

Script-Based Computation System for Lagrange Interpolating Polynomials

Katarzyna Wiltos

Faculty of Applied Mathematics, Silesian University of Technology, Kaszubska 23, 44100 Gliwice, Poland

Abstract

The goal of this paper is to present a script-based system for calculating Lagrange Interpolating Polynomials from data points provided in text files. This simple solution utilizes Batch for the management of text files, Python for performing calculations, and HTML for presenting the obtained results in an aesthetically pleasing manner. This straightforward approach shows how one could easily and effectively implement such mathematical concepts with an interactive interface.

Keywords

Data processing, HTML, Batch scripting, Python computation, Lagrange Interpolating Polynomials

1. Introduction

Interpolation is a mathematical concept used for the construction of an approximating function based on given data points [1]. It has been aptly described as a mathematical art of reading between the lines [2]. It is predominantly utilized in various engineering fields such as physics or computer science, but also in others such as finance.

Since ancient times interpolation played an important role in calculating the planetary movement [3]. It served astronomers of ancient Greece and Babylon as an immensely helpful method for obtaining information about the planets' position (such as the sun or the moon) in unfavorable atmospheric conditions [4]. Such lists (ephemerides) with planetary position data points, would then serve as a basis for farmers to plan their farming strategies effectively throughout the coming seasons. Later, emerging from the late medieval times, in the early modern era, the epochal breakthrough started off by Copernicus and then preserved by other researchers such as Kepler, Galileo, and Newton have significantly contributed to the mathematical theory called "classical interpolation theory" [4].

Nowadays it has even more widespread usage. In medicine, it is implemented e.g. to obtain 3D images out of 2D magnetic resonance imaging (MRI) or computed tomography (CT) scans [5, 6]. In technology application such models are used for insulation modeling in electric systems [7, 8], as well as for robotic applications [9, 10, 11].

Interpolation is also applied in digital signal processing - for the process known as upsampling (transforming the sampled signal into a higher sampling frequency) [12],

image processing - for finding the unknown or missing pixels e.g. in the process of converting an image between resolutions without affecting the image quality [13, 14], in geographic information systems (GIS) [15], in finance - to estimate unknown price or potential value of an asset [16] in big data or machine learning - among others for improving models accuracy [5]. In [17] interpolation model was designed as part of hyperchaotic fusion strategy for efficient image encryption. Data analytic systems very often use also other methods, ie. [18] presented deep neural network model for imbalanced medical data or neural network applications [19] for learning representations.

One type of interpolation is Lagrange interpolating polynomial, which is widely applied in numerical analysis. This method was published and named after by Joseph-Louis Lagrange in the eighteenth century [20]. In this paper, it is used as an interpolation method for data points given in text files. There are various types of interpolation methods such as Newton's divided difference formula, and spline or linear interpolation among others. However, for the simplicity of this scrip-based system proposal, Lagrange's method was utilized.

The proposed system is managed by batch scripting. Batch is a language dedicated for Windows operating systems. It works through simple executable commands that allow for automation of repetitive tasks [21]. Recommendation systems are also often based on XAI-driven models [22].

A simple file-based approach is suggested for the purpose of efficiency and straightforwardness making data handling and management more accessible. The mathematical calculations are performed with the help of Python programming language. To present the obtained results, the HTML was used for creating a table report.

This paper is structured in the following manner: firstly the theory of Lagrange interpolating polynomials is presented, secondly, the proposed system architec-

ICYRIME 2023: 8th International Conference of Yearly Reports on Informatics, Mathematics, and Engineering. Naples, July 28-31, 2023

✉ wiltos.katarzyna@polsl.pl (K. Wiltos)

© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).



ture is described, then a case study is demonstrated with the use of examples, and last but not least a conclusion with possible future improvement or further development ideas listed.

2. Lagrange Interpolating Polynomial

Lagrange interpolation is applied to estimate or find the polynomial that best fits with the given data points.

