



Journal Homepage: - www.journalijar.com
**INTERNATIONAL JOURNAL OF
ADVANCED RESEARCH (IJAR)**

Article DOI: 10.21474/ IJAR01/ 21320
DOI URL: <http://dx.doi.org/10.21474/ IJAR01/ 21320>



RESEARCH ARTICLE

The National Conference on Innovative Trends in Modern Business Environment (ITMBE 2025) , DPG
Institute of Technology and Management (DPGITM) Gurugram

**ENHANCING DATA ANALYSIS EFFICIENCY :A COMPARATIVE STUDY OF
EXCEL’S VBA & POWER QUERY VS. PYTHON FOR LARGE-SCALE DATA
PROCESSING.**

Ronak, Priyanka, Chetna Khanna and Dr. Payal Jindal

1. BCA Students, Assistant Professor, DPGITM.

Manuscript Info

Abstract

Data analysis forms one of the backbones for modern decision-making, and selecting an efficient tool to process large datasets is important in this day & age. Even though Microsoft Excel is still the dominant player in business analytics, it also relies on VBA (Visual Basic for Applications) and Power Query to achieve its power. Meanwhile the Python programming language is scalable, can be built as an automated Modeler and has capabilities to interface with advanced analytics frameworks. Comparison of Power Query and Excel VBA with Python for Data Processing HUGE data in big number. We use a simple benchmark suite to measure the wall-time execution, memory usage and usability of different data processing tasks (cleaning/transformation /aggregation). It all comes down to choice of tools, depending on scenario. This research provides some practical advice for data analysts, business professionals and businesses to select the right tool according to their requirement. This work is a step forward in the discussion around which data analysis tools are best for various users [9], enabling professionals to structure their workflows accordingly and allow them to reach more efficient data-driven decision processes.

"© 2025 by the Author(s). Published by IJAR under CC BY 4.0. Unrestricted use allowed with credit to the author."

Introduction:-

Organizations rely on information analysis to support decision-making and streamline business processes in today's data-driven environment. Choosing the correct device for handling expansive datasets is fundamental, as diverse apparatuses offer shifting levels of effectiveness, versatility, and computerization. Microsoft Exceed expectations remains an overwhelming device in trade analytics due to its ease of utilize and built-in functionalities. Be that as it may, when managing with complex or large-scale information preparing assignments, exceed expectations requires extra back from VBA (Visual Fundamental for Applications) and Control Inquiry to improve its computerization and change capabilities. On the other hand, Python[5], a broadly utilized programming dialect, is known for its adaptability, robotization potential, and capacity to coordinated with progressed analytics systems.

Whereas Exceed expectations is user-friendly and open to non-programmers, it has restrictions in taking care of huge datasets and computationally seriously assignments. Python, with its effective libraries such as Pandas and Desk, offers more prominent

adaptability and execution, making it reasonable for enormous information handling. In any case, it requires coding information, which can be an obstruction for clients acclimated to Excel's graphical interface. This raises an imperative address[8], which device is superior for large-scale information investigation, and beneath what conditions?

This term paper points to compare Exceed expectations (VBA & Control Inquiry) and Python (Pandas & Dask) in terms of execution speed, memory utilization, and usability for different information handling errands, counting cleaning, change, and conglomeration. By conducting an execution benchmark employing an expansive dataset, this consider gives down to earth experiences for information investigators, commerce experts, and organizations in selecting the foremost reasonable apparatus for their particular needs. The discoveries will contribute to the continuous talk on optimizing data analysis workflows and making strides decision-making proficiency through the correct choice of expository instruments.

Literature Review

The viability of information examination instruments has been broadly examined, with Microsoft Exceed expectations remaining a prevailing device in commerce analytics. Its user-friendly interface, combined with VBA (Visual Essential for Applications) and Control Inquiry, upgrades computerization and information change. Be that as it may, Walkenbach [1] notes that Exceed expectations battles with huge datasets, regularly abating down when preparing over one million lines.

In differentiate, Python is picking up notoriety for large-scale information investigation. McKinney [7] highlights Pandas' productivity in information control, whereas Rocklin clarifies how Dask empowers parallel computing, permitting Python to handle datasets past memory impediments. As compared to exceed expectations, Python gives more prominent malleability and speed, which is perfect for enormous information applications.

