e-ISSN: 2686-2573
DOI: 10.21512/emacsjournal.v6i2.11736

Computer Resource Utilization Analysis for Microsoft Excel and Python in Data Processing

Kelvin[1*], Wahidin Wahab[2], Meirista Wulandari[3]

[1-3]Electrical Engineering Department, Faculty of Engineering,
Universitas Tarumanagara,
Jakarta, Indonesia 11440
kelvin.525210001@stu.untar.ac.id; wahidinw@ft.untar.ac.id;
meiristaw@ft.untar.ac.id

*Correspondence: kelvin.525210001@stu.untar.ac.id

Abstract – *Data analysis is essential for gaining insights and making informed decisions. A crutial step in data analysis is data processing, which involves preparing and filtering raw data to ensure accuracy, consistency, and structure. While Microsoft Excel is commonly used for data processing, it is susceptible to human errors and has limitations in handling large datasets. Python provides an alternative by automating data processing through scripts executed by the interpreter. The superior software for data processing is obtained by comparing the computer resource utilization based on statistical theory approach, Wilcoxon signed-rank test. This test is appropriate because it does not require the assumption of a normal distribution, providing flexibility in comparing computer resource utilization between Microsoft Excel and Python. Microsoft Excel and Python proceed *.csv and *.xlsx files, then Task Manager recorded the data of computer resource utilization for each processing step. The Wilcoxon signed-rank test analyzes the data and evaluating two hypotheses. H0 (there is no any significant differences in computer resource utilization between Microsoft Excel and Python are calculated for each data processing) and H1 (there is significant differences in computer resource utilization between Microsoft Excel and Python are calculated for each data processing). The sum of ranks in Wilcoxon test are compared to the critical value from the Wilcoxon distribution table to determine the accepted hypothesis. Based on the Wilcoxon test results, hypothesis H1 is accepted, indicating a significant difference in computer resource utilization between Microsoft Excel and Python.*

Keywords: *Data Processing; Python; Microsoft Excel; Wilcoxon; Computer Resource Utilization*

I. INTRODUCTION

Data has become indispensable in modern society, permeating every aspect of daily life and driving decision-making processes across industries. In today's interconnected world, organizations rely on data to understand customer behavior, enhance efficiency, and gain a competitive edge (Cuenca et al., 2021). Data provides invaluable insights that help businesses understand consumer preferences, market trends, and operational inefficiencies (Menghinello et al., 2020). This understanding allows companies to tailor their products and services to better meet customer needs and improve overall efficiency. Data also drives innovation by revealing patterns and connections that were previously unseen, leading to the development of new technologies, products, and services. Effective collection, analysis, and utilization of data are paramount for driving progress and shaping the future.

Data analysis is crucial for extracting valuable insights and making informed decisions across various domains (Fanelli et al., 2023). It enables organizations to understand patterns, trends, and correlations within their datasets, unlocking opportunities for optimization, innovation, and strategic planning. By employing statistical methods, machine learning algorithms, and visualization techniques, data analysis helps uncover hidden patterns, identify outliers, and predict future trends, thereby empowering businesses to improve efficiency, enhance customer experiences, and drive growth (Menghinello et al., 2020). Numerous tools are available for data analysis, ranging from traditional spreadsheet software like Microsoft Excel (Sousa et al., 2021) to advanced platforms such as Python and R programming languages with libraries like

Pandas, NumPy, and scikit-learn, as well as specialized tools like Tableau and Power BI for data visualization (Rahmany et al., 2020). These tools offer diverse functionalities to handle various data types, perform complex analyses, and create compelling visualizations, catering to the diverse needs of analysts and decision-makers alike.

Before data analysis, data processing is a critical step that involves preparing and refining raw data to make it suitable for analysis (Fanelli et al., 2023). This process ensures that the data is accurate, consistent, and structured in a way that facilitates meaningful analysis. This process typically includes data collection from diverse sources, data cleaning to address errors and inconsistencies, data integration to merge disparate datasets, data transformation to standardize formats and apply necessary calculations, and finally, data storage in suitable repositories for easy access. Data processing is generally conducted using Microsoft Excel (Sousa et al., 2021).

However, using Microsoft Excel for data processing often encounters several issues. Microsoft Excel tends to slow down when processing very large datasets and sometimes causes the program to crash or stop functioning, requiring it to be reloaded. Additionally, Microsoft Excel is prone to human errors, such as inputting data incorrectly and improper data processing procedures. Another alternative for data processing is using the Python programming language (Mckinney, 2022). Python enables automated data processing using scripts executed by an interpreter. One of the commonly used Python libraries for data processing is Pandas.

