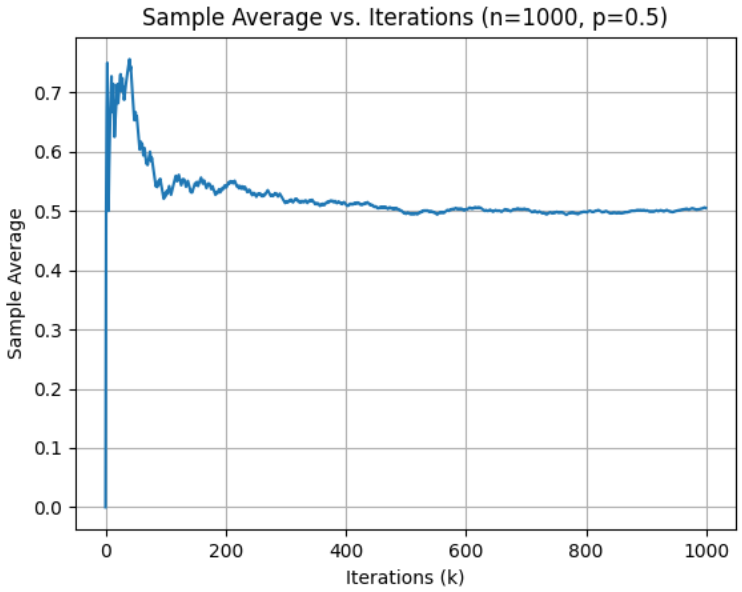
Data-driven Modelling - Machine Learning

Programming Exercise 1: Probability Theory and Statistics

Problem: 1a) A fair coin is tossed n times. Simulate a Bernoulli random variable with success probability p = 0.5. At each iteration k, compute a sample average of all k sampled elements. Afterwards, produce a plot of the average vs iterations. According to the law of large numbers, the sample average approaches the mathematical expectation, as n → ∞. Take n = 10^3.

Given: p=0.5, n = 10^3

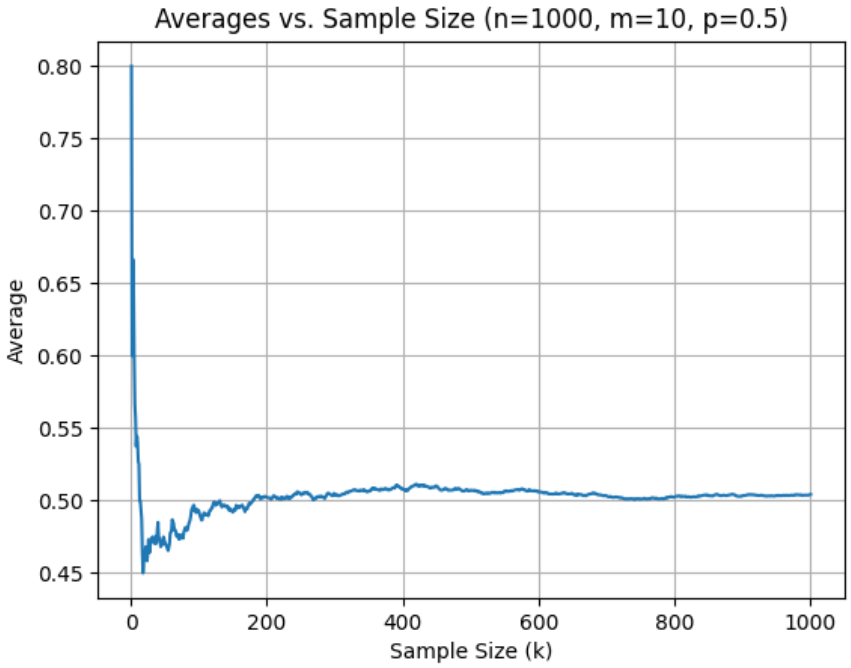
Inference: The value of the average approaches 0.5 as the number of iterations approaches a large value (n → ∞), as predicted by the law of large numbers. This is depicted in the Average vs Iterations plot below.

