PHSX 815 Project 3:
Sharks, Minnows, and MLEs:
Estimating the Rate Parameter of a Soccer Team's Performance through Maximum Likelihood Estimations (MLE)

Ashley Lieber

April 10, 2023

1 Introduction

As the year 2022 drew to a close, much of the world had their eyes set on the television screen to see which country's soccer team would rise to the top and win the FIFA World Cup. This event captivated millions across the globe and serves as the inspiration for the simple simulation and analysis described in this paper.

The first stage of this project follows a simulated experiment of a soccer team's performance by generating data of the number of goals a team scores over many seasons. This provides a way for one to simulate more observations of games than one could ever hope to realistically observe. Once the data has been generated, a secondary script will analyze the data and attempt to estimate the overall rate of goals for the team. This rate is known when the data is generated, but the analysis script will attempt to estimate that parameter from the data alone. When compared to the scenarios presented in Projects 1 & 2, this scenario of attempting to estimate the rate parameter directly from the simulated, experimental data is much more realistic and certain than hypothesis testing between two guesses at the rate. The methods that are used to make this estimation will be described throughout the course of this paper. Beyond simply making this estimation for a single data set, this paper will also attempt to answer supplemental questions such as: How does the number of game observations (measurements) affect the accuracy of the estimation?, How does the number of experiments or seasons measured affect the analysis?, and lastly, How does the code's output change when it analyzes data based on extreme values of the rate parameter?. This paper will walk through the analysis of each of these scenarios.

This paper is organized as follows: Sec. 2 explains the experimental setup, models used, and methods for data creation. Sec. 3 walks through the analysis code and the computations it performs on a given data set. Next, Sec. 4 explains the different scenarios that were tested in this experiments and the findings that can be taken from those tests. The following Sec. 5 presents an overall summary of the conclusions of this simulation. Lastly, Sec. 6 provides the link to the GitHub repository for this project which contains all pertinent code scripts, referenced data sets, and figures as well as instructions on how to use and understand the various items.

2 Model Details and Data Generation

In order to conduct this analysis, a data set needs to be generated. To obtain these, a code was written that will randomly sample data from a Poisson distribution. Each measurement, or number pulled from the Poisson distribution is akin to sitting and watching a full soccer game and recording the number of goals a particular team achieved. The setup gives the great advantage of simulating the observation of a high number of games (e.g. 1000, 10,000, 100,000...) that one could never expect to realistically observe. Based on the scenario we have set forth, there are certain limits on the different parameters involved. For example, a single measurement value can only take on values from $[0, \infty)$ since a negative score makes no logical sense. Additionally, the decision to randomly sample the data from a Poisson distribution was made because the distribution was aptly suited to the situation at hand. In order to use a Poisson distribution, several criteria must be met which are as follows: (1) the "individual events must occur at random and independently in a given interval of time or space", and (2) "the mean number of occurrences of events in an interval (time or space) is finite and known, [1]. The particular experiment being simulated in this project fits both of these criteria for the following reasons. First, the individual data points or measurements of goals scored by a team are inherently random events which are recorded once per game. Each game is, in and of itself, a independent event. Secondly, the "mean number of occurrences", often referred to at the rate parameter with the symbol λ, is a value which is finite and known.

It should be noted at this point, that when the data is generated, it must be based off of a given rate parameter (e.g. 5 goals per game) in order to generate data. This is because we need to generate artificial data in order to run the experiment, but in a real-world experiment that would be an unknown value. So, even though the analysis won't factor in this true rate value when it is estimating the rate, since we generated the data we know the true rate value and can use that as a point of comparison to credibly analyze the success of the analysis. Beyond simply meeting the criteria of Poisson distributions, it is known in statistical practices that the Poisson distribution is an apt choice for event data as it is a "discrete probability distribution that describes probabilities for counts of events" [2]. After confirming that the scenario being tested fulfilled the base criteria of utilizing the Poisson distribution, the code to begin generating our data sets was written.

The data will be generated by randomly sampling values from the distribution to simulate each and every measurement in our data. The Poisson Probability Distribution equation is given by the following formula which calculates the probability of data point, x, occurring in a given interval based on some rate parameter λ

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

where λ is the average number of measurements per interval (rate), e is the constant Euler's number which is approximately 2.78, and x which takes discrete values and corresponds to our data [3]. In terms of our scenario, x is the measurement of the number of goals scored by a team in one game (e.g. 5 goals) and λ is the rate parameter which corresponds to the teams performance (e.g. 5 goals/game). An example of what a Poisson distribution looks like is shown in Figure 1.

The data can be generated according to this Poisson distribution by randomly sampling data using the code script named $GoalDataGeneration.py$ which can be found in the repository (linked in 6). When running this script, there is the ability to specify a number of parameters in order to generate data

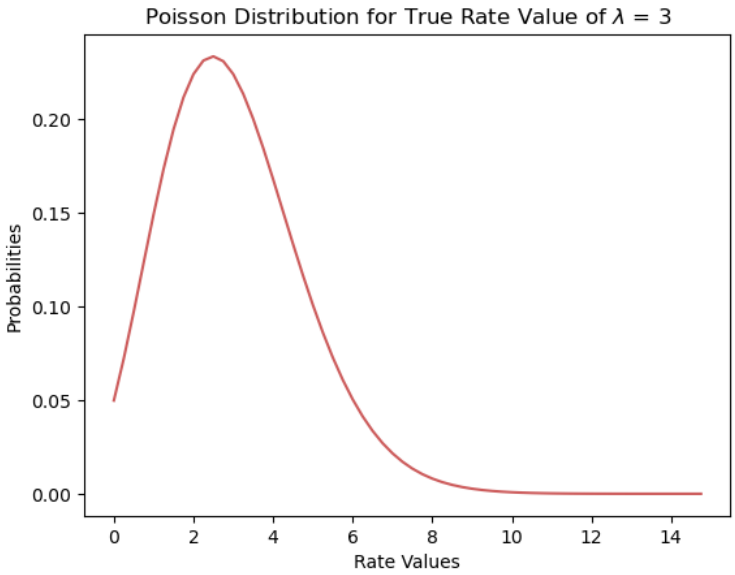

Figure 1: Example Poisson Distribution with a rate value of $\lambda = 3$. This is an example of the distribution from which data is randomly sampled from.

for different scenarios. These parameters are the number of measurements (Nmeas), the number of experiments (Nexp), seed, rate parameter (lambda), as well as the filename to save the dataset to. In the context of the soccer scenario, this means we can input parameters for the number of games we would like to observe (Nmeas) for every season, or sets of games, (Nexp), and the true rate parameter for the dataset. Since we are simulating the experiment rather than actually taking data ourselves, we can choose to simulate as many measurements and experiments as we would like. For example, the dataset $TestData-1000-1000-4.txt$, which can be found in the Project 3 repository 6, simulates the experiment of conducting 1000 experiments of measuring 1000 games, so 1,000,000 total measurements based on a Poisson distribution with a mean rate parameter of 4 goals per game.

