of Variable 1 Series of Dataset 3
Density
Log returns of Asset Prices
0
2
4
6
8
10
12
14
−0.10
−0.05
0.00
0.05
0.10


Probability Distribution of Variable 93 Series of Dataset 3
Density
Log returns of Asset Prices
0
2
4
6
8
10
12
−0.15
−0.10
−0.05
0.00
0.05
0.10
0.15


Probability Distribution of Variable 194 Series of Dataset 3
Density
Log returns of Asset Prices
0
5
10
15
20
25
30
35
−0.04
−0.02
0.00
0.02
0.04

b) Hurst Exponent of r(t)

```
hc_1_0, _, _ = compute_Hc(df1[0], kind = 'change',simplified = True)
print("Hurst Component of 1st variable in Dataset 1: ", hc_1_0)
hc_1_92, _, _ = compute_Hc(df1[92], kind = 'change',simplified = True)
print("Hurst Component of 93rd variable in Dataset 1: ", hc_1_92)
hc_1_193, _, _ = compute_Hc(df1[193], kind = 'change', simplified = True)
print("Hurst Component of 194th variable in Dataset 1: ", hc_1_193)
```

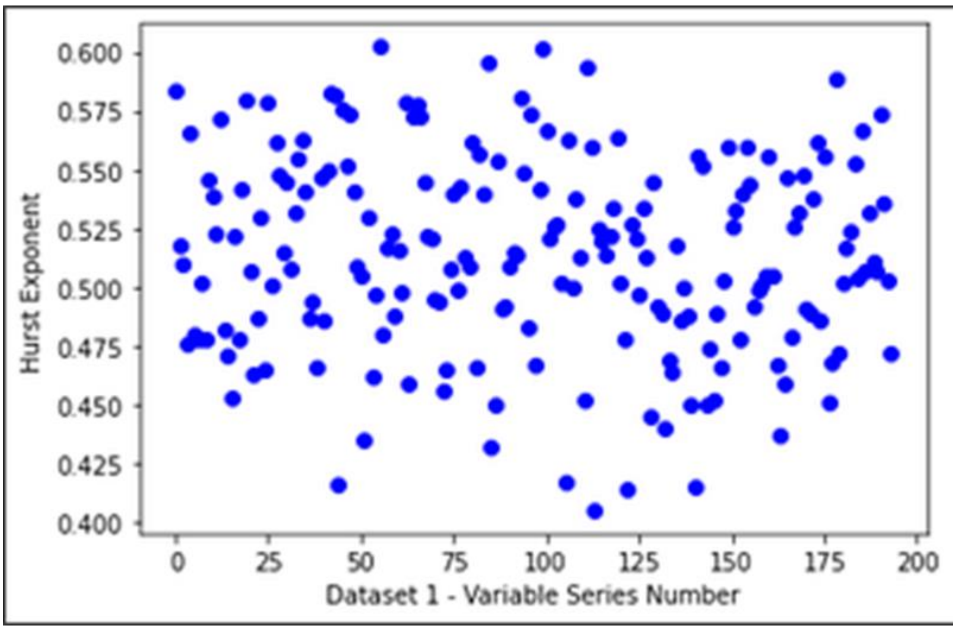
```
Hurst Component of 1st variable in Dataset 1:  0.5840960780126716
Hurst Component of 93rd variable in Dataset 1:  0.5139900098428936
Hurst Component of 194th variable in Dataset 1:  0.4717164739908118
```

