AST325H1F Lab 1: Basic Photon Statistics

Parampreet Singh
parampreet.singh@mail.utoronto.ca
Lab Group G

October 2nd, 2019

Abstract

In this lab report, various methods of computing the statistical properties of photons are investigated. The statistical data for photons in this lab originate from the Photo Electric Effect and are physically measured through devices known as Charged Coupled Devices (CCDs). Through the provided photon data sets from the CCD, statistical analysis such as scatter plots, histogram plots, and Poisson distributions were conducted in order to study the nature of photon counting statistics. This was done through given photon count rate data from stars of various distances. It will be shown that various photon count rate data sets can provide distinct statistical properties. One of the major observations being that for large photon count rates, the data increasingly represents a continuous Gaussian distribution. Also explored are potential sources of error and reduction methods for these errors.

1 Introduction

The Photo Electric Effect is a well known phenomenon in physics. In simple terms, the effect is based on the idea that a photon is able to provide energy to an electron on a metal surface, causing the electron to be emitted or ejected. This effect proves to be very useful for astronomical purposes since one of the main tasks in the field of astronomy is to investigate and gather information about light from distant sources. CCDs employ the Photo Electric Effect as a basic method of detecting photons from astronomical sources. It is also well noted that this process of photon detection and counting is not perfect. CCDs have flaws that introduce various systematic errors with the process of measurement. In this experiment, we attempt to visualize data and calculate basic statistical parameters on photon count rate data sets. This is done by examining inputs of photons counted by a CCD image sensor. While it will be shown that the statistical properties for small photon count rate data sets resemble a Poisson distribution (when modeled with parameters as number of photons vs number of measurements). Our experiment will show that large photon count rate data sets resemble a Gaussian distribution (when modeled with the same parameters). Various methods of assembling data and data visualization for photon count rates will also be explored.

The division of labour this lab was split evenly between the Author (Parampreet Singh) and lab group G members Nicholas Clark, Juan Pablo Alfonzo, and Lucas Louwerse. The methods and codes used to obtain and analyze data were determined as a partnership between all four group members. Individual members had their own approach to compile the given data and produced their own unique results and plots. These were compared between the group members to ensure the accuracy of the experiment.

2 Observations and Data

The data for this lab was taken from the assumption that a series of astronomical observations using a CCD were done. Also in the assumption is that the experiments which yielded the data were preformed many times in order to produce sufficient data sets. In total, there were four data sets which were considered. These include determining the distance to a nearby star, and three data sets of measuring the amount of incoming photons from an astronomical source. The exact instructions and reference to the explanation of these data sets and CCDs can be found on the AST325 course website [1].

Table 1: Distance Measurements With Uncertainties.

Beginning of Data	
Distance Measurement (pc)	Uncertainty (pc)
31.91	1.41
37.14	0.36
38.19	0.69
41.03	3.53
34.86	2.64
... 36.51	0.99
End of Data	

Table 2: Photon Count Rates.

Beginning of Data	
Measurement	Photon Count Rate
1	13
2	17
3	18
4	14
5	11
... 30	10
End of Data	

Shown in Table 1 and Table 2 above are samples of the provided data. Included is one data set corresponding to the distance measurements with uncertainties and another corresponding to photon count rates. As you can see, many trials were conducted for each experiment so the tables have been shortened in order to effectively display the data. Also, only one data set corresponding to a photon count rate was shown. This is because all three data sets closely resemble one another and showing one suffices in gaining an understanding of the type of data set which was provided to analyze. For distance measurements, a series of distances were pre-determined with their relative uncertainties. A total of 30 distance measurements and 30 uncertainties were provided. For the photon count rates, a series of measurements were pre-conducted with the number of photons counted for each measurement. One photon count rate data set contained 30 measurements while the other two contained 1000 measurements each. Data sets for this lab were common between all group members, meaning each individual in the Author's group will have all four of the same data sets.