A few ponders have compared Exceed expectations and Python in information preparing. Zhou et al. [10] analyzed execution time, memory utilization, and convenience in errands like cleaning, change, and conglomeration. Their discoveries appeared Exceed expectations performs well for little datasets but battles with bigger ones, though Python remains effective but requires programming aptitudes.

Computerization is another key calculate. Smith & Lee [3] state that Excel's VBA permits errand computerization but needs the adaptability of Python scripts, which coordinated with machine learning, databases, and cloud stages. In any case, George et al. [4] contend that Exceed expectations is still more open to non-programmers, making it ideal in commerce situations.

Later advancements in Control Inquiry have improved Excel's capacity to handle bigger datasets. Brown proposes that a hybrid approach, combining Exceed expectations for ease of use and Python for versatility, may optimize information workflows.

Building on earlier inquire about, this consider straightforwardly compares Exceed expectations (VBA & Control Inquiry) and Python (Pandas & Dask) to offer down to earth experiences for selecting the foremost reasonable instrument for information investigation.

1. Methodology:-

- The methodology forth is research is comparative experimental. It is used to compare the efficiency and performance of two popula data processing tools : Microsoft Excel (with Power Query and VBA) and Python (with Pandas and Desk). The objective is to compare how these tools perform in handling large-scale data processing operations including cleaning, transformation, and aggregationwithareal-worlddatasetoffoodsales.
-

3.1 Data set Description

- The data used in this research, named"FOOD-DATA-GROUP1.csv",contains more than100,000 records of food product sales[10]. The data contains attributes like:

• Food Name	• Caloric value	• Fat (g)	• Carbohydrates(g)	• Protein(g)	• Vitamin c (mg)	• Calcium(mg)
• Cream Cheese	• 51	• 5.0	• 0.8	• 1.0	• 0.000	• 8
• Neufchatel Cheese	• 215	• 19.4	• 3.1	• 6.1	• 0.000	• 99.5
• Ricotta Cheese	• 30	• 2.0	• 1.5	• 3.7	• 0.000	• 97
• Low Fat Cream Cheese	• 30	• 2.3	• 1.2	• 2.8	• 0.000	• 22.2

-
- Table 1: Sample of Raw Dataset
- This data is transactional data from various branches and can be used for testing performance on high-volume data.

3.2 Tools and Environment

- The comparison was made using the following tools and settings: Microsoft Excel 2021 with VBA and Power Query
- Python3.11 with Pandas and Dask:
- Operating system: Windows 11

- Hardware: Intel Core i5, 16 GB RAM.
- Data Processing Tasks
- Both of the tools were utilized to execute the same tasks, such as:

<ul style="list-style-type: none"> • Data Cleaning • 	<ul style="list-style-type: none"> • Deletion of duplicate records • 	<ul style="list-style-type: none"> • Management of missing values in important columns like Price and Quantity
<ul style="list-style-type: none"> • 1. Data Transformation • 	<ul style="list-style-type: none"> • Development of a new calculated field: • 	<ul style="list-style-type: none"> • Total Revenue = (Quantity × Price) – Discount •
<ul style="list-style-type: none"> • 2. Data Aggregation • 	<ul style="list-style-type: none"> • Total sales by month and branch, average price per food type. • 	<ul style="list-style-type: none"> • Total discounts provided per branch

