File - binomialPoisson.py

```python
# Python code for Poisson and Binomial distribution
#   Prof. Jonathan Eckstein

import numpy
import math


# Returns the distribution of a Poisson variable with mean "mean",
# truncated so its largest possible value is "maxVal".
#
# Uses that first term is exp(-M) and then
# Each successive term is previous one times M/i

def poisson(mean,maxVal) :
    dist     = numpy.zeros(maxVal+1)
    probLeft = 1.0
    term     = math.exp(-mean)
    for i in range(maxVal) :
        dist[i]   = term
        probLeft -= term
        term     *= (mean/(i + 1))
    dist[maxVal] = probLeft
    return dist


# Returns the binomial distribution with n trials and p chance of
# success per trial
#
# Uses symmetry of the binomial coefficients (n choose k) and (n choose n-k)
# Only loops up through floor(n/2) and computes two terms per iteration
# Computes each binomial coefficient based on the previous one

def binomial(n,p) :
    dist = numpy.zeros(n+1)
    factorialTerm = 1
    k = 0
    while True :
        # Following two statements are redundant when k == n/2, but no matter
        dist[k]   = factorialTerm*(p**k)*(1-p)**(n-k)
        dist[n-k] = factorialTerm*(p**(n-k))*(1-p)**k
        factorialTerm *= (n - k)
        k += 1
        if k > n/2 :
            break
        factorialTerm /= k
    return dist

```