**Lab ­Statistics for Astronomical Applications**

**By Via Nielson**

**29 September 2020**

**Abstract**

We conduct numerical experiments to test the frequentist technique against the analytic predictions of both the binomial and Poisson distribution types for a coin flip simulation. This was accomplished by first writing a simple function to numerically calculate the binomial probability from multiple “coin flip” experiments where the probability of 2 discrete outcomes are equal. Then, the computer simulation of a large number of trials was compared to the exact analytical solution using the binomial PDF. Lastly, we considered a binomial PDF where the probability of success is very small and a large number of trials are used; this was finally compared to the exact solution using the Poisson PDF. We find, upon overlaying various predictions on the numerically calculated, that there is little to no variation among models. We conclude that the frequentist technique is a very sufficient method of modeling results.

**Introduction**

The theory that is being tested through the numerical experiments is the frequentist statistical theory. The theory acts as a philosophical interpretation of probability. It states that “as the number of trials increases, the change in the relative frequency will diminish. Hence, one can view a probability as the limiting value of the corresponding relative frequencies” [1]. This simply means that the probability of some event, (for the purposes of these experiments, this means either a head or a tail) is defined by the frequency of that event based on previous observations.

The motivation for this work, then, is to test this frequentist theory against the known analytic solutions for different types of distributions; namely, the binomial and Poisson probability distributions. All approaches allow for the interpretation of evidence for probability claims, however, the frequentist approach only treats random events probabilistically whereas the Poisson and binomial approaches define probability distributions over possible values of a parameter. Therefore, the overall goal of these experiments is to either prove or disprove that averaging over a large set of M realizations is an adequate model for known probability distribution curves of both a fair and unfair coin toss.

**Theory**

The binomial distribution, defined as , is one of the main equations used in these experiments. This traditional equation for the binomial distribution was determined in a proof written by Swiss mathematician Jakob Bernoulli in 1713. It is used as shown to produce a probability of an independent event happening a certain way for any fixed number (n) of coin flip trials that produces a certain outcome with the same probability with each run.

To derive this formula, we first take n to be the number of coins. If we toss the first coin, then the probability of having the second coin flip result in a heads is now impacted by (n − 1) instead of n. Finally, flipping the last coin we see it is impacted by (n − x + 1). Thus there are n(n−1)(n−2)...(n−x+1) variations in determining what x turns out to be. This becomes the following:

However, we can then divide by x! to get rid of repetition in the coin flips and turn them into events independent of each other. The statistical representation of this idea is included:

The probability that we should observe x heads and (n-x) tails equals the value multiplied by the probability that each of the x coins are heads, , and the probability that (n-x) coins are tails . Considering the manipulation of p (e.g. an unfair coin), we can derive the second half of the equation for the binomial distribution. If, for the purposes of this derivation, we have

p(H) = a and p(T) = b,

then to determine p(exactly x H in n flips), we must consider an example where the coin toss result in 2 heads and 4 tails out of 6 flips:

H H T T T T = a\*a\*b\*b\*b\*b =

Therefore, and so .

To generalize, p(exactly ‘x’ heads in ‘n’ flips) where p is equal to the probability of a head. The outcome is identical to the formula given for the binomial probability distribution in lecture and is therefore a valid derivation.

The other main equation used in these experiments is the Poisson probability distribution defined as . In the limit of small p, we can derive the Poisson probability distribution function from the binomial PDF as shown:

The Poisson equation, named after French mathematician Siméon Denis Poisson, is a distribution function similar to the binomial PDF and is useful for characterizing events with specifically low probabilities. It is a discrete function that describes the probability that an very improbable independent event will occur, but the number of trials is large enough that the event actually occurs a couple of times.

**Experiment and Methodology**

The first experiment conducted explored the binomial distribution for p = 0.5. I plotted the frequentist technique results in the form of a bar graph, and overplotted the analytical prediction of the binomial probability distribution function in the form of a line to compare (Figure 1).

