International Journal of Computer (IJC)

ISSN 2307-4523 (Print & Online)

https://ijcjournal.org/index.php/InternationalJournalOfComputer/index

Integrating Python Into Power BI for Analyzing and Predicting Digital Development: Case Study – Balkan Countries

Teodora Siljanoska[a]*, Snezana Savoska[b], Ilija Jolevski[c]

[a,b,c]*Faculty of Information and Communication Technology, University St. Kliment Ohridski – Bitola, Partizanska bb., Bitola, 7500, North Macedonia*

[a]*Email: siljanoska.teodora@uklo.edu.mk*

[b]*Email: snezana.savoska@uklo.edu.mk*

[c]*Email: ilija.jolevski@uklo.edu.mk*

Abstract

The modern era with emerging technologies leads to the generation of an abundance of data, in the correct interpretation of which visualization plays a key role. The accelerated growth of the volume and complexity of data emphasizes the need for advanced ways of their transformation, visualization and analysis. By applying quantitative, empirical and qualitative methods, this scientific paper investigates the time required to perform complex data transformations in the Power BI visualization tool, in the case when they are provided manually and by applying Python code, i.e. integrating a Python script, and analyzes the efficiency and practicality in both cases through a specific example of analysis and prediction of the digital development of selected Balkan countries. The research is based on processing secondary data on the determinants of digital progress taken from official sources, empirical research and observation of IT and BI professionals for whom the time of performing assigned tasks was measured and subjective assessment of personal perceptions through an interview. All results pointed to the fact that the use of Python scripts within Power BI significantly reduces the required work time and increases efficiency while improving the accuracy, user experience and practicality of this tool, which is an important step towards adopting new advanced practices in visual data analysis.

Keywords: Integration; Power BI; Python; Data Visualization; Visual Data Analysis (VDA).

Received: 4/30/2025
Accepted: 6/12/2025
Published: 6/23/2025

----

* Corresponding author.

1. Introduction

In today's modern world driven by data and information, information visualization is a transformative tool that is key to understand data and interpret complex data and information from a variety of sources. Converting data and information from a complex form, as it is obtained from the source, into a visual format such as graphs, charts, maps, and the similar forms, helps to simplify complex information while making it easily accessible and understandable. Visualization as a technique is widespread and crucial in a large number of fields, ranging from economics and business to health, medicine, scientific research, and education, supporting decision-making processes, revealing certain inconsistencies or trends, and effectively conveying the results obtained in a form that can be easily understood and interpreted.

In healthcare and medicine, visualization is used to monitor patient outcomes, manage medical resources, and present test results in an intuitive manner. For researchers, visual data representation and information helps to see complex relationships and connections. In education, when information is presented visually, abstract concepts become clearer, facilitating the process of understanding concepts and learning.

The economy and business, with its large and complex data sets, rely heavily on visualization. When it comes to its application in this area, visualization provides real-time tracking and monitoring of key performance indicators, tracking market trends, and supporting strategic decision-making. Various tools such as interactive dashboards and charts, plots, and similar forms, provide management with a clear picture of the current situation, allowing them to quickly react to market changes and respond appropriately to all changes.

The government sector uses visualization and VDA in policymaking, ensuring transparency, demographic statistics, analyzing civic inclusion, and making “evidence-based and transparent decisions.” By transforming complex data into a visual format and generating interactive dashboards, trends can be easily identified, the effectiveness of policies and programs can be evaluated, and advanced strategies can be created that will yield better results.

In addition to the large number of available platforms and visualization tools, in this scientific paper emphasis is placed on Power BI which, with its intuitive interface, advanced functionalities and techniques, also enables the integration of Python scripts. The presented integration of Python scripts in this tool is applied in order to provide complex data transformations, statistical modeling and prediction of future trends and at the same time data visualization from the model and the obtained results. The power of Python in data processing and the availabe libraries and Power BI in data visualization is presented in the context of analysis and prediction of the future development of digitalization through a case study covering selected Balkan countries. Based on data on the E-Government Index, E-Inclusion, the Online Services Index, Human Capital and Telecommunications Infrastructure, an analysis of the digital development of the Balkans in the period from 2014 to 2024 and a prediction of future trends were made through the use of an interactive dashboard that includes various diagrams and visual displays aimed at better understanding the progress that supports digital planning in this region.

