Probability Theory and Stochastic Processes: Coursework

Nikita Korolev
MSc Software Engineering
Institute of Graduate Studies, Final International University

03.07.2023

1 Introduction

Probability theory and stochastic processes are both branches of mathematics that are concerned with computing the odds of certain events either happening or not. Their field of influence is not limited to just mathematics; probabilistic issues are very important in all sciences, giving to statistics. It allows to produce hypotheses, and based on results of statistical analysis of these results make conclusions on whether hypotheses are true in general.

One particular problem that arises quite often is analysis of confidence intervals for random samples; it allows us to compute a range, for which the true population mean lies (with the given probability or confidence level). In this very paper lies the description for such analysis, as well as for the software component produced to solve this problem.

JEstimator is a statistical CLI tool that can be used for visualization of normal distribution data and confidence interval computation. Complete source code along with description and executable jar can be found on project's github repository: JEstimator Github

2 Literature Review

In order to create JEstimator, a moderate literature and concept review has been carried out. It focuses on the main ideas of standard normal distribution and numerical methods of integration (used to evaluate the area under the normal curve).

2.1 Standard Normal Z-Distribution

Most scientists, statisticians and mathematicians in the community argue, that the most important probability distribution is *the standard normal distribution*[1]. It describes many processes/phenomena that happen in nature; some of them are meteorological experiments, rainfall studies, as well as some phenomena from the world of humans - errors in scientific measurements, etc. The PDF of the standard normal distribution is as follows:

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{\frac{-x^2}{2}}, \mu = 0, \sigma = 1, -\infty < x < +\infty$$

As shown on Figure 2.1, the graph of the CDF of standard normal distribution is a bell curved shape. Normal distribution, despite it's mathematical significance doesn't have a way to be integrated so that the integral is an analytical function (no easy way of getting CDF).

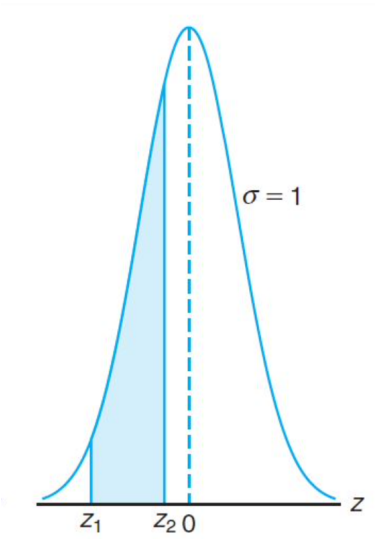

Figure 1: PDF of Standard normal distribution

However, certain mathematical techniques allow to compute approximated values for the area under the standard normal curve; these methods are called *numerical*.

2.2 Numeric Integration Methods

The most important and popular in mathematical community are the following three numerical methods: rectangular, trapezium and Simpson's. [2]

1) *Rectangular* - based on the idea of separating the area under the curve into thin rectangles/strips and summing them together to approximate the area under the function; has three variations, based on which side of the rectangle is used for calculations (left, right or middle):