The use of Python as an alternative to Microsoft Excel in data processing is discussed in Crowe et al., n.d article. Crowe et al., n.d. article explores the consistency of data obtained from financial data processing based on Microsoft Excel and Python. The article indicates data consistency between Microsoft Excel and Python after data processing. In addition, Mansour et al., 2024 article also discusses the use of Python as an alternative to Microsoft Excel in financial modeling. This article uses runtime as a parameter to assess the efficiency of using Microsoft Excel and Python in financial modeling. The final results show that Python improves efficiency by more than 90%. A new approach can be taken to measure the performance of Microsoft Excel and Python in data processing by monitoring computer resource utilization. Computer resource utilization is a crucial factor determining the efficiency of both Microsoft Excel and Python (Sun et al., 2021). Therefore, this research assesses the efficiency of Microsoft Excel and Python in data processing by testing their computer resource usage.

Analyzing computer resource usage to compare Microsoft Excel and Python for data processing is essential for making an informed choice between the two tools (Sun et al., 2021). By monitoring central processing unit (CPU) and graphical processing unit (GPU) utilization during data processing tasks, organizations can gain insights into the performance characteristics and resource requirements of each tool (Sousa et al., 2021). For Microsoft Excel, resource analysis may reveal limitations in handling large datasets or complex calculations, leading to sluggish performance or system crashes. In contrast, Python's resource usage analysis might highlight its efficient memory management, multi-threading capabilities, and scalability for processing large volumes of data. Additionally, examining resource usage patterns over time can help identify trends and optimize resource allocation strategies for both Excel-based and Python-based data processing workflows. Ultimately, leveraging insights from computer resource usage analysis enables organizations to select the tool that best aligns with their performance requirements, scalability needs, and resource constraints for effective data processing.

One of the tools used for analyzing computer resource usage is Task Manager (Essien & Constant Umo, 2024). Task Manager is a system monitoring utility provided by Microsoft Windows operating systems. It offers real-time information about the processes and programs running on a computer, as well as the overall performance of the system. Users can view details information about CPU and GPU utilization for each process. Task Manager also provides options to end tasks, start new tasks, and monitor system services.

Applying the Wilcoxon signed-rank test further enhances the comparison between Microsoft Excel and Python for data processing (Dao, 2022). This statistical analysis evaluates the differences in resource usage between the two tools systematically (Walpole & Myers, 2023). By conducting the Wilcoxon test on CPU and GPU utilization data collected during processing tasks, organizations can determine if there are statistically significant disparities in resource utilization. If the test reveals significant differences, it provides quantitative evidence to support the selection of one tool over the other based on resource efficiency. Moreover, the Wilcoxon test allows for a nuanced understanding of which aspects of resource usage contribute most to performance variations between Excel and Python.

The computer resources could be monitored to determine the performance of each resource, such as Python's scalability and efficient memory management compared to potential challenges in Microsoft Excel with handling a process. This comprehensive approach empowers organizations to make data-driven decisions when choosing the most suitable tool for their specific data processing needs, ensuring optimal performance and resource utilization in their workflows.

II. METHODS

This research was conducted by collecting testing data for computer resource utilization in data processing using both Microsoft Excel (Sousa et al., 2021) and Python (Molin, 2019). The tested data processing tasks included reading CSV files, reading XLSX files, using formulas and `IF` logic functions, pivot tables, and saving data in XLSX format. Each data processing task was tested with computer resource utilization parameters, including CPU and GPU, which were observed in the Task Manager software (Martyshkin, 2020). This data was recorded in tables and analyzed using the Wilcoxon signed-rank test (Hogg et al., 2019). To perform the Wilcoxon test, hypotheses were formulated: the null hypothesis (H_0) stating that there is no significant difference, and the alternative hypothesis (H_1) stating that there is a significant difference. The results of the Wilcoxon test consist of a test value obtained by comparing the smallest sum of ranks with the Wilcoxon test table. This result is compared against the critical value obtained using the Wilcoxon distribution table, shown in Table I (Hogg et al., 2019), to determine which hypothesis is accepted. If H_1 is accepted, then the superior software for data processing can be determined by calculating the highest sum of ranks. Figure 1 illustrates the flowchart of the research methodology employed to analyze the computer resource usage of Microsoft Excel and Python in data processing.

