[Original Article]

A Quasi-Experimental Study of Parallel Python-R Learning in Data Science Education: Implementation and Assessment in a BYOD Environment

Naruki Shirahama[1]

[1]Shimonoseki City University

要旨

本研究では、データサイエンス教育において、Python と R 言語を並行して学ぶ教育手法の有効性を検証し、Windows Subsystem for Linux（WSL）と Jupyter Notebook を活用した BYOD 環境での実践的学習環境の構築を目指す。さらに、ビジネスとヘルスケアの専門科目に対する本手法の教育効果についても測定・分析する。研究方法として、1 年次の「DS プログラミング入門」において両言語を並行して学ぶカリキュラムを実施し、言語理解度テスト、課題解決能力テスト、学生の言語選好アンケートなどを用いて効果を測定する。また、WSL と Jupyter Notebook を用いた BYOD 環境の有効性を評価し、専門分野別のプログラミング教育手法の最適化を図る。本研究の成果は、データサイエンス教育の新たなベストプラクティスとして、高度なスキルを持つデータサイエンティスト人材の育成に貢献することが期待される。

キーワード： データサイエンス教育、並行学習、Python、R 言語、BYOD、WSL、Jupyter Notebook、ビジネス、ヘルスケア、プログラミング教育

Abstract

This study aims to verify the effectiveness of a parallel learning approach for Python and R programming languages in data science education and to develop a practical learning environment using the Windows Subsystem for Linux (WSL) and Jupyter Notebook in a Bring Your Own Device (BYOD) setting. Additionally, it measures and analyzes the educational impact of this method on specialized courses in business and healthcare. The research methodology includes implementing a curriculum for simultaneous learning of both languages in the first-year "DS Programming Introduction" course, with effectiveness measured through language comprehension tests, problem-solving ability tests, and student language preference surveys. Furthermore, the study evaluates the efficacy of the BYOD environment using the WSL and Jupyter Notebook and optimizes programming education methods for specific fields. The findings of this study are expected to contribute to the development of highly skilled data science professionals by establishing new best practices in data science education.

Keywords: Data Science Education, Parallel Learning, Python, R Language, BYOD, WSL, Jupyter Notebook, Business, Healthcare, Programming Education

1. Introduction.

With the advent of Society 5.0, the demand for human resources possessing the capability to effectively analyze large and complex data sets and make precise decisions is rapidly increasing. In response to this societal demand, universities across Japan are establishing new data science faculties and departments. Shimonoseki City University also established the Faculty of Data Science in 2024. Python, which demonstrates particular advantages in data analysis, machine learning, and

R language, which specializes in statistical analysis, has been adopted as the primary teaching language at these educational institutions.

However, substantial challenges remain for contemporary data science education. Numerous students tend to rely exclusively on the initial programming language they acquire and experience difficulty adapting to alternative languages and utilizing multiple languages effectively. This phenomenon has impeded the development of a versatile and adaptable workforce in the rapidly evolving field of data science.

Furthermore, in actual work environments, the ability to use multiple languages and tools according to a situation is often required. Therefore, there is an urgent need in the educational field to develop human resources that not only have a deep understanding of a single language, but are also able to flexibly use multiple languages and take advantage of their respective strengths. The purpose of this study was to propose a new approach to this issue and to examine its effectiveness.

1.1 Background of the Study

Data science is an interdisciplinary field that derives meaningful insights from large and complex datasets. The field requires a wide variety of tasks, including data collection, pre-processing, analysis, and visualization, to interpret the results. It is known that different programming languages and tools are best suited to the different characteristics of each task. For example, Python is widely adopted for data processing and machine-learning tasks because of its rich set of libraries (NumPy, Pandas, Scikit-learn, etc.). R, on the other hand, has advantages in statistical analysis and advanced data visualization and is particularly useful in academic research and biostatistics [1].

In recent years, the role of data scientists in the industry has evolved, and the required skill set has changed significantly. According to Kumar and Aithal (2020)[2], modern data scientists need flexibility and broad technical knowledge that is not dependent on a specific programming language [2]. This trend is a prerequisite for adapting to rapidly evolving technology and business environments.

Against this backdrop, there is an urgent need to develop effective methods for learning multiple programming languages in data science education. In particular, Python and R languages are highly suitable for parallel learning because their strengths are complementary and can cover the entire data science workflow. Donoho (2017) provided a comprehensive analysis of the scope and evolution of data science, arguing that data science encompasses much more than just statistics and machine learning, including data preparation, representation, computation, visualization, and the scientific study of data analysis itself [3].