The subject of this paper is the comparative analysis of the time required for complex data transformations when they are performed manually and using scripts written in the Python programming language, as well as the "power" and capabilities of Power BI in both cases. The main goal is to "measure" the added value, effectiveness and efficiency that Python scripts provide within visualization tools, specifically Power BI, through analysis and prediction of the determinants of digitalization in the Balkans. By automating the integration of data from different sources, their processing and transformation, and econometric modeling, the focus is on proving how Python scripting can significantly expand the functionalities of conventional visualization tools, strengthening their processing and analytical power. Using the measured time for collecting and pre-processing digital indicators and applying the VAR model for analyzing and predicting future trends and visualizing past and future data, the goal is to compare the efficiency of manual and Python-based workflows in terms of time, effort, and relevance. All this, in order to prove the main hypothesis that the integration of Python scripts into Power BI significantly enhances its functionalities and at the same time enriches its power, especially when analyzing time series and other complex analyses.

2. Materials and Methods

The rapid modern technologies development has undoubtedly imposed an environment with huge amount of data in various forms and from various sources. According to Sarker, the modern digital world contains a multitude of data in various formats that originate and relate to all spheres of everyday life – starting from medical and healthcare data, economic and financial data, security data, to data related to the Internet of Things, smart computing, artificial intelligence and so on [1]. Significant technological development and the increasingly rapid and extensive generation of data have led us to find ways and research how and for what we can use it to simplify our daily lives, making the world a better place not only in the future, but also today [2].

In this context and in order to optimally use data, it is necessary to understand its meaning, to insight into data, to interpret them appropriately. Marchena Sekli and De La Vega state that the process of understanding things is called analysis, and the process of understanding data is called data analysis, which is fundamentally based on collecting them from different sources and processing using a variety of techniques and tools. Analysis provides the opportunity to extract and understand the essential correlations and trends between the data [3]. Sonavane, on the other hand, claims that data analysis means “collecting essential data” from all available “unfiltered” data and processing it in order to extract only the essential meanings in order to create better strategies [4].

The most powerful and effective tool for understanding the meaning of data is its graphical representation provided through visualization, with which they can be easily understand and interpret. With the increasing scope of data in a range of domains, the importance of data visualization also increases [5]. In a modern society surrounded by a huge number of visualizations, software tools and electronic distribution are essential in creating effective visual data representations. Although there are countless software implementations, many of them can be improved in simple ways [6]. In this regard, Python stands out significantly in “working” with data and has a rich ecosystem of libraries [7]. As a language with a wide range of applications that has become particularly popular recently, it stands out for its simplicity, readability, precision, and versatility, making it the best choice for a wide range of applications [8]. At the same time, with its high expressiveness and simplicity, it provides a range of possibilities and execution of various functionalities in a short time [9].

Wade illustrates how the R and Python languages simplify the execution of extremely complex tasks that are impossible when using conventional tools like Power BI. He points out that these languages are complementary to Power BI and provide the most powerful data transformation and processing techniques that are not available in the classic configuration of this software package, transforming it from a simple visualization tool into a tool for advanced data analysis [10].

For these reasons, the subject of research of this scientific paper is the analysis of the time required to perform complex data transformations in two cases - manually and when integrating Python scripts within Power BI in order to expand its capabilities and improve its analytical and predictive capabilities through the specific example of the digital development of selected countries in the Balkan region. The primary goal includes measuring and assessing the efficiency and effectiveness, as well as the added value that Power BI receives when using Python code within its framework, through the specific example of analyzing and predicting the digital progress of the countries in the Balkans. The specific goals relate to comparing the time required for data collection, preprocessing and transformation, the display of the extended capabilities and capacities of Power BI during such integration and visualization of the development of digitalization in the Balkan region. The primary hypothesis of this paper is that the integration of Python scripts into Power BI greatly increases efficiency and precision compared to manual work. Additional hypotheses attempt to show that the number of steps and time spent working with Power BI are significantly lower when using Python code than manually, and that the interactivity, quality, and clarity of visualizations obtained when using Python scripts within Power BI are significantly greater than when they are run independently as code.

For the purposes of this scientific paper, combined methods were applied in terms of research, i.e. a combination of quantitative, empirical and qualitative methods. The methodology is based on the implementation of econometric techniques, comparison of the efficiency of workflows and integration practices. Quantitative methods include econometric modeling and statistical analysis of secondary data using the Python programming language, the VAR model and visualization of time series. Meanwhile, the empirical methods refer to observation and research conducted on a total of 10 respondents – IT specialists and business intelligence specialists of different genders and ages between 23 and 58 years, in the period from April 23 to 30, 2025, during which the time required and the complexity of the tasks (collection, purification, i.e. pre-processing of data, their transformation, processing, as well as the application of the VAR model, forecasting and analysis of future trends in digital development) were measured when they were performed in the Power BI tool, manually and with the help of integrated Python scripts. The research took place in two phases – that is, each respondent performed the assigned tasks, first manually, and then with the help of the integrated Python code within the Power BI tool. Finally, the respondents through a direct interview with them applied qualitative methods – for the evaluation of the subjective experience and satisfaction and assessment of practicality. The interview consisted of closed-ended questions divided into 3 categories: perception of efficiency, user experience assessment, and practicality assessment.