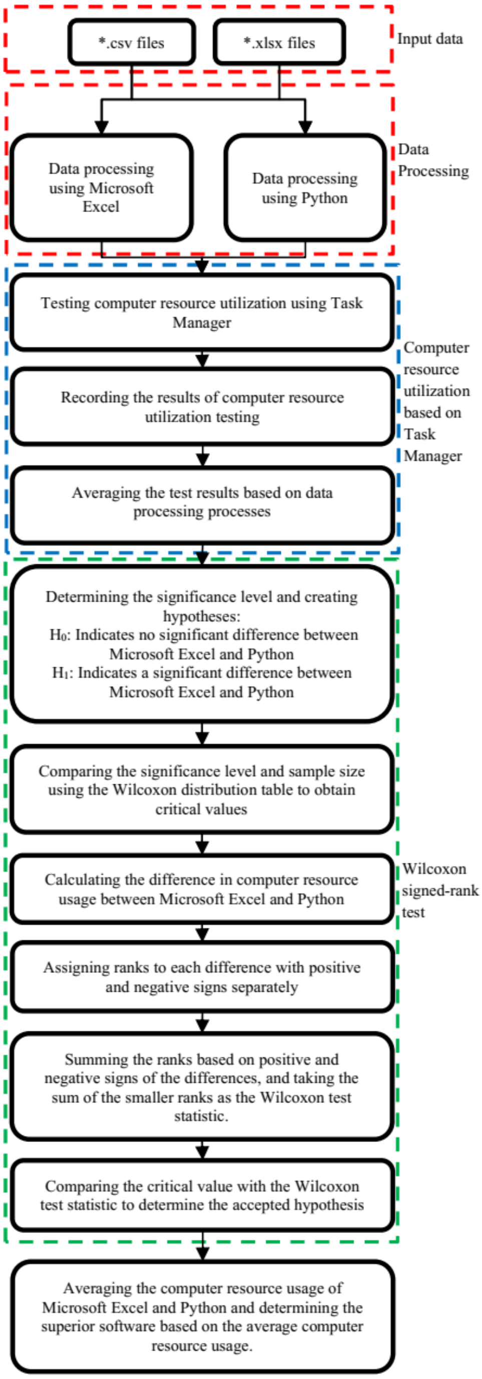
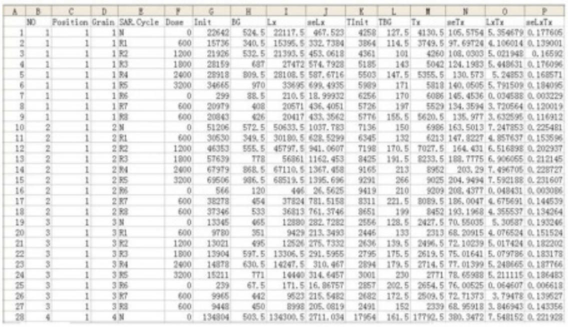

*.csv files
*.xlsx files
Input data
Data processing using Microsoft Excel
Data processing using Python
Data Processing
Testing computer resource utilization using Task Manager
Recording the results of computer resource utilization testing
Averaging the test results based on data processing processes
Computer resource utilization based on Task Manager
Determining the significance level and creating hypotheses:
H0: Indicates no significant difference between Microsoft Excel and Python
H1: Indicates a significant difference between Microsoft Excel and Python
Comparing the significance level and sample size using the Wilcoxon distribution table to obtain critical values
Calculating the difference in computer resource usage between Microsoft Excel and Python
Assigning ranks to each difference with positive and negative signs separately
Summing the ranks based on positive and negative signs of the differences, and taking the sum of the smaller ranks as the Wilcoxon test statistic.
Comparing the critical value with the Wilcoxon test statistic to determine the accepted hypothesis
Wilcoxon signed-rank test
Averaging the computer resource usage of Microsoft Excel and Python and determining the superior software based on the average computer resource usage.

Figure 1. Computer Resource Utilization of Microsoft Excel and Python in Data Processing Analyze Process

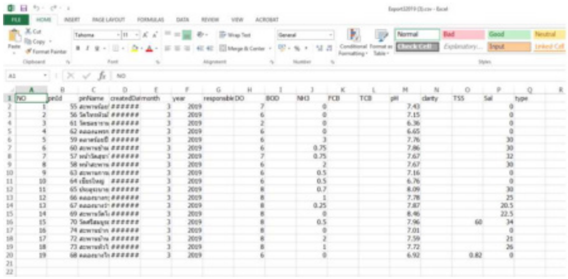
2.1 Data Processing Using Microsoft Excel