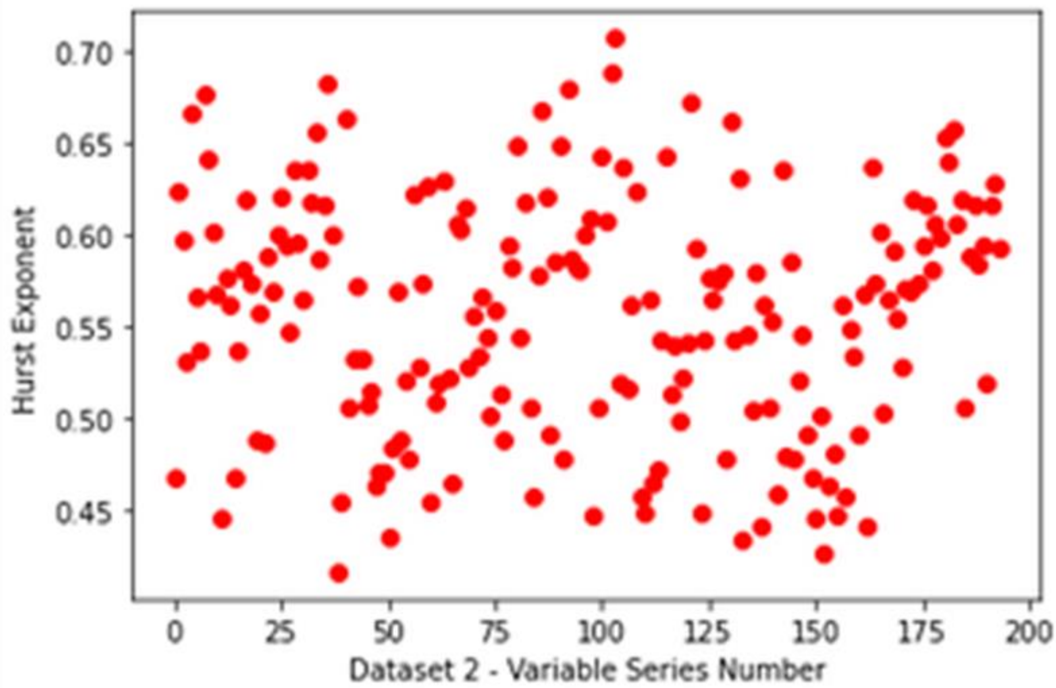
```
hc_2_0, _, _ = compute_Hc(df2[0], kind = 'change', simplified = True)
print("Hurst Component of 1st variable in Dataset 2: ", hc_2_0)
hc_2_92, _, _ = compute_Hc(df2[92], kind = 'change', simplified = True)
print("Hurst Component of 93rd variable in Dataset 2: ", hc_2_92)
hc_2_193, _, _ = compute_Hc(df2[193], kind = 'change', simplified = True)
print("Hurst Component of 194th variable in Dataset 2: ", hc_2_193)
```

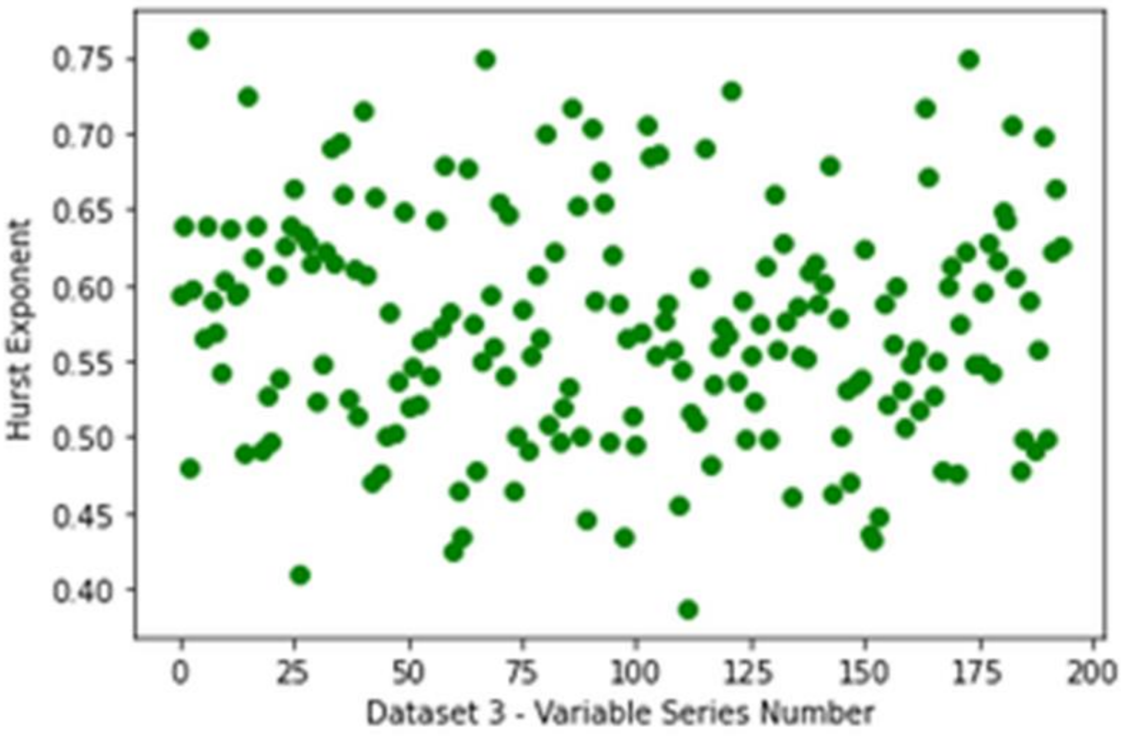
```
Hurst Component of 1st variable in Dataset 2:  0.46825434821055284
Hurst Component of 93rd variable in Dataset 2:  0.6788376239592012
Hurst Component of 194th variable in Dataset 2:  0.5933998207206423
```

```
hc_3_0, _, _ = compute_Hc(df3[0], kind = 'change', simplified = True)
print("Hurst Component of 1st variable in Dataset 3: ", hc_3_0)
hc_3_92, _, _ = compute_Hc(df3[92], kind = 'change', simplified = True)
print("Hurst Component of 93rd variable in Dataset 3: ", hc_3_92)
hc_3_193, _, _ = compute_Hc(df3[193], kind = 'change', simplified = True)
print("Hurst Component of 194th variable in Dataset 3: ", hc_3_193)
```

