Project #4

Dylan Robertson
PHSX 815

May 10, 2023

Abstract

Shot noise is a source of noise in electronic equipment that arises due to the discrete charge of the electron and Poisson Fluctuations. This experiment ran simulations that estimated the value of the Poisson parameter based on data that was generated from a Poisson Distribution with a parameter of $\lambda = 5.00$. Each experiment had 10,000 measurements, and 100 experiments were run in total. The mean value of experiment 1 was estimated to be $\lambda = 5.027 \pm 0.022$, and the overall mean value from all experiments was estimated as $\lambda = 5.032 \pm 0.023$. This then allowed the charge of the average electron to be calculated as $q = 1.612 \pm 0.023\ 10^{-19}$ C

1 Introduction

The Poisson Distribution is described as follows.

$$P(X|\lambda) = \frac{\lambda^X e^{-\lambda}}{X!}, \tag{1}$$

where λ is a positive, real number, representing the average rate that an event occurs, and X is a whole number that represents the number of events that occurred within some fixed time interval.

The Poisson Distribution can be seen for various values of λ in Fig. 1. As λ gets sufficiently large, the distribution approaches that of a normal distribution, as predicted by the Central Limit Theorem.

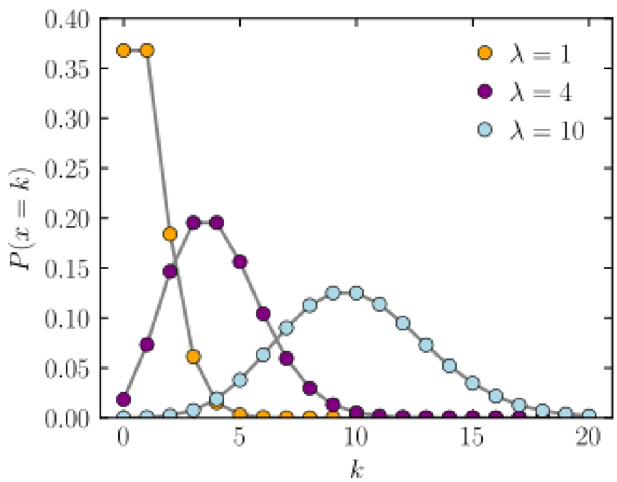

Figure 1: Poisson distributions with various values of λ. It will be shown that the expected peak in the distribution (the mean) occurs at the value of λ.

The likelihood of the above distribution is related by Baye's Theorem.

$$P(\lambda|X) \approx \prod_{i=1}^{N_{meas}} P(X_i|\lambda) = \prod_{i=1}^{N_{meas}} \frac{\lambda^{X_i} e^{-\lambda}}{X_i!} \tag{2}$$

where X_i is each individual measurement made in a single experiment and N_{meas} is the total number of measurements in the given experiment.

A more useful quantity is the logarithm of the likelihood, as it allows us to turn the product into a summation using properties of logarithmic functions. Going through the calculation results in the following expression.

$$ln(P(\lambda|X)) = -N_{meas}\lambda + ln(\lambda) \sum_{i=1}^{N_{meas}} X_i - \sum_{i=1}^{N_{meas}} X_i! \tag{3}$$

where the last term is of no importance for calculating the analytical solution since it does not depend upon our parameter λ that we are trying to estimate. However, the Stirling Approximation for the last term for the numerical analysis of the code. The final formula for the likelihood is the following.

$$ln(P(\lambda|X)) = \sum_{i=1}^{N_{meas}} [x_i ln(\lambda) - \frac{1}{2} ln(2\pi x_i) - x_i ln(x_i) + x_i - \lambda] \tag{4}$$

The most common way to estimate a parameter is to find the value of the parameter that maximizes the likelihood (or the log likelihood). For a Poisson Distribution, this can be done analytically by taking partial derivatives and setting them equal to 0.

Maximizing the log likelihood with respect to the parameter λ yields,

$$\lambda_{most-likely} = \frac{1}{N_{meas}} \sum_{i=1}^{N_{meas}} x_i \tag{5}$$

which is the mean value of the experiment, as expected from Fig. 1. This result will be used as a comparison for our numerical estimation.