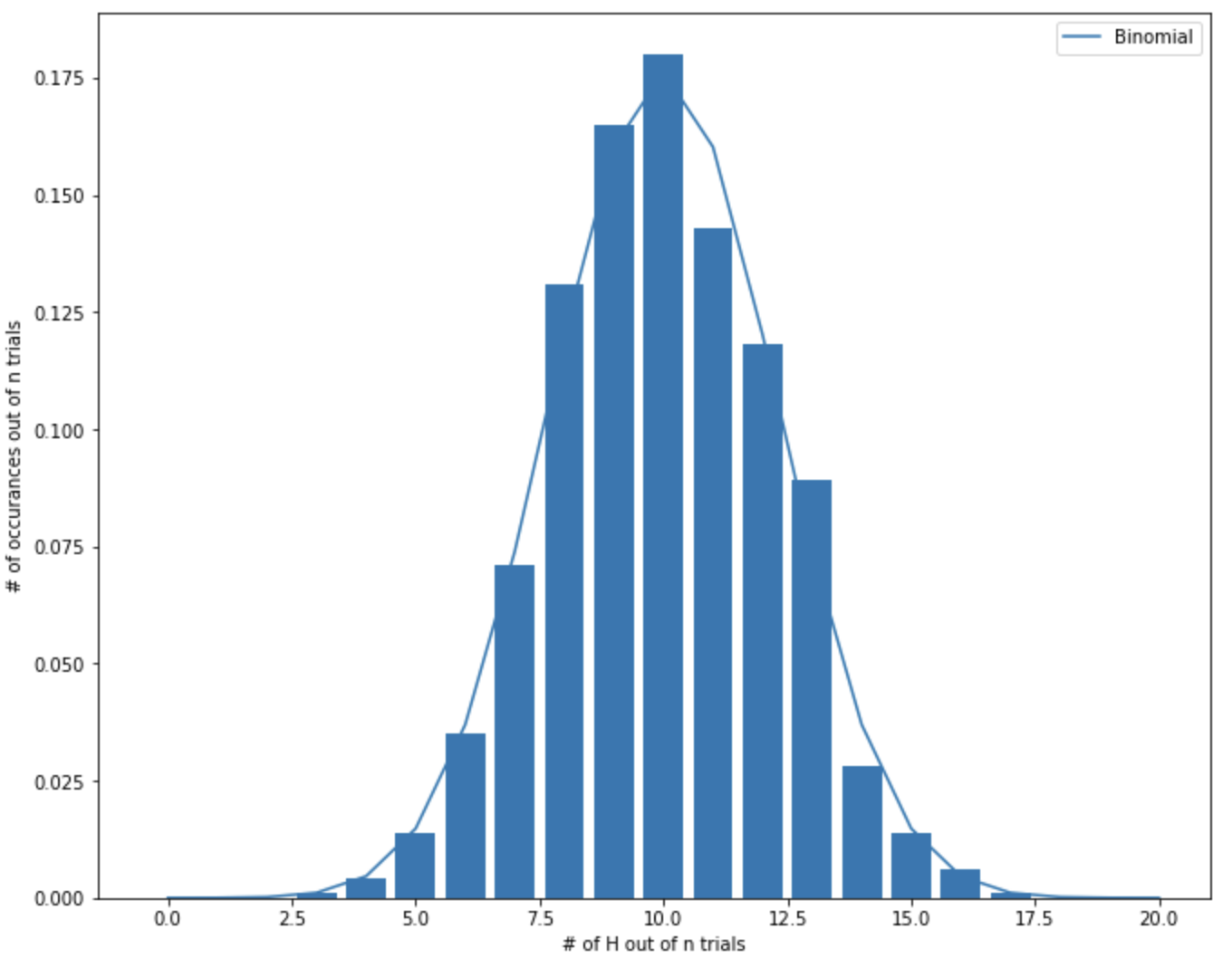
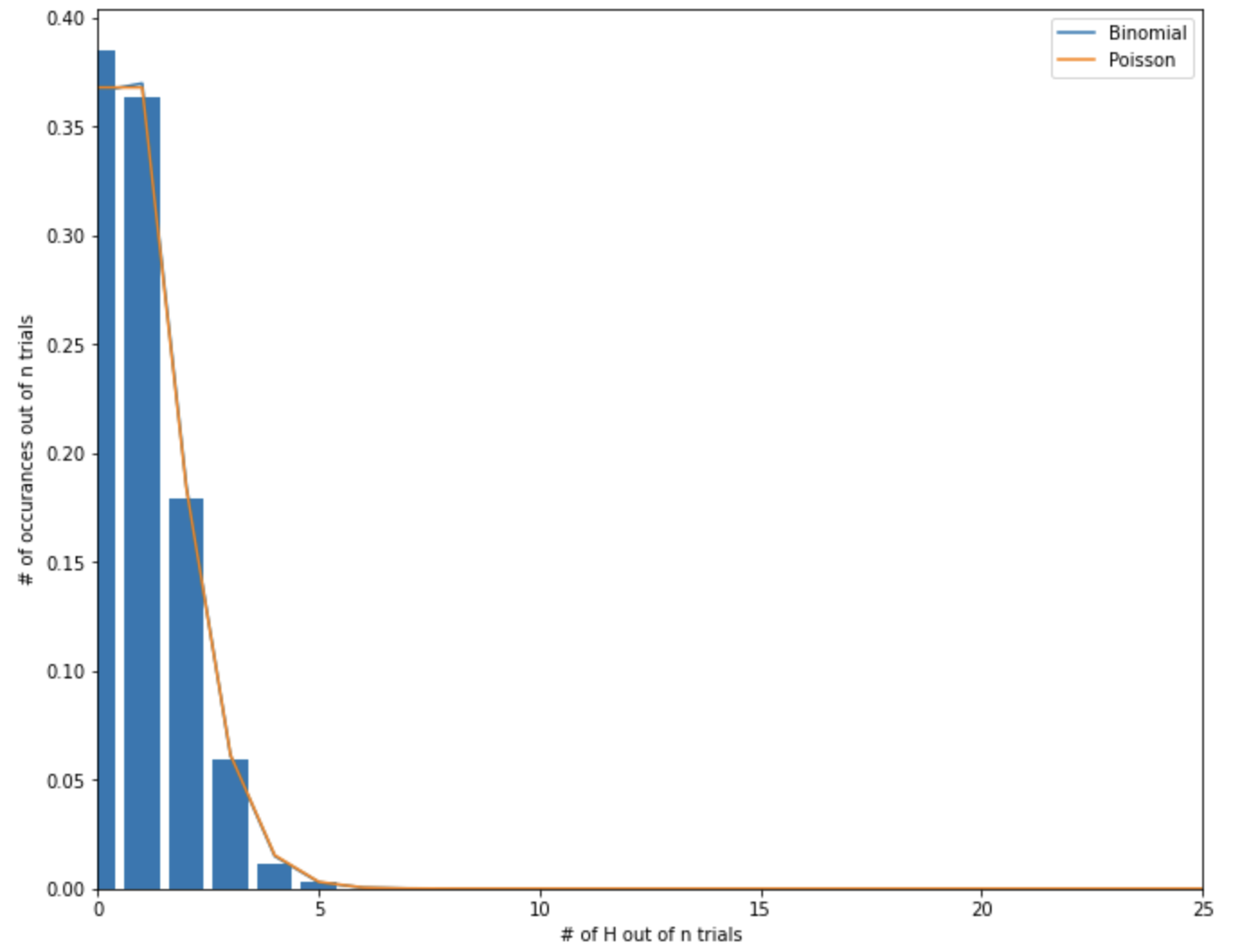
To begin my simulation, I defined three variables: N, x, and n. N represents the number of trials, n represents the number of flips per trial (same as number of coins), and x represents the number of successes out of N. Essentially, inside of the ‘for’ loop, the draw command loops through individual coins, flipping each. The ‘for’ loop itself works to flip those coins all a certain number of times, increasing the number of trials from 1 to N. So, 1000 times, twenty coins were flipped and every time the code drew a ‘1’ it would represent a head and every time it would draw a ‘0’ it would represent a tail. For each successful “draw of a head” I added to an array detailing how many occurrences there were of 0/20 heads in the first column, 1/20 heads in the second column, 2/20 heads in the third column, and so on. From there, I graphed the number of head successes over N to retrieve a bar graph to see the number of times each occurrence of the number of H out of n happened. I chose N to be as large as it is to have the graph appear as evenly distributed as possible. Upon observing both the frequentist strategized plot and then the analytical prediction of P(20,x), I observed the peak of the distribution to be at 10.0, meaning, there was on average 10 / 20 heads achieved per trial. This result is as I expected because n\*p = 20.0\*0.50 is equal to 10.0. The formula I used for the binomial distribution is identical to the one presented in class. 