The data file in the repository demonstrates what this output looks like. Essentially, the first line records the true rate used to generate the data, and then each subsequent line in the file represents an experiment, and each value on each line is a single game measurement. An example of such an output is as follows, if we sampled data with $\lambda = 2$ with $N_{meas} = 5$ and $N_{exp} = 10$ the data would appear in the format shown in Figure 2.

```
(base) ashleylieber@PHSX-MILLS-22 PHSX815_Project1 % python3 GoalData.py -rate 3 -Nmeas 5 -Nexp 10
3.0
2 4 7 5 3
5 4 2 3 1
5 1 0 3 3
0 1 1 2 3
0 1 2 4 6
1 4 4 1 3
5 3 2 5 3
3 1 5 0 1
3 4 4 6 3
1 3 6 4 5
```

Figure 2: Example of simulated data output if $\lambda = 2$, $N_{meas} = 5$, and $N_{exp} = 10$

In order to sample values from a Poisson Distribution, the script $GoalDataGeneration.py$ utilizes the

external package Scipy.Poisson [4] which encodes the equation shown earlier which helps to condense and simplify our code since there was no need to write the algorithm again from scratch. For each measurement needed, the code will sample a number from the Poisson distribution. This utilizes a nested for loop to sample each measurement. The resulting data will be a discrete value that is a non-negative integer $(e.g. 0, 1, 2, ...)$ [2]. These measurement values are then stored in a persistent data file format (.txt) which can then be read in by the analysis program.

In this project, rather than generating and analysis a single data set, I decided to generate three sets of data to demonstrate different aspects and abilities of the analysis code. Essentially, I would keep two of the three main variables (Nexp, Nmeas, and lambda) constant while greatly varying the other variable to understand how that variable affected the analysis of data as a while.

1. The first set of data I generated varied the number of experiments (Nexp) which meant that the true rate value and number of measurements per experiment were held constant. In this case, the rate was set at 3 goals per game with 1000 measurements per experiment. A data set with those parameters was generated for 10, 100, 1000, and 10,000 experiments.
2. The second set of data varied the number of measurements while holding the true rate value and number of experiments constant. The rate used for this data was 4 goals per game and used 1000 experiments. Similarly, a data set was generated for 10, 100, 1000, and 10,000 measurements.
3. Lastly, I wanted to test how the analysis reacted to extreme values of the rate parameter. In this case, the number of experiments and measurements were held constant. The values of lambda used were 5, 50, and 500 for these testing purposes. I had intended to try 0.5, but since a "half-goal" makes no logical sense in the scenario we are testing, I decided that it would not be necessary to test an impossible rate value. However, it is a value that should theoretically work for the Poisson distribution in general even if it is not functional in this specific case. The goal of this data set is to test the limits of the analysis even though a rate parameter of 500, or even 50 for that matter, is unlikely to represent a true rate value for a soccer team's performance.

3 Analysis Methods

After generating the randomly sampled data, the analysis can begin which can be found in the file $GoalDataAnalysis.py$ in the repository. The analysis code can handle a single data file at a time, so this section will outline the algorithms and computations that are performed in order to estimate the true rate parameter for the given data set. First the code gathers a few parameters from the data set such as the true rate parameter, number of measurements, and number of experiments. The remainder of the analysis in order to estimate the rate parameter follows the method of maximum likelihood estimation (MLE). This maximum likelihood estimate of λ is the value of λ that maximizes the likelihood – "that is, makes the observed data 'most probable' or 'most likely' [5]. The first step of this method is write down a function of the parameter of interest (λ) that is proportional to the likelihood of the function given some data. The following equation shows how this likelihood was written down and simplified.

$$L(x) = \prod_{i}^{Nmeas} (Pois(x_i|\lambda) \tag{2}$$

$$= \prod_{i}^{Nmeas} \frac{\lambda^x e^{-\lambda}}{x!} \tag{3}$$

We can then take the log of this likelihood function in order to get it in a format that is more amenable to functioning within the code. The following derivation shows how this was achieved.

$$Log(L(x)) = Log(\prod_{i}^{Nmeas} \frac{\lambda^x e^{-\lambda}}{x!}) \tag{4}$$

$$= \sum_{i=1}^{Nmeas} (X_i log\lambda - \lambda - logX_i!) \tag{5}$$

$$= log(\lambda) \sum_{i=1}^{Nmeas} X_i - (Nmeas)\lambda - \sum_{i=1}^{Nmeas} log(X_i! \tag{6}$$

The value that will be the most likely estimate of the λ parameter is the value that maximizes this function, or equivalently, minimizes the negative of the log likelihood function. Since I preferred to use a minimization routine within the code, I opted to minimize the negative of the log likelihood function [6].

$$\lambda_{estimate} = argmin\left[- (log(\lambda) \sum_{i=1}^{Nmeas} X_i - (Nmeas)\lambda - \sum_{i=1}^{Nmeas} log(X_i!)\right] \tag{7}$$

$$\tag{8}$$

Within the analysis code, a value for λ is estimated for each experiment and stored as a list. The calculated values of the negative log likelihood estimate is also stored in an array to keep track of those values. Lastly, each data point is stored in a list so that we can visualize the spread of the data in tandem with the rate parameter estimations.