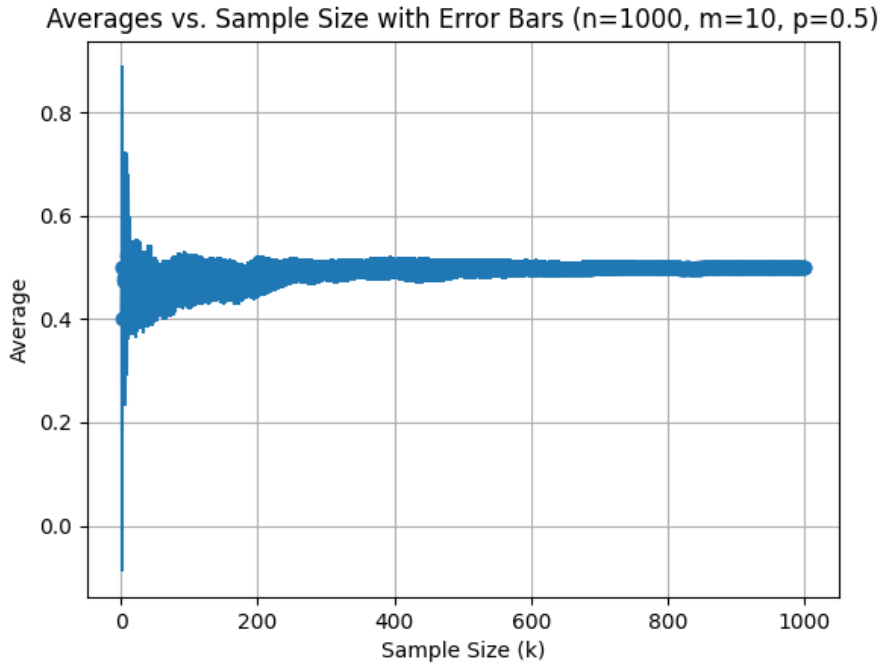


Problem: 1b) Make n following experiments. At k-th experiment (k = 1, . . ., n), draw k elements from a Bernoulli distribution m time, compute an average of k elements for each time, and save the result into a corresponding row of an n-by-m matrix (thus, you will have m averages in a k-th row). Afterwards, plot the results as the average vs the sample size using Matplotlib's plot and error bar functions. According to the law of large numbers, the deviation of the average should decrease with the increase of the sample size. Take n = 10^3, m = 10.

Given: p=0.5, n = 10^3, m=10

Inference: The value of the average approaches 0.5 as the sample size approaches a large value (n → ∞). Also with a larger sample size, the deviation of average (error) is lessened. Hence the result is as predicted by the law of large numbers. This is depicted in the average vs sample size plot below.