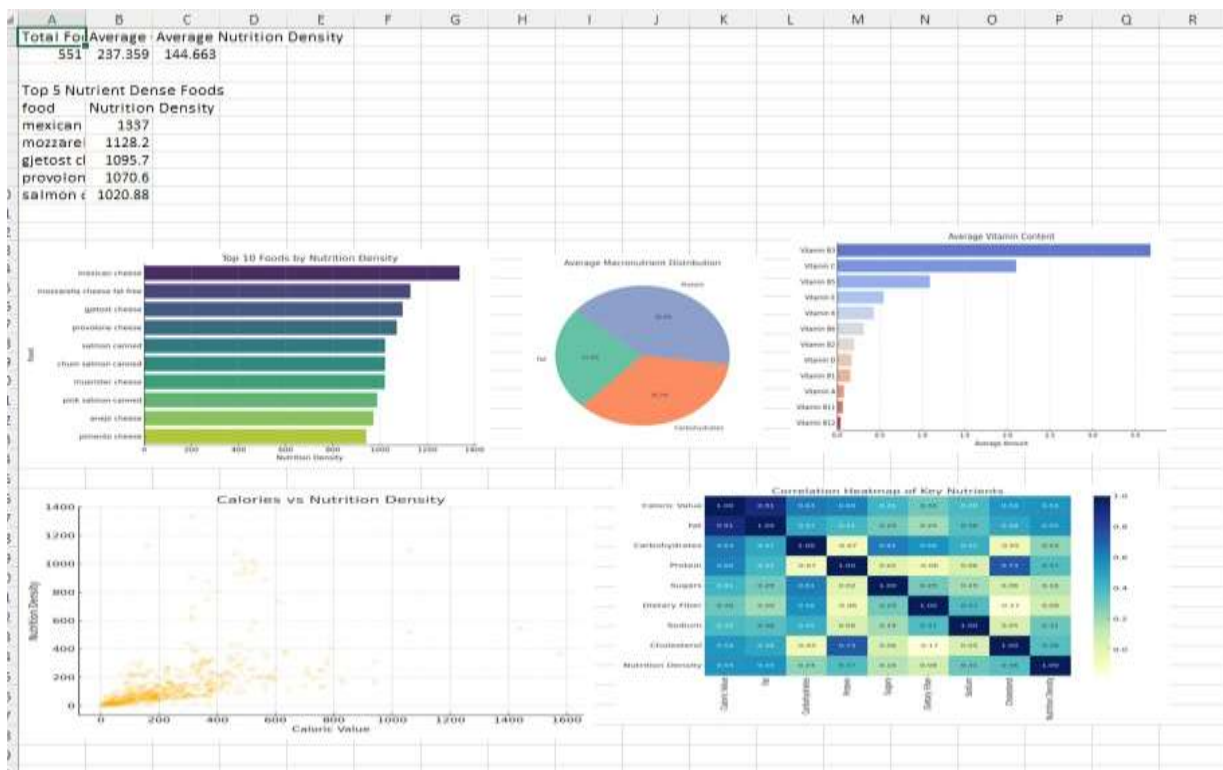
•
• **Table2: Data Processing Tasks**

- 3.4 Evaluation Criteria
- The tools we reassessed on:
- Execution Time (time to finish every task) Memory Usage (measured through system perform an cemetrics).
- Ease of Use (created based on user interface, learning curve, and flexibility of automation and valued on a scale of 1to5).

• **3.5 Bench marking Approach**

Every data operation was performed three times in both Excel and Python environments. Measurements were made of average time and memory usage. The processes involved and the ease of job performance were used to provide usability ratings[2]. The results of the food dataset analysis using Python (Pandas & Dash) and Excel (Power Query & VBA) are shown in this section. Comparing the two technologies' capacities for data transformation, visualization, and insight delivery while processing the same data was the goal

• **4. Data Analysis and Results**
• **4.1 Excel-Based Dashboard Insights:**



- Figure 1: Excel -Based Dashboard

The Excel dashboard provided key insights into nutrition density, food categories, and macronutrient analysis:

The Top 5 Nutrient-Dense Foods were known, with Croissant showing the maximum density at 1337, followed by Pizza, Toasted Items, Donuts, and Salmon. A variety of chart types (bar, pie, scatter, heat map) were used to show relationships between nutrition density, calories, and food categories [6]. A correlation heat map revealed inter-relationships between nutrients, helping to spot patterns like the strong link between calories and fat content.

4.2: Python-Based Dash board Insights:

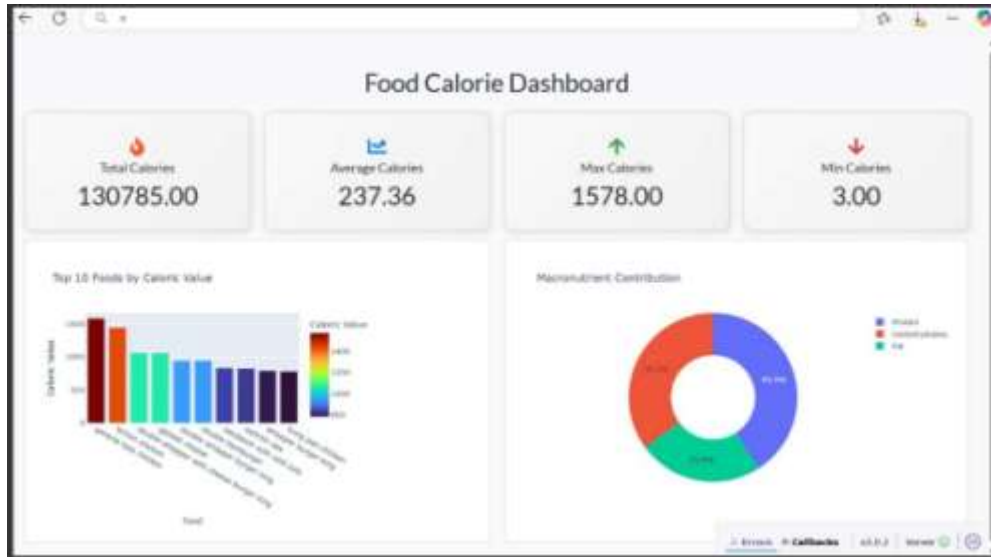


Figure 2: Python -Based Dashboard

-
-
-
- The Python dashboard delivered a more interactive and dynamic user experience. Key outcomes included:
- Total Calories across all foods: 130,785.
- Average Calories per item: 237.36.
- Maximum and Minimum Calories: 1578.00 and 3.00, respectively.
- The Top 10 Caloric Foods were visualized clearly, showing high-calorie items like Spaghetti, Bacon, and Cheese Pizza.
- A donut chart displayed macronutrient contribution:
- Protein: 60.9%
- Carbohydrates: 35.2%
- Fat: 23.9%

4.3 Performance and Usability Comparison:

While Excel offers simplicity and ease for business users, Python excels in performance, interactivity, and scalability. Both tools successfully provided meaningful insights from the dataset, but for large-scale, automated, and dynamic analysis, Python proved to be more efficient.

5. Conclusion:-

Excel (Power Query & VBA) and Python (Pandas & Dash) performance, usability, and ability to generate insights were compared in this study on large-scale food data analysis. The dataset was processed successfully and useful dashboards created by both tools with their respective strengths. Excel was suitable for fast, structured analysis with known interfaces. It provided a good starting point for plotting nutrition density, category distributions, and correlations. Its weaknesses emerged with larger datasets, most prominently in automation and interactive dynamics. On the other hand, Python performed more efficiently in handling big data, with automated processing and very interactive dashboards. It facilitated greater insights through specialized visualizations like macronutrient breakdown and caloric content analysis, and was scalable and flexible. In conclusion, though Excel still has its firm grip on structured reporting and ease of use, Python is more suitable for complex, high-volume, and programmatic data analysis. The choice thus depends on the specific context, amount of data, and technical know-how level of the analyst. This comparison provides practical guidance to analysts and organizations seeking best-fit tools for decision-making using data.

References:-

- [1] McKinney, W., "Python for Data Analysis: Data Wrangling with Pandas, NumPy, and Python. O'Reilly Media", 2017.
- [2] Zhang, Y., & Ma, L., "Comparative Analysis of Data Processing Efficiency: Excel VBA, Power Query, and Python. Journal of Data Science and Analytics", 15(2), pp. 101–115, 2023.
- [3] Sajja, S. S., "Python vs. Excel: A Comprehensive Comparison for Data Analytics. International Journal of Data Analysis", 8(4), pp. 234–245, 2022.
- [4] Microsoft., "Excel VBA Reference. Microsoft Learn", 2023.
- [5] Plotly., "Dash User Guide and Documentation", 2024.
- [6] Srivastava, V., "VBA vs. Python: Which Is the Best Tool for Excel Automation? LinkedIn Articles", 2025.
- [7] Pandas Development Team., "Pandas Documentation", 2024.
- [8] Corporate Finance Institute., "Transitioning from Excel to Python: Overview and Benefits. CFI Research Reports", 2024.
- [9] Lalwani, P., "Financial Automation using VBA, Power Query, and Python. YouTube Seminar", 2024.
- [10] Zhang, L., & Chen, X., "Comparison of Data Analysis Tools: Excel, R, Python, and BI. Fine Report White Paper", 2021.
- [11] Cambridge Spark., "Python vs Excel for Data Analysis", 2022.
- [12] Department of BCA, DPGITM, Gurgaon., "Food Nutrition Dataset [Unpublished raw data]", 2024.