In order to estimate the rate parameter λ for a whole data set, rather than each experiment, the analysis utilizes and compares two different estimation methods that make use of the data stored in various lists as described in the earlier paragraph. These methods also allow us to estimate the uncertainty or error within our analysis to quantify the confidence in the final result. The first method is to create a histogram of the λ estimations and analyze the spread of the data. This distribution should peak at the value of the true rate parameter (λ_{true}). The width of this distribution will allow us to estimate the uncertainty by quantifying the variance and the 1σ standard deviation from the distribution's mean. The next section, 4 will describe how varying different parameters within the data set can affect these uncertainty calculations. The second method is to plot the negative log likelihood versus the rate parameter λ and ascertain which value of λ minimizes the negative log likelihood curve. In order to calculate these values, the code takes each experiments estimated parameter and calculates a negative log likelihood result for the complete data set. This is very similar to the initial method of calculating a λ for each experiment but instead of estimating a lambda based off of the data in a single experiment, it is taking a lambda and computing its negative log likelihood result across the entire data set. A similar method with subtle differences. These two methods the lambda histogram and the negative log likelihood curve should give roughly the same result which will be outputted for comparison in a table by the script.

In addition to these two methods, since the Poisson function is a well-defined and well-behaved function, we can compare our maximum likelihood estimations to the analytical solution to the Poisson distribution.

In this case, the most likely rate parameter is also equal to the average of the data points [7].

$$\lambda = \overline{X} = \left(\frac{\sum_{i=1}^{N_{meas}} X_i}{N_{meas}} \right) \tag{9}$$

As an example, for a data set with the parameters, $\lambda = 4$, $N_{exp} = 1000$, and $N_{meas} = 1000$, the analysis code with output the following plots Fig. 3 and tabulated values Fig. 4. The plots visually show the distribution of the data, the lambda histogram distribution, the negative log likelihood curve, and finally a Poisson distribution curve for the data given the true rate parameter. The table shows the computed values for the estimated rate parameter for the simulated data set for each aforementioned method (Lambda histogram, Negative log likelihood curve minimization, and finally the analytically derived average) as well as any associated variances or errors determined.

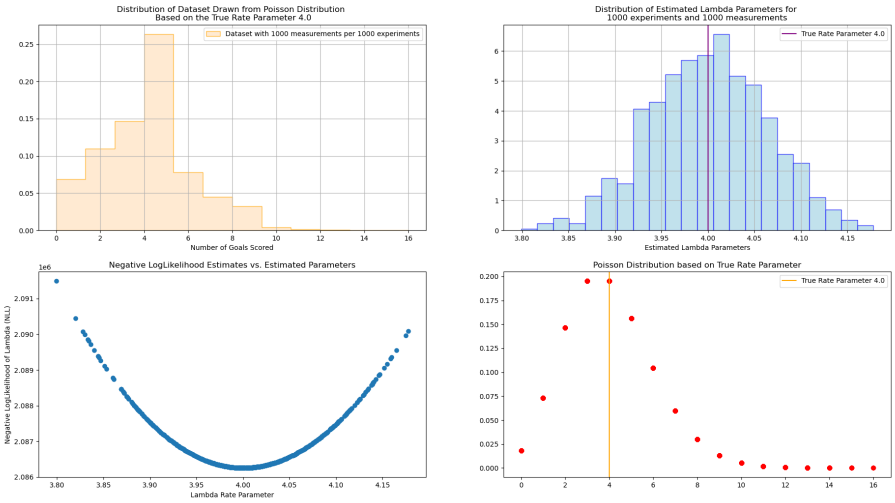

Figure 3: Output analysis plots for a simulated data set which was generated off of the parameters: rate $\lambda = 4$, $Nexp = 1000$, $Nmeas = 1000$. The top left panel shows a histogram of the data points. The top right panel shows a histogram of the estimated rate parameters which were calculated for each experiment. This histogram should peak at the true value. The bottom left panel shows a comparable plot of the parameter estimations versus the negative log likelihood values. The estimated rate parameter for the dataset should be the λ that minimizes this curve. Lastly, the bottom right panel shows a depiction of the Poisson distribution for the data with the true rate indicated by a vertical line.

Description	Value
True Rate Parameter for Data	4.0
Number of Experiments	1000
Number of Measurments	1000
Lambda Histogram Estimated Mean:	4.000369212217579
Lambda Histogram Estimated Variance:	0.004018757633934757
Lambda Histogram Estimated StDev:	0.06339367187610098
Minimum of LogLikelihood Curve Estimate of Lambda:	4.000000028511696
Analytically derived Average:	4.000328
Analytically derived StDev:	2.000081998319069

Figure 4: A comparison table of the analysis calculations for a simulated data set which was generated off of the parameters: rate $\lambda = 4$, $Nexp = 1000$, $Nmeas = 1000$. The first section of this table shows characteristics of the data set ($\lambda, Nexp, Nmeas$). The following section shows the estimated mean (rate parameter), variance, and standard deviation for the histogram technique. The next section shows the rate parameter if we were to minimize the negative log likelihood curve. The final section shows the results compared to the analytically derived results since those can be computed for a Poisson distribution.

This concludes the analysis that is determined for each and every data set put into the GoalDataAnalysis.py script. The following section will discuss what trends and tests were conducted to analyze the behavior of different parameters within the data set and their affect on the analysis outputs.

4 Analysis Testing

In addition to simply analyzing a single, simulated data set, there are many other questions that can be analyzed about the behavior of this model and analysis methods. The questions I decided to assess are as follows:

1. How does the accuracy of the estimated rate parameter change as the number of measurements changes within a data set?
2. How does the precision of the estimated rate parameter change as the number of experiments changes?
3. How does the analysis react to data that is generated with extreme values?

For each of these tests, different data sets were generated to help demonstrate what changes, if any, are present from these changes. These different simulated experiments will help to assess how well the rate parameter can be estimated from that data.

4.1 Varying the Number of Measurements

In this test, the goal was to assess how the accuracy of the rate parameter changes as the number of measurements per experiments increases. In order to test this, I generated data sets that had $N_{meas} = 10, 100, 1000, 10000$. These data sets each had the same number of experiments ($N_{exp} = 1000$) and same true rate value ($\lambda = 4$). The resulting data tables and plots are shown in **Figures** 5, 6, 7, 8, and 9.