Figure 1

*Exercise 1 graph depicting the frequentist data compared to the binomial distribution prediction. The data shows the amount of times a certain number of heads was obtained in the prescribed trial size.*

Statements that appear more than once in a program should generally be made into a function. You should use a function when you are going to call the same lines of code multiple times. It does not make sense to use a function if you only plan on executing the lines contained once. It also does not make sense to use a function if you do not have a clear idea of the inputs and outputs needed from it. For example, functions are useful when made from equations to be run over and over again with different inputs. But a function would not be useful for statements trying to set labels once or initialize a variable. I used a function P(N,x) to contain the formula for the analytical prediction of the binomial distribution, for instance. This is so that I can simply call the function in other cells so that I do not have to continue to rewrite the same lines of code multiple times.

The second experiment began to incorporate fine-tune features to allow for the eventual comparison of the binomial distribution and the Poisson distribution to the frequentist results. I began the simulation by re-writing my code for the experimental binomial probability to allow for a variable probability of success. To be able to control the value of p, the first change I made to the code was changing what constitutes a successful draw of a head. Instead of having the code draw either a 0 (T) or a 1 (H), I removed the draw for the tails and instead had heads\_count draw a decimal between zero and one. The probability, p, is defined before the loop. If the value of the draw is smaller than the given p value, then the program counts it as a successful draw of a head and continues on with the array as explained in the last exercise. To analyze these results, I overplotted both the analytic prediction of P(N,x) using the binomial probability formula and the analytic prediction of P(N,x) using the Poisson probability formula (Figure 2). The results were exactly as expected with the distribution centered on x = 1.0 because n\*0.01 = 1.0.