It is important to note that although data sets were fully provided, there could possibly be a string of errors associated with each of these data sets. For example, it is not certain how exactly the data was collected. This means that there could be many variations and systematic errors associated with the equipment used to gather the provided data. The data set could also be completely random, in which case processing the given data would be the only important factor for this lab. As a result, the accuracy and precision of the data sets is very questionable. Since there is no way to counter these errors as someone who's main task is to process the data, this is just something that we have to keep in mind moving forward through the lab.

3 Data Reduction and Methods

For this lab, the provided data was analyzed using numerous Python coding packages. These include `matplotlib`, `numpy`, and `pandas`, as well as many algorithms and functions, created by the Author, which heavily incorporate these packages for the purpose of processing the given data and providing the required analysis. For this lab in particular, the required tasks were to perform a reduction of data following instructions which were given through a lab handout. What is unique is how each group member went about tackling the method for reducing and processing their data. The main goal for this lab was to familiarize ourselves with basic statistical principles needed for professional data reduction. This includes performing tasks such as plotting histograms to visualize the data and computing the mean, standard deviation, weighted mean, and to understand the Poisson distribution to investigate if the data is best represented by this distribution.

The reduction of the given data mainly focused on computing means, standard deviations, and weighted means. These quantities are best visualized through histogram plots, which was one of the main data visualization techniques that was used for this lab, combined with Poisson and Gaussian distributions for these histogram plots, which is discussed in Section 5.3.2. Throughout the lab, it was required that for a given data set, the mean and the standard deviation should be calculated. Also, for the distance data, alongside the mean and standard deviation, it was also required to calculate the weighted mean and weighted standard deviation. For all of these calculations, simple Python codes were implemented in order to manipulate the data in order calculate for a given quantity. The calculations were based on the following formulae.

Firstly, the mean (or average) of all data sets was calculated with the following equation where μ represents the mean and N represents the number of samples or trials:

$$\mu = \frac{\sum_{i=1}^{N} x_i}{N} \tag{1}$$

Secondly, the standard deviation of all data sets was calculated with the following equation where σ represents the standard deviation, N represents the number of trials, and μ represents the mean:

$$\sigma = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N} x_i - \mu} \tag{2}$$

Thirdly, the weighted mean of the distance data set was calculated with the following equation where $\bar{x}$ represents the weighted mean, σ_i represents the error of the i^{th} trail, and N represents the number of trials:

$$\bar{x} = \frac{\sum_{i=1}^{N}(x_i/\sigma_i)}{\sum_{i=1}^{N}(1/\sigma_i)} \tag{3}$$

Lastly, the weighted standard deviation of the distance data was calculated with the following equation where $\sigma_{\bar{x}}$ represents the weighted standard deviation, σ_i represents the error of the i^{th} trail, and N represents the number of trials:

$$\sigma_{\bar{x}} = \frac{1}{\sqrt{\sum_{i=1}^{N}(1/\sigma_i^2)}} \tag{4}$$

There are possible ways to shorten the amount of calculation steps that are required for these quantities through various shortcuts in Python programming, however, since one of the main objectives of this lab was to obtain knowledge on basic Python programming skills, all calculation steps were shown thoroughly through code which is provided in Appendices A-C. Gaussian and Poisson distributions were also calculated completely by functions created by the Author which use the quantities determined by Eq. 1 and Eq. 2, this is discussed in Section 4.

Another thing to note is that the factorial term which appears in the Poisson distribution of Eq. 5 also had to be calculated by the author entirely. This means that the definition of factorial, as interpreted by the code, is displayed as something which is not commonly looked at as a factorial. The code for this definition of the factorial is provided in Appendix E.

Lastly, recalling the fact that Gaussian and Poisson distributions represent probabilities, this means that they must be normalized (the sum under their respective graphs must equal to unity). As a result of this, an appropriate scaling factor was introduced which ensures that this remains the case. For the purpose of the lab, this scaling factor was also to ensure that both distributions align with their appropriate histogram plots. The code for this can be found under Appendix E.

4 Data Analysis and Modeling

Recall that one of the main goals of this lab was to understand the Poisson distribution and investigate if the given data is represented well by this distribution. To do this, it was crucial to determine the mean and standard deviation of each data set prior to investigating anything about the Poisson distribution associated to the respectable data set. These values were calculated using Eq. 1 and Eq. 2.