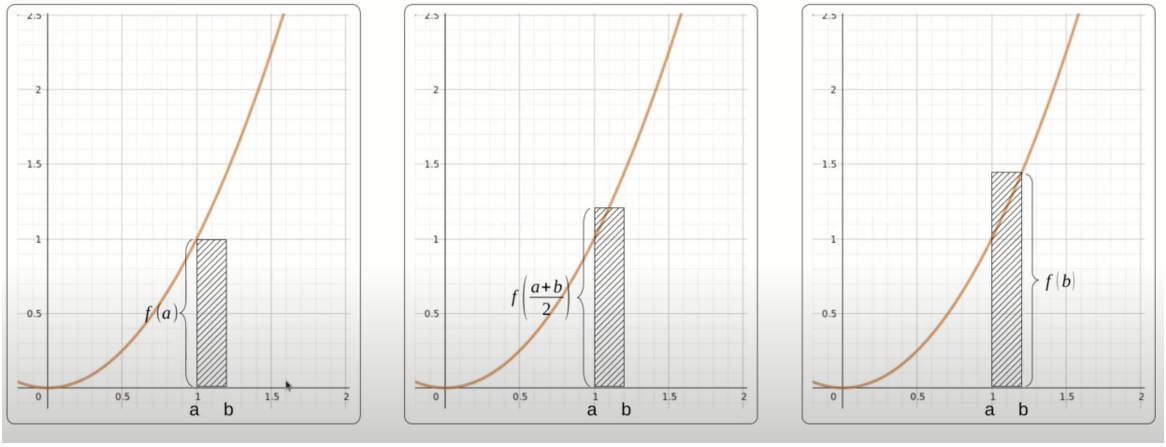

Figure 2: Rectangular methods

The formulas for methods in Figure 2.2 for the calculations of areas are as follows, with formulas for errors of these methods specified further below:

$$Left : \int_{x_1}^{x_2} f(x)dx \approx \sum_{i=0}^{n-1} f(x_1 + i \times \Delta x) \times \Delta x$$

$$Middle : \int_{x_1}^{x_2} f(x)dx \approx \sum_{i=0}^{n-1} f(x_1 + (i + \frac{1}{2}) \times \Delta x) \times \Delta x$$

$$Right : \int_{x_1}^{x_2} f(x)dx \approx \sum_{i=1}^{n} f(x_1 + i \times \Delta x) \times \Delta x$$

where $\Delta x = (b - a)/n$ is defined as the horizontal step for the entire interval from a to b with n being the number of steps 2.2.

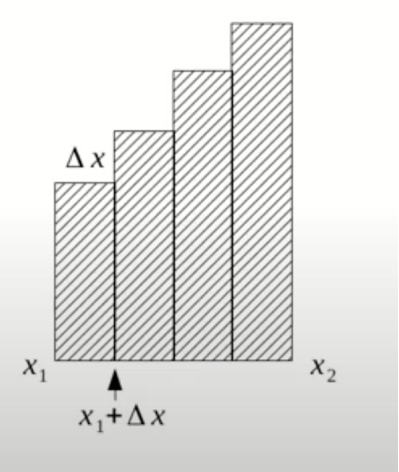

Figure 3: Horizontal step

$$Left/RightErr \leq |max(|f'(x)|) \times \frac{(x_2 - x_1)^2}{2n}|$$

$$MiddleErr \leq |max(|f''(x)|) \times \frac{(x_2 - x_1)^3}{24n^2}|$$

Thus it is clear, that the most accurate of the rectangular methods is the middle rectagnles, since the degree of it's error depends more on the number of divisions of the interval.

2) *Trapezium* - similar to Rectangular method, with the idea of using trapeziums instead of rectangles to approximate area (more accurate result for area). Thus, the problem of finding the area comes down to summing up the area under trapeziums (shown in Figure 2.2)

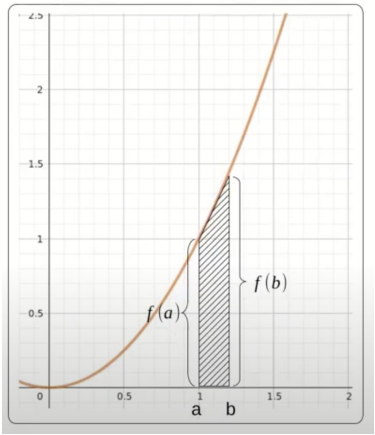

Figure 4: Trapezium method

$$\int_{x_1}^{x_2} f(x)dx \approx \sum_{i=0}^{n-1} \left(\frac{f(x_1+i\times\Delta x)+f(x_1+(i+1)\times\Delta x)}{2}\right) \times \Delta x$$

$$Err \leq |max(|f''(x)|) \times \frac{(x_2-x_1)^3}{12n^2}|$$

3) *Simpson's* - further improvement of rectangular and trapezium methods, based on using second order polynomials to approximate the area under the target function. This means that curvilinear trapezium (not regular) is being used for approxmation, which better suits the case and is to be of higher precision (as shown in Figure 2.2).