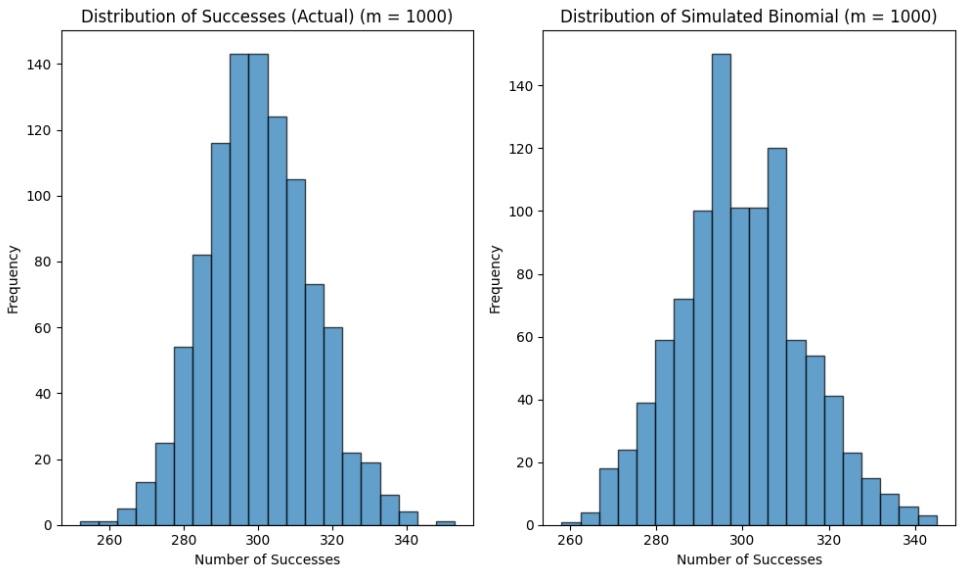


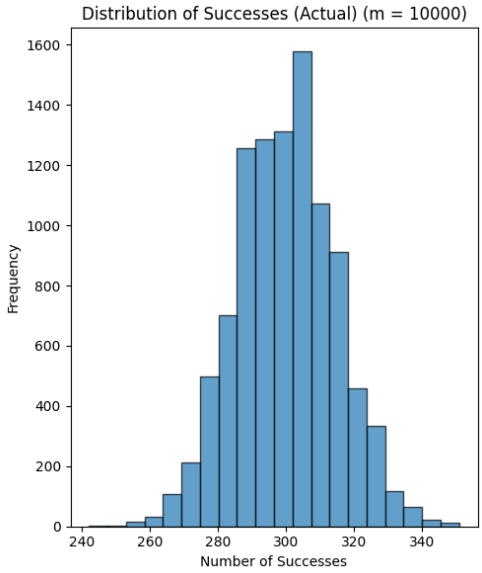
Problem: 2a) The sum of n independent Bernoulli random variables with success probability p has a binomial distribution with parameters n, p. Conduct the following procedure m times. Use the program from Problem 1 a., compute the number of successes for n tosses, and save this number into a corresponding row of an m-by-1 vector. After the procedure is done, draw a binomially distributed random variable m times with parameters n, p, and save it into another m-by-1 vector. For both vectors, plot the resulting probability distributions with the help of a histogram and compute the mean-square error of the difference. Use the following parameter values: n = 10^3, p = 0.3, m = {10^3, 10^4, 10^5}.

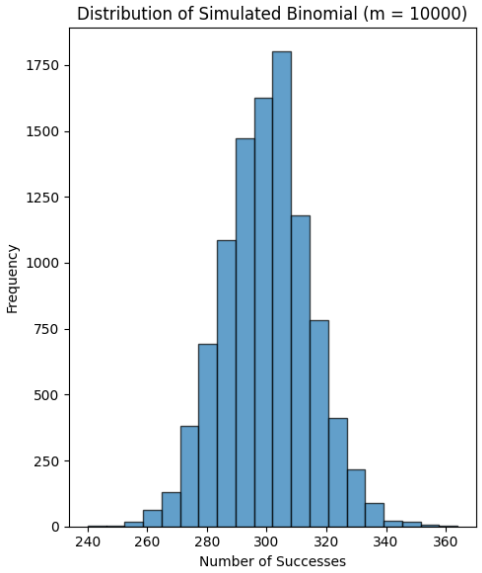
Given: p=0.3, n = 10^3, m = {10^3, 10^4, 10^5}