For given set of data points x_i , where $i = 0, 1, \dots, n$ it can be obtained $n + 1$ Lagrange interpolating polynomials :

$$l_i(x) = \prod_{j=0, j \neq i}^n \frac{(x - x_j)}{(x_i - x_j)}$$

For given interpolating points, (x_i, y_i) , where $i = 0, 1, \dots, n$ formula for estimating the interpolating polynomial states as follows:

$$p(x) = \sum_{i=0}^n y_i l_i(x)$$

After simplifying the result, a polynomial will be obtained for which the following mathematical relationship will be satisfied:

$$\forall i \in N : p(x) = y_0 l_0(x) + y_1 l_1(x) + \dots + y_n l_n(x) = y_i$$

The algorithm is presented below:

Algorithm 1 Lagrange Interpolating Polynomial Algorithm

Require: xv = list of arguments x, yv = list of values y

Ensure: length(xv) == length(yv)

$x = \text{Symbol}('x')$

unsimplified = 0

for $i = 0$ to $\text{length}(xv)$ **do**

$\Phi = 1$

for $j = 0$ to $\text{length}(xv)$ **do**

if $j \neq i$ **then**

$$\Phi = \Phi \cdot \frac{(x - xv_j)}{(xv_i - xv_j)}$$

end if

end for

 unsimplified = unsimplified + $yv_i \cdot \Phi$

end for

result = simplify(unsimplified)

3. System Architecture

The program calculates Lagrange interpolating polynomials for given sets of data points extracted from a text file IN.txt and saves obtained results into OUT.txt file, finally, it displays the results in a table format on a simple website from Program-Report.html file. The system consists of four files, and during operation, it creates two more files. Additionally, if the user decides to perform the "backup" option, a BACKUP directory will be created with a copy of the appropriate files.

3.1. Program structure

The general file organization of the proposed system is as follows:

- Fundamental files:
 1. IN.txt - file that contains the input data points for calculations
 2. Batch.bat – script that controls and manages the workings of the whole system
 3. Program1.py – processes the input data and creates the OUT.txt with output data
 4. Program2.py – extracts data from both IN.txt and OUT.txt files to generate a report with obtained calculation results in the form of a simple website.
- Files generated during the program execution:
 1. OUT.txt – containing the results of calculations from Program1.py file
 2. Program-Report.html – HTML file with the website format of the report.

3.2. Input data requirements

Subsequent data sets for calculations should be included in the IN.txt file and meet the following requirements :

- one set of data points should be in one line of the text file
- numbers should be separated by the traditional comma

For example :

$$x_1, y_1, x_2, y_2, \dots, x_n, y_n$$

$$x_1, y_1, x_2, y_2, \dots, x_m, y_m$$

...

Where numbers m and n state the data point number.

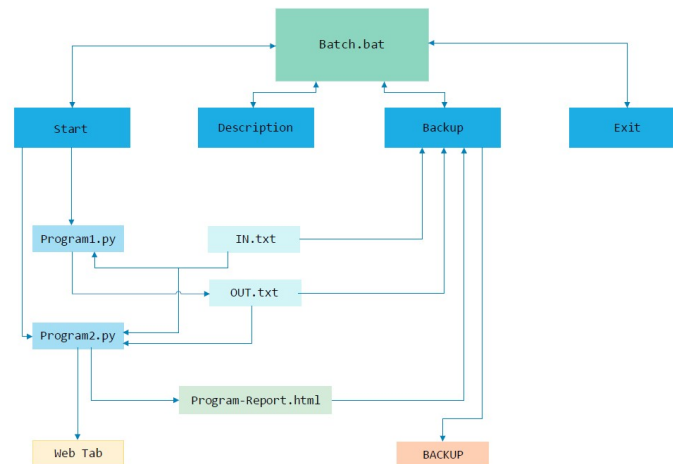


Figure 1: The program flowchart.

3.3. Program workflow

After the Batch.bat script file is executed and the Start option is chosen, the Program1.py will be executed. It processes the input data extracted from the IN.txt file and saves the obtained results in an OUT.txt file. Next, the Program2.py is executed and it extracts data from both IN.txt and OUT.txt to then process them and create a simple HTML website saved to Program-Report.html file. After successful program execution, a new window will be opened in the default browser with a report presenting the results of program operation. Additionally, after selecting the "Backup" option, a copy of the IN.txt and OUT.txt files will be created and Program-Report.html, if they exist, otherwise an appropriate message will be displayed. The copy will be saved to the directory with the current date and time.