Microsoft Excel is spreadsheet software used to manage, analyze, and visualize data in table form (Jelen & Syrstad, 2022). With Microsoft Excel, users can create spreadsheets consisting of rows and columns to store data and apply various formulas and functions to perform mathematical calculations and data analysis. Microsoft Excel can read files in *.csv and *.xlsx formats. Comma-separated value (CSV) is a simple text format used for Microsoft Excel data in tabular form, with each row representing a record and each value separated by a comma or other delimiter. CSV files are widely used because of their simplicity and can be read by various applications such as Microsoft Excel and Python. Figure 1 (Peng & Li, 2017) illustrates Microsoft Excel file in *.csv format. On the other hand, *.xlsx is a specific file format from Microsoft Excel for data in a more complex spreadsheet format, including text, numbers, formulas, and cell formatting. *.xlsx format can also store worksheets with multiple sheets, charts, and other objects, and provide support for more complex layout and formatting settings. Figure 2 (Kaewrat et al., 2019) illustrates Microsoft Excel file in *.xlsx format.



A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
	NO	Position	Grain	SAR.Cycle	Dose	Init	BG	Lx	seLx	TInit	TBG	Tx	seTx	LxTx	seLxTx
1	1	1	1	N	0	22642	524.5	22117.5	467.523	4258	127.5	4130.5	105.5754	5.354679	0.177605
2	1	1	1	R1	600	15736	340.5	15395.5	332.7384	3864	114.5	3749.5	97.69724	4.106014	0.139001
3	1	1	1	R2	1200	21926	532.5	21393.5	453.0618	4361	101	4260	108.0303	5.021948	0.16592
4	1	1	1	R3	1800	28159	687	27472	574.7928	5185	143	5042	124.1983	5.448631	0.176096
5	1	1	1	R4	2400	28918	809.5	28108.5	587.6716	5503	147.5	5355.5	130.573	5.24853	0.168571
6	1	1	1	R5	3200	34665	970	33695	699.4935	5989	171	5818	140.0505	5.791509	0.184095
7	1	1	1	R6	0	299	88.5	210.5	18.99932	6256	170	6086	145.4536	0.034588	0.003229
8	1	1	1	R7	600	20979	408	20571	436.4051	5726	197	5529	134.3594	3.720564	0.120019
9	1	1	1	R8	600	20843	426	20417	433.3562	5776	155.5	5620.5	135.977	3.632595	0.116912
10	2	1	2	N	0	51206	572.5	50633.5	1037.783	7136	150	6986	163.5013	7.247853	0.225481
11	2	1	2	R1	600	30530	349.5	30180.5	628.5299	6345	132	6213	147.8227	4.857637	0.153596
12	2	1	2	R2	1200	46353	555.5	45797.5	941.0607	7198	170.5	7027.5	164.431	6.516898	0.202937
13	2	1	2	R3	1800	57639	778	56861	1162.453	8425	191.5	8233.5	188.7775	6.906055	0.212145
14	2	1	2	R4	2400	67979	868.5	67110.5	1367.458	9165	213	8952	203.29	7.496705	0.228727
15	2	1	2	R5	3200	69506	986.5	68519.5	1395.696	9291	266	9025	204.9494	7.592188	0.231607
16	2	1	2	R6	0	566	120	446	26.5625	9419	210	9209	208.4377	0.048431	0.003086
17	2	1	2	R7	600	38278	454	37824	781.5158	8311	221.5	8089.5	186.0047	4.675691	0.144539
18	2	1	2	R8	600	37346	533	36813	761.3746	8651	199	8452	193.1968	4.355537	0.134264
19	3	1	3	N	0	13345	465	12880	282.7282	2556	128.5	2427.5	70.55035	5.30587	0.193246
20	3	1	3	R1	600	9780	351	9429	213.3493	2446	133	2313	68.20915	4.076524	0.151524
21	3	1	3	R2	1200	13021	495	12526	275.7332	2636	139.5	2496.5	72.10239	5.017424	0.182202
22	3	1	3	R3	1800	13904	597.5	13306.5	291.5955	2795	175.5	2619.5	75.01641	5.079786	0.183178
23	3	1	3	R4	2400	14878	630.5	14247.5	310.467	2894	179.5	2714.5	77.01399	5.248665	0.187766
24	3	1	3	R5	3200	15211	771	14440	314.6457	3001	230	2771	78.65988	5.211115	0.186483
25	3	1	3	R6	0	239	67.5	171.5	16.86757	2857	202.5	2654.5	76.00525	0.064607	0.006618
26	3	1	3	R7	600	9965	442	9523	215.5482	2682	172.5	2509.5	72.71373	3.79478	0.139527
27	3	1	3	R8	600	9448	450	8998	205.0819	2491	152	2339	68.95918	3.846943	0.143356
28	4	1	4	N	0	134804	503.5	134300.5	2711.034	17954	161.5	17792.5	380.3472	7.548152	0.221928