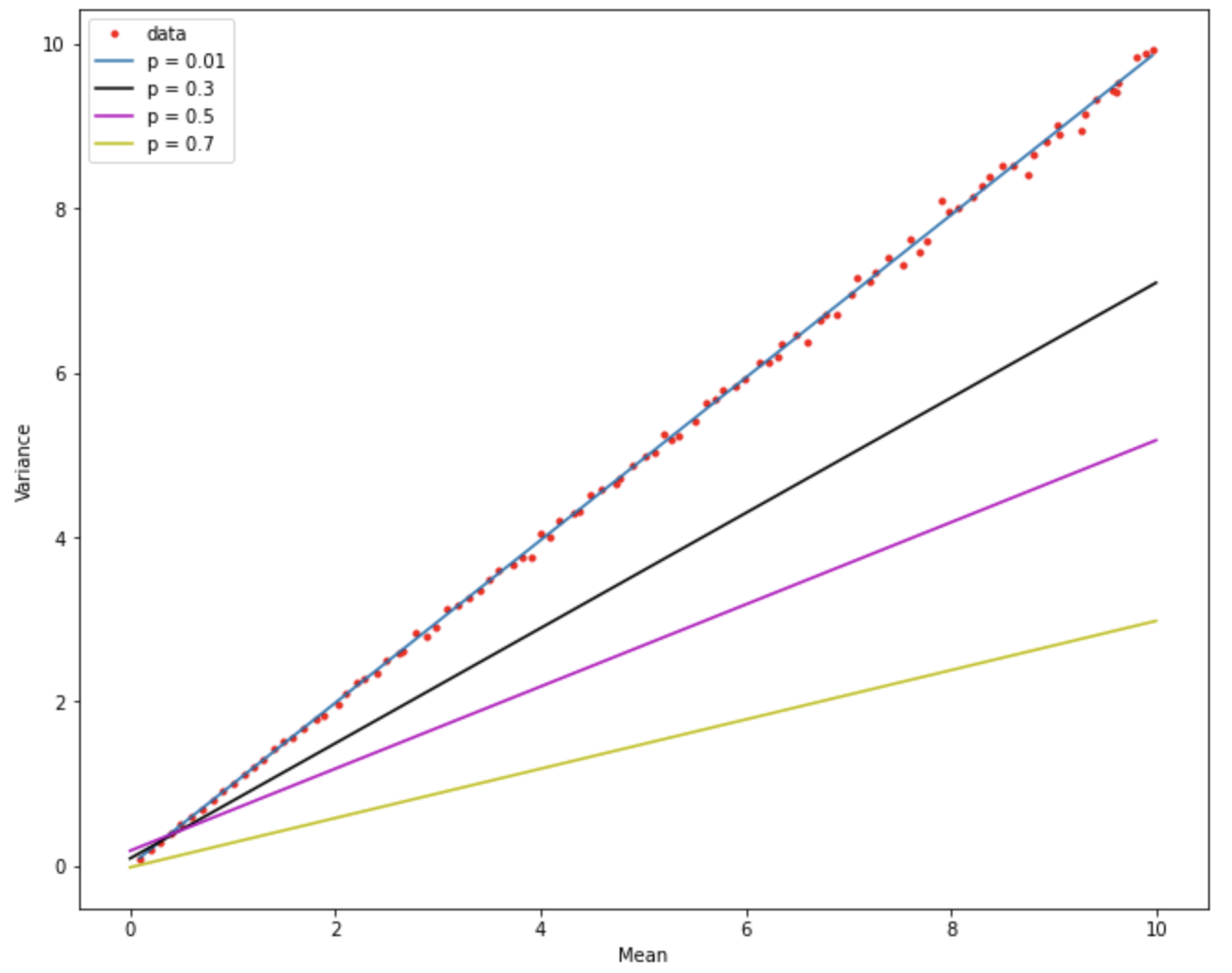
Figure 2

*Exercise 2 graph depicting the frequentist data compared to both the binomial and Poisson distribution predictions.*

My empirical histogram (Figure 2) of P(100,x) for small p (p = 0.01 in this exercise) and the exact prediction using the analytical binomial probability distribution correspond very neatly. The curve tends to move right through the center line of each bar in the graph, which is indicative of a great fit. The prediction of the analytical Poisson distribution follows suit. The Poisson distribution and the binomial distribution overlap almost exactly except for a small, consistent variation between x = 1 and x = 2. Inside of this interval, the binomial distribution curve reaches a slightly larger y-value than the Poisson curve.

Using the simulation from exercise two for small p, the third exercise explored Poisson Noise. Setting N, the number of trials, to 10,000, I hoped to obtain the most accurate results possible. To begin the simulation, I calculated the mean counts detected for a range of n, the number of coins being flipped per trial. To accomplish this, I included a ‘while’ loop just outside of the ‘for’ loop that has been described above. The ‘for’ loop was slightly modified to have the array newArr store the successes, but then after that I instead filtered the mean and variance from newArr into their own respective arrays, ‘mean’ and ‘var’ for easier calculations. Because the ‘for’ loop flips a certain number (n) of coins, a certain number (N) of times, adding a ‘while’ loop just outside of it means the program now runs a number of trials for each new ‘n’ the user decides on. I began with n = 10 coins and incremented the variable by ten at the end of all the trials (at the bottom of the ‘while’ loop). This means I ran multiple trials for each n = 10, n = 20, n = 30, and so on all the way up until the program was flipping 1,000 coins. I then plotted (Figure 3) the variance (y-axis) vs the mean (x-axis) and fit a line to the curve using NumPy’s “polyfit” option.