Description	Value
True Rate Parameter for Data	4.0
Number of Experiments	1000
Number of Measurments	10
Lambda Histogram Estimated Mean:	4.006307595255428
Lambda Histogram Estimated Variance:	0.395084435149489
Lambda Histogram Estimated StDev:	0.6285574239077039
Minimum of LogLikelihood Curve Estimate of Lambda:	4.0
Analytically derived Average:	4.0027
Analytically derived StDev:	2.000674886132177

(a) $N_{meas} = 10$

Description	Value
True Rate Parameter for Data	4.0
Number of Experiments	1000
Number of Measurments	100
Lambda Histogram Estimated Mean:	4.0030923235509155
Lambda Histogram Estimated Variance:	0.03865662003455534
Lambda Histogram Estimated StDev:	0.19661286843580544
Minimum of LogLikelihood Curve Estimate of Lambda:	4.0
Analytically derived Average:	4.00233
Analytically derived StDev:	2.000582415198134

(b) $N_{meas} = 100$

Description	Value
True Rate Parameter for Data	4.0
Number of Experiments	1000
Number of Measurments	1000
Lambda Histogram Estimated Mean:	4.000369212217579
Lambda Histogram Estimated Variance:	0.004018757633934757
Lambda Histogram Estimated StDev:	0.06339367187610098
Minimum of LogLikelihood Curve Estimate of Lambda:	4.000000028511696
Analytically derived Average:	4.000328
Analytically derived StDev:	2.000081998319069

(c) $N_{meas} = 1000$

Description	Value
True Rate Parameter for Data	4.0
Number of Experiments	1000
Number of Measurments	10000
Lambda Histogram Estimated Mean:	4.000281316427376
Lambda Histogram Estimated Variance:	0.0003644668381668757
Lambda Histogram Estimated StDev:	0.019091014592390727
Minimum of LogLikelihood Curve Estimate of Lambda:	4.000299360585067
Analytically derived Average:	4.0002797
Analytically derived StDev:	2.0000699237776662

(d) $N_{meas} = 10000$

Figure 5: A comparison table of the analysis calculations for a simulated data set which was generated off of the parameters: rate $\lambda = 4$, $Nexp = 1000$, $Nmeas = 10(a), 100(b), 1000(c), 10000(d)$. The first section of this table shows characteristics of the data set ($\lambda, Nexp, Nmeas$). The following section shows the estimated mean (rate parameter), variance, and standard deviation for the histogram technique. The next section shows the rate parameter if we were to minimize the negative log likelihood curve. The final section shows the results compared to the analytically derived results since those can be computed for a Poisson distribution.

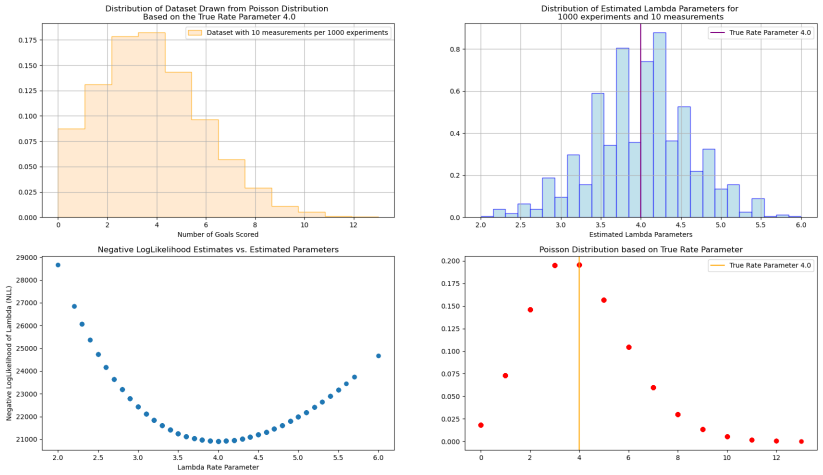

Figure 6: Output analysis plots for a simulated data set which was generated off of the parameters: rate $\lambda = 4$, $Nexp = 1000$, $Nmeas = 10$. The top left panel shows a histogram of the data points. The top right panel shows a histogram of the estimated rate parameters which were calculated for each experiment. This histogram should peak at the true value. The bottom left panel shows a comparable plot of the parameter estimations versus the negative log likelihood values. The estimated rate parameter for the data set should be the λ that minimizes this curve. Lastly, the bottom right panel shows a depiction of the Poisson distribution for the data with the true rate indicated by a vertical line.

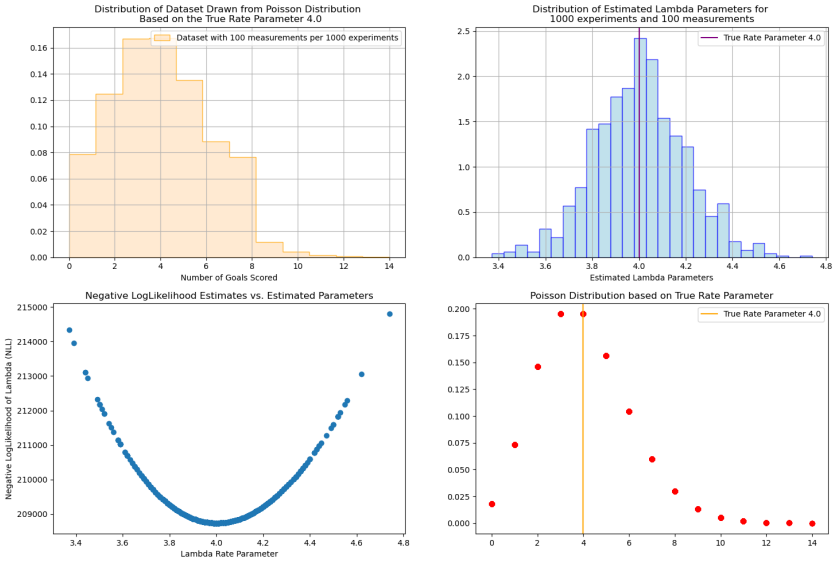

Figure 7: Output analysis plots for a simulated data set which was generated off of the parameters: rate $\lambda = 4$, $Nexp = 1000$, $Nmeas = 100$. Please see Fig. 6 for a full description on what each plot depicts.

