Monte-Carlo and Statistical Distributions

TURO, B[*, 1]

[1] *Physics Department, Florida International University*
[*] *Corresponding author: bturo001@fiu.edu*

Compiled May 10, 2021

More and more we are observing an inability to calculate analytical solutions to integrals; this is detrimental when a numerical solution is required to integrate continuous or discrete probability distributions, as often is done in particle physics. Described is a powerful and efficient computational method for solving numerical integration by relying on random sampling. Due to the difficulty in gathering a set of truly random data, we demonstrate the use of pseudo-random data that can be obtained from common probability distributions.

1. INTRODUCTION

A. Monte-Carlo

The Monte-Carlo technique relies on the fact that random sampling is a uniform distribution. Cut a square piece of paper and enclose a circle. Now, blindfold yourself and let 10,000 darts fall onto the board. By counting the number of darts that land inside the circle compared to the number that land outside we can actually calculate the area of the circle. That is:

$$\frac{A_c}{A_s} = \frac{N_c}{N_s} = \frac{\pi r^2}{(2r)^2} = \frac{\pi}{4} \tag{1}$$

This idea can be generalized to much more than the area of a circle. Later we shall touch upon calculating the 'volume' of a N-dimension sphere, which will generalize to solving any integral.

B. Probability Distributions

Often times the probability distribution that comes to mind is the Gaussian distribution, this continuous distribution is most commonly used due to the Central Limit Theorem which states: if you have a population with mean μ and standard deviation σ and take sufficiently large random samples from the population with replacement, then the distribution of the sample means will be approximately normally distributed. The most important take away from these distributions is that due to the conservation of probabilities, a random set can be drawn from probability density deviates, that is:

$$|p(r)\Delta r| = |P(x)\Delta x| \therefore \int_{r=0}^{r} 1dr = \int_{x=-\inf}^{x} P(x)dx = r \tag{2}$$

The method of obtaining random samples r mentioned above is called the transformation method. Another method is the rejection method, which is what was used to calculate the area of a circle in equation 1. We simply reject unwanted points.

2. MONTE-CARLO AND SPHERES

For a N-dimensional sphere of $r = 1$ the inequality 3 must be satisfied.

$$\sum_{i=0}^{N} x_i^2 \leq 1 \tag{3}$$

Therefore, if we choose random x_i we can simply check if the inequality is met in order to consider the point to be within the sphere. Doing this will produce results presented in fig. 1 To determine the accuracy of our rejection method, let us calculate the value of π and compare