2 Shot Noise

Shot Noise is a fundamental source of noise that arises in electronics due to the discrete nature of the electron's charge. This source of noise is most prevalent at low currents, and small timescales (wide bandwidths).

A current can be modeled as the flow of electrons, and the average current is given by,

$$I = \frac{Nq}{t}, \tag{6}$$

where N is the number of electrons that pass in a given time interval t, and q is the fundamental charge of the electron.

The root mean square fluctuations in the current can be shown to be on the order of,

$$\sigma_I = \sqrt{\frac{qI}{t}}. \tag{7}$$

Rearranging Eqns. 6 and 7 allows one to solve for the charge of an electron based upon a signal and it's variance.

$$q = \frac{t\sigma_I^2}{I}. \tag{8}$$

In reality, shot noise is just Poisson Fluctuations as current is simply the average flow of discrete packets of charge, called electrons, in a given time interval. The relationship between the parameter of a Poisson Distribution and the charge of an electron is the following.

$$\lambda = Nq = It, \tag{9}$$

showing that the physical interpretation of the rate parameter λ is the total amount of charge delivered in a given time interval.

Performing error analysis on Eqn. 9 yields the following relationship between the uncertainties,

$$\frac{\sigma_\lambda^2}{\lambda^2} = \frac{\sigma_I^2}{I^2} + \frac{\sigma_t^2}{t^2}, \tag{10}$$

where each σ_i is the uncertainty in each variable.

For a small current of $I = 100$ mA, the average rms flucutation in a time interval of $t = 1.00 \pm 0.01$ s is $\sigma_I \approx 0.1$ nA. which gives the charge of the electron being on the order of $q = 1 * 10^{-19}$ C, as expected. Plugging these values into Eqn. 10 lets us know that we should expect our uncertainty in our estimated parameter to be on the order of $\sigma_\lambda = 0.01 * \lambda$.

3 Experimental Procedure

Our first section gave us an introduction to the Poisson Distribution and how to analytically estimate the parameter λ using the maximum likelihood method. Now, we will randomly generate data from this distribution, and then analyze the data to numerically estimate the parameter.

This experiment first simulated random draws from a Poisson Distribution with a rate parameter $\lambda = 5.00$. There were $N_{meas} = 10,000$ measurements per experiment, and $N_{exp} = 100$ experiments ran in total.

A couple of important things should be noted about the above information. First, a rate value of $\lambda = 5.00$ means that the the number of measurements per experiment should be $N_{meas} = 3.125 * 10^{19}$ (from Eqn. 9) in order to measure the charge of the electron accurately. Second, if we are using a current of $I \approx 100$ mA, then we are using a time interval of $t \approx 50$ s. While not the ideal conditions to do this simulation under, my computer can't handle that many measurements so we will instead add the following correction factor and use the numbers listed above.

$$q = \frac{N'}{N} q', \tag{11}$$

where q' is the estimated value of the electron's charge from the simulation using Eqn. 9, q is the actual value for the electric charge, N' is the number of measurements per experiment used in the simulation, and N is the number that should have been used.

The data was generated using **Poisson.py** and was then graphed using **PoissonPlot.py**. The graph of this data, and the corresponding Poisson curve can be seen in Fig. 2.

A random measurement from the Poisson Distribution was done using a standard function from the numpy library.

```
#Poisson Distribution
def Poisson(self, lamb):
    return np.random.poisson(lam=lamb, size=1)
```