Now that the mean and standard deviation are present for each given data set, it is feasible to create the respectable Poisson and Gaussian distributions.

Firstly, the Poisson distribution function of the photon count rate data sets was determined with the following equation where $P(x, \mu)$ represents the Poisson distribution function, μ represents the mean from Eq. 1, x represents the number of outputs we are interested in, and A represents the appropriate scaling factor (note that $P(x, \mu)$ is discrete by nature due to it's inputs of x, this is good for the case of this lab since the majority of Poisson distributions will be used to model properties of photons which are also discrete by nature):

$$P(x, \mu) = A\frac{\mu^x}{x!}e^{-\mu} \qquad (x = 1, 2, 3, ...) \tag{5}$$

Secondly, the Gaussian distribution function of the photon count rate data sets was determined with the following equation where $P(x, \mu, \sigma)$ represents the Gaussian distribution function, σ represents the standard deviation from Eq. 2, μ represents the mean from Eq. 1, x represents the number of outputs we are interested in, and A represents the appropriate scaling factor (note that $P(x, \mu, \sigma)$ is continuous by nature in contrast to $P(x, \mu)$) :

$$P(x, \mu, \sigma) = A\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\dfrac{(x-\mu)^2}{2\sigma^2}} \tag{6}$$

The codes to how the modelling of both the Poisson and Gaussian distributions are provided in Appendix E.

5 Discussion

5.1 Mean and Standard Deviation

The first step in understanding basic statistical calculations for this lab was to calculate the two quantities known as the mean and standard deviation. Both of these quantities were determined for all four of the provided data sets. An example of the raw data that was provided in which these calculations were applied to can be seen in Table 1 and Table 2. The Python programming code for the calculation of these quantities can be found under Appendices A-B. The calculation for these quantities are represented by Eq. 1 and Eq. 2. The reason that these quantities are determined is because they are extremely important in any statistical analysis. For the purpose of this lab, the mean and standard deviation provide the foundations to allow distributions such as the Poisson and Gaussian functions to exist. From Eq. 5, we can see that the Poisson distribution contains a variable specifically for the mean, while on the other hand, the Gaussian distribution contains variables for both the mean and the standard deviation.

Table 3: Means and Standard Deviations of Data Sets.

Beginning of Data		
Data Set	Mean	Standard Deviation
Distance Data	37.21	2.65
Photon Count Rate (30 Measurements)	13.20	3.80
Small Photon Count Rate (1000 Measurements)	3.38	1.78
Large Photon Count Rate (1000 Measurements)	24.00	4.88
End of Data		

Shown in Table 3 are the mean and standard deviation outputs for each given data set as calculated by the Python programming code (see Appendices A-B). It is clear that the mean represents the average (or middle) value for a given data set. Another way to interpret the mean is to order the data in ascending or descending order and to choose a value where half of the data lies above and the other half of the data lies below that data point. The significance of the mean for this lab is further discussed in Section 5.3. The standard deviation represents how measurements for a given data set are spread out from the mean. A low value for the standard deviation means most data points are close to the mean, in contrast, a high value for the standard deviation means that data points are more spread out (or further from the mean). The significance of the standard deviation for this lab is also further discussed in Section 5.3.

Notice that these values completely rely on measurements. Various systematic errors can greatly effect the outcome of such values since these errors deal with direct measurement and measurement devices. However, for the case of this lab, it is unsure exactly how the data was collected. This means that it is not distinguishable whether or not certain outliers in the data sets are a result of errors or just simply a result of correct measurement, as this greatly effects the outcomes of the mean and standard deviation calculations. For example, if a data set contains a few large or a few small outlier values, the mean is allowed to sway in the favour of these outlier values whether the measurement was correct or not. Since the standard deviation is dependent on the mean, according to Eq. 2 and as explained before, this may cause the standard deviation to increase or decrease in favour of these outlier data points.

For further investigation, more information about how the four given data sets were collected would need to be provided. This would allow for additional judgements on the reasoning behind the calculated values for the mean and standard deviation of these data sets. For now and for the purpose of this lab, these values represent essential tools required in order to determine the Poisson and Gaussian distribution curves with respect to each data set.