The scientific paper includes a data framework consisting of primary and secondary data. Data on which the quantitative methods are applied are secondary data collected from official sources - the United Nations e-Government Knowledge Base [11] and the World Bank [12] which represent values relating to the key determinants of digital development, namely: E-Government Index, E-Participation Index, Human Capital Index, Online Service Index and Telecommunication Infrastructure Index for selected Balkan countries (Albania, Bosnia and Herzegovina, Bulgaria, Serbia, North Macedonia, Slovenia, Montenegro and Croatia for the period from 2014 to 2024, as well as their predicted values in the next 5 years, i.e. until 2029. In contrast, the data of the empirical (observation and research) and qualitative methods (for subjective evaluation) are primary data relating to demographic characteristics and time data (measured execution time in seconds) in the empirical, i.e. subjective assessment of qualitative methods.

Secondary data in the form of time series obtained from different sources and in different formats were combined into a single unified table. In doing so, unnecessary data (data not related to digitization) were discarded and missing values were filled in using the Forward Fill and Backward Fill methods that fill the data according to the values that follow, or the previous values, respectively [13] and the MICE (Multiple Imputation by Chained Equations) method that is based on estimating the values according to the remaining observed values [14]. On the data thus arranged, their change over time, or more precisely their stationarity, was tested using the Augmented Dickey-Fuller (ADF) test [15] and wherever necessary, differentiation was performed in order to apply the econometric VAR model. The optimal lag is defined by the BIC criterion that evaluates the efficiency of the model [16]. By applying the VAR model to the transformed data, the values of the aforementioned determinants were predicted in the next 5 years. In this way, a solid foundation was obtained for creating visualizations – Line Chart, Clustered Column Chart, Filled Map, KPI, Matrix and Slicer. After measuring the time to perform each of the tasks manually and using Python, each of the respondents was interviewed to give their assessment of the efficiency, satisfaction and potential for working with Power BI in both cases, in order to determine whether the integration of Python scripting in Power BI can really significantly upgrade its functionalities, strengthening their processing and analytical power.

3. Results

This scientific paper is based on couple of key steps:

- **Extracting the necessary data from the appropriate sources and merging them into one unified table:** from the various tables containing a large number of determinants, only the data related to digital development were selected, and specifically for specific countries in the Balkans. More specifically, the values of the E-Inclusion, E-Government, Online Services Index, Human Capital and Telecommunications Infrastructure indices for selected Balkan countries (Albania, Bulgaria, Bosnia and Herzegovina, Croatia, Montenegro, North Macedonia, Serbia and Slovenia), for the specific period, starting from 2014 to 2024, were extracted from the available tables and synchronized into one universal table;
- **Data preprocessing and cleaning:** data that were not related to the digital development of the countries and countries for which there was not enough data for the considered indices were discarded and the data for each country were sorted and standardized by year;
- **Filling the missing values:** using value propagation methods based on the previous – Forward Fill [17] and those that follow it – Backward Fill methods [17] and based on the Multivariate imputation of chained equations (MICE) method that predicts the data according to the remaining observed data [18], the empty values

of the indices were filled in;

- **Testing data stationarity and their differentiation:** each value was checked for changes over time, i.e. its stationarity was also tested with the Augmented Dickey Fuller (ADF) test [19] which indicates the possibility of autocorrelation in the error, if the data are non-stationary. For these reasons, non-stationary data were differentiated;
- **Application of the VAR econometric model and prediction of future trends:** the Vector Autoregression (VAR) model is applied to the appropriately transformed data, which is one of the most widely used models for predicting macroeconomic factors in the form of time series [20]. In this case, based on the digital indices in the period from 2014 to 2024, the trend of the digital development of the Balkan countries in the next 5 years was predicted;
- **Data visualization and results:** to easier data understanding, interpreting and analyzing as well as obtained results, several types of visualizations were generated: Line Chart, Clustered Column Chart, Filled Map, KPI, Matrix and Slicer.