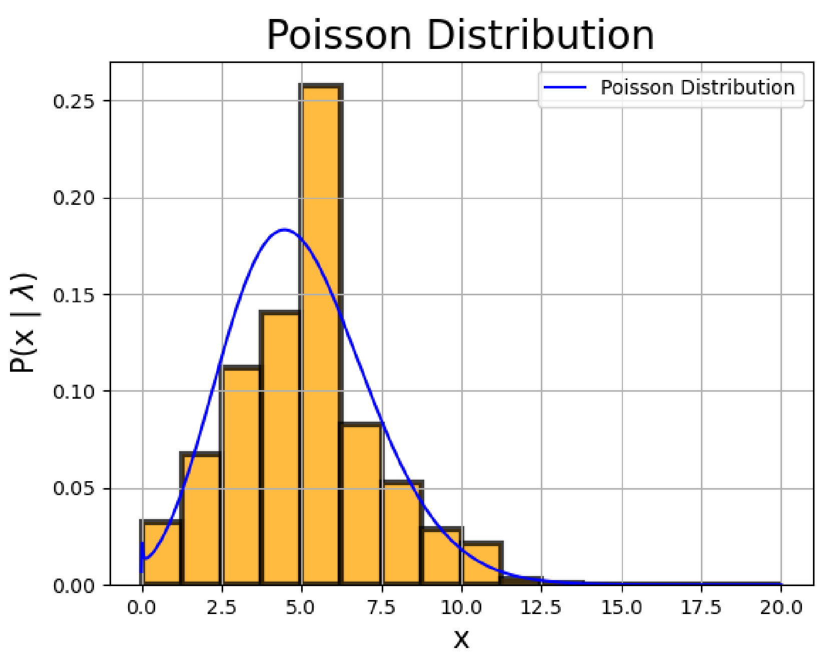

Figure 2: A histogram of the random data generated from a Poisson Distribution with $\lambda = 5.0$. The curve in blue is the actual distribution.

4 Results

Results are summarized in Figs. 3 and 4. These graphs were created with **PoissonAnalysis.py**, which is shown in section 6.

The numerical estimation for the mean parameter from experiment one was $\lambda = 5.027 \pm 0.0224$, and the average estimation from all 100 experiments was $\lambda = 5.032 \pm 0.0225$. The numerical maximization was done using the scipy.optimize library and minimizing the negative of the log likelihood function for a single experiment (since we are trying to maximize the log likelihood in reality). The error bars on the numerical estimations were computed with a simple root finding algorithm that solved for the x-values when the log likelihood was equal to 1/2.

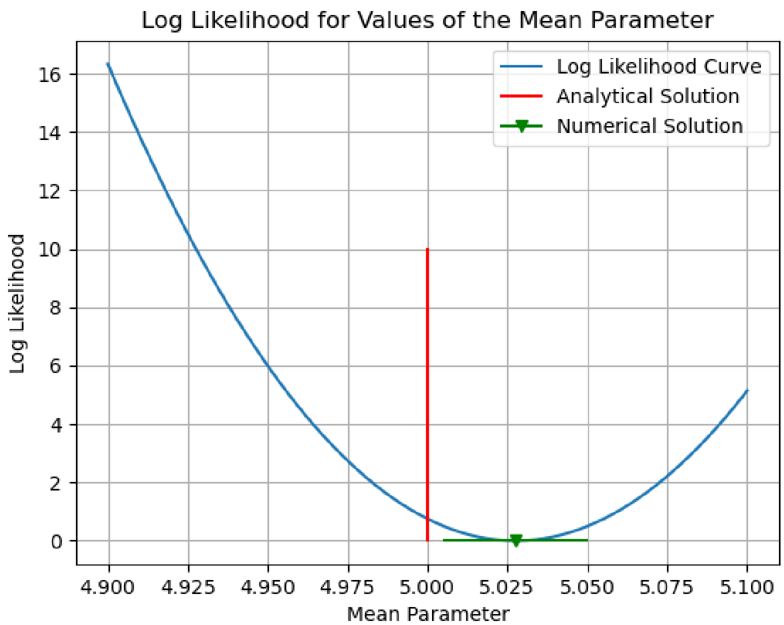

Figure 3: A graph of the log likelihood versus the mean parameter using the data from experiment 1 and comparing it to the analytical solution. The calculated mean value was $\lambda = 5.027 \pm 0.022$. The analytical solution derived in Section 1 was $\lambda = 5.00$. The charge of the electron from this estimation is $q = 1.610 \pm 0.022 \; 10^{-19}$ C.