Figure 2. Example of Microsoft Excel File in *.csv Format (Peng & Li, 2017)



A1 NO

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	NO	pnId	pnName	createdDat	month	year	responsible	DO	BOD	NH3	FCB	TCB	pH	clarity	TSS	Sal	type
2	1	55	[illegible]	######	3	2019		7		0			7.43			0	
3	2	56	[illegible]	######	3	2019		6		0			7.15			0	
4	3	61	[illegible]	######	3	2019		2		0			6.36			0	
5	4	62	[illegible]	######	3	2019		6		0			6.65			0	
6	5	59	[illegible]	######	3	2019		6		3			7.76			30	
7	6	60	[illegible]	######	3	2019		6		0.75			7.86			30	
8	7	57	[illegible]	######	3	2019		7		0.75			7.67			12	
9	8	58	[illegible]	######	3	2019		6		2			7.67			30	
10	9	63	[illegible]	######	3	2019		6		0.5			7.16			0	
11	10	64	[illegible]	######	3	2019		6		0.5			6.76			0	
12	11	65	[illegible]	######	3	2019		8		0.7			8.09			30	
13	12	66	[illegible]	######	3	2019		8		1			7.78			25	
14	13	67	[illegible]	######	3	2019		8		0.25			7.87			20.5	
15	14	69	[illegible]	######	3	2019		8		0			8.46			22.5	
16	15	70	[illegible]	######	3	2019		8		0.5			7.96		60	34	
17	16	74	[illegible]	######	3	2019		8		0			7.01			0	
18	17	72	[illegible]	######	3	2019		8		2			7.59			21	
19	18	73	[illegible]	######	3	2019		8		1			7.72			26	
20	19	68	[illegible]	######	3	2019		6		0			6.92		0.82	0	
21																	
22																	

Figure 3. Example of Microsoft Excel File in *.xlsx Format (Kaewrat et al., 2019)

The use of formulas and `IF` logical functions is applied to multiple columns in Microsoft Excel. The formulas and functions are written on the first row of each column. Some formulas applied include `LEFT` and `RIGHT`, used for extracting characters from a cell within a column, as well as combining two columns using the '`&`' operator. Computer resource usage is measured when the autofill feature is used to automatically copy formulas or data to related cells. Meanwhile, vertical lookup is performed separately using the `VLOOKUP` function. Microsoft Excel also features pivot tables, enabling users to quickly and easily summarize, analyze, and aggregate data in table format. With Pivot Table, users can swiftly organize and analyze data in various ways, including subtotal calculations, cross-tab analysis, and generating informative data-based reports.

Pivot Table allows users to quickly change data perspectives and explore hidden patterns and trends, making it a valuable tool for in-depth data analysis.

Microsoft Excel can be saved in *.XLSX file format, designed to store information in a complex spreadsheet format. The *.XLSX format provides various features including storing text, numbers, formulas, and complex cell layout and formatting settings. *.XLSX also supports storing worksheets with multiple sheets, charts, objects, and further layout settings. By using the *.XLSX format, users can save their data in well-structured files compatible with newer versions of Microsoft Excel, facilitating sharing and collaboration with other users.

2.2 Data Processing Using Python

Python is one of the alternatives that can be used for data processing (Molin, 2019). One of the Python libraries commonly used for data processing is Pandas. Pandas provide powerful data structures such as series and data frames. The series data structure is a one-dimensional array with labels that allow flexible indexing and is similar to columns in Microsoft Excel spreadsheets. Meanwhile, the data frame is a two-dimensional data structure in the form of a table, with each column called a series. This allows users to store and manipulate data in a format similar to Microsoft Excel spreadsheets. These data structures enable users to easily load, clean, and manipulate data.

