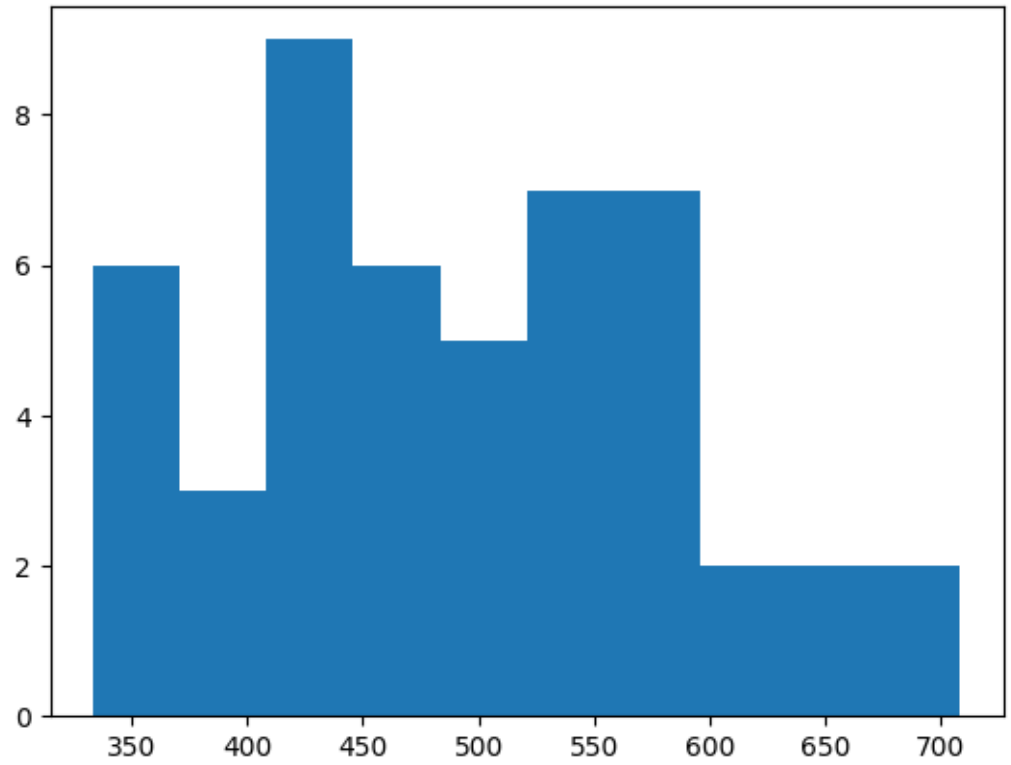
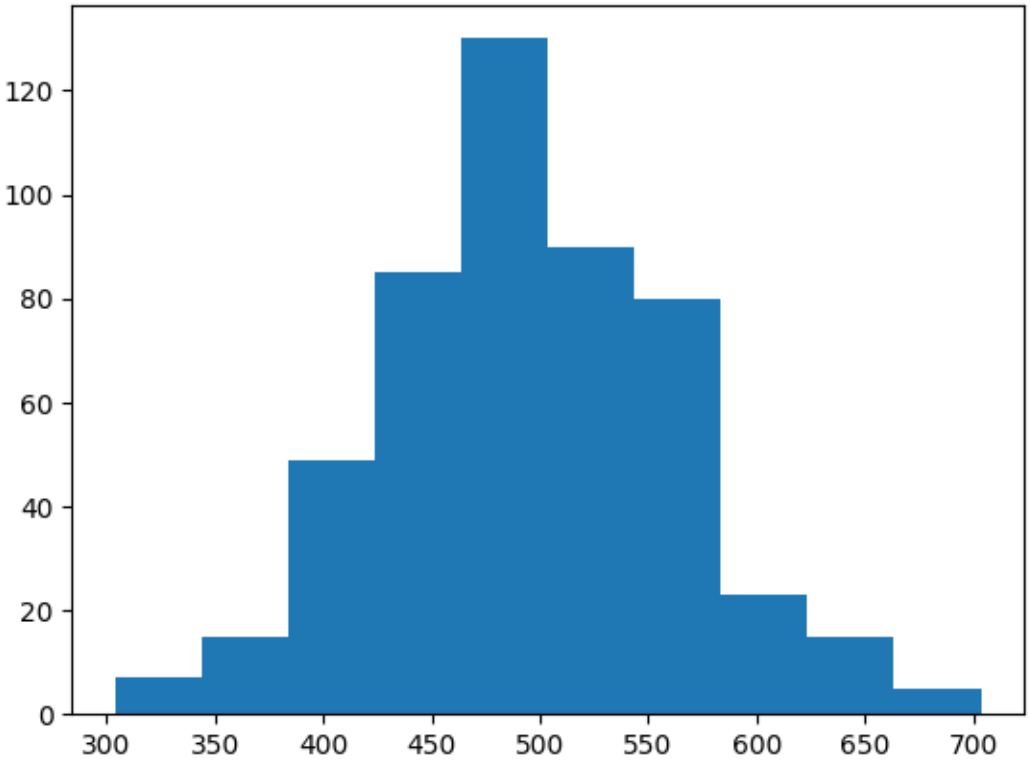
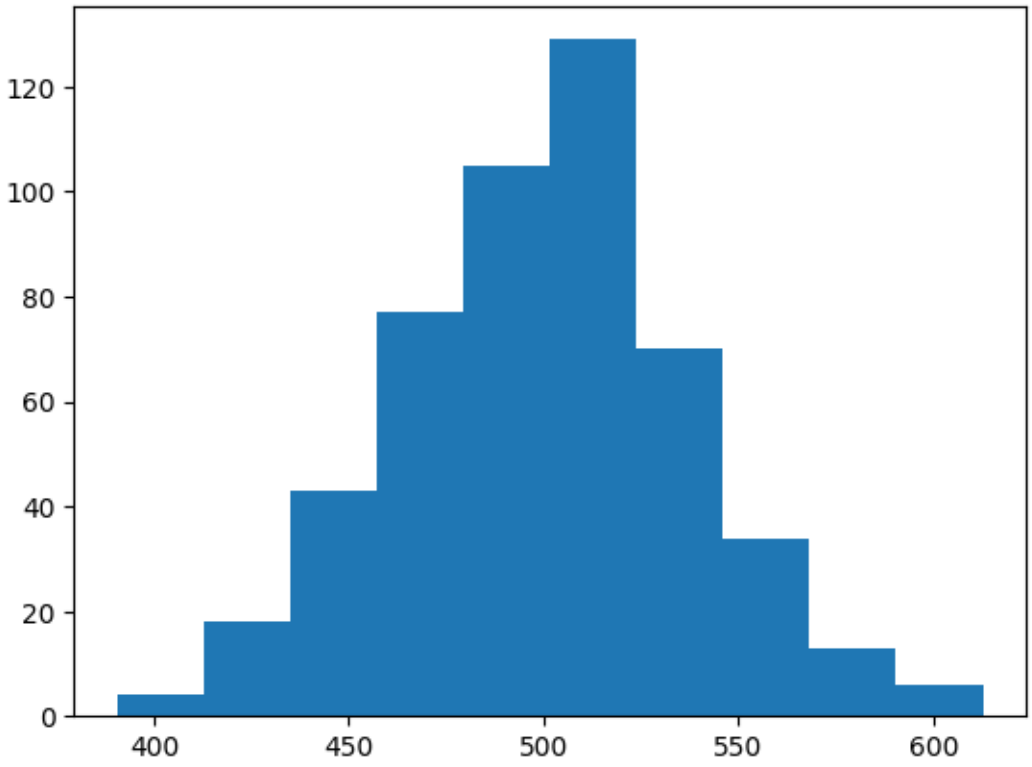
sampling-dist