My program found that the best fit line is y = 0.9889359854125808 x + 0.004097586047852299 for a p = 0.01. The graph (Figure 3) clearly shows that the variance increases as mean increases with a slope close to one, and the y-intercept is nearly zero. This result is exactly as expected. I found that my results depended heavily on the value of p. As long as p is relatively low, you obtain this result. However, as p begins to increase, the slope begins to decrease and the y-intercept increases. I plotted the fits obtained by the program for all of p = 0.01, p = 0.3, p = 0.5, and p = 0.7. The results show that for large p, the fit is poor. The theory is that for the Poisson probability distribution, the variance should equal the mean for small p limits.

Figure 3

*Exercise 3 graph depicting the best fit line to the data plotted using the mean and its corresponding variance from each trial of varied coins flipped (n). This means the data points further to the left represent the mean and variance for small n and the data on the right depicts the data of larger n. The assorted lines represent the best selected fit for a range of p values.*

Poisson noise is then the amount of variation observed between the model and the actual data. Poisson noise may be dominant when the finite number of coin flips is sufficiently small so that uncertainties due to the Poisson distribution are of significance. For large numbers, the Poisson distribution approaches a [normal distribution](https://en.wikipedia.org/wiki/Normal_distribution) about its mean. Moreover, when N is very large, the signal-to-noise ratio is very large as well, and any relative fluctuations in N due to other sources are more likely to dominate over Poisson noise. However, when the other noise source is at a fixed level or grows slower than , increasing N can lead to dominance of Poisson noise [2]. Moreover, based on the derivation of the Poisson probability distribution, we see that Poisson distributions are unimodal, exhibit positive skew (that decreases as 𝝺 increases), are roughly centered on the value of 𝝺, and have variance that increases as 𝝺 increases [3]. Because 𝝺 = Np, we can expect a poorer fit from a higher p value, and this is exactly as we observe.

***Summary***

When using the frequentist technique to average over a large set of N realizations, we find that the analytical solution that the binomial probability distribution formula provides lines up nicely and provides a fairly accurate prediction for the data. N was chosen to be large to provide for a more accurate and even distribution.

In the limit of small p, we conclude that the Poisson probability distribution also provides an accurate prediction for the frequentist data. As shown in the graphs produced by the program, the two models line up almost exactly over the histogram of the obtained data. This result was numerically determined by altering the code to allow for a varying parameter p.

In the last exercise, altering the code once more to allow for the variation of variable n, we determined that Poisson noise increases and the fit determined by the numerically produced data worsens as the value of p increases. This is made evident by the derivation of the Poisson distribution from the binomial distribution as 𝝺 = Np. We find that the variance should equal the mean for small p limits, resulting in a slope of close to one and a y-intercept of near zero seen in the final graph.

In summation, we conclude that upon overlaying various analytical predictions on the data obtained by the frequentist technique, that there is little to no variation among models. We conclude that the frequentist technique is a very sufficient method of defining probability for events concerning the modeling of coin tosses. We conclude this because the graphs show that both distribution formulas used in these exercises are very accurate in modeling the frequentist results.

Works Cited

[1] von Mises, Richard (1939) Probability, Statistics, and Truth(in German) (English translation, 1981: Dover Publications; 2 Revised edition. [ISBN](https://en.wikipedia.org/wiki/ISBN_(identifier)) [0486242145](https://en.wikipedia.org/wiki/Special:BookSources/0486242145)) (p.14)

[2] “Shot Noise.” *Wikipedia*, Wikimedia Foundation, 19 June 2020, en.wikipedia.org/wiki/Shot\_noise.

[3] *Lecture 5: The Poisson Distribution*. 11 Nov. 2015, www.stats.ox.ac.uk/~filippi/Teaching/psychology\_humanscience\_2015/lecture5.pdf.