Python scripts can be written using Jupyter Notebook running on Visual Studio Code (Molin, 2019). Jupyter Notebook is an application used in data development due to its ability to combine code, visualization, and narrative in one interactive document. Using Jupyter Notebook provides an interactive environment that allows users to run Python code step by step, view results instantly, and include narrative text, visualizations, and explanatory notes in one easily accessible and shareable document. Jupyter Notebook has a processing engine that can execute code interactively behind the scenes, known as a kernel. The kernel is responsible for executing code written in Jupyter Notebook, generating output, and storing all variables and environments in a specific session. Python can read Microsoft Excel files in CSV or XLSX formats. Python reads CSV files using the `read_csv` function, while XLSX files are read using the `read_excel` function. Python can also execute formulas and `IF` logical functions and can be applied automatically to all cells in a column. Some formulas in Microsoft Excel that are used to extract characters from a cell in a column can be executed using the `str()` function in Python. Vertical lookup can also be performed using the `merge()` function. Python also has the `pivot_table()` function used to quickly and easily summarize, analyze, and aggregate data in table format. Additionally, Python can save its data frame to a Microsoft Excel file in XLSX format using the `to_excel()` function.

2.3 Computer Resource Utilization Data Collection

The utilization of computer resources by Microsoft Excel and Python in data processing is monitored using Task Manager (Essien & Constant Umo, 2024). Task Manager provides information about hardware resource utilization and performance related to individual applications and processes on the computer system. This information includes information about the utilization of the central processing unit (CPU) and graphic processing unit (GPU) (Sun et al., 2021). The computer resource utilization data was tested and recorded in a table.

CPU responsible for processing data, instructions, and controlling system operations (Ledin, 2020). It consists of the control unit, which regulates computer operations, and the arithmetic/logic unit, which performs calculations and logical operations. The efficiency of CPU usage can significantly impact the overall performance of the computer system.

GPU is a computer component initially designed for graphics processing and image rendering, but it is now frequently utilized in data processing due to its high capability in handling parallel processing (Ledin, 2020). GPUs have hundreds to thousands of cores that can work simultaneously, making them ideal for handling computationally intensive tasks such as machine learning, deep learning, and big data analysis. The ability of GPUs to process many operations concurrently allows for significant acceleration in data processing compared to CPU, which has fewer cores and works serially. Therefore, in data analysis and processing, GPUs are a crucial resource that enhances computational efficiency and speed.

The utilization of computer resources (CPU and GPU) are test for tasks such as reading *.csv format files, reading *.xlsx files, using formulas and IF logic functions, pivoting tables, merging files, and saving *.xlsx files. The utilization of computer resources for *.csv files reading, data sorting, and column data replacement is obtained by averaging these data processing processes across 6 *.csv files. The utilization involves applying formulas and IF logical functions, which include character removal from a column cell, character extraction from a column cell, column merging, IF logic usage, and simultaneous vertical lookup. The utilization of computer resources for pivot table data processing is obtained by averaging 12 data summarization processes with different files and indices. The utilization for file merging data processing is obtained by averaging data processing processes across 4 worksheets in each *.csv file. Meanwhile, the utilization for *.xlsx file storage data processing is obtained by averaging the storage of 4 *.xlsx files.

2.4 Wilcoxon Signed-Rank Test

The Wilcoxon signed-rank test is a non-parametric comparative statistical method used to compare two related samples (Hogg et al., 2019). The Wilcoxon signed-rank test provides a more flexible approach to comparing the usage of computer resources by Microsoft Excel and Python. The Wilcoxon test does not rely on assumptions that the data is normally distributed or has the same variance between the two data groups. The usage data of computer resources by Microsoft Excel and Python may have unexpected distributions or different variabilities.

In its application, researchers formulate a null hypothesis (H0) stating that there is no significant difference between the two compared conditions or treatments and an alternative hypothesis (H1) stating that there is a significant difference between the two compared conditions or treatments (Walpole & Myers, 2023). The significance level is determined in the Wilcoxon signed-rank test as a measure indicating whether there is a significant difference. The significance level represents the percentage of rejecting the null hypothesis. The significance level is compared against the sample size using the Wilcoxon distribution table, as shown in Table I (Walpole & Myers, 2023), for critical values.

Table I. Wilcoxon Distribution Table

n	Significance value				
	0.005	0.010	0.025	0.050	0.100
5	-	-	-	-	0
6	-	-	-	0	2
7	-	-	0	2	3
8	-	0	2	3	5
9	0	1	3	5	8
10	1	3	5	8	10
11	3	5	8	10	13
12	5	7	10	13	17
13	7	9	13	17	21
14	9	12	17	21	25
15	12	15	20	25	30
16	15	19	25	29	35
17	19	23	29	34	41
18	23	27	34	40	47
19	27	32	39	46	53
20	32	37	45	52	60

The Wilcoxon signed-rank test can be calculated from this data. The result is compared with critical values from the Wilcoxon distribution corresponding to the sample size and selected significance level to determine whether the null hypothesis is rejected or not. The significance value indicates the percentage of rejecting the null hypothesis. If the test value is smaller than the critical value, then the null hypothesis is rejected, and the conclusion is made that there is a significant difference between the two groups.