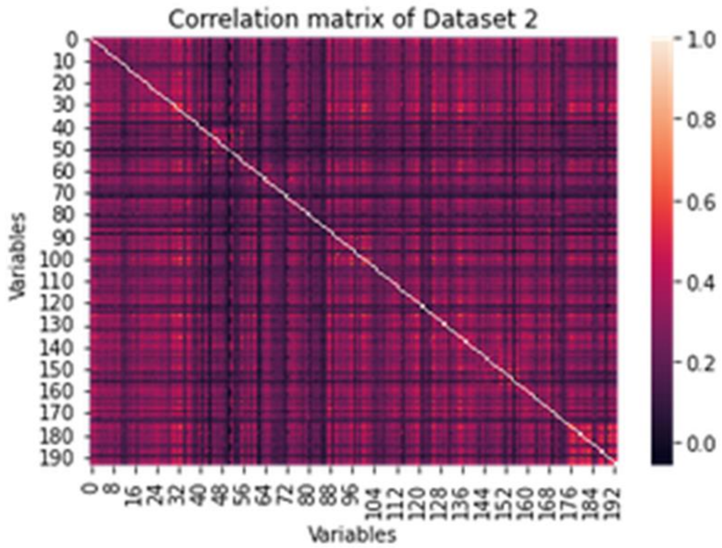
```
Hurst Component of 1st variable in Dataset 3:  0.5928936873420295
Hurst Component of 93rd variable in Dataset 3:  0.6757201968772152
Hurst Component of 194th variable in Dataset 3:  0.6267787383792501
```

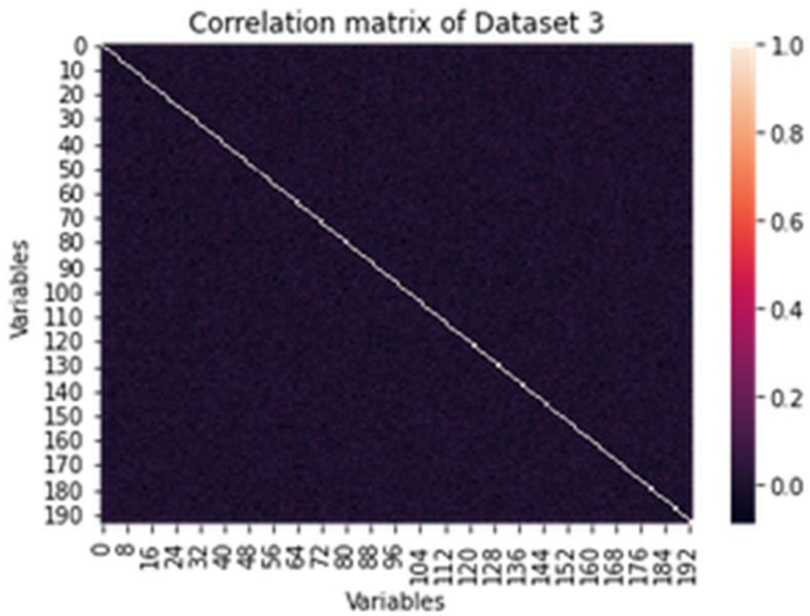



Hurst Exponent
0.70
0.65
0.60
0.55
0.50
0.45
0 25 50 75 100 125 150 175 200
Dataset 2 - Variable Series Number