The above chart represents how files interact with each other. The core of the whole system is based on the main batch script file called Batch.bat, which manages each subsequent or chosen action performed by the program. It is expected that the user begins with the Start option, which initiates the calculations by extracting the input data from the IN.txt file, then processes them appropriately, and finally saves the results into the OUT.txt file and presents them in a table on an HTML site. User has also an option to get acquainted with the program workings instructions that are included in the Description option. After successful calculations, the user can also save the whole program log data into a given folder by choosing the Backup option that will save there all related files. By choosing the Exit option the user can finish the process.

4. Case Study

The program can be executed by opening the Batch.bat script. Then the main window will appear as follows:



Figure 2: The menu window of the program.

There are 4 options to choose:

1. Start
2. Description

- 3. Backup
- 4. Exit

The Start option begins the calculations and the program execution. The successful operation finishes with a new web browser window with the report being opened.



Figure 3: „Start”window after the successful program execution.

- - - REPORT - - -

. LAGRANGE INTERPOLATING POLYNOMIAL CALCULATOR

Data Set	Polynomial
(-2, 6) (-1, 3) (1, 3) (2, 6)	$p(x) = x^2 + 2$
(0, -1) (1, 0) (2, -1)	$p(x) = -(x - 1)^2$
b, c, d	Invalid Data

Figure 4: Excerpt of the report webpage.

If any issues are detected the program will display an appropriate warning message.

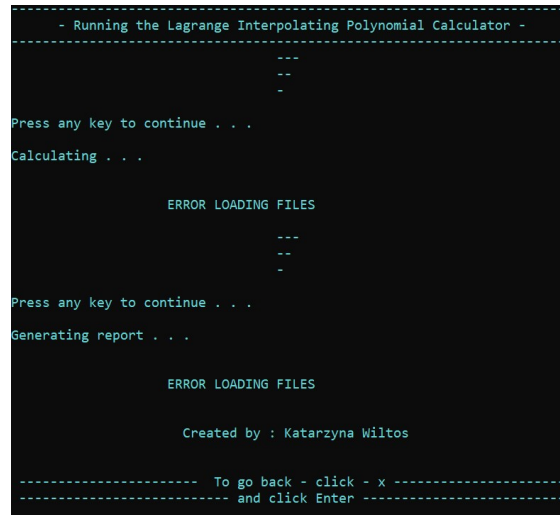


Figure 5: „Start” after failed execution - no IN.txt file detected, it does not exist. Program input data does not exist.

The program is protected against incorrect input data, as shown in the example below, where both incorrect and correct values are given as subsequent sets of points.

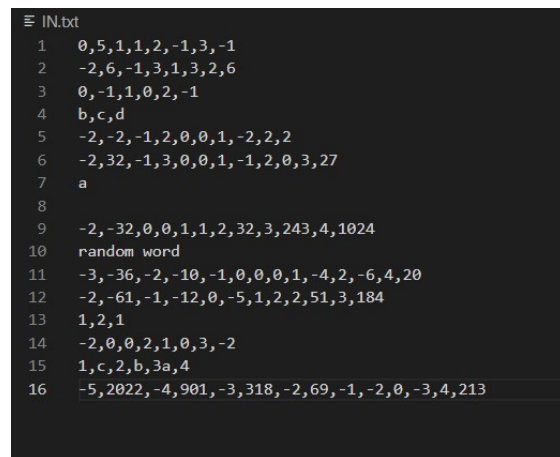


Figure 6: Test input data.