5.2 Weighted Mean and Weighted Standard Deviation

One unique step which was required as a part of the distance data analysis was to calculate the weighted mean and the weighted standard deviation. The reason that this was unique for the distance data set is because according to Eq. 3 and Eq. 4, the weighted mean is calculated by incorporating the uncertainty (or error) associated with each measurement. Since the distance data set is the only data set which displays the errors associated with each measurement, the weighted mean and the weighted standard deviation could only be found for this particular data set. The reason that these quantities were found was mostly to gain experience in calculating and programming such operations. For the purpose of this lab, the weighted mean and the weighted standard deviation served no further purpose, however, it was still important to do this since one of the main goals for this lab was to familiarize ourselves with basic statistical principles, this being one of them.

Table 4: Weighted Means and Weighted Standard Deviations of Data Sets.

Beginning of Data		
Data Set	Weighted Mean	Weighted Standard Deviation
Distance Data	37.50	0.021
Photon Count Rate (30 Measurements)	NA	NA
Small Photon Count Rate (1000 Measurements)	NA	NA
Large Photon Count Rate (1000 Measurements)	NA	NA
End of Data		

Shown in Table 4 are the weighted mean and weighted standard deviation outputs for each given data set as calculated by the Python programming code (see Appendix C). Like the mean, the weighted mean represents an average, however, in a weighted mean calculation, the errors associated with each measurement are also incorporated, same as with the weighted standard deviation. The effect is that some data points in a given data set are "worth" more than other or carry a greater weight. This aspect gets included in such a calculation. The result is a value which provides a more precise representation of an average value. This is because outlier data points in a weighted mean tend to carry less "worth" or less weight than those data points which are more consistent, thus providing for a more precise middle value. Note the difference between a precise average and a accurate average.

For a weighted mean, the average may be more precise, but it does not ensure that the average will become any more accurate. This is because the accuracy of the average depends on experimental processes and/or knowledge of any errors or the absence of errors for a given experiment. To investigate this statistical calculation more comprehensively would require one to conduct an experiment where measurements yield uncertainties. The weighted mean and the weighted standard deviation would then prove to be insightful statistical tools. Again, for the purpose of this lab, what was focused more heavily was the ability to perform such a calculation if required.

5.3 Poisson and Gaussian Statistics

The ultimate goal of this lab was to investigate the properties of the Poisson and Gaussian distributions. This is the reason why quantities such as the mean and standard deviation were determined (aside from just being useful statistical analyzing tools). What will now be discussed is how the mean and standard deviation of the given data sets were used in order to determined the respectable distribution curves.

5.3.1 Scatter Plot Data

Before studying any Poisson or Gaussian distributions, it was required to determine scatter plot data of each of the data sets.

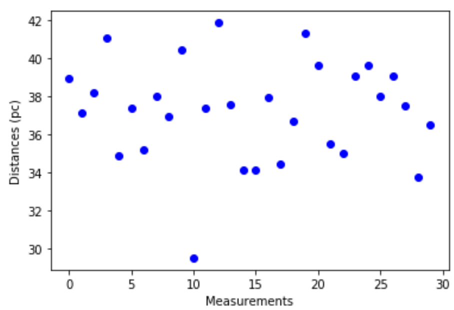

Figure 1: Scatter Plot for Distance Data

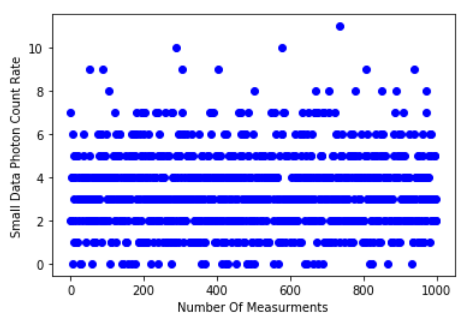

Figure 2: Scatter Plot for Small Photon Count Rate Data

Displayed in Figure 1, Figure 2, and Figure 3 are the scatter plot data associated with their respective data set. The plots are fairly similar, they show the number of measurements versus the measured quantity for each experiment. Since Figure 1 contains 30 measurements, it is much easier to read data from this plot. On the other hand, Figure 2 and Figure 3 both contain 1000 measurements each making them much more difficult to read. From this, it is noted that the number of measurements for a given experiment effects the readability for the respective scatter plot associated with that data.