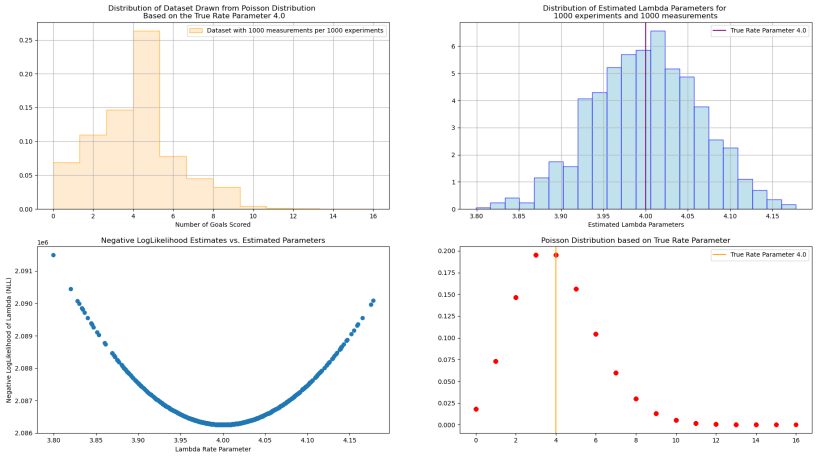

Figure 8: Output analysis plots for a simulated data set which was generated off of the parameters: rate $\lambda = 4$, $Nexp = 1000$, $Nmeas = 1000$. Please see Fig. 6 for a full description on what each plot depicts.

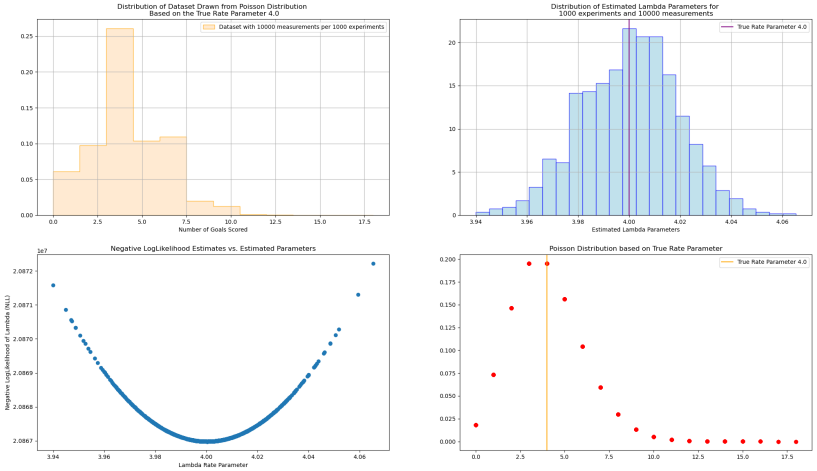

Figure 9: Output analysis plots for a simulated data set which was generated off of the parameters: rate $\lambda = 4$, $Nexp = 1000$, $Nmeas = 10000$. Please see Fig. 6 for a full description on what each plot depicts.

Thus from these figures we can make a conclusion that as you increase the number of measurements in the data set, irrespective of the number of experiments, the estimation of the rate parameter λ becomes more and more accurate with uncertainty decreasing.

4.2 Varying the Number of Experiments

In this test, the goal was to assess how the precision of the rate parameter changes as the number of measurements per experiments increases. In order to test this, I generated data sets that had $N_{exp} = 10, 100, 1000, 10000$. These data sets each had the same number of measurements ($N_{meas} = 1000$) and same true rate value ($\lambda = 3$). The resulting data tables and plots are shown in Figures 10, 11, 12, 13, and 14. By varying this parameter we are able to notice is that the histogram of estimated rate parameter values for each experiment becomes a more well defined distribution around the true mean. While this method does not increase our accuracy or confidence in the overall estimated parameter, it does create a much smoother function for further calculations to be conducted on. Each new measurement adds a data point for the histogram to place, so it is logical to think that the more data points you have to put in the histogram, the more well defined the distribution will be.

Description	Value
True Rate Parameter for Data	3.0
Number of Experiments	10
Number of Measurments	1000
Lambda Histogram Estimated Mean:	2.990799949716998
Lambda Histogram Estimated Variance:	0.0009147572755789045
Lambda Histogram Estimated StDev:	0.030244954547476253
Minimum of LogLikelihood Curve Estimate of Lambda:	2.9939999471539567
Analytically derived Average:	2.9911
Analytically derived StDev:	1.7294796905427945

(a) $N_{meas} = 10$

Description	Value
True Rate Parameter for Data	3.0
Number of Experiments	100
Number of Measurments	1000
Lambda Histogram Estimated Mean:	3.004518468883495
Lambda Histogram Estimated Variance:	0.0029963287022780195
Lambda Histogram Estimated StDev:	0.05473873128122372
Minimum of LogLikelihood Curve Estimate of Lambda:	3.004999833452502
Analytically derived Average:	3.00481
Analytically derived StDev:	1.7334387788439487

(b) $N_{meas} = 100$

Description	Value
True Rate Parameter for Data	3.0
Number of Experiments	1000
Number of Measurments	1000
Lambda Histogram Estimated Mean:	3.001240611684371
Lambda Histogram Estimated Variance:	0.0031458296073150513
Lambda Histogram Estimated StDev:	0.05608769568555167
Minimum of LogLikelihood Curve Estimate of Lambda:	3.000999997624483
Analytically derived Average:	3.001165
Analytically derived StDev:	1.732387081457259

(c) $N_{meas} = 1000$

Description	Value
True Rate Parameter for Data	3.0
Number of Experiments	10000
Number of Measurments	1000
Lambda Histogram Estimated Mean:	2.99893663884982
Lambda Histogram Estimated Variance:	0.0029450650928835474
Lambda Histogram Estimated StDev:	0.05426845393857786
Minimum of LogLikelihood Curve Estimate of Lambda:	2.9989998546422227
Analytically derived Average:	2.9989621
Analytically derived StDev:	1.7317511657279172

(d) $N_{meas} = 10000$

Figure 10: A comparison table of the analysis calculations for a simulated data set which was generated off of the parameters: rate $\lambda = 3$, $Nmeas = 1000$, $Nexp = 10(a), 100(b), 1000(c), 10000(d)$. The first section of this table shows characteristics of the data set ($\lambda, Nexp, Nmeas$). The following section shows the estimated mean (rate parameter), variance, and standard deviation for the histogram technique. The next section shows the rate parameter if we were to minimize the negative log likelihood curve. The final section shows the results compared to the analytically derived results since those can be computed for a Poisson distribution.