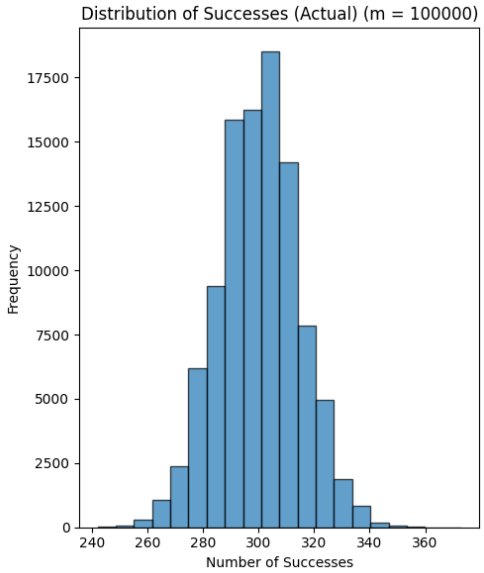
Inference: The resulting probability distribution is plotted as Number of successes vs frequency graph for the sum of Bernoulli random variables and Binomial Random variable vs frequency for the Binomial distribution. The plot of Binomial distribution closely resembles the plot of sum of independent Bernoulli random variables. The difference in the values (computed as mean-square error) was 425.588 when (m=10^3), 411.485 when (m=10^4) and 419.087 when (m=10^5).

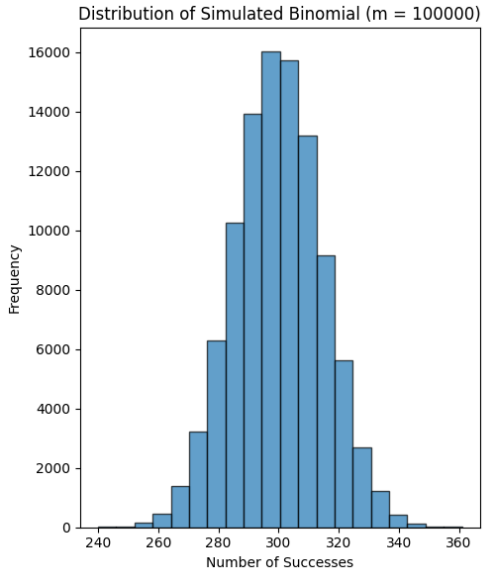
```
Running analysis with m = 1000
Mean Sample Average: 299.91261512799446
Mean Squared Error (MSE): 425.588

Running analysis with m = 10000
Mean Sample Average: 300.0398743144677
Mean Squared Error (MSE): 411.4852

Running analysis with m = 100000
Mean Sample Average: 300.0733591937479
Mean Squared Error (MSE): 419.08652
```