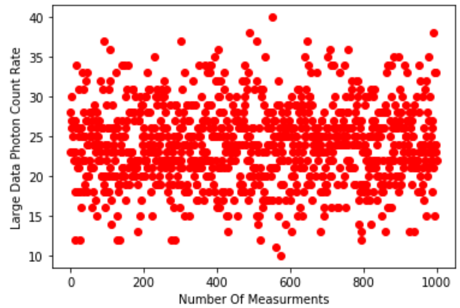

Figure 3: Scatter Plot for Large Photon Count Rate Data

As explained in Section 5.3.2, histogram plots provide a better way to visualize data, regardless of the number of measurements, in contrast to scatter plots which can become hard to read with many data points.

5.3.2 Histogram and Distribution Data

The histogram data as well as the Poisson and Gaussian distribution data corresponding to respective data sets will now be displayed and discussed.

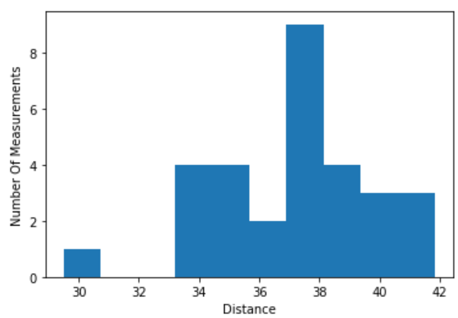

Figure 4: Histogram for Distance Data

Figure 4 simply shows data compiled from distance measurements into a histogram plot. For the Python programming code which displays the creation of a histogram plot, refer to Appendix D. When observing a plot such as this, it is important to understand all the parameters involved. In contrast to a scatter plot, a histogram plots data in accordance to how often a data point arises as a result of measurement. For example, a measurement of a distance of 39pc appears roughly four times throughout the entire data set, and, a measurement of 32pc never appears throughout the entire data set.

This type of plot is important because it allows you to quickly deduce statistical quantities, one of the easiest being the mode (the value which occurs most frequently in a given set of data). It is also possible to determine the mean through a more extensive observation. This can be done by roughly estimating a value at which you believe half of the data points lie to the left of that chosen point and half of the data points lie to the right of that chosen point.

In Figure 5, alongside a histogram plot, a Poisson distribution is displayed for this data set. For the Python programming code which is used to display a Poisson distribution, refer to Appendix E. From Eq. 5, the Poisson distribution depends upon the mean of a data set. A Poisson distribution represents how many times an event is likely to occur within a specified period of time. It is used for independent events which occur at a constant rate within a given interval of time [2]. This is precisely why this type of distribution is perfect for experiments involving photon counting. This is because when counting photons, there is a rate and an interval of time which are considered (e.g 43 photons/sec).

According to Figure 5, the event which is likely to occur would be to detect photons in the range of where the distribution peaks, or, where most of the data points are contained.

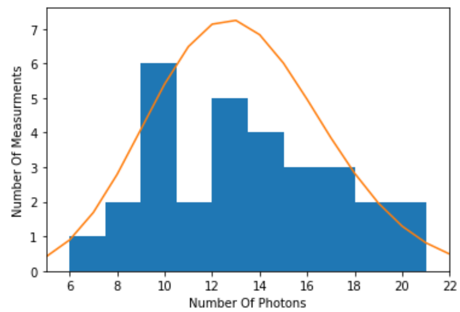

Figure 5: Histogram and Poisson Distribution (Yellow) for Photon Count Rate Data (30 Measurements)

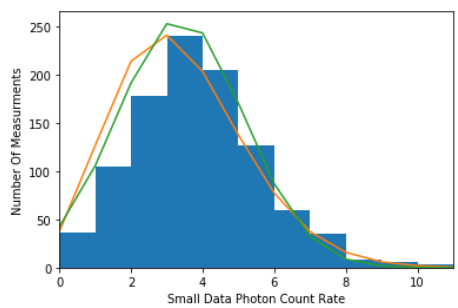

Figure 6: Histogram, Poisson Distribution (Yellow), and Gaussian Distribution (Green) for Small Photon Count Rate Data (1000 Measurements)