Furthermore, Bialystok et al. (2009) suggested that the concurrent learning of multiple languages contributes to improved problem-solving skills [4]. This approach has the potential to foster student adaptability and creative thinking in data science education.

While existing research has extensively examined programming language education, several critical gaps remain.

- Limited empirical studies on simultaneous acquisition of multiple programming languages
- Insufficient research on BYOD implementation in data science education
- Lack of domain-specific studies in business and healthcare contexts

This study addressed these gaps through a systematic investigation of parallel language learning in a controlled educational environment.

1.2 Purpose of the Study

The main objective of this study was to empirically verify the effectiveness of a parallel learning approach for Python and R languages in data science education. Furthermore, we aim to evaluate the impact of a Bring Your Own Device (BYOD) environment integrating the Windows Subsystem for Linux (WSL) and the Jupyter Notebook on practical data science learning. To achieve this objective, the following three research questions will be set

Primary Research Question

How does the parallel learning of Python and R affect students' programming competency and problem-solving abilities in data science education?

Secondary Research Questions

1. What impact does a BYOD environment with WSL and Jupyter Notebook have on learning outcomes?
2. How do domain-specific applications (business or healthcare) influence the effectiveness of parallel language learning?
3. What factors contribute to the successful implementation of the parallel learning approach?

This investigation aims to examine the optimization of the concurrent acquisition of Python and R programming languages in the context of data science applications across various disciplines. The objective was to formulate an educational framework that addresses the specific requirements and demands of each field of study.

Through these research topics, this study aims to provide a new perspective on conventional data science education methods and contribute to the establishment of more effective and practical educational approaches. Furthermore, the results of this research are expected to serve as the basis for an educational model to foster highly adaptable human resources in the rapidly evolving field of data science.

1.3 Significance of the Study

This study aims to contribute to the development of educational methodology in this field by presenting a new paradigm for data science education and empirically verifying its effectiveness. The significance of this study can be viewed from the following perspectives from multiple perspectives.

This apprenticeship model of learning aligns with the established practices in other scientific fields. This approach contributes to improving students' adaptability and creative problem-solving skills and can be positioned as an attempt to apply the cognitive advantages identified in the context of programming education [4]. This study is expected to contribute to the development of highly adaptive personnel in the rapidly evolving field of data science.

Second, the construction and evaluation of a BYOD environment integrating WSL and the Jupyter Notebook aims to design an educational environment based on practical learning theory [5] and demonstrate its effectiveness. This effort bridges theory and practice and provides students with learning opportunities that closely resemble the actual work environment. This is expected to effectively implement Kolb's (2014) experiential learning cycle [6] and facilitate the acquisition of deep understanding and practical skills.

Third, the development of educational methods specific to the different disciplines of business and healthcare is based on situated learning theory [7], which aims to develop context-dependent applied competencies in data science. This approach bridges the gap between industry needs and academic education, and contributes to the development of a more effective human resource development model.

Furthermore, through the qualitative improvement of data science education, this research will contribute to the realization of an advanced data-utilizing society, which is the goal of Society 5.0. The development of human resources to drive digital transformation will strengthen industrial competitiveness and promote social innovation, and its social significance is extremely significant. The results of this research can be applied not only to data science education but also to education in other fields requiring complex skill sets, and are expected to contribute widely to the development of practical and effective educational models in higher education.

2. Pilot studies

2.1 Current Status and Issues in Data Science Education

Data science education has developed rapidly in response to the demands of the digital age and has come to occupy a core position in the curricula of higher education institutions. A comprehensive survey by Shimbaru (2023) documented that the Mathematics, Data Science, and AI Education Program Accreditation System (MDASH) had certified 78 educational institutions by August 2021, including 66 universities, 2 junior colleges, and 10 technical colleges [8]. However, the survey revealed marked differences in the curriculum structure and completion requirements of each institution, making the standardization and quality assurance of educational content an urgent issue.

Sasajima et al. (2023) conducted a five-year study implementing problem-based learning (PBL) exercises using real data in undergraduate data science education and reported positive outcomes in practical skill development [1]. Their longitudinal study showed that the systematic introduction of PBL exercises from lower grades had a statistically significant positive impact on students' acquisition of practical skills ($p < 0.01$). However, the study also pointed out that individual differences in students' basic programming skills may affect the learning effectiveness of PBL, suggesting the need for further research in this regard.

A survey conducted by Nakada and Gherghel (2024) found that only 33% of humanities students in the Faculty of Foreign Studies showed interest in data science, whereas other faculties showed higher levels of interest (e.g., 52% in the Faculty of Business) [9].