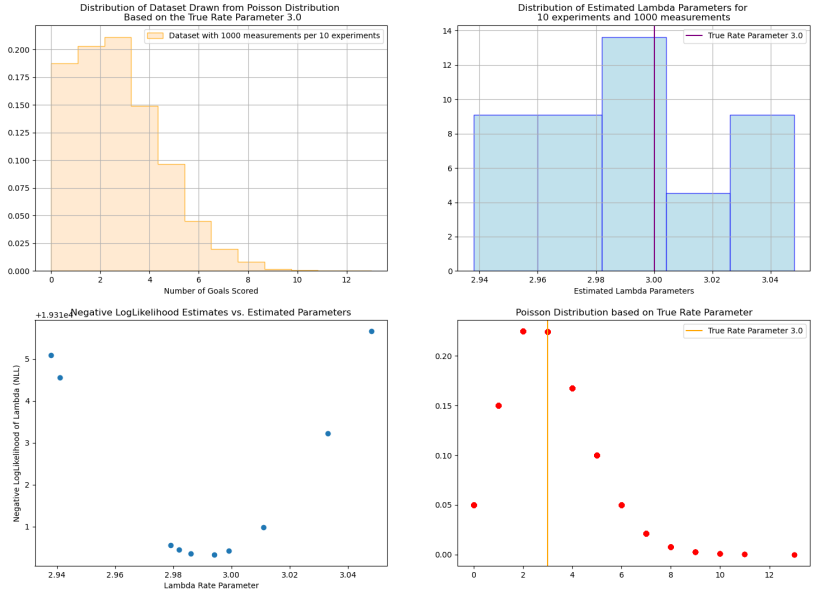

Figure 11: Output analysis plots for a simulated data set which was generated off of the parameters: rate $\lambda = 3$, $Nexp = 10$, $Nmeas = 1000$. Please see Fig. 6 for a full description on what each plot depicts.

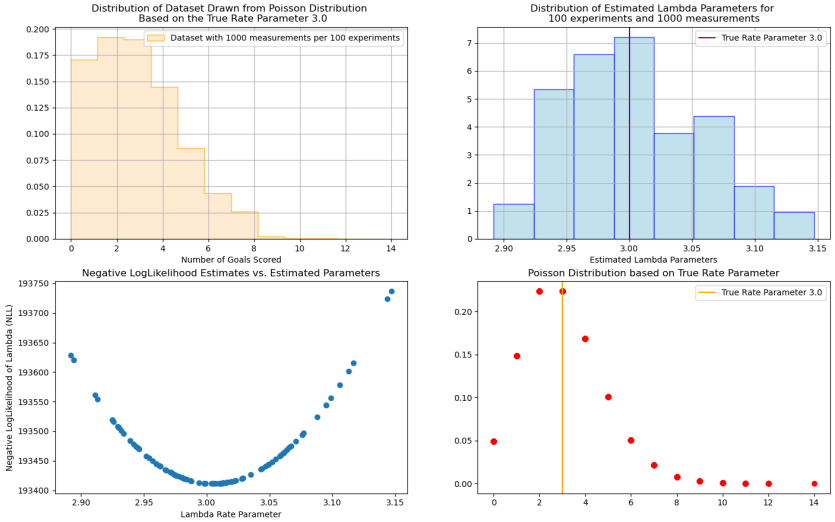

Figure 12: Output analysis plots for a simulated data set which was generated off of the parameters: rate $\lambda = 3$, $Nexp = 100$, $Nmeas = 1000$. Please see Fig. 6 for a full description on what each plot depicts.

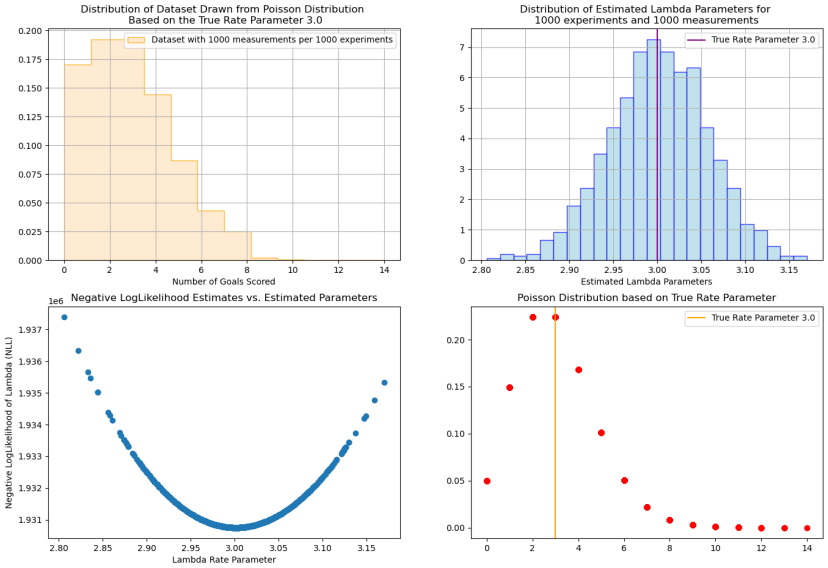

Figure 13: Output analysis plots for a simulated data set which was generated off of the parameters: rate $\lambda = 3$, $Nexp = 1000$, $Nmeas = 1000$. Please see Fig. 6 for a full description on what each plot depicts.