In the Wilcoxon signed-rank test, the differences between the measured values in each pair are recorded with either a positive or negative sign (Walpole & Myers, 2023). A positive difference occurs when the second value is greater than the first, while a negative difference occurs when the second value is smaller than the first. These differences are ranked based on the magnitude of absolute values (smaller differences receive lower ranks, and so on). In the Wilcoxon signed-rank test, if there are ties (i.e., identical differences) among several pairs of values, the differences receive the same rank. In such cases, tied ranks are assigned by averaging the ranks that would have been assigned if the differences were not tied.

The ranks of the absolute differences are summed separately for positive and negative differences. The sum of ranks with the smaller value between positive and negative differences is used as the Wilcoxon test statistic. This statistic is then compared against critical values to determine acceptance of the null hypothesis. If the Wilcoxon test statistic is smaller than the critical value, the null hypothesis is rejected, indicating a significant difference between the two groups.

III. RESULTS AND DISCUSSION

The data on computer resource usage by Microsoft Excel and Python is analyzed using the Wilcoxon signed-rank test method. This method uses the null hypothesis and the alternative hypothesis to determine the significance of the difference. In this research, the null hypothesis (H0) states that there is a significant difference between Microsoft Excel and Python computer resource utilization during data processing. The alternative hypothesis (H1) states that there is a significant difference between Microsoft Excel and Python computer resource utilization during data processing. This hypothesis is rejected or accepted by comparing the test statistic result from the Wilcoxon signed-rank test with the critical value obtained based on the significance level and sample size. The significance level and sample size are matched using the Wilcoxon distribution table. A Significance level of 0.05 is used with a sample size of 10. Based on the Wilcoxon distribution table, the critical value used is 8. This critical value is compared against the smaller of the sums of positive and negative ranks. The results of the Wilcoxon test can be seen in Table II.

Table II. Wilcoxon Signed-Rank Test Data

Process	Ex	Py	RD	AD	ADR	PSR	NSR
			CPU (%)				
CSV file reading	19.78	8.48	11.30	11.30	7	7	
XLSX file reading	15.70	15.10	0.60	0.60	1	1	
Using formulas and IF logic functions	89.55	15.30	74.25	74.25	10	10	
Pivoting Tables	87.78	14.41	73.37	73.37	9	9	
Saving data in XLSX format	50.67	16.07	34.60	34.60	8	8	
			GPU (%)				
CSV file reading	9.30	5.65	3.65	3.65	3	3	
XLSX file reading	0.4	6.60	-6.20	6.20	5		5
Using formulas and IF logic functions	8.4	7.50	0.90	0.90	2	2	
Pivoting Tables	10.75	5.1	5.65	5.65	4	4	
Saving data in XLSX format	15.67	5.9	9.77	9.77	6	6	
Mean	**30.80**	**10.01**					
Total Ranking Sum						**50**	**5**

Explanation of Columns:

Ex	: Microsoft Excel's CPU and GPU utilization
Py	: Python's CPU and GPU utilization
RD	: Relative Difference
AD	: Absolute Difference
ADR	: Absolute Difference Rank
PSR	: Positive Sign Rank
NSR	: Negative Sign Rank

The difference in Table II is calculated by subtracting the value of Microsoft Excel from Python. Thus, positive ranks indicate Microsoft Excel, while negative ranks indicate Python. Based on the sum of positive and negative ranks, the smallest sum is 5, which comes from the sum of negative ranks. This smallest sum is smaller than the critical value determined, which is 8. Therefore, the null hypothesis (H0) stating no significant difference is rejected, and the alternative hypothesis (H1) stating a significant difference is accepted. With the acceptance of hypothesis H1, it is proven that there is a significant difference between Microsoft Excel and Python in terms of computer resource usage for data processing. Based on the average computer resource usage shown in Table 2, it is evident that Microsoft Excel has a much higher average computer resource usage compared to Python, at 30.80%. On the other hand, Python demonstrates significantly lower average computer resource usage compared to Microsoft Excel, at 10.01%. This indicates that Python is superior in terms of computer resource efficiency for data processing compared to Microsoft Excel.