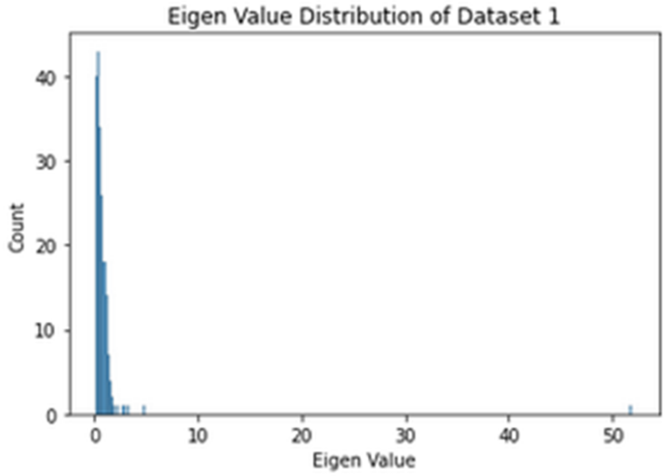

Hurst Exponent
0.75
0.70
0.65
0.60
0.55
0.50
0.45
0.40
0 25 50 75 100 125 150 175 200
Dataset 3 - Variable Series Number

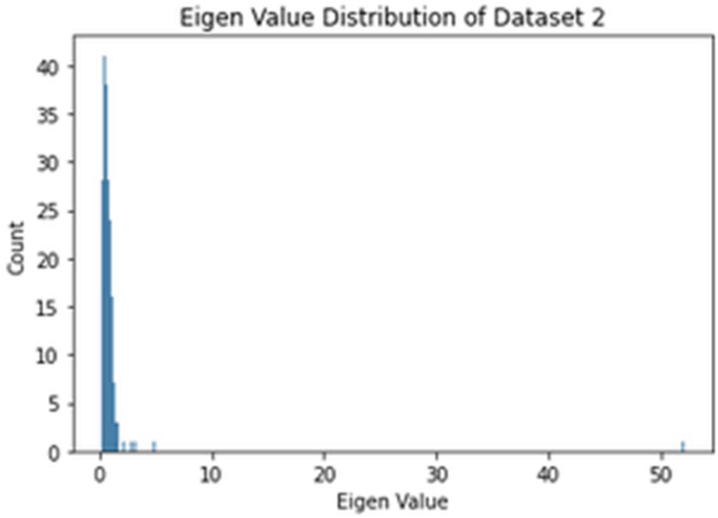
d) Pearson correlation matrix constructed from datasets.

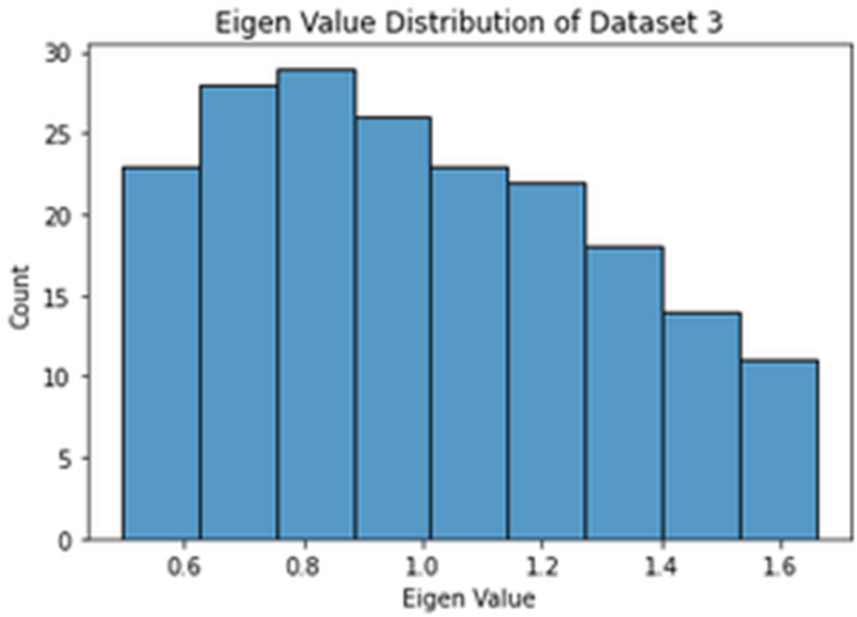



e) Distribution of eigen values of the Pearson correlation matrix.



