Bernoulli and Binomial Distribution - Lab

Introduction

In this lab, you'll practice your newly gained knowledge on the Bernoulli and Binomial Distribution.

Objectives

You will be able to:

- Apply the formulas for the Binomial and Bernoulli distribution
- Apply NumPy to randomly generate Binomial and Bernoulli trials
- Use Matplotlib to generate Binomial and Bernoulli trials with various probabilities

Apply the formulas for the Binomial and Bernoulli distribution

When playing a game of bowling, what is the probability of throwing exactly 3 strikes in a game with 10 rounds? Assume that the probability of throwing a strike is 25% for each round. Use the formula for the Binomial distribution to get to the answer. You've created this before, so we provide you with the function for factorials again:

In [1]:
```
def factorial(n):
    prod = 1
    while n >= 1:
        prod = prod * n
        n = n - 1
    return prod
```

In [2]:
```
p_3_strikes = (factorial(10)/(factorial(7)*factorial(3)))*(0.25)**3*(0.75)**7
p_3_strikes
```

Out[2]:
```
0.25028228759765625
```

Now, create a function for the Binomial distribution with three arguments n, p and k just like in the formula.

In [3]:
```
def binom_distr(n,p,k):
    p_k = (factorial(n)/(factorial(k)*factorial(n-k)))*(p**k*(1-p)**(n-k))
    return p_k
```

Validate your previous result applying your new function.

In [4]:
```
binom_distr(10,0.25,3)
```

Out[4]:
```
0.25028228759765625
```

Now write a for loop along with your function to compute the probability that you have five strikes or more in one game. You'll want to use numpy here!

In [5]:
```
import numpy as np
prob = 0
for i in np.arange(5,11):
     prob += binom_distr(10,0.25,i)
```

Use a simulation to get the probabilities for all the potential outcomes

Repeat the experiment 5000 times.

In [6]:
```
np.random.seed(123)
n = 5000
iteration = []
for loop in range(n):
    iteration.append(np.random.binomial(10, 0.25))
    np_it = np.array(iteration)
```

In [7]:
```
values, counts = np.unique(np_it, return_counts=True)
print(values)
print(counts)
```

```
[0 1 2 3 4 5 6 7 8]
[ 310  941 1368 1286  707  297   78   11    2]
```

Visualize these results

Make sure to set an appropriate title and appropriate y-axis label

In [8]:
```
import matplotlib.pyplot as plt
plt.bar(values, counts/5000, align='center', alpha=0.5)
plt.xticks(values)
plt.ylabel('Fraction')
plt.title('Total number of strikes in a bowling game');
```

You can see that, with a 25% strike hit rate, even when simulating 5000 times, an almost perfect and perfect game of 9 and 10 strikes didn't even occur once! If you change your seed, however, you'll see that occasionally perfect games will show up randomly.

Summary

Congratulations! In this lab, you practiced your newly gained knowledge on the Bernoulli and Binomial Distribution.