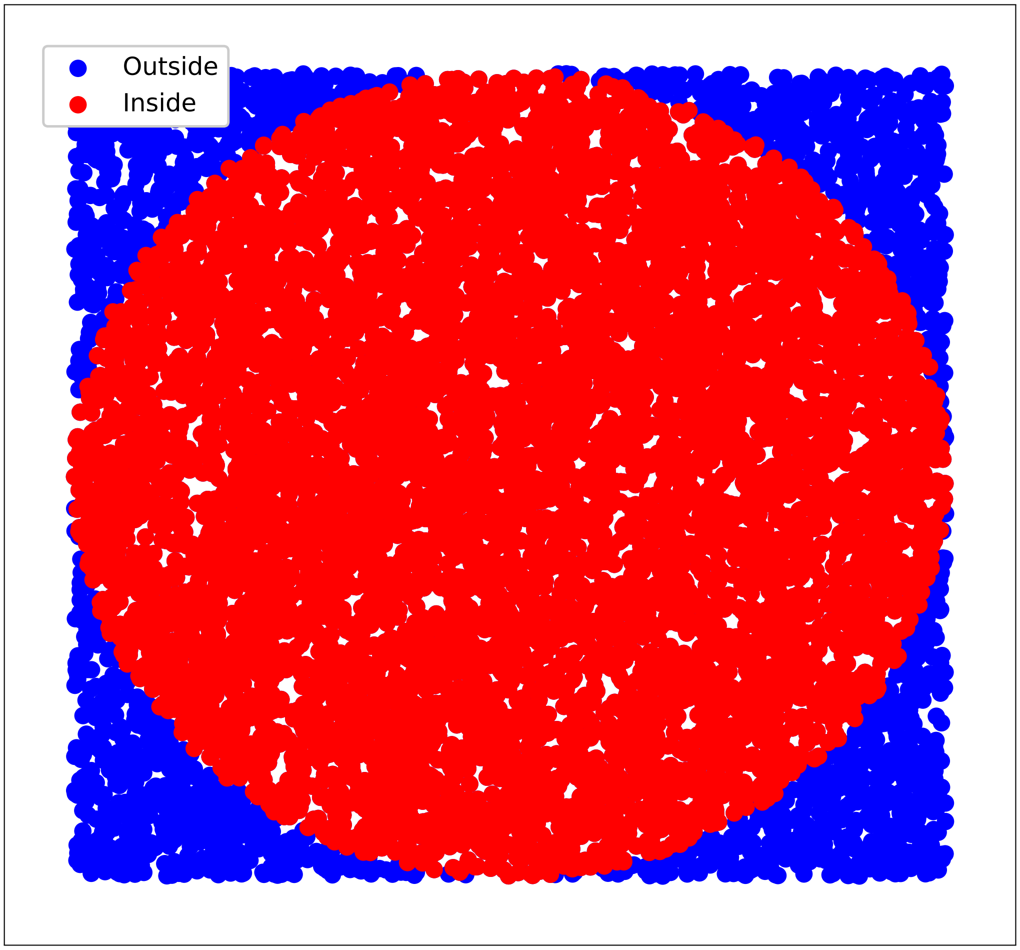


Fig. 1. Rejection method being used to calculate the area of a circle, red are random points that landed inside the circle and blue are random points that landed outside.

to accepted values. Using equation 1 we have $\pi = 4\frac{N_c}{N_s}$ Since this process is a binomial one, in other words there exists a probability of success and a probability of failure, we can use the binomial uncertainty in order to calculate the uncertainty of our calculation depending on the number of samples used.

$$\sigma_p = \sqrt{\frac{p(1-p)}{N_s}} \tag{4}$$

Where $p = \frac{N_c}{N_s}$

A. Calculating π

A.1. Uncertainties

Propagating the uncertainty using Eqn. 4 we get

$$\sigma_\pi = 4\sqrt{\frac{p(1-p)}{N_s}}$$

This generates a confidence interval with 68% confidence. The formula can be used to calculate the number of iterations N_s necessary to have a certain uncertainty of π when $4p = \pi$

$$n = \frac{16 \cdot p(1-p)}{\sigma^2} = \frac{16 \cdot \frac{\pi}{4}(1-\frac{\pi}{4})}{\sigma^2} \tag{5}$$

A.2. Results

To get an uncertainty of 1%, using equation 5, we must use nearly 27,000 random points. Doing this we get a result of

$$\pi = 3.14 \pm 0.01$$

For uncertainties of 0.1%, 0.01% and 0.00001% we must use 2.7 million, 270 million and 270 trillion points, respectively. The value of π for an uncertainty of 0.1% we get a result of

$$\pi = 3.142 \pm 0.001$$

and for an uncertainty of 0.01% we get a result of

$$\pi = 3.1415 \pm 0.0001$$

It would be silly to attempt the calculation for 270 trillion points. With a code complexity of O(n) [1], a linear algorithm, the time it would take is in the scale of decades.

B. Calculating N-Sphere Volumes

Using Eqn. 3 we can calculate the "volume" of N-dimensional spheres. Let's use a simple algorithm for this:

Algorithm 1. N-Sphere algorithm

```
procedure VOLUME(N, μ)
    for i = 1 to N do
        if Σ_{i=0}^{N} x_i^2 ≤ 1 then
            insideCounter + = 1
    totalCounter + = 1
    result = insideCounter/totalCounter          ▷ This is the ratio p = N_s/N_c
```

Now to get the volume all we must do is account for the circle to square ratio as shown in Eqn. 6

$$V = 2^N \frac{N_c}{N_s} \tag{6}$$

B.1. Uncertainties

Our uncertainty stems from the binomial uncertainty presented in Eqn. 4. We simply propagate by multiplying 2^n

B.2. Results and confirmation

In order to theoretically calculate the volume of an N-Dimensional sphere we will use Eqn. 7, this equation can be proven by generalizing the integral calculation of a sphere and replacing the integral with a gamma function.

$$V_n(R) = \frac{\pi^{\frac{n}{2}}}{\Gamma\left(\frac{n}{2} - 1\right)} R^n \tag{7}$$

The following data is presented with $N_s = 100,000$. Keep in mind that $R = 1$

N	Calculated Volume	Uncertainty	Theoretical Value	Relative Error
1	2.00	0	2	0%
2	3.152	0.005	$\pi \approx 3.142$	0.328%
3	4.186	0.013	$\frac{4\pi}{3} \approx 4.189$	0.072%
4	4.937	0.023	$\frac{\pi^2}{2} \approx 4.935$	0.040%
5	5.256	0.037	$\frac{8\pi^2}{15} \approx 5.264$	0.152%

