Data Analysis at LCLS

Cristina E. Garcia

STAR Intern

[2]Linac Coherent Light Source, SLAC National Accelerator Laboratory, 2575 Sand Hill Road, Menlo Park, CA 94025, USA.

[+]Contact: yoon82@stanford.edu

Introduction

Picture on the left illustrates a lysozyme protein crystal in real time. Picture on the right presents the same protein crystal under a microscope.

A protein's structure determines its function. In order to better understand a protein's function one must analyze in depth its atomic structure. Serial femtosecond crystallography is the process for which an ultrafast free-electron x-ray laser, like such at LCLS, diffracts a protein crystal into its atomic structural basis and takes pictures of protein crystals in motion. From these atomic structures, scientists can better understand the functions of the diffracted proteins. This is important because the protein structures obtained from this process can also provide valuable information that can help redesign medicinal drugs that will target diseases caused by certain proteins.

Research

Objectives

Objective 1: This research aims to analyze protein crystal diffraction data using programs from the crystallography toolbox (CCTBX) at LCLS to solve the protein structure.

Objective 2: A protein crystal diffraction pattern contains numerous strong peaks on top of Gaussian noise. We calculated the probability of finding a peak on random Gaussian images using mathematical theory and testing our theory with a computer stimulation.

Methods

For data analysis we manually ran python scripts from the CCTBX for indexing and integration. The data collected was inputted into an excel worksheet for analysis. The data was used to analyze the difference in data sets we collect by using different parameters for each run. Also, we check for completeness, number of observations, and the averaging of dark images.

Using the standard normal distribution function, we will calculate the probability of finding a peak in a set of images. We will use a python script to verify the derivation and accuracy of the mathematical theory. Also, we will investigate the sigma level needed to obtain a fixed number of peaks.

Mathematical Derivation of Probability

Based on a random Standard Normal distribution function with a fixed mean of 10 and a standard deviation of 3 we can calculate the probability of finding values that are equal to $(\mu + 3\sigma)$ and $(\mu + 2\sigma)$ right next to each other. This would indicate we have found a peak in a random array of images.

Table 1

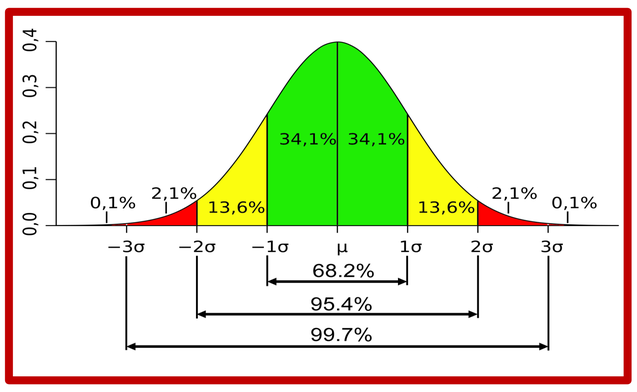

Table 1 presents the Standard Normal Distribution graph used as reference for the mathematical derivation of probability theory.

First, using the Normal Standard Distribution graph above we can find the probability of finding a value that is equal to $(\mu + 3\sigma)$.

$$P(\mu + 3\sigma) = 1 - (.997) = .003$$

Since we are only looking for the top values about 3σ, we have to divide this probability by 2 to get the actual probability of finding that value.

So, $P(\mu + 3\sigma) = \frac{.003}{2} = .0015$

This means that in an array of 1000x1000 we would expect around 1,500.

Working off this probability we can calculate the probability of finding at least one 2σ pixel next to the 3σ pixel. The probability of finding "at least one" 2σ pixel is equal to one minus the complement of probability of the event not occurring.

$$P(\mu + 2\sigma) =$$
$$1 - (1 - .0227)^4 = 1 - (.9773)^4$$
$$1 - .912 = .088$$

Therefore,

$$P(\mu + 3\sigma) \times P(\mu + 2\sigma) =$$
$$.0015 \times .088 = .000132$$

This means that in an array of 1,000x1,000 we would expect around 132 2 sigma pixels.

Figure 1

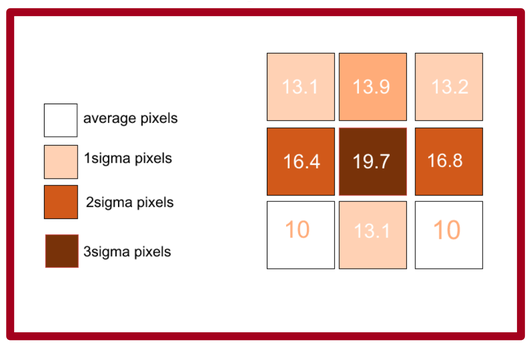

Figure 1 illustrates a portion of the matrix of images around a 3sigma pixel found.

Python Stimulation

The mathematical theory was verified through a python stimulation. First the program generates a random array of values. Each value represents how dark an image is. In order to control the randomness of the values we used the Standard Normal Distribution function which was also a fundamental part of our mathematical derivation. To find a peak, we must find joints of high values like nineteen and sixteen. So first the program locates the values that are greater than or equal to nineteen, which are values that are three sigma away from the mean. It prints out location, actual value, and counts the total found. Then it analyzes the values around it in search of values that are greater than or equal to sixteen, these are the values that are two sigma's away from the mean. Once the program finds these joints, it prints out the total number of peaks and calculates its percentage from the total.

Table 2

Trial:	(μ+3σ):	Percentage f	(μ+2σ)+(μ+3σ):	Percentage:
1	1285	0.129 %	105	0.0105 %
2	1315	0.135 %	125	0.0125 %
3	1347	0.135 %	108	0.0108 %
4	1313	0.131 %	121	0.0121 %
5	1344	0.134 %	116	0.0116 %
6	1373	0.137 %	133	0.0133 %
7	1382	0.138 %	128	0.0128 %
8	1317	0.132 %	117	0.0117 %
9	1445	0.145 %	130	0.0130 %
10	1345	0.135 %	110	0.0110 %

Table 2 shows the results of each trial done on the stimulating program.

The computer stimulation also allowed us to solve for the sigma level needed to only find five peaks in an automated peak finding threshold, which was 2.25sigma.

Conclusions

The probability obtained from the mathematical theory was coherent with the values presented in the python script stimulation. This verified the accuracy of our theory and also allowed us to determine the sigma level necessary in an automated peak finding threshold.

In order to determine the efficiency of the data being presented by the images collected we must look at the amount of false peaks it presents. Through our python stimulation we found that using a 2.25 sigma we could locate an average of 5 false peaks.

Acknowledgments

This project was made possible by the funding provided by the STAR Teacher Research Program.

Date: 08/05/2017