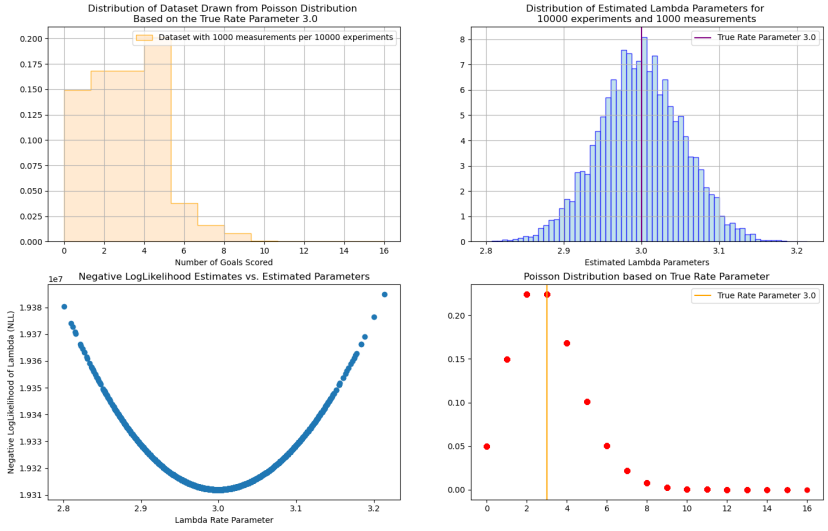

Figure 14: Output analysis plots for a simulated data set which was generated off of the parameters: rate $\lambda = 3$, $Nexp = 10000$, $Nmeas = 1000$. Please see Fig. 6 for a full description on what each plot depicts.

4.3 Extreme Values of the Rate Parameter λ

Finally, in this last testing measure, the aim is to see how the analysis methods react to analyzing data that is generated from "extreme" values of the rate parameter λ. For these data sets the number of experiments and measurements were held constant and both were assigned a value of 1000. The true rate parameters that were tested were 5, 50, and 500. Each of these data sets were then analyzed by the analysis code. Once again, the output tables and plots are shown below in Figures 15, 16, 17, and 18.

After conducting this test, it was found that as the value of the true rate parameter increases, the variance and standard deviation for the estimation also increased. However, the estimations were still very close to the true value as seen in Figure 15. Additional, it can be seen that at high values of lambda the distribution of the Poisson does appear to reflect more of a Gaussian or normal curve which is to be expected. Finally, it should be noted that within the context of a soccer team's performance and average rate of 5 goals per game is already rather stellar performance. In reality, a rate parameter of 50 or 500 is rather ludacris, but it is a good check to see how the model and analysis behaves at these values.

Description	Value
True Rate Parameter for Data	5.0
Number of Experiments	1000
Number of Measurments	1000
Lambda Histogram Estimated Mean:	5.000702396731414
Lambda Histogram Estimated Variance:	0.004767129210191625
Lambda Histogram Estimated StDev:	0.06904440028120763
Minimum of LogLikelihood Curve Estimate of Lambda:	4.999999564932427
Analytically derived Average:	5.00039
Analytically derived StDev:	2.236155182450449

(a) $N_{meas} = 10$

Description	Value
True Rate Parameter for Data	50.0
Number of Experiments	1000
Number of Measurments	1000
Lambda Histogram Estimated Mean:	49.998787741513105
Lambda Histogram Estimated Variance:	0.0515346599296857
Lambda Histogram Estimated StDev:	0.22701246646315637
Minimum of LogLikelihood Curve Estimate of Lambda:	49.997975596368
Analytically derived Average:	49.998364
Analytically derived StDev:	7.070952128249774

(b) $N_{meas} = 100$

Description	Value
True Rate Parameter for Data	500.0
Number of Experiments	1000
Number of Measurments	1000
Lambda Histogram Estimated Mean:	499.98245464115115
Lambda Histogram Estimated Variance:	0.515801273477303
Lambda Histogram Estimated StDev:	0.7181930614238089
Minimum of LogLikelihood Curve Estimate of Lambda:	499.98392197564266
Analytically derived Average:	499.982015
Analytically derived StDev:	22.360277614555685

(c) $N_{meas} = 1000$

Figure 15: Comparison tables of the analysis calculations for a simulated data set which was generated off of the parameters: rate $\lambda = 5, 50, 500$, $Nmeas = 1000$, $Nexp = 1000$. The first section of this table shows characteristics of the data set ($\lambda, Nexp, Nmeas$). The following section shows the estimated mean (rate parameter), variance, and standard deviation for the histogram technique. The next section shows the rate parameter if we were to minimize the negative log likelihood curve. The final section shows the results compared to the analytically derived results since those can be computed for a Poisson distribution.

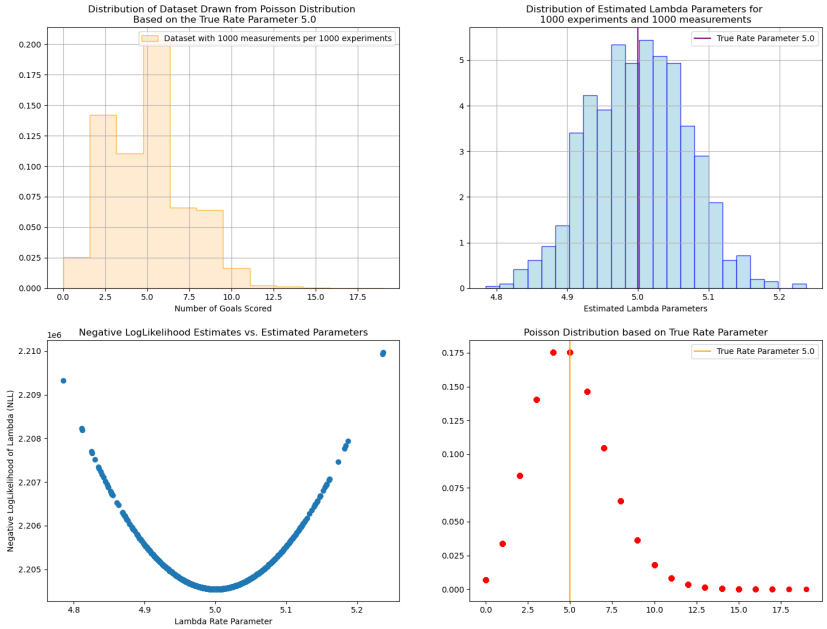

Figure 16: Output analysis plots for a simulated data set which was generated off of the parameters: rate $\lambda = 5$, $Nexp = 1000$, $Nmeas = 1000$. Please see Fig. 6 for a full description on what each plot depicts.