The program operation finished successfully and the results were saved to the OUT.txt file:

Then, a website with the report is displayed in a new browser window:

The program is protected against incorrect data, both in the IN.txt file and during the operation of the Batch.bat script when selecting subsequent options.

```

OUT.txt
1  p(x) = x^2 - 5x + 5
2  p(x) = x^2 + 2
3  p(x) = -(x - 1)^2
4  Invalid Data
5  p(x) = x(x^2 - 3)
6  p(x) = x^3(x - 2)
7  Invalid Data
8  Invalid Data
9  p(x) = x^5
10 Invalid Data
11 p(x) = x(x^2 - 2x - 3)
12 p(x) = 7x^3 - 5
13 Invalid Data
14 p(x) = (x - 1)(x + 2)(4x - 15)/15
15 Invalid Data
16 p(x) = 2x^4 - 6x^3 + 3x^2 + 10x - 3
17
    
```

Figure 7: Output data.

- - - REPORT - - -

. LAGRANGE INTERPOLATING POLYNOMIAL CALCULATOR

Data Set	Polynomial
(0, 5) (1, 1) (2, -1) (3, -1)	$p(x) = x^2 - 5x + 5$
(-2, 6) (-1, 3) (1, 3) (2, 6)	$p(x) = x^2 + 2$
(0, -1) (1, 0) (2, -1)	$p(x) = -(x - 1)^2$
b, c, d	Invalid Data
(-2, -2) (-1, 2) (0, 0) (1, -2) (2, 2)	$p(x) = x(x^2 - 3)$
(-2, 32) (-1, 3) (0, 0) (1, -1) (2, 0) (3, 27)	$p(x) = x^2(x - 2)$
a	Invalid Data
	Invalid Data

Figure 8: Report - part 1

5. Conclusion

The aim of this paper has been achieved. A simple script-based system for the calculation of Lagrange interpolating polynomials has been proposed. User-friendly scripts allow potential users to input their data points in a text file and then generate a Lagrange interpolating polynomial that can be used to estimate intermediate values of

(-2, -32) (0, 0) (1, 1) (2, 32) (3, 243) (4, 1024)	$p(x) = x^5$
random word	Invalid Data
(-3, -36) (-2, -10) (-1, 0) (0, 0) (1, -4) (2, -6) (4, 20)	$p(x) = x(x^2 - 2x - 3)$
(-2, -61) (-1, -12) (0, -5) (1, 2) (2, 51) (3, 184)	$p(x) = 7x^3 - 5$
1, 2, 1	Invalid Data
(-2, 0) (0, 2) (1, 0) (3, -2)	$p(x) = (x - 1)(x + 2)(4x - 15)/15$
(1, c) (2, b) (3a, 4)	Invalid Data
(-5, 2022) (-4, 901) (-3, 318) (-2, 69) (-1, -2) (0, -3) (4, 213)	$p(x) = 2x^4 - 6x^3 + 3x^2 + 10x - 3$

Figure 9: Report - part 2

the given function.

Through testing, it has been found that the suggested approach is accurate and effective as it generates proper results and is capable of handling large datasets without any lag in performance. In the future, the project can be further improved by adding more functionalities such as displaying visualizations of estimated functions along with the data points on a graph next to the already presented results in the table on the website.

This project demonstrates how Lagrange interpolating polynomials can serve as a tool for the estimation of function and analysis. However, the script-based approach and text file data storage make it more accessible for entry-level users and allow for easy modification and customization.

References

- [1] G. Mastroianni, G. V. Milovanović, Interpolation processes: Basic theory and applications, volume 194, Springer, 2008.
- [2] Interpolation: the mathematical art of reading between the lines, *Chemical & Engineering News Archive* 40 (1962) 3–4. URL: <https://doi.org/10.1021/cen-v040n021.p003>. doi:10.1021/cen-v040n021.p003.
- [3] E. W. Woolard, The calculation of planetary motions, *National Mathematics Magazine* 14 (1940) 179–189. URL: <http://www.jstor.org/stable/3028385>.
- [4] E. Meijering, A chronology of interpolation: from