Displayed in Figure 6, alongside a histogram and Poisson distribution, is a Gaussian distribution for this data set. For the Python programming code which is used to display a Gaussian Distribution, refer to Appendix E. From Eq. 6, the Gaussian distribution depends upon the mean and the standard deviation. A Gaussian distribution is centered on the mean and the width of the distribution represents the standard deviation. So a small standard deviation means that there is a high probability of obtaining a measured value close to the mean value [2]. A Gaussian distribution is one of the most fundamental tools used in statistical analysis as it provides a great deal of statistical information.

From the two distributions in Figure 6, which distribution better represent the data? For the case of small data points, a Poisson distribution proves to be a better representation of the data. This is because as stated before, a Poisson distribution is ideal for photon counting. This is clearly the case in Figure 6 where the two distributions are quite discrete and do not exactly align. If this is the case, then the question arises of why this would change for distinct data set points.

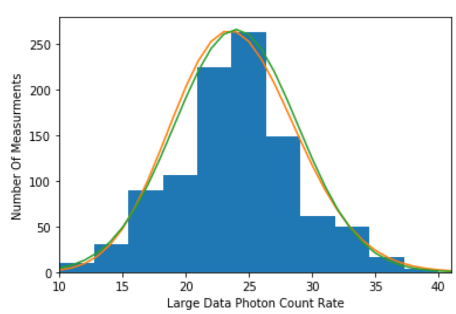

Figure 7: Histogram, Poisson Distribution (Yellow), and Gaussian Distribution (Green) for Large Photon Count Rate Data (1000 Measurements)

Displayed in Figure 7, alongside a histogram and Poisson distribution, is a Gaussian distribution for this data set. In this figure, the Gaussian distribution greatly resembles the Poisson distribution. This is because the data set which was considered for these distributions contained large data points. As the data points become large, a Poisson distribution begins to approximate a Gaussian distribution [2]. In contrast to the small data points, it is clear the both distributions align very well in this case.

6 Conclusion

In this lab, basic statistical analysis of data sets was performed. From provided data sets, tasks such as determining quantities known as the mean, standard deviation, weighted mean, and weighted standard deviation were approached through the writing of manual Python programming codes. Determining these quantities was required in order plot the necessary distribution curves, those being the Poisson and Gaussian distributions, as well as gaining the basic programming knowledge needed in order to perform such tasks. Overall, it was determined that for large data points within a given data set, a Poisson distribution curve will start to resemble a Gaussian distribution curve. It was also discovered that troubleshooting errors for this particular lab was out of the control of the Author and fellow group members.

7 Appendix A Mean Calculation Code

Below is a sample of code which is used to calculate the mean of a data set.

```
#Mean
MeanSmall=sum(DataSmallList)/1000 #Add all of the elements in the list
                                  #together and divide the sum by the
                                  #number of results
print("The Mean Is:", MeanSmall) #Print the mean
```

8 Appendix B Standard Deviation Calculation Code

Below is a sample of code which is used to calculate the standard deviation of a data set.

```
# Standard Deviation
i=0 #Create an index for the
    #loop that will calculate the standard deviation
SDlistSmall=[] #Create an empty list to store the
               #difference between the elements
               #of DataSmallList and the mean squared
while i<1000: #While loop which fills SDlistSmall
              #with the difference between the elements
              #of DataSmallList and the mean squared
    SDlistSmall.append((DataSmallList[i]-MeanSmall)**2)
    i=i+1

sdSmall=(sum(SDlistSmall)/999)**0.5 #Apply the standard
                                    #deviation formula and call this sdSmall
print("The Standard Deviation Is:" , sdSmall) #Print standard deviation
```