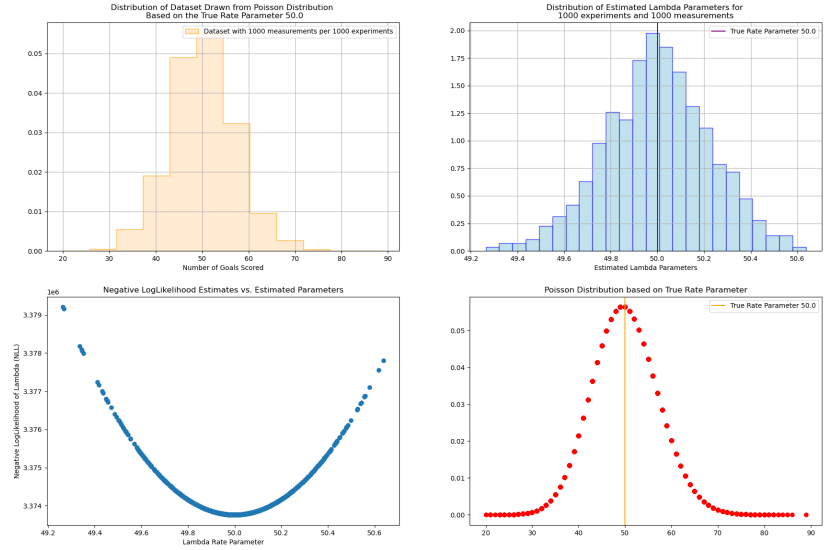

Figure 17: Output analysis plots for a simulated data set which was generated off of the parameters: rate $\lambda = 50$, $Nexp = 1000$, $Nmeas = 1000$. Please see Fig. 6 for a full description on what each plot depicts.

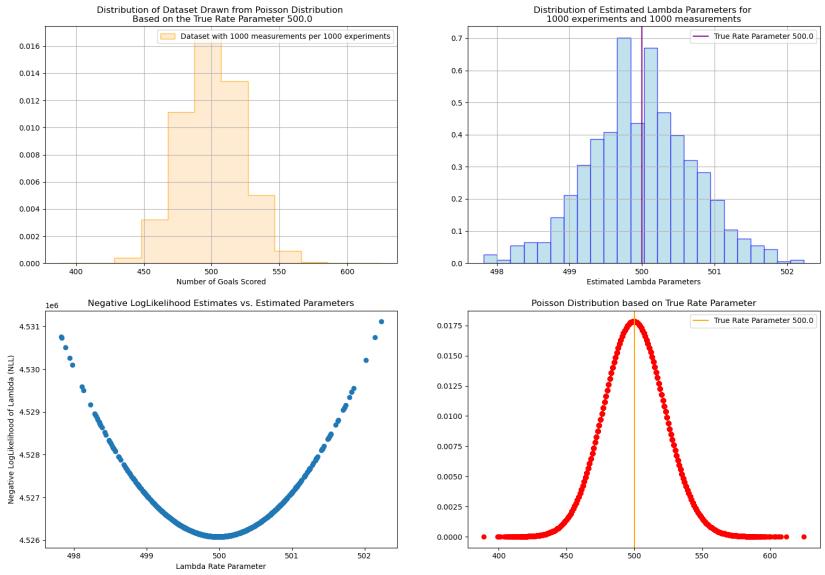

Figure 18: Output analysis plots for a simulated data set which was generated off of the parameters: rate $\lambda = 500$, $Nexp = 1000$, $Nmeas = 1000$. Please see Fig. 6 for a full description on what each plot depicts.

5 Conclusion

Overall, this project aimed to estimate the rate parameter based solely off of the simulated data set from this experiment. This is a great improvement upon the first two projects as it tests a continuous range of values – those acceptable to a Poisson distribution – rather than comparing two discrete values of the parameter against each other as we saw in hypothesis testing. In order to perform this analysis, I employed the technique of maximum likelihood estimation, or equivalently, minimization of the negative log likelihood estimation. This allows for an estimation of the rate parameter to be made for each experiment. Beyond simply generating the data and analyzing each experiments most likely rate parameter, I also employed more techniques to estimate the mean for the entire data set. This was done by quantifying the spread of the parameter estimation data in a histogram. Additionally, a negative log likelihood curve was calculated and plotted for each data point across the range of potential rate parameters — whose minimum should equal the true rate parameter. These methods were successful and sufficient for the analysis of a single data set, but three additional questions were considered. First, how the number of measurements affects the data set's analysis for which it was concluded that the more measurements are done the more accurate the parameter estimation is. The second question considered how the number of experiments affected the analysis. For this, it was found that the more experiments that were included in the data set did not make the parameter estimation more accurate, but did help make the distribution of lambdas more precise and well defined. Lastly, it was considered how the range of true values of lambda might affect the analysis measures. For this, multiple values of the rate parameter were tested and it was found that the analysis is less confident in estimations that are being conducted at very high values of lambda, but that the estimations were still essentially correct.

This code effectively demonstrates how an essential parameter of a model's distribution can be estimated from data alone which is an incredibly useful result for working with real world data since the true rate parameter would be unknown for a raw, observed data set. When considering the application to the soccer team's performance, this analysis method would be much more efficient than testing discrete hypotheses. So as we've discussed in each project thus far, the next time you tune in to watch your favorite soccer team compete (or watch in person), ponder taking note of the goal data and try applying this method to that data set to see what the true average rate of goals for the team may be and best of luck as you try to quantify the performance of your soccer team this season!

6 Repository Link

GitHub Repository Link: `https://github.com/aelieber1/PHSX815_Project3`

References

[1] A. Kumar, *Poisson distribution explained with python examples*, Oct, 2021.

[2] J. Frost, *Poisson distribution: Definition amp; Uses*, May, 2022.

[3] O. Eaton, *Modelling the Distribution of Football Goals*,.

[4] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors,

SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python, *Nature Methods* **17** (2020) 261–272.

[5] *chapter 8: estimation of parameters and fitting of probability distributions*.

[6] B. Lindsey, *Understanding maximum likelihood estimation*, Nov, 2020.

[7] S. Towers, *Maximum likelihood estimation (MLE)*.