- ancient astronomy to modern signal and image processing, *Proceedings of the IEEE* 90 (2002) 319–342.
- [5] A. M. Silas Bamigbola, Andrew Latham, *Interpolation: Explained, types, and real-world examples*, 2024. URL: <https://www.supermoney.com/encyclopedia/interpolation-in-statistics>.
- [6] G. Capizzi, G. L. Sciuto, C. Napoli, E. Tramontana, A multithread nested neural network architecture to model surface plasmon polaritons propagation, *Micromachines* 7 (2016). doi:10.3390/mi7070110.
- [7] F. Bonanno, G. Capizzi, C. Napoli, Some remarks on the application of rnn and prnn for the charge-discharge simulation of advanced lithium-ions battery energy storage, 2012, pp. 941 – 945. doi:10.1109/SPEEDAM.2012.6264500.
- [8] A. Sikora, A. Zielonka, M. F. Ijaz, M. Woźniak, Digital twin heuristic positioning of insulation in multimodal electric systems, *IEEE Transactions on Consumer Electronics* (2024).
- [9] V. Ponzi, S. Russo, V. Bianco, C. Napoli, A. Wajda, Psychoeducative social robots for a healthier lifestyle using artificial intelligence: a case-study, volume 3118, 2021, pp. 26 – 33.
- [10] G. De Magistris, R. Caprari, G. Castro, S. Russo, L. Iocchi, D. Nardi, C. Napoli, Vision-based holistic scene understanding for context-aware human-robot interaction 13196 LNAI (2022) 310 – 325. doi:10.1007/978-3-031-08421-8_21.
- [11] S. Pepe, S. Tedeschi, N. Brandizzi, S. Russo, L. Iocchi, C. Napoli, Human attention assessment using a machine learning approach with gan-based data augmentation technique trained using a custom dataset, *OBM Neurobiology* 6 (2022). doi:10.21926/obm.neurobio1.2204139.
- [12] C. Ciancarelli, G. De Magistris, S. Cagnetta, D. Appetito, C. Napoli, D. Nardi, A gan approach for anomaly detection in spacecraft telemetries 531 LNNS (2023) 393 – 402. doi:10.1007/978-3-031-18050-7_38.
- [13] S. Fadnavis, Image interpolation techniques in digital image processing: an overview, *International Journal of Engineering Research and Applications* 4 (2014) 70–73.
- [14] D. Połap, M. Woźniak, C. Napoli, E. Tramontana, Real-time cloud-based game management system via cuckoo search algorithm, *International Journal of Electronics and Telecommunications* 61 (2015) 333 – 338. doi:10.1515/eletel-2015-0043.
- [15] N. Brandizzi, S. Russo, G. Galati, C. Napoli, Addressing vehicle sharing through behavioral analysis: A solution to user clustering using recency-frequency-monetary and vehicle relocation based on neighborhood splits, *Information (Switzerland)* 13 (2022). doi:10.3390/info13110511.
- [16] W. Kenton, What is interpolation, and how do investors and analysts use it?, 2022. URL: <https://www.investopedia.com/terms/i/interpolation.asp>.
- [17] W. Feng, J. Zhang, Y. Chen, Z. Qin, Y. Zhang, M. Ahmad, M. Woźniak, Exploiting robust quadratic polynomial hyperchaotic map and pixel fusion strategy for efficient image encryption, *Expert Systems with Applications* 246 (2024) 123190.
- [18] M. Woźniak, M. Wiczorek, J. Siłka, Bilstm deep neural network model for imbalanced medical data of iot systems, *Future Generation Computer Systems* 141 (2023) 489–499.
- [19] Q. Ke, X. Jing, M. Woźniak, S. Xu, Y. Liang, J. Zheng, Apgvae: Adaptive disentangled representation learning with the graph-based structure information, *Information Sciences* 657 (2024) 119903.
- [20] J. Lagrange, *Leçon cinquieme: sur l’usage des courbes dans la solution des problemes, Séances des Écoles Normales recueillies par les sténographes et revues par les professeurs*, Reynier, Paris (1795).
- [21] Writing a windows batch script, 2022. URL: <https://www.geeksforgeeks.org/writing-windows-batch-script/>.
- [22] P. Naga Srinivasu, M. F. Ijaz, M. Woźniak, Xai-driven model for crop recommender system for use in precision agriculture, *Computational Intelligence* 40 (2024) e12629.