9 Appendix C Weighted Mean and Weighted Standard Deviation

Below is a sample of code which is used to calculate the weighted mean and weighted standard deviation of a data set. This particular algorithm was used for only the distance measurement data set.

```
# Weighted Mean
E=[1.41, 0.36, 0.69, 3.53, 2.64, 0.17, 2.34, 0.46, 0.57,
   2.91, 8.00, 0.17, 4.34, 0.03, 3.38, 3.39, 0.44, 3.07,
   0.82, 3.81, 2.11, 2.02, 2.52, 1.55, 2.12, 0.46, 1.52,
   0.03, 3.74, 0.99] #Create a list of the errors
j=0 #Create and index for the loop
    #which will calculate the weighted mean
WMlistNumerator=[] #Create an empty list for the
                   #numerator of the weighted mean calculation
WMlistDenominator=[] #Create an empty list for the numerator
                     #of the weighted mean calculation
while j<30: #While loop which fills the two empty
            #numerator and denominator lists with the appropriate calculaton
    WMlistNumerator.append(M[j]/(((E[j])**2)))
```

```
    WMlistDenominator.append(1/((E[j])**2))
    j=j+1

wmN=sum(WMlistNumerator) #Takes the sum of all elements in the numerator list
wmD=sum(WMlistDenominator) #Takes the sum of all elements in the denominator list
wm=wmN/wmD #Calculates the weighted mean
print("The Weighted Mean Is:" , wm) #Print the Weighted Mean

print("") #Skip a line in the text


#Weighted Standard Deviation
wsd=1/((sum(WMlistDenominator))**0.5) #Use the denominator list from the Weighted Mean
                                       #and apply it to the Weighted Standard Deviation
print("The Weighted Standard Deviation Is:" , wsd) #Print the Weighted Standard Deviation
```

10 Appendix D Scatter Plot and Histogram Code

Below is a sample of code which is used to generate a scatter plot and a histogram plot for a given data set.

```
import matplotlib.pyplot as plt #Import class to create plots

#Plots
#Scatter Plot
print("A Scatter Plot of Number Of Measurements vs Distance")
plt.subplot() #Allows the creation of multiple plots
plt.plot(M, 'bo') #Adds axis elements from the list of Measurments M
                  #which was defined previously
plt.xlabel('Measurements') #x-axis title set
plt.ylabel('Distances (pc)') #y-axis title set
plt.show() #Display the plot

print("") #Skip a line in the text
#Histogram
print("A Histogram of Distance vs Number Of Measurements")
plt.subplot()
plt.hist(M) #Generate histogram plot
plt.xlabel('Distance')
plt.ylabel('Number Of Measurements')
plt.show()
```

11 Appendix E Poisson and Gaussian Distribution Code

Below is a sample of code which is used to create a Poisson and Gaussian distribution from their respectable function equations for a given data set.

```
import matplotlib.pyplot as plt #Import class to create plots
import numpy as np #Import class to take exponential in possion
                   #and gaussian distribution

#Distributions Large
FunOutLarge=[] #Create An Empty list to store outputs of poisson function
```

```
FunOutLargeG=[] #Create an empty list to store outputs of guassian function
b=0 #Create an index for the while loop
while b<100: #While loop to fill poisson distrubution outputs
    factorialLarge=1 #Reset the base value of the factorial

    for t in range(1, b+1):
        factorialLarge = factorialLarge*t #Evaluates the
                                          #factorial for the position

    FunOutLarge.append(135*((MeanLarge**(b+1)/factorialLarge)
    *np.exp(-1*(MeanLarge)))) #Formula for poisson distribution

    FunOutLargeG.append(3250*(1/((2*np.pi*(sdLarge)**2)**0.5)
    *np.exp(-1*((b-MeanLarge)**2)/(2*(sdLarge)**2)))) #Formula for gaussian distribution
    b=b+1 #Add one to the index

plt.plot(FunOutLarge) #Plot poisson distribution
plt.plot(FunOutLargeG) #Plot gaussian distribution
plt.show() #Show all plots
```

References

[1] AST325/326 Course Website. http://www.astro.utoronto.ca/ astrolab/.

[2] Dae-Sik Moon. Basic Statistics. http://www.astro.utoronto.ca/ astrolab/.