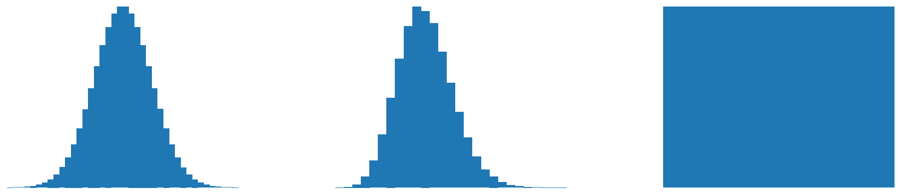
Relative error is calculated by using

$$\frac{|V_{true} - V_{observed}|}{V_{true}}$$

3. PROBABILITY DISTRIBUTION FUNCTIONS

When talking about probability distributions functions (PDFs) it is useful to use the concept of a deviate. But, before we do that let's graph a Gaussian 8, Poisson 9 and Uniform 10 distribution and compare them.

$$f(x;\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)}{\sigma}^2} \quad \textbf{(8)} \qquad f(x;\mu) = \frac{\mu^x e^{-\mu}}{x!} \quad \textbf{(9)} \qquad f(x) = \frac{1}{b-a} \quad \textbf{(10)}$$



The deviate is the interval between each bin presented in the histogram.
As mentioned in the introduction, the power of these distributions is the ability to obtain a randomized set of data. Using equation 2 we can see that all is needed is a linear transformation. Since this is commutative we will demonstrate the ability to generate the distributions from pseudo-random numbers.

A. Poisson Probability Distribution

First, we will utilize the rejection method in a similar fashion as before. Let's modify Eq. 2 by replacing the integral with a sum.

$$r = \sum_{x=0}^{x} P_p(x:\mu) = \sum_{x=0}^{x} \frac{\mu^x}{x!} e^{-\mu} \qquad \textbf{(11)}$$

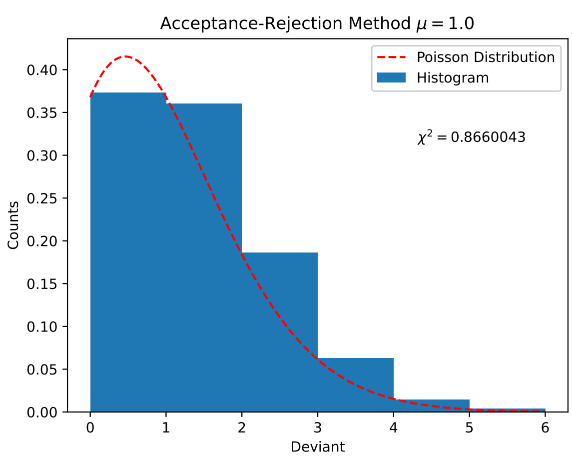
Then use the algorithm presented below Alg.2 to solve for x.

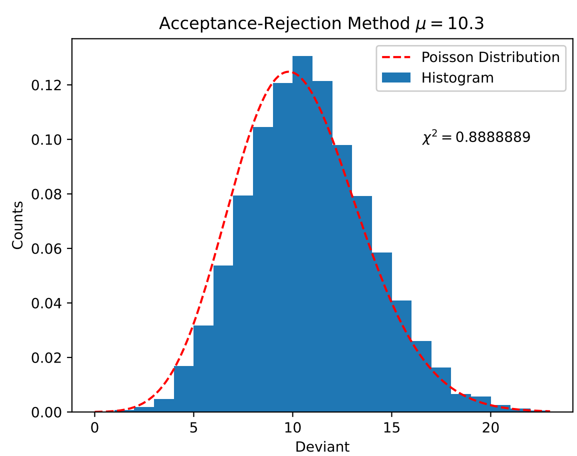
Algorithm 2. Poisson algorithm

```
procedure POISSON(N, μ)                ▷ Generate Poisson data set from uniform distribution
    f(x, μ) ←Poisson Function
    for i = 1 to N do
        x ← 0
        r ← random
        while f(x, μ) < r do
            x ← x + 1
        store x                        ▷ The Poisson deviate is the data set of x
```