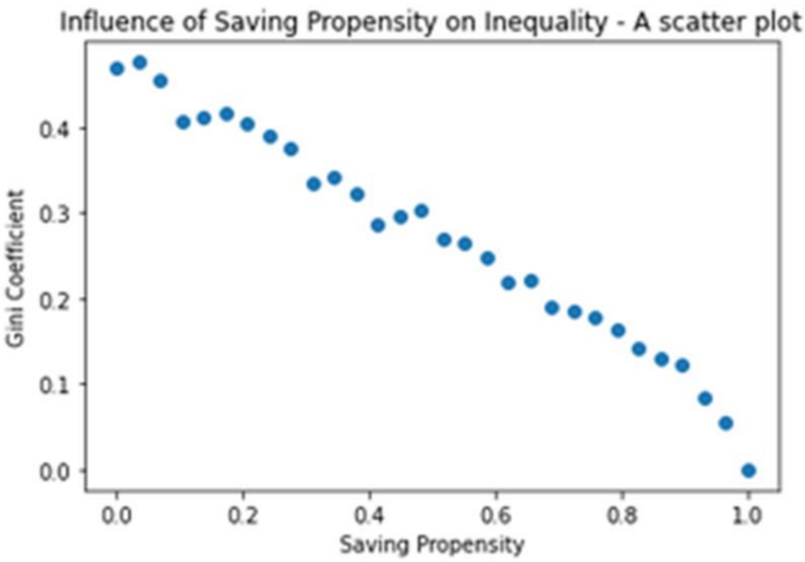
Analysis:

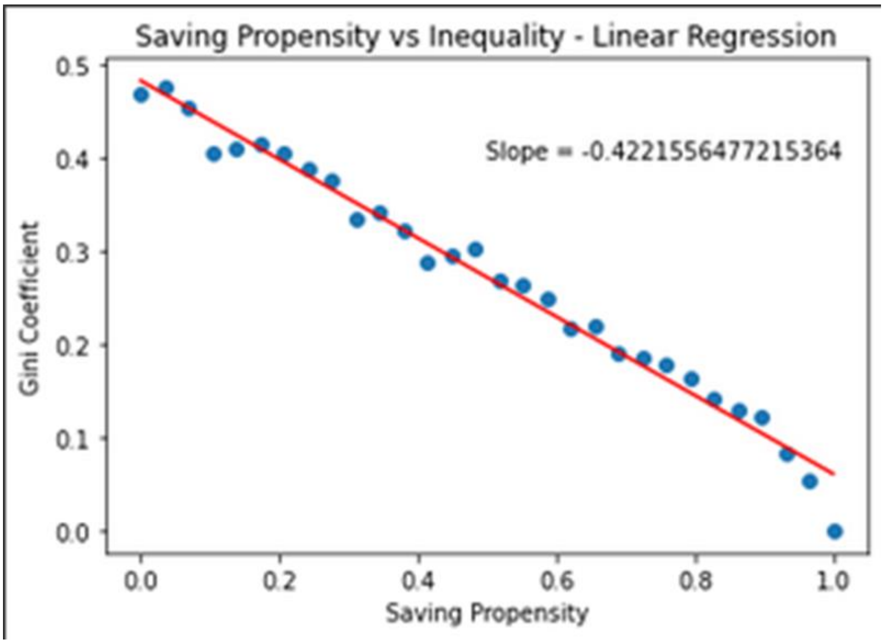
The probability distributions of the 3 datasets looks quite similar to each other and gaussian. Hence, it is not possible to classify the datasets based on the probability distributions.

Generally, if the Hurst exponent > 0.5, it is persistent/trending. If is < 0.5, then it is anti-persistent/mean-reverting. If it is 0.5, it is a random walk. However, the Hurst exponent of the 3 datasets vary too much from 0.4 to 0.6. Hence, no classification of datasets could be made using it.

Correlation matrix shows that the 3rd dataset has almost no correlation. So, it must be the uncorrelated Gaussian noise.

Similarly, we can analyse the distribution of eigen values. Eigen values reflect the proportion of variances of the underlying correlation matrix. In both dataset 1 and dataset 2, we see that a few eigen values have very large frequency. This implies that the variables share substantial common variance. Hence, dataset 1 and dataset 2 are correlated. However, in dataset 3, distribution of eigen values is spread across many far away eigen values which shows that there are not just a few large eigen values. This shows that there is no redundancy among the variables. So, as the eigen values are spread across a wide range of values, it means the variables do not share a great amount of variance. This shows that the variables are uncorrelated.

2. Effect of saving propensity on the inequality in wealth distribution
    a. Hypothesis: Inequality, hence the Gini coefficient decreases as saving propensity increases.
    b. Proof using linear regression
    c. 

d. 


e. Proof using Correlation coefficient – An almost equal to -1 correlation coefficient shows that inequality is inversely proportional to saving propensity.

## Testing hypothesis using Correlation Coefficient

```
[ ]
correlation_matrix = np.corrcoef(x,y)
print(correlation_matrix)
print('Correlation Coefficient = ', correlation_matrix[0,1])
```

```
[[ 1.         -0.99076783]
 [-0.99076783  1.        ]]
Correlation Coefficient =  -0.9907678288236917
```