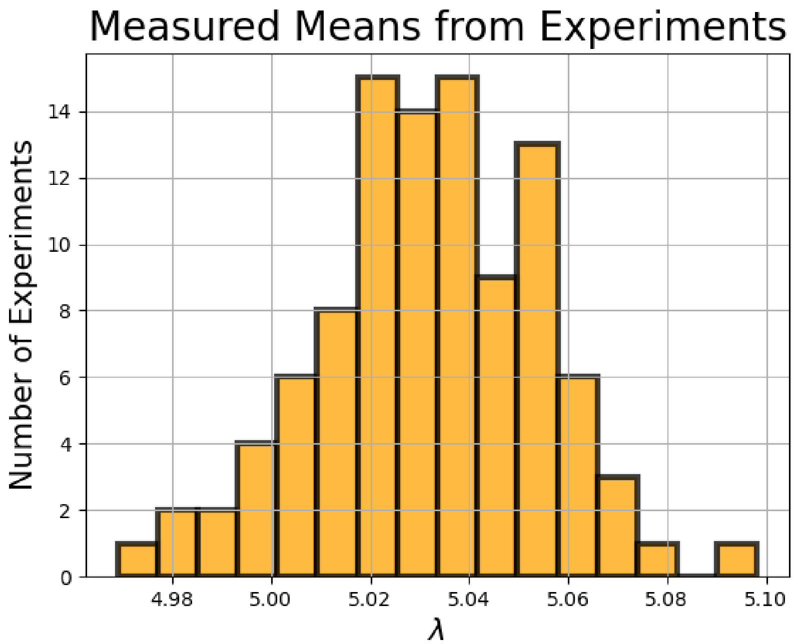

Figure 4: The distribution of all of the numerical estimations of the mean parameter from 100 experiments. As expected, it follows a Gaussian Distribution. The peak of the distribution is the most likely value of the mean parameter and was found to be $\lambda = 5.032 \pm 0.023$.

5 Summary

This simulation numerically estimated the rate parameter of a Poisson Distribution and the charge of an electron using some data. This was done by finding the "most likely value" of the mean parameter λ by maximizing the log likelihood of our probability distribution.

The estimated charge of the electron for experiment one is given by Eqn. 9 and was found to be $q = 1.610 \pm 0.022\ 10^{-19}$ C after accounting for the correction factor in Eqn. 11. The average charge of the electron from 100 experiments was found to be $q = 1.612 \pm 0.023\ 10^{-19}$ C.

6 PoissonAnalysis.py

Here are the key parts of the code file **PoissonAnalysis.py** used to do the analysis in this simulation.