IV. CONCLUSION

Microsoft Excel and Python are software that is often used for data processing. Microsoft Excel has limitations in terms of speed and smoothness in data processing. Therefore, the Python programming language can be used as an alternative. One of the factors that determine the performance of Microsoft Excel and Python in data processing is the utilization of computer resources. The utilization of computer resources is analyzed using Wilcoxon signed-rank test to determine the accepted hypothesis.

Based on the Wilcoxon test results in Table II, it can be concluded that there is a significant difference in the utilization of computer resources for data processing between Microsoft Excel and Python. This is indicated by the test value of 5 obtained from the smallest sum of ranks, which is the negative rank. This test value is significantly smaller than the critical value determined based on the sample size and significance level, which is 8. With the acceptance of hypothesis H1, it is proven that there is a significant difference between Microsoft Excel and Python in terms of computer resource usage for data processing. Based on the average computer resource utilization, it can also be concluded that Python is superior in computer resource efficiency compared to Microsoft Excel. This is evidenced by Microsoft Excel's average computer resource utilization in data processing at 30.80%, whereas Python's average computer resource usage in data processing is 10.01%.

Further research can explore the random-access memory and disk utilization needed by Microsoft Excel and Python during data processing. The goal is to gather information about the efficiency of memory and disk usage between the two platforms, complementing the analysis of CPU and GPU utilization efficiency already conducted.

REFERENCES

Crowe, R., Winney, K. M., & Avila, M. I. (n.d.). A Comparison of R, Python, and Excel Power Query with Open-Source Financial Data.

Cuenca, V., Urbina, M., Córdova, A., & Cuenca, E. (2021). Use and Impact of Big Data in Organizations. Advances in Intelligent Systems and Computing, 1327 AISC, 147–161.

Essien, N. P., & Constant Umo, M. (2024). Conceptual Analysis of Optimization Strategies in Operating System (Windows and Unix) in the Current IT Trend from 2015 to 2023. INTERNATIONAL JOURNAL OF CONTEMPORARY AFRICA RESEARCH, 2(1).

Fanelli, S., Pratici, L., Salvatore, F. P., Donelli, C. C., & Zangrandi, A. (2023). Big Data Analysis for Decision-Making Processes: Challenges and Opportunities for the Management of Health-Care Organizations. Management Research Review, 46(3), 369–389.

Jelen, B., & Syrstad, T. (2022). Microsoft Excel VBA and macros : (Office 2021 and Microsoft 365).

Kaewrat, J., Janta, R., Thammarak, K., Rattikansukha, C., & Sichum, S. (2019). People and Data: Two Factors for Sustainable Development of Water Quality Management in Pak Phanang River Basin. Convention Center.

Ledin, Jim. (2020). Modern Computer Architecture and Organization. Packt Publishing.

Mansour, M. F., Aly, Dr. T., & Gheith, Prof. M. (2024). Python Based End User Computing Framework to Empowering Excel Efficiency. International Journal for Research in Applied Science and Engineering Technology, 12(4), 2719–2729.

Mckinney, W. (2022). Python for Data Analysis: Data Wrangling with Pandas, NumPy & Jupyter (3rd ed.). O'Reilly. www.allitebooks.com

Menghinello, S., Pritchard, A., Ravindra, D., Blancas, A., Alcantara, G. A. D., Hermans, H., & Al-Kafri, S. (2020). A Strategic and Data Production Frameworks for the Development of Business Statistics. Statistical Journal of the IAOS, 36(3), 607–613.

Peng, J., & Li, B. (2017). Single-aliquot Regenerative-Dose (SAR) and Standardised Growth Curve (SGC) Equivalent Dose Determination in a Batch Model Using the R Package "numOSL." Ancient TL, 35(2).

Rahmany, M., Mohd Zin, A., & Sundararajan, E. A. (2020). Comparing Tools Provided by Python and R for Exploratory Data Analysis. IJISCS, 13–1.

Sousa, R., Miranda, R., Moreira, A., Alves, C., Lori, N., & Machado, J. (2021). Software Tools for Conducting Real-Time Information Processing and Visualization in Industry: an Up-to-Date Review. In Applied Sciences (Switzerland) (Vol. 11, Issue 11). MDPI AG.

Sun, Y., Ou, Z., Chen, J., Qi, X., Guo, Y., Cai, S., & Yan, X. (2021). Evaluating Performance, Power and Energy of Deep Neural Networks on CPUs and GPUs. Communications in Computer and Information Science, 1494 CCIS, 196–221.

Walpole, R. E., & Myers, R. H. (2023). Probability & Statistic for Engineers & Scientists (9th ed.). Pearson.