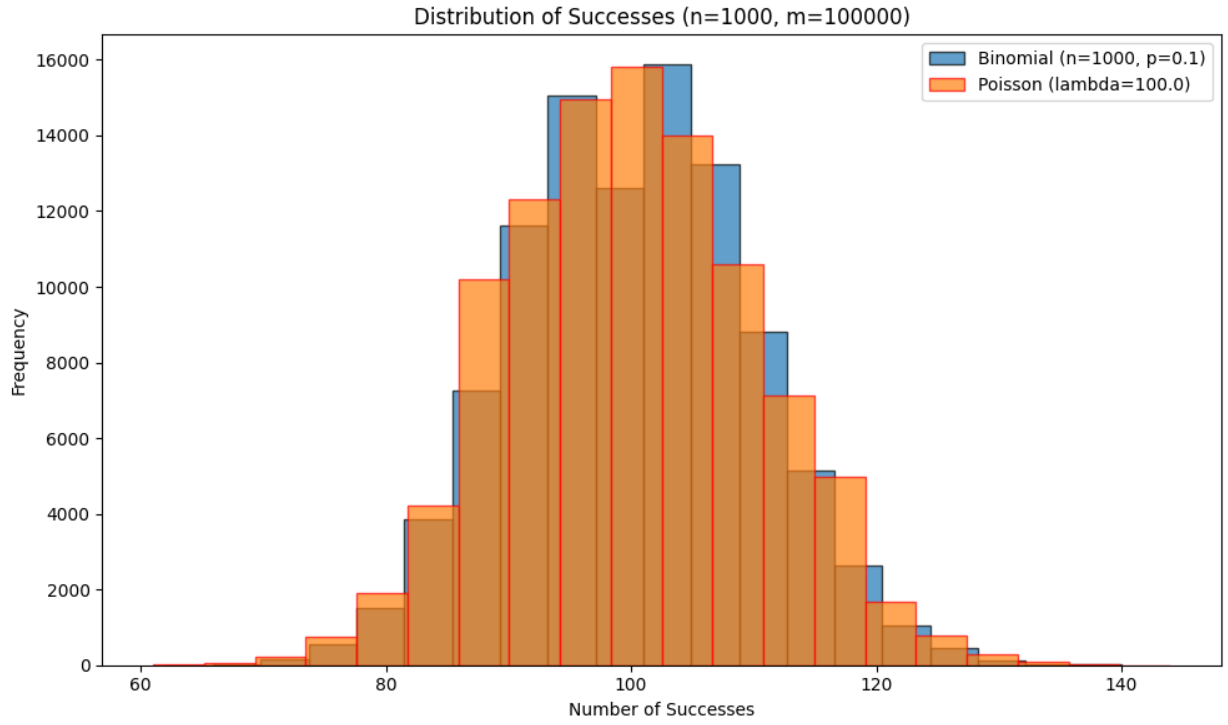


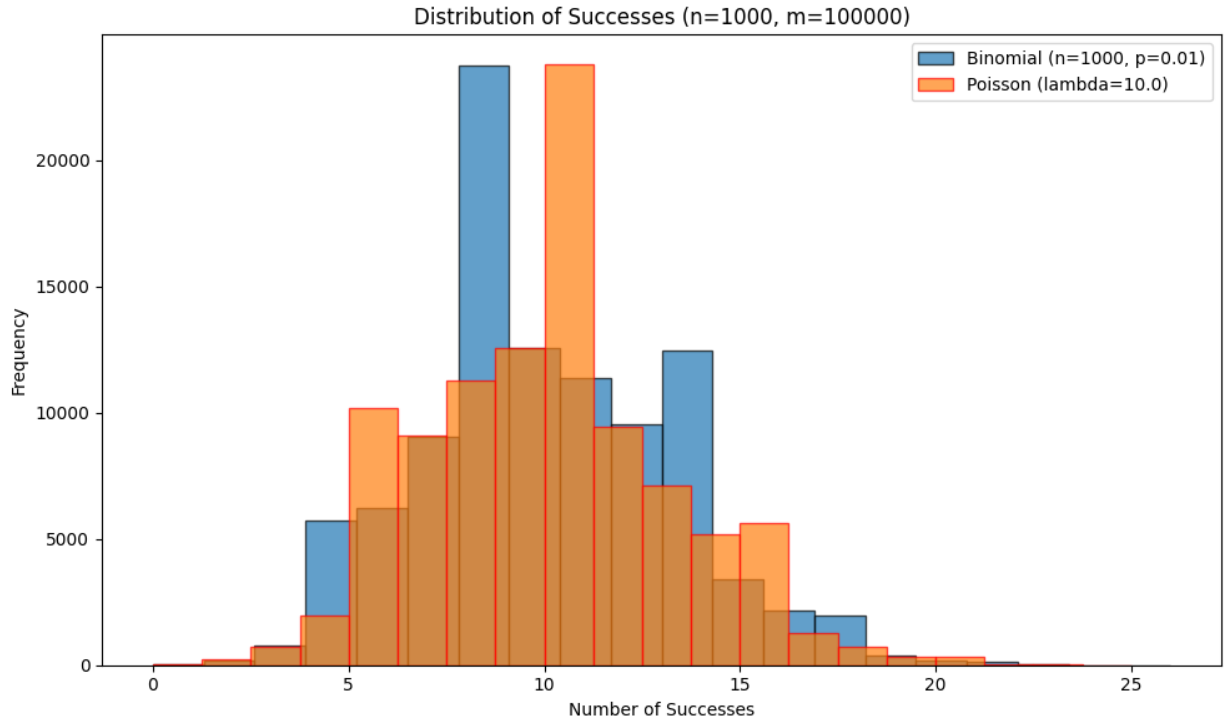


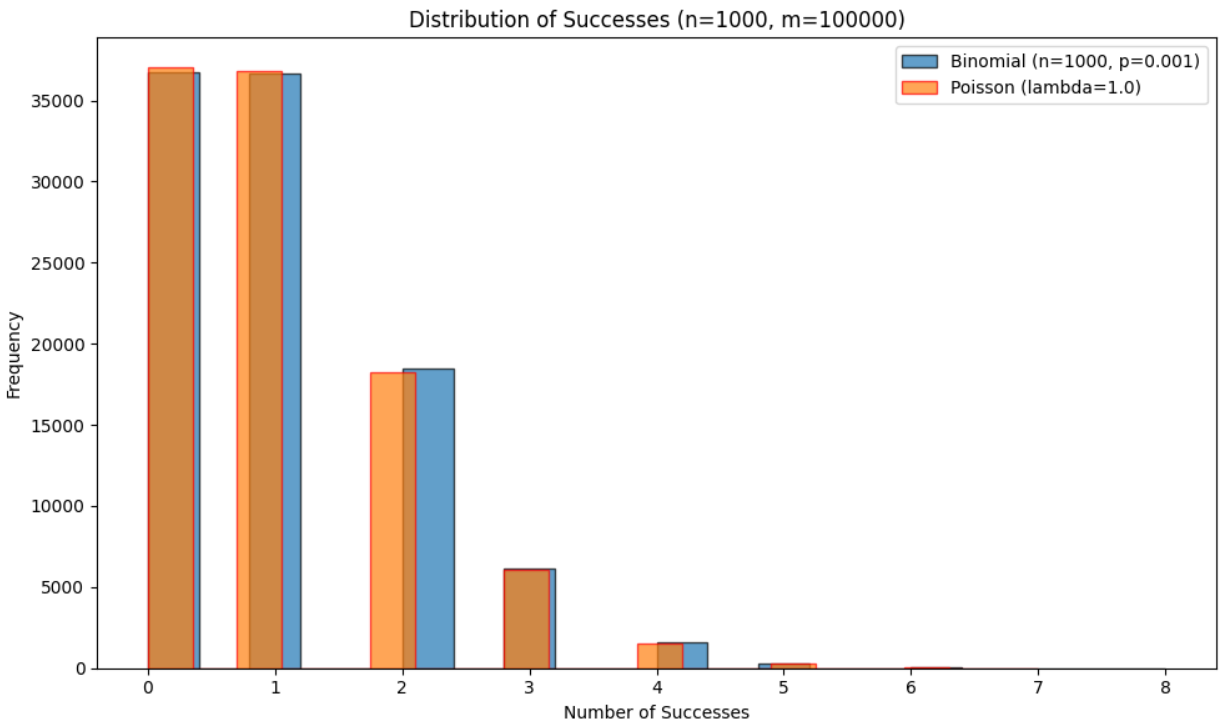
Problem: 2b) According to the Poisson limit theorem, as n → ∞ and p → 0, the Binomial (n, p) distribution approaches Poisson(np) distribution. Assume n = 10^3, p = {10^−1, 10^−2, 10^−3}, m = 10^5 and modify the previous program in order to compare Binomial and Poisson distributions.

Given: p = {10^−1, 10^−2, 10^−3}, m = 10^5, n = 10^3

Inference: The plot of Binomial distribution and Poisson distribution shows similar values as n $\rightarrow \infty$ and p $\rightarrow$ 0. This can be seen in the plot below of Binomial distribution and Poisson distribution.






Distribution of Successes (n=1000, m=100000)
Binomial (n=1000, p=0.001)
Poisson (lambda=1.0)
Frequency
Number of Successes
0
5000
10000
15000
20000
25000
30000
35000
0
1
2
3
4
5
6
7
8