The code reads in the data file generated by **Poisson.py** and then finds a numerical estimate for the mean parameter λ for a single experiment, and then for all experiments. The numerical minimization/maximization was done using standard scipy libraries. Error on the mean parameter was found using a simple root finding algorithm.

```python
#Log Likelihood for a single experiment with Nmeas measurements.
current_exp = 0 #experiment 1

def LogLikelihood(lamb):
    f = 0 #initialize value
    lamb = np.sqrt(lamb * lamb) #lambda must be positive

    for d in data_0[current_exp]: #measurements in a single experiment
        if d == 0:
            f += 0

        else:
            f += d * np.log(lamb) - 1/2 * np.log(2 * np.pi * d) - d * np.log(d) + d - lamb

    return -1 * f #need to return the negative in order to use minimization package.

#analytical solution for maximum log likelihood
x_max = lamb
y_max = 0

print(f"Log likelihood is maximized at x = {x_max} for the analytical solution")

#numerical solution for single experiment
result = optimize.minimize_scalar(LogLikelihood)
print(f'Scipy minimization was successful: {result.success}') # check if solver was successful
```

```

x_num = result.x #the numerical solution for a single experiment
#print(f'Numerical x-value found to maximize log likelihood for experiment 1: {x_num}')

num_result = LogLikelihood(x_num) #normalization factor for graph below
#print(f'Maximimum log likelihood: {num_result}')

#Log Likelihood Curve
x2 = np.linspace(lamb - 0.1, lamb + 0.1, 1000) #bounds are close to actual value of lambda
y2 = []

#Analytical Solution
x3 = []
y3 = np.linspace(0, 10, 1000)

#Initializtion for error analysis
yerrlo = 0.5
yerrhi = 0.5
xerrlo = lamb - 0.1
xerrhi = lamb + 0.1

for i in range(len(x2)):
    y2.append(LogLikelihood(x2[i]) - num_result)
    x3.append(x_max) #need an array of all the same values to plot a vertical line on the graph.

    #find lower bound for 1 sigma error bar
    if x2[i] < x_num:
        if np.abs(y2[i] - 1./2.) < yerrlo:
            yerrlo = np.abs(y2[i] - 1./2.)
            xerrlo = x2[i]

    #find upper bound for 1 sigma error bar
    if x2[i] > x_num:
        if np.abs(y2[i] - 1./2.) < yerrhi:
            yerrhi = np.abs(y2[i] - 1./2.)
            xerrhi = x2[i]
```

```

#Numerical error bars on x for a single experiment
sig_num = (xerrhi - xerrlo)/2

#print(f'Uncertainty of numerical solution for experiment 1:
    {sig_num}')
print(f'Numerical Mean value for experiment 1: {x_num} +-
    {sig_num}')
print(f'68% CI of numerical solution for experiment 1:
    [{xerrlo}, {xerrhi}]')

#plot of Log Likelihood for a single experiment
plt.plot(x2, y2, label = 'Log Likelihood Curve')
plt.plot(x3, y3, color = 'red', label = 'Analytical Solution')
plt.errorbar(x_num, 0, xerr = sig_num, color = 'green', marker
    = 'v', label = 'Numerical Solution')

plt.xlabel('Mean Parameter')
plt.ylabel('Log Likelihood')
plt.title('Log Likelihood for Values of the Mean Parameter')
plt.grid()
plt.legend(loc = 'upper right')

plt.savefig('LogLikelihood.png')
plt.show()

#Estimate charge of electron for Experiment 1
q = 1.602 * 10**-19 #actual value of electron
N = lamb/q #number of Nmeas that should have been used given
    the lambda value

correction = Nmeas/N #correction factor

q_prime = x_num/Nmeas #value measured based upon parameter
    estimation of lambda

e_val = q_prime * correction
e_unc = sig_num

print(f'Charge of an electron from experiment 1: {e_val} +-
```

```
    {e_unc}')

#Numerical Solution for all Experiments
result_list = [] #list of all means

Mean_exp = 0
Mean_exp_squared = 0

for exp in range(0, Nexp):
    current_exp = exp
    result = optimize.minimize_scalar(LogLikelihood) #result for each experiment
    result_list.append(result.x)

    Mean_exp += result.x
    Mean_exp_squared += result.x * result.x

Mean_exp = Mean_exp/Nexp
Mean_exp_squared = Mean_exp_squared/Nexp

Sigma_exp = np.sqrt(Mean_exp_squared - Mean_exp**2)

print(f'Numerical Mean value of all experiments: {Mean_exp} +- {Sigma_exp}')

#Histogram of all means
result_list = np.asarray(result_list)
n, bins, patches = plt.hist(result_list, 16, edgecolor = 'black', linewidth = 3, density = False, facecolor = 'orange', alpha=0.75)

#Plot Formatting
plt.xlabel('$\lambda$', fontsize = 15)
plt.ylabel('Number of Experiments', fontsize = 15)
plt.title('Measured Means from Experiments', fontsize = 20)
plt.grid(True)

plt.savefig('MeanDistribution.png')
plt.show()
```

```
    #Estimate charge of electron from all experiments
    q_mean = Mean_exp/Nmeas #value measured based upon parameter
        estimation of lambda

    e_val_exp = q_mean * correction
    e_unc_exp = Sigma_exp

    print(f'Average charge of an electron from all experiments:
        {e_val_exp} +- {e_unc_exp}')
```