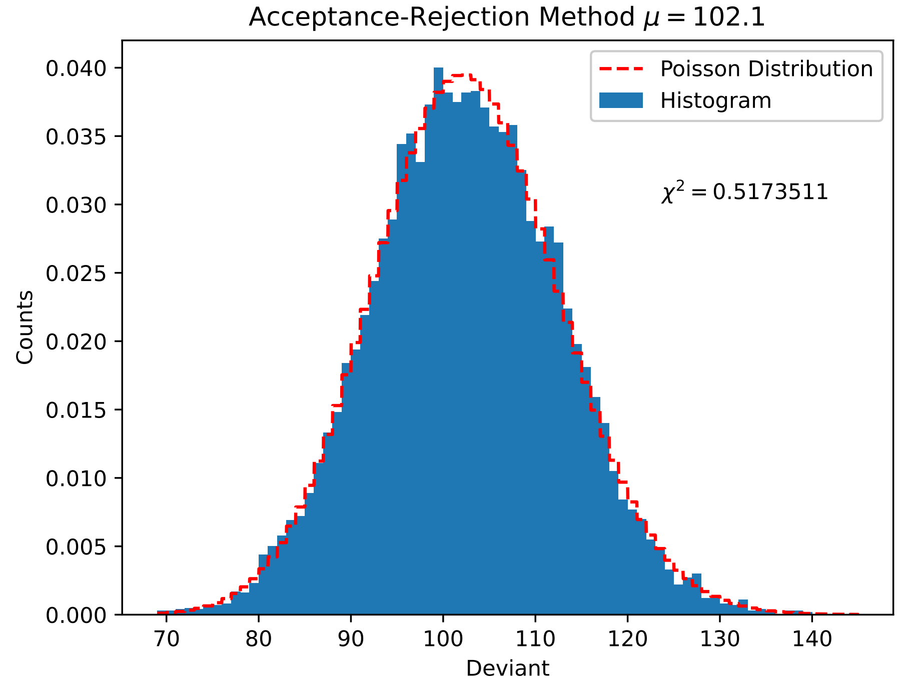
A.1. Results

This process was repeated for three values of μ: 1.0, 10.3, 102.1 with 10,000 deviates





Here a key property of the Poisson distribution is clearly visible: it is a discrete function for small values of n. As $n \to \infty$ we get a Gaussian distribution which is seen in the $\mu = 102.1$ data set. Due to the computational limitations involved with calculating Eq. 9 the Poisson Distribution plot is not as smooth as lower μ. Another consequence is seen when inspecting the χ^2 value, it is not reliable due to the limitations of our distribution plot.



B. Gaussian Probability Distribution

As previously seen, the rejection method is impressively powerful, however, limitations in computational power make it unreliable to use. In these cases, we can use the original transformation method alluded to in Eq. 2. Quickly, we realize there is a problem as the Gaussian function is rather difficult to integrate. However, a method developed by Box and Müller uses the two-dimensional Gaussian Distribution.[2]

$$f(z_1, z_2) = \frac{1}{2\pi} \exp\left(-\frac{(z_1^2 + z_2^2)}{2}\right) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z_1^2}{2}\right) \times \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z_2^2}{2}\right) \tag{12}$$

Further, from this equation the authors integrated and solved for Gaussian deviates z_1 and z_2 from two uniform random deviates r_1 and r_2.

$$z_1 = \sqrt{-2\ln r_1} \cos 2\pi r_2 \tag{13}$$

$$z_2 = \sqrt{-2\ln r_1} \sin 2\pi r_2 \tag{14}$$

The algorithm 3 is used to generate a Gaussian distribution from random deviates.

Algorithm 3. Gaussian Deviate Algorithm

```
procedure GAUSSIAN(N, μ, σ)                    ▷ Generate standard distribution
    f(x, μ, σ) ← Gaussian Function
    for i = 1 to N do
        r1, r2 ← random
        z1 ← sqrt(−2ln(r1)) cos(2πr2)
        z2 ← sqrt(−2ln(r1)) sin(2πr2)
        store z1, z2                           ▷ The Gaussian deviates are z1, z2
```