November 1, 2024

[1]:
```
# Import necessary libraries
import numpy as np # For numerical operations (not used in this snippet)
import matplotlib.pyplot as plt # For creating plots
from random import sample # To sample elements from a list randomly
from statistics import mean # To calculate the mean of a list of numbers
# Define a function to plot the distribution of sample means
def plot(arr, N, n_samples):
    x = [] # Initialize an empty list to store sample means
    # Loop to take samples and calculate means
    for i in range(1, n_samples): # Iterate n_samples times (excluding the␣
 ↪first index)
        # To find N samples from the arr
        smp = sample(arr, N) # Randomly sample N elements from the array 'arr'
        mu = mean(smp) # Calculate the mean of the sampled elements
        x.append(mu) # Append the calculated mean to the list x
    plt.hist(x) # Create a histogram of the sample means
    plt.show() # Display the histogram
    # Example data (population)
arr = [i for i in range(1000)] # Create a list of integers from 0 to 999
# Variations of sampling and plotting
plot(arr, 5, 50) # Plot sample means for 50 samples of size 5
plot(arr, 20, 500) # Plot sample means for 500 samples of size 20
plot(arr, 50, 500) # Plot sample means for 500 samples of size 50
# Explanation of the observed results:
# As the number of samples (n_samples) increases, the distribution of the means␣
 ↪tends to become normal
# As the sample size (N) increases, the spread (flatness) of the distribution␣
 ↪decreases, indicating more precision in the estimates
```


0
2
4
6
8
350
400
450
500
550
600
650
700


0
20
40
60
80
100
120
300
350
400
450
500
550
600
650
700


120
100
80
60
40
20
0
400
450
500
550
600