Specifically, 54.3% of liberal arts students indicated that they were "not very interested" or "not interested at all" in data science, a statistically significant difference from science students (23.7%) ($\chi^2(1) = 45.62$, $p < 0.001$). This result suggests that current data science education may be biased toward science fields, and highlights the need for an integrated arts and sciences approach.

While these previous studies demonstrate the importance and prevalence of data science education, they raise three main issues: standardization of the curriculum, effective ways to develop practical skills, and the need for a comprehensive approach that is not limited to the humanities and sciences. This study aimed to develop and validate a new educational model to address these issues.

2.2 Methodology of Programming Language Education

With regard to programming language teaching methodologies, there are two main paradigms: the traditional approach, which emphasizes a deep understanding of a single language, and the emerging approach, which encourages the parallel learning of multiple languages. Each approach is based on a different theoretical foundation and pedagogical goal.

Yusupova et al. (2022) empirically examined the impact of adaptive e-learning environments using machine learning algorithms on programming education [10]. Their study showed that e-learning environments that provide personalized learning pathways significantly improve student learning outcomes compared to traditional uniform teaching methods ($t(198) = 3.74$, $p < 0.001$, Cohen's $d = 0.53$). In particular, they concluded that flexibility in meeting students' individual needs is a key factor in this effect.

Kumar & Aithal (2020), on the other hand, through a qualitative analysis of the demand for human resources in the data science field, revealed the diversity of skill sets required of data scientists [2]. They emphasized that industry demands require data scientists to be proficient in multiple programming languages and tools, highlighting the importance of versatile technical skills in the field. This finding suggests the effectiveness of a concurrent learning approach for multiple languages, and has important implications for the design of educational curricula.

In addition, Bialystok et al. (2009) demonstrated that bilingualism is associated with enhanced executive control functions, particularly in tasks requiring inhibition of distracting information and switching between tasks, although bilingual speakers also show some disadvantages in lexical access and vocabulary size compared with monolinguals [4].

While research on programming education is expanding, Robins (2019) notes that teaching and learning programming continue to face well-documented challenges [11]. In particular, longitudinal studies on the effects of parallel learning and comparative studies that consider different learner characteristics are lacking.

While these previous studies suggest the potential effectiveness of the multiple-language parallel learning approach in programming language education, they also highlight the need to rigorously examine its effectiveness. This study aims to address this research gap and empirically examine the effectiveness of multiple language parallel learning in data science education.

2.3 Educational Use of WSL and Jupyter Notebook

Research on the educational use of the Windows Subsystem for Linux (WSL) and Jupyter Notebook is an emerging area of computer science education. The WSL provides the ability to run Linux directly in a Windows environment, greatly simplifying the construction of a cross-platform development environment. Jupyter Notebook, on the other hand, is an open-source web application that allows users to interactively execute code and instantly visualize results.

Perez and Granger (2015), in describing Project Jupyter, emphasized the importance of integrating code and explanatory text into a single interactive document for scientific computing and data analysis [12]. According to their qualitative analysis, a key advantage of the Jupyter Notebook is its ability to integrate code and explanatory text into a single interactive document. This characteristic is highly valued in both reproducible research and teaching, suggesting that it is particularly effective in developing computational thinking skills. Rule et al. (2019) provide guidelines for using Jupyter Notebooks to enhance reproducibility and transparency in computational research through proper documentation and sharing practices [13].

Ochkov et al. (2022) demonstrated how Jupyter Notebook and JupyterLab can be effectively integrated into STEM education, supporting various learning activities including problem formulation, scientific calculations, computational experiments, and interactive teaching materials [14]. On the other hand, empirical research on the combined educational use of WSL and the Jupyter Notebook remains limited; Ochkov et al. (2022) provide a comprehensive analysis of the impact of these environments on STEM education. Their study showed that the Jupyter Notebook and JupyterLab support a wide range of learning activities, from problem-setting formulation to scientific computation, computational experimentation, and even the creation of interactive teaching materials. In particular, the ability to integrate multiple programming languages (e.g., Python and R) and markup languages (e.g., Markdown and LaTeX) in a single environment has been pointed out as having the potential to deepen learners' understanding and promote practical skill acquisition. This finding provides theoretical support for the effectiveness of the parallel learning approach for Python and R languages proposed in this study and suggests that an integrated learning environment may contribute to the acquisition of multiple languages and practical data science skills. A survey of freshmen university students conducted by Kambe et al. (2024) (n = 102) reported significant individual differences in students' ICT utilization skills in a 14-item self-assessment of basic computer skills [15].

While these previous studies show that WSL and the Jupyter Notebook have great potential for data science education, they also highlight the lack of empirical studies on their effective implementation and long-term learning effects. This study aims to address this research gap and systematically examine the impact of a learning environment that integrates WSL and Jupyter Notebook on concurrent learning of multiple programming languages.