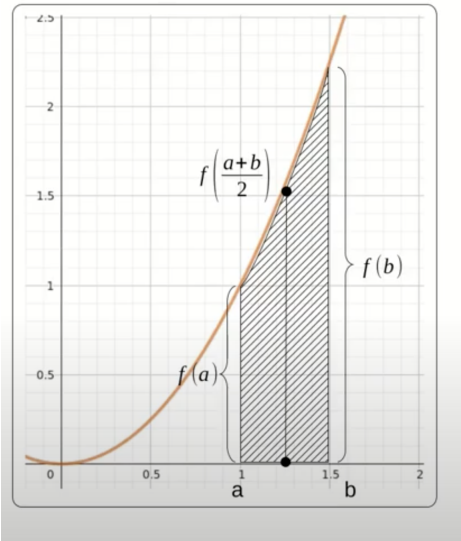

Figure 5: Simpson's Method

$$\int_{x_1}^{x_2} f(x)dx \approx \sum_{i=0}^{n-1} (f(x_1 + i \times \Delta x) + 4 \times (f(x_1 + (i + \tfrac{1}{2}) \times \Delta x)) + f(x_1 + (i + 1)\Delta x)) \times \frac{\Delta x}{6}$$

$$Err \leq |max(|f^4(x)|) \times \frac{(x_2-x_1)^5}{2880n^4}|$$

Therefore it can be noticed, that the most accurate method of the described three is Simpson's, followed by trapezium and Rectangular coming right in the end. Each of the method's has been defined as the general formula to be used inside the JEstimator.

3 Work Done and Results

This part of the report is concerned with giving the reviewer better understanding of what JEstimator's internal structure and how to run it with CMD arguments.

3.1 Software Requirements

In this section, the requirements for JEstimator are listed:

1) User should be able to produce the z-distribution (standard normal) table
2) User should be able to get the corresponding z-values ($\frac{\pm\alpha}{2}$), provided the confidence level
3) User should be able to get corresponding x-values (confidence interval computation)

3.2 Architecture

Despite the importance of such software, programming functions and various computations is rather trivial. The real art of software development process is to develop an efficient architecture, that will not cause significant issues at the evolution stage. JEstimator apart from providing the essential statistical functionality has been designed with architectural significance in mind.
The architecture of JEstimator is based around the following concepts: representing the standard normal data in a structure, development of methods for manipulating that structure, and providing a simple interface for supporting computations and data search.

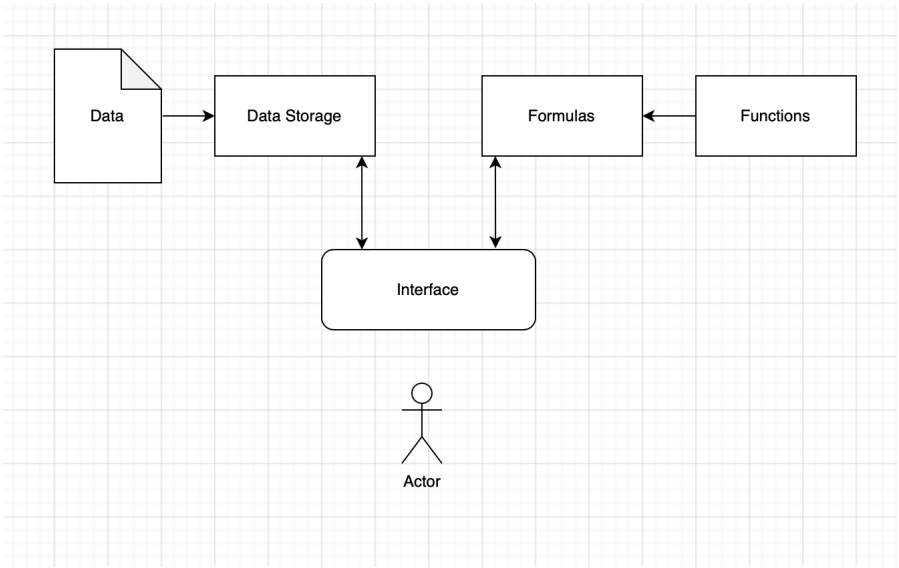

Figure 6: JEstimator's Architecture

Figure 3.2 shows the abstract architecture that JEstimator is implementing. Certain packages with semantic names have been formed to reflect the essence of the proposed architecture:

1) com.company.data - package stores the object that emulates the z-value in a table ObjectZ, with it's respective data structure for storage TableStorage;
2) com.company.formulas - package stores functional formulas, both for curves/distributions (Curves) and numerical methods (Formulas) and the corresponding parameters storage (FormulaParams) for numerical methods;
3) com.company.interfaces - package that helps user interact with formulas and data, containing methods for branch execution and console output (Actions), and various calculations (Calculations) needed to execute the key functionality from formulas and storage packages.

3.3 Implemented functionality

With respect to software requiremtns, and personal motivation of the developer (mrkorolev), certain functionality has been added to initial software requirements. As a result, the following functionality has been implemented:

1) Output z-distribution table to console
2) Output z-distribution table to a PDF file
3) Calculate the CDF-value of z-distribution for a given z-value
4) Calculate the confidence interval
5) Quit the software package

3.4 Installation

Here listed are the details, that are useful for users to run the software. Due to , the maximum degree to which an executable could be created is an executable jar file. In order to do that, one needs to first download jdk (version 17.0.5): openjdk-17.0.5

After that is done and java (as well as JAVA HOME) has been configured, launch the application with the command: java -jar JEstimator.jar
On it's own, the command will output the cmd arguments that JEstimator is expecting from the user. The minimal arguments to enter are: decimal places (for table computations for rounding in table), number of divisions (used in numerical methods), and the numerical method (used for generating the table), in that specific order.

3.5 Command Line Arguments

Here are listed all the command line arguments that come in forms of keys used by the famous library, JCommander:

-h, - -help (gives the outline for all the keys, and their respective usage)
-dp, - -decimalPlaces (number of d.p.s used in standard normal table)
-divs, - -divisions (number of divisions used in a chosen numerical method)
-rect, - -rectangular (sets the active numerical formula to rectangular)
-tr, - -trapezium (sets the active numerical formula to trapezium)
-smp, - -simpson (sets the active numerical formula to Simpson's)

Another important thing to consider here are the limitations for the Command Line Arguments, as well as order for them. They are provided below:

$decimalPlaces \in [5, 8],\ divisions \in [10^3, 10^5]$
for each of numerical methods, only one key can be provided at a time!

In order to find out more about the arguments, their description and orden during runtime, provide -h key.

4 Conclusion and Recommendations

4.1 Conclusion

In conclusion, moderate research has been conducted in order to find out standard normal distribution properties, confidence intervals and data structures. JEstimator, a statistical CLI has been designed in order to visualize the standard normal distribution data, as well as compute confidence intervals for the data.
As a simplification, only the part of the standard normal distribution for positive arguments has been modelled (it becomes easier to obtain results for negative arguments).
Software also provides persisting the current table into a pdf file, as well as calculating CDF values for a particular argument. Speaking in terms of improvements, more formulas for numeric integration could be added; furthermore, more probability distributions could be added, providing further statistical analysis for JEstimator's users.

4.2 Self-evaluation

All implemented functionality has been checked for correctness. Here are the results:
1) for output of table to the console, if decimalPlaces cmd arg is higher than 5 the table is no longer nicely printed (shifts);
2) for pdf output, sometimes the file doesn't show up in the working directory until option 5 is pressed (there mmight be some delays when writing to file);
3) cdf computations work in expected ways
4) for certain confidence intervals, not exact values from the table ($z_{alpha/2}$) get computed; reason for this are varying decimalPlaces arg values used as input; also, the parameters for calculating confidence interval (sample data) is unchecked.

References

[1] Prof. Dr. Orhan Gemikonakli
"Some Continuous Probability Distributions", p. 12-13

[2] "Algorithms. Numerical integration", YouTube