Next steps:

Generate code with df

Simulacion de Monte Carlo

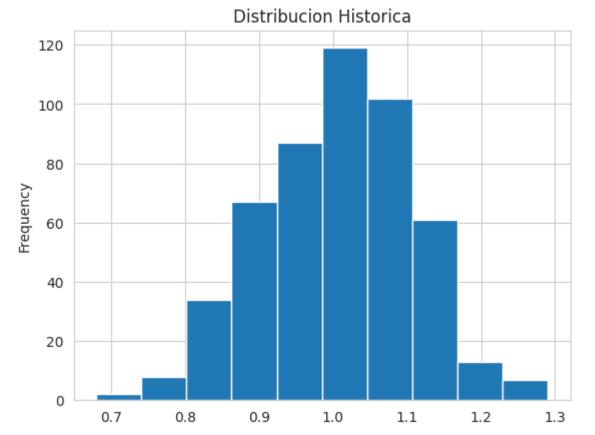
```
1 import pandas as pd
2 import numpy as np
3 import seaborn as sns
1 sns.set style('whitegrid')
1 # Definir las variables de porcentajes
2 \text{ avg} = 1
3 \text{ std dev} = .1
4 \text{ num reps} = 500
5 \text{ num simulations} = 1000
1 # Generar datos
3 pct to target = np.random.normal(avg, std dev, num reps).round(2)
1 pct_to_target[0:20]
   array([0.97, 1.05, 0.94, 0.82, 1.05, 0.85, 0.88, 1.1 , 1.08, 1.16, 1.04,
           1.05, 0.68, 0.95, 0.99, 0.99, 1.1, 1.01, 1.16, 1.01])
1 # Mas datos
2 sales target values = [75 000, 100 000, 200 000, 300 000 ,400 000, 500 000]
3 sales_target_prob = [.3, .3, .2 , .1, .05, .05]
4 sales_target = np.random.choice(sales_target_values, num_reps, p=sales_targe
1 sales_target[0:3]
   array([300000,
                   75000, 75000])
1 df = pd.DataFrame(index=range(num reps), data={'PCT To Target': pct to target'
2 df.head()
                                       \mathbf{H}
       PCT_To_Target SALES_Target
    0
                 0.97
                             300000
                                       ılı.
    1
                 1.05
                              75000
    2
                 0.94
                              75000
    3
                             500000
                 0.82
    4
                 1.05
                              75000
```

1 of 2 5/10/24, 15:57

View recommended plots

1 df['PCT_To_Target'].plot(kind='hist', title='Distribucion Historica')

<Axes: title={'center': 'Distribucion Historica'}, ylabel='Frequency'>



2 of 2 5/10/24, 15:57