3. Research methodology

In this study, we adopted a quasi-experimental approach (quasi-experimental approach) using naturally formed learning patterns to examine the effectiveness of parallel learning of Python and R languages in data science education. This approach is suitable for implementation in an educational setting because it exploits naturally occurring differences in conditions without experimental manipulation, and is also appropriate from the perspective of ethical considerations [16].

3.1 Research Subjects

The research subjects will be first-year students enrolled in the 2025 academic year at the Faculty of Data Science at Shimonoseki City University. The same curriculum will be offered to all students, and learning patterns that naturally form during the one-year learning process will be observed.

3.2 Learning Environment

The following integrated learning environments are provided for all students:

1. Ubuntu 22.04 LTS environment using WSL2 (Windows Subsystem for Linux 2) on Windows 11

   This allows students to use a Linux-based development environment integrated directly into Windows, providing a modern development platform that bridges the Windows and Linux environments (Leeks, 2020) [17].
2. Jupyter Notebook and RStudio Server
   - Jupyter Notebook: A web-based interactive computing environment that enables the creation and sharing of documents containing live code, equations, visualizations, and narrative text, supporting multiple programming languages including Python and R (Kluyver et al., 2016) [18]. These 'computational notebooks' allow students to combine code

execution with rich documentation and immediate visualization of results.

- RStudio Server: An integrated development environment that provides a comprehensive platform for R-based data analysis and supports reproducible research workflows through integrated tools for code development, data analysis, visualization, and document preparation (Gandrud, 2020) [19]. This environment allows the seamless integration of coding, analysis, and presentation of results in a single interface.

3. Visual Studio Code（VS Code)

A lightweight cross-platform code editor that supports multiple programming languages, including Python and R, provides seamless integration with the WSL. Its extensible architecture allows it to function effectively across different development scenarios, including web, desktop, and cloud development (Del Sole, 2022) [20].

Students have autonomy in selecting and utilizing these environments, based on their individual preferences. This flexible environment setting will allow students to choose languages according to their personal learning styles and preferences, and to observe natural learning patterns. These environments provide students exposure to professional computational tools and practices (Nolan & Lang, 2015), while facilitating practical data science skills [21]."

3.3 Curriculum

The curriculum for this study will be designed to allow students to learn the basics of Python and R languages in parallel, and will be built on the constructivist alignment theory proposed by Biggs et al. (2023) [22]. This theory emphasizes the coherence among learning objectives, instructional activities, and assessment methods to promote deep learning. The curriculum consists of six main topics, with appropriate study periods for each topic.

1. Fundamental data types and variables
   - Python: int, float, str, list, dict
   - R: numeric, character, vector, list, data.frame
   - Learning Objectives: Understand the basic data structures of both languages and to use them appropriately.
2. Control structure (conditional branching, loops)
   - Python: if-else, for, while
   - R: if-else, for, while, apply functions
   - Learning Goal: To Implementing basic algorithms using conditional branches and loops.
3. Functions and Modules
   - Python: def, lambda, import
   - R: function(), source(), library()
   - Learning Objectives: Implement reusable code as functions and make appropriate use of external libraries.
4. Manipulation of data frames
   - Python: pandas (DataFrame, Series)
   - R: dplyr, tidyr
   - Learning Objectives: To acquire efficient manipulation and pre-processing techniques for structured data.
5. Data visualization:
   - Python: matplotlib, seaborn
   - R: ggplot2
   - Learning Goal: Select and implement appropriate visualization techniques according to data characteristics.
6. Basics of statistical analysis
   - Python: scipy.stats, statsmodels
   - R: basic statistics, lm(), glm()
   - Learning Objectives: Understand basic statistical methods and perform analyses using appropriate libraries.

Each topic will be structured around interactive exercises using Jupyter Notebook and RStudio

Server, adopting a Problem-Based Learning (PBL) approach (Hmelo-Silver, 2004) [23]. This allows students to learn by comparing the characteristics of both languages through practical problem-solving. The PBL approach was chosen because it promotes flexible knowledge acquisition, effective problem-solving skills, self-directed learning, collaboration, and intrinsic motivation, all of which are essential in data science education.

The effectiveness of educational approaches that utilize real-world datasets has been demonstrated in multiple studies. Vandergon et al. (2018) examined the impact of computer-supported collaborative science (CSCS), a teaching method that engages students in the collection, analysis, and interpretation of real-world data [24]. Their research showed that working with real-world data significantly increases students' motivation and engagement in learning. Based on these findings, the use