The empirical research included a group of 10 respondents – men and women between the ages of 23 and 58, IT specialists and business intelligence specialists working with the Power BI tool. The research took place in 2 key phases and with the use of Power BI: In the first phase, the respondents performed tasks 1 to 5 manually, and in the second phase, they were asked to complete the same tasks using a Python script within Power BI. During both phases, the time taken by each respondent to complete the given steps was measured, in the first phase using a stopwatch, while in the second phase the measurement was within the code and the time was recorded in a separate table. Finally, the respondents were interviewed with closed-ended questions divided into 3 categories that related to their subjective perception of efficiency, evaluation of the user experience and evaluation of practicality during both phases, i.e. when they performed the given tasks manually and "automatically" with code. In the first phase of the manually running, the following results for time measuring are obtained, shown in Table 1.

Table 1: Measured time of manual work

	Data Extraction & Merging	Preprocessing & Cleaning	Missing Values Imputation	Stationarity Testing & Differentiation	VAR Model Application & Forecasting	Total Time
Participant 1	38min (2280s)	25min (1500s)	30min (1800s)	42min (2520s)	61min (3660s)	196min (11760s)
Participant 2	32min (1920s)	27min (1620s)	28min (1680s)	38min (2280s)	58min (3480s)	183min (10980s)

Participant 3	37min (2220s)	26min (1560s)	31min (1860s)	42min (2520s)	61min (3660s)	197min (11820s)
Participant 4	34min (2040s)	28min (1680s)	29min (1740s)	41min (2460s)	59min (3540s)	191min (11460s)
Participant 5	36min (2160s)	24min (1440s)	30min (1800s)	39min (2340s)	63min (3780s)	192min (11520s)
Participant 6	33min (1980s)	25min (1500s)	28min (1680s)	40min (2400s)	57min (3420s)	183min (10980s)
Participant 7	35min (2100s)	27min (1620s)	29min (1740s)	41min (2460s)	64min (3840s)	196min (11760s)
Participant 8	34min (2040s)	26min (1560s)	30min (1800s)	39min (2340s)	59min (3540s)	188min (11280s)
Participant 9	36min (2160s)	28min (1680s)	32min (1920s)	42min (2520s)	62min (3720s)	200min (12000s)
Participant 10	33min (1980s)	25min (1500s)	28min (1680s)	40min (2400s)	58min (3480s)	184min (11040s)

As can be seen from Table 1, the first step of data extraction and merging takes from 32 to 38min, or from 1920 to 2280s; Cleaning and preprocessing when done manually take from 24min (1440s) to 28min (1680s); For filling in the blank values, the respondents spent from 28min (1680s) to 32min (1920s); The measured time for testing stationarity and differentiation of non-stationary data ranges from 38min (2280s) to 42min (2520s);

While, the respondents spent the most time in calculating the VAR econometric model and forecasting, between 57min (3420s) and 64min (3840s), which was expected due to the complexity of the model. In contrast, the shortest time was required for data preprocessing and cleaning, due to the routine nature of the task. According to the table above, the total time for manual execution of the tasks ranges from 183min (10980s) to 200min (12000s).

Table 2: Average time of manual work

	Data extraction and merging	**Data pre-processing and cleaning**	**Filling the missing values**	**Testing for stationarity and differentiation**	**Applying VAR model and forecasting**	**Average Total time**
Average	35.2min (2110s)	26.1min (1566s)	29.7min (1784s)	40.3min (2416s)	60.3min (3618s)	191.6min (11494s)

According to results in Table 2, the respondents needs average time of around 11494s for manually execution of the given steps. Average, each of them devoted the most time to econometric modeling with the VAR model, or about 3618s, and the least to sorting, i.e. discarding data that is not relevant to digital development and standardizing the data for each country. They filled in the missing values in about 1784s, extracted the data and merged it into one table in 2110s, and tested for stationarity and differentiation in about 3618s.

In the second phase, the respondents performed the same steps automatically, namely they were asked to integrate and execute a given Python script in the Power BI tool that performs the same tasks (1-5). When starting the tool, selecting “Get data from another source” and Python script, they integrated the script in the following way shown in Figure 1:

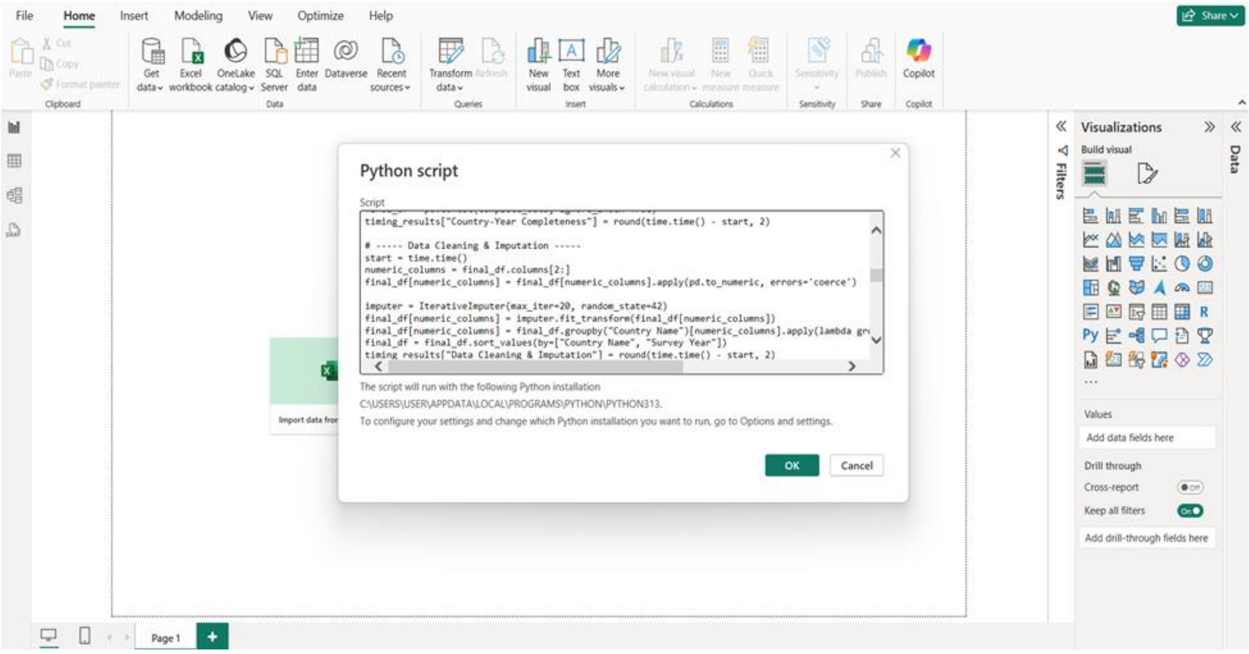

Figure 1: Integration of Python script in Power BI

Figure 1 shows the key integration of the Python script into Power BI which is of essential importance for this scientific paper. Namely, after the appropriate choice to load data from other sources, the window shown in the figure opens, where the respondents insert the given code which is the same for all and performs the same tasks at the same time measuring the time and storing it in a separate table. After completing the second phase of the research, the following results were obtained Table 3:

Table 3: Execution time after Python integration

	Data extraction and merging	**Data pre-processing and merging**	**Filling into Missing values**	**Testing for stationarity and differentiation**	**Applying VAR model and forecasting**	**Total time**
Time	0.02s (20ms)	0.07s (70ms)	0.02s (20ms)	0.03s (30ms)	0.01s (10ms)	0.15s (150ms)

After integrating the Python code within Power BI, it can be seen that significantly, better results were obtained and the steps were performed in a much shorter time (Table 3). If in the first phase the most complex task that took the respondents the longest to complete was the application of the VAR model, in this phase it was performed in 10ms (0.01s), precisely due to the power of Python and the availability of various ready-made libraries with a range of capabilities. Extracting and merging the data and filling in the missing values required 20ms (0.02s), applying the ADF test (testing stationarity and differentiation) took 30ms (0.03s), and the most time was observed in the preprocessing and cleaning of the data 70ms (0.07s), since the main transformation, conversion of data types and their standardization are performed. All in all, executing the given steps in the second phase took much shorter time – only 0.15s (150ms).

The comparison of the time required for each task, the total time, and the improvement are presented in Table 4:

Table 4: Time comparison of manual work and Python integration

	Manually execution (s)	**Integration of Python script (s)**	**Improvement (s)**	**Improvement (%)**
Data extraction and merging	2110	0.02	2109.98	99.99905%
Data pre-processing and merging	1566	0.07	1565.93	99.99553%
Filling into Missing values	1784	0.02	1783.98	99.99888%
Testing for stationarity and differentiation	2416	0.03	2415.97	99.99876%
Applying VAR model and forecasting	3618	0.01	3617.99	99.99972%
Total time	11494	0.15	11493.85	99.99870%

Table 4 provides a comparison between the measured execution time of each of the tasks in manual and "automated" work. For example, when manually extracting and merging the data, it took 2110s, and the Python script did it 0.02s or 2109.98s (99.99905) faster than when it was done manually. The same applies to the other tasks, which clearly shows that the insertion of the Python script greatly shortens the time required for each of the steps, i.e. there is an improvement in time efficiency by almost 100%, which is of particular importance when it comes to complex tasks (such as the application of VAR econometric modeling).

After completing both phases, the respondents were asked to visualize the data obtained through manual work and Python integration. They aimed to create a dashboard that would include several types of visualizations: Line Chart, Clustered Column Chart, Filled Map, KPI, Matrix, and Slicer as dashboard (Figure 2):

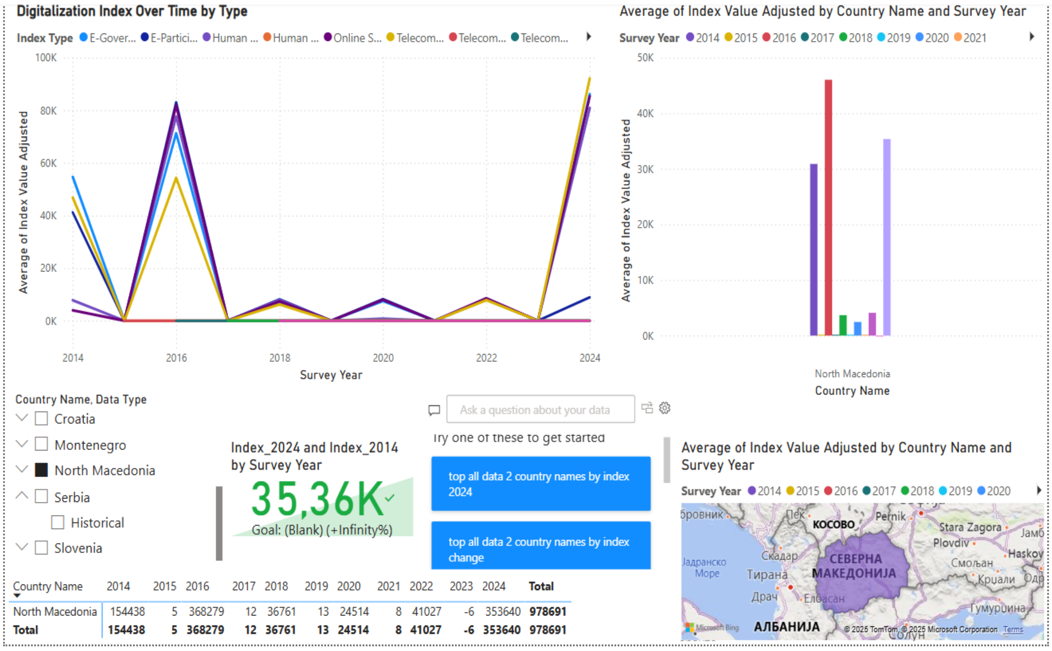

Figure 2: Data visulization of digital trends in the Balkans

Figure 2 shows the dashboard that respondents were asked to create. In the upper left corner, there is a Line Chart, and to the right of it is a Clustered Column Chart – both for a detailed and overview view of the development of digitalization (the movement of the values of digital determinants) over the years. Below the Line Chart is a Slicer that provides interactivity in a way that allows users to choose what they want to see – in this case, they can select a display of the current or projected values of the indices or a display of their values for a specific country. Right next to the Slicer, there is a KPI that provides the opportunity to select a goal and monitor key performance indicators – in this specific example, the selected goal was the optimization of the telecommunications infrastructure index. A Q&A section was added to answer the most frequently asked questions, and a Filled Map next to it aimed to "geographically" distribute the data under consideration, i.e. to display the digital indices by the countries covered. Finally, a Matrix was provided for tabulating the index values for the selected country (using Slicer) in the considered range of years (2014 – 2024).

After all key steps were completed, a direct interview was conducted with the respondents, which included closed-ended questions divided into three categories that related to their perceptions of efficiency, evaluation of the user experience, and the practicality of the two methods, i.e. the choice of one of the two "ways of working" - manually or with Python integration.

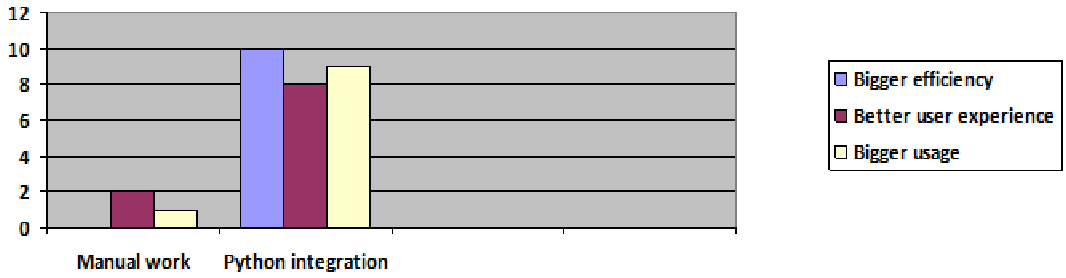

Figure 3: Interview results of personal experience

From Figure 3 it can be seen that to the question "Which way of working, according to your perception, is more efficient?", all gave the answer that the Python integration is more practical for work. Regarding the second question "Which way of working, according to you, is simpler and better?", 8 of the respondents preferred Python, while 2 were more inclined to manual work. To the third question "Which method, according to you, is more practical?" – 9 of them answered that it is the integration of Python code, and only 1 declared for the manual way of working. The 3 respondents who, in relation to the second and third questions, chose manual work, the reason for such answers was the need for additional verification and testing of the data and the obtained results in the case when Python code integration is applied.

4. Discussion

The results obtained indicate that the integration of Python into the Power BI visualization tool greatly increases efficiency, i.e. significantly reduces the time of work compared to manual execution of the same tasks. This is especially evident in more complex tasks that require additional investment of effort, time and greater precision, as in the case of the application of econometric modeling with the VAR model. The significant increase in efficiency leads to the modernization and automation of data visualization tools, where traditional GUI-based processes are enriched with scripting functionalities in order to handle complex and time-consuming tasks much more easily and efficiently. As the most complex and demanding task that required an average of 60.3min for manual calculations, with the integration of the Python script it was performed in only 10ms, which proves the high performance of Python, i.e. the execution of a wide range of functionalities in a short time, as indicated by Gorelick and Ozsvald [9]. It is precisely its "skill" in working with huge amounts of data and the large range of available libraries that are the main features of this programming language, determined by Addepalli, Gaurav and other authors [7], that are the reason for such a reduction in the time required to complete the given steps. The readability and simplicity of implementation and high precision that, according to Rayhan and Gross, make it the most suitable choice when it comes to various applications [8], with a single step of loading the script and checking and comparing the results of both ways of working, were confirmed in the research conducted within the framework of this scientific paper. In addition, the precision of Python, despite its high speed, played a key role because the probability of errors during manual work is much higher. As shown in Table 4, all tasks from data extraction and cleaning to econometric modeling, with Python integration indicate an improvement of even greater than 99.99%, which is a significant degree of efficiency improvement while maintaining the accuracy and relevance of the obtained results. All this proved the main hypothesis of the paper, i.e. confirmed that the integration of Python scripts within Power BI significantly increases efficiency and accuracy compared to manual work. These findings are in agreement with the work of Wade [10], who emphasized the valuable impact of scripting with Python on Power BI. Moreover, the study adds empirical validation through time (efficiency) tracking and user preference analysis. While prior studies largely focus on theoretical and technical benefits of integrating scripting with BI platforms, this study focuses on both - theory and applied practice by using measured performance metrics and user experience data.

When observing the respondents during both phases (manual and Python "automated" work), it was determined that using ready-made libraries reduces the number of steps that must be performed during manual work, especially when testing stationarity, applying the ADF test, differentiation and VAR modeling, which with the integration of the script are performed with only a few lines of code. This also proved the hypothesis that using Python code reduces the number of steps and time of working with Power BI.

In line with the claims of Wang, Sundin and a group of authors that the most powerful tool for interpreting and understanding the meaning of data is their graphical representation [5]. the respondents were asked to build a dashboard with multiple types of visualizations (Figure 2) in order to move on to subjective assessments of efficiency, user experience and practicality through an interview. The results of the conducted interview shown in Figure 3 indicate that in all three categories, respondents prefer the integration of the Python script when working with Power BI rather than manual work, which confirms that the quality, interactivity and clarity of the resulting visualizations with Python integration is significantly higher than when it is done independently – which was stated in the last hypothesis. Nonetheless, it is important to acknowledge the potential constraints of this study, including the limited sample size - which although diverse in age and profession, limits the generalized results to broad user population and participant preferences may have been influenced by their existing familiarity with scripting tools, introducing potential response bias in the qualitative phase.

Such results indicate the importance of accepting new advanced technologies, techniques and methods in data visualization and analysis in order to maximize their use.

5. Conclusion

Based on the conducted research and the analysis of the obtained results, it can be concluded that the integration of Python code within the Power BI visualization tool represents a step forward towards improving the efficiency, accuracy, computational and analytical capabilities and user experience when working with such tools and leads to the automation of analytical processes. The results of the quantitative methods show an enormous reduction in execution time and an improvement in the time component by 99.99%, but also a simultaneous reduction in the number of steps, maximizing precision and minimizing the possibility of errors compared to manual work. This is complemented by the subjective perceptions of the respondents, which indicate the fact that such "automation" offers an advanced user experience and are a confirmation of the intuitiveness, simplicity of use and the high degree of interactivity of the obtained visualizations. It was also determined that the use of such modern techniques and tools is necessary, especially for efficient and sound decision-making based on data. Well, the results that emerged from this paper concisely indicate the necessity and value of applying modern technological solutions, such as Python, in data visualization and analysis in tools such as Power BI.

References

[1] I.H. Sarker, "Data Science and Analytics: An Overview from Data-Driven Smart Computing, Decision-Making and Applications Perspective," *SN COMPUT. SCI.*, vol. 2, no. 377, pp. 32-34, July 2021.

[2] L. Tredinnick and C. Laybats, "Living in a data driven world," *Business Information Review*, vol. 38, no. 2, pp. 58-59, June 2021.

[3] G. F. Marchena Sekli and I. De La Vega, "Adoption of Big Data Analytics and Its Impact on Organizational Performance in Higher Education Mediated by Knowledge Management," *Journal of Open Innovation: Technology, Market, and Complexity*, vol. 7, no. 4, pp. 23-25, December 2021.

[4] A. Kumar and K. Sonavane, "Study of Emerging Role of Data," *Design Engineering*, vol. 2, no. 6, pp. 2552 - 2558, 2021.

[5] Z. Wang, L. Sundin, D. Murray-Rust, and B. Bach, "Cheat Sheets for Data Visualization Techniques," in *CHI Conference on Human Factors in Computing Systems*, Honolulu, 2021, pp. 1-13.

[6] S. R. Midway, "Principles of Effective Data Visualization," *Patterns*, vol. 1, no. 9, pp. 68-70, December 2020.

[7] L. Addepalli et al., "Assessing the Performance of Python Data Visualization Libraries: A Review," *International Journal of Computer Engineering in Research Trends*, vol. 10, no. 1, pp. 28-39, January 2023.

[8] D. Gross and A. Rayhan, "The Rise of Python: A Survey of Recent Research," *Preprint*, vol. 1, no. 1, pp. 14-17, 2024.

[9] M. Gorelick and I. Ozsvald, *High Performance Python: Practical Performant Programming for Humans*. Sebastopol: O'Reilly Media, Inc., 2025.

[10] R. Wade, *Advanced Analytics in Power BI with R and Python: Ingesting, Transforming, Visualizing*. New York: Apress Medisa LLC, 2020, vol. 1.

[11] (2025, April) UN E-Government Knowledgebase. [Online]. https://publicadministration.un.org/egovkb/en-us/Data-Center

[12] (2025, April) World Bank Group. [Online]. https://databank.worldbank.org/source/world-development-indicators

[13] K. Kotan and S. Kırışoğlu, "Cyclical hybrid imputation technique for missing values in data sets," *Scientific Reports*, vol. 15, no. 1, pp. 3-5, February 2025.

[14] Y. Zhou, S. Aryal, and M. Bouadjenek, "A Comprehensive Review of Handling Missing Data: Exploring Special," *ArXiv*, pp. 20-22, April 2024.

[15] M. Musyoki, D. Alilah, and D. Angweyi, "Updated Vector Autoregressive Model Incorporating new Information Using the Bayesian Approach," *SCIENCE MUNDI*, vol. 4, no. 2, pp. 178-197, 2024.

[16] J. Kasali and A. Alaba Adeyemi, "Model-Data Fit using Akaike Information Criterion (AIC), Bayesian

Information Criterion (BIC), and The Sample-Size-Adjusted BIC," *Square : Journal of Mathematics and Mathematics Education*, vol. 4, no. 1, pp. 43-50, April 2022.

[17] M. S. Ribeiro and C. Leite de Castro, "Missing Data in Time Series: A Review of Imputation Methods and Case Study," *Learning and Nonlinear Models - Journal of the Brazilian Society on Computational Intelligence (SBIC)*, vol. 20, no. 1, pp. 31-46, 2022.

[18] A. Hallam, D. Mukherjee, and R. Chassagne, "Multivariate imputation via chained equations for elastic well log imputation and prediction," *Applied Computing and Geosciences*, vol. 14, pp. 78-93, 2022.

[19] A. Roza, E. Silvino Violita, and S. Aktivani, "Study of Inflation using Stationary Test with Augmented Dickey Fuller & Phillips-Peron Unit Root Test (Case in Bukittinggi City Inflation for 2014-2019)," *Eksakta: Berkala Ilmiah Bidang MIPA*, vol. 23, no. 2, pp. 106-116, June 2022.

[20] M. Lenza and E. G. Primiceri, "HOW TO ESTIMATE A VAR AFTER MARCH 2020," NATIONAL BUREAU OF ECONOMIC RESEARCH, Cambridge, NBER WORKING PAPER SERIES 2020.