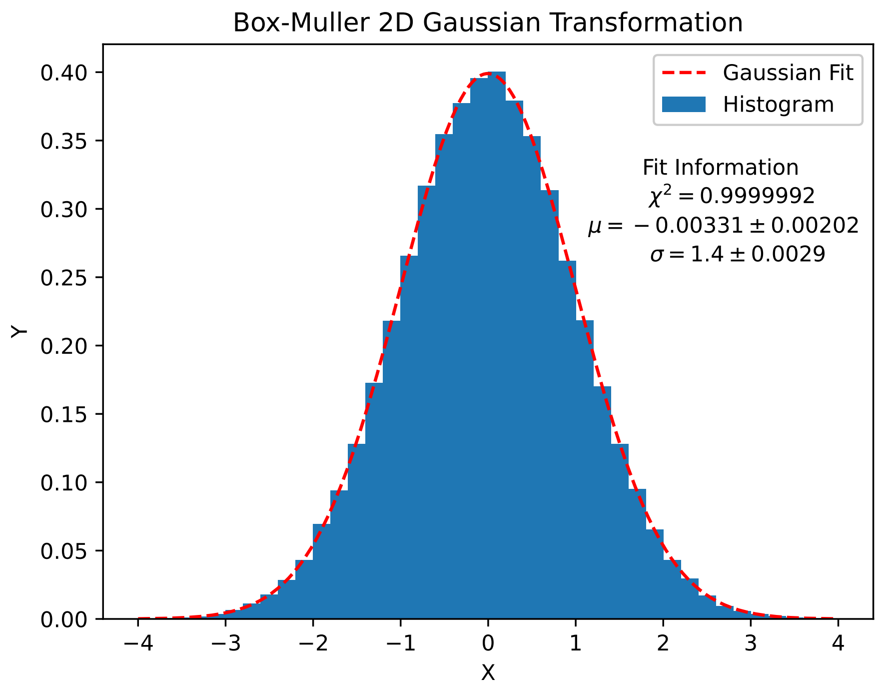


Fig. 2. A Histogram generated by Alg. 3 then fitted with a Gaussian PDF

B.1. Results

Plotting a histogram of all our z_1, z_2 deviates we get Fig. 2 Here, we see a 1:1 correlation with the Gaussian PDF, demonstrated by our χ^2 (a measure of correlation).

$$\chi^2 = \sum_{j=1}^{n} \frac{\left[g\left(x_j\right) - h\left(x_j\right)\right]^2}{h(x_j)} \tag{15}$$

B.2. Covariance Matrices and Uncertainties

Given a parent function $h(x)$ it is possible to estimate the correlation between another function $g(x)$ by simply squaring the difference and dividing by the expected output of the parent function.

$$\sigma_x^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2 \tag{16}$$

It is also possible to estimate the correlation between points of freedom or different variables by slightly adjusting Eqn. 16 This is presented as the covariance of the two variables and is estimated by Eqn. 17

$$\sigma(x,y) = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}) \tag{17}$$

Using matrices, we can create a Variance-Covariance Matrix by using Eqn. 18 as our definition.

$$C_{i,j} = \sigma(x_i, x_j) \tag{18}$$

In Fig. 2 we estimated the correlation between our Gaussian least squares fit and our histogram data, in the process of doing this we had three fitting parameters: μ, σ and the amplitude, A. To get the Variance-Covariance matrix of this fit to our data we simply estimate a three points of freedom variance-covariance matrix demonstrated by Eqn. 19

$$C = \begin{pmatrix} \sigma(x,x) & \sigma(x,y) & \sigma(x,z) \\ \sigma(y,x) & \sigma(y,y) & \sigma(y,z) \\ \sigma(z,x) & \sigma(z,y) & \sigma(z,z) \end{pmatrix} \tag{19}$$

Computing this for our histogram in fig. 2 using the least squares method we obtain the covariance matrix below. This can be used to calculate the total standard deviation or variance for the fit and help determine whether the fit correlates with the data or not.

Covariance Matrix		
3.3269e-06	-6.5484e-12	2.4250e-12
-6.5484e-12	6.6539e-06	3.3198e-06
2.4249e-12	3.3198e-06	4.9689e-06

For the sake of brevity, we will not dive further into the covariance matrix but instead, it will be left as motivation for the reader to explore.

4. CONCLUSION

In this paper, it was demonstrated how to calculate the value of an N-dimensional sphere using probability and a pseudo-random set and confirmed it by verifying the value of pi and the volume of a 3, 4 and 5D sphere. Additionally, it was explored how to obtain a Poisson and standard probability distribution from random sets by using a rejection method and a Box-Muller 2D Gaussian transformation. The results correlate almost 1:1 to what was expected. These methods are extremely powerful and will be explored further in the future.

REFERENCES

1. P. E. Black, "big-o notation"," https://www.nist.gov/dads/HTML/bigOnotation.html. Online; accessed 20 January 2021.
2. P. R. Bevington and D. K. Robinson, *Data reduction and error analysis for the physical sciences* (McGraw-Hill, 2010).
3. NIST, "Digital Library of Mathematical Functions," http://dlmf.nist.gov/5.19#E4 (2013). Online; accessed 20 January 2021.
4. University of Utah, "SciPy curve fitting," http://www.physics.utah.edu/~detar/lessons/python/curve_fit/node1.html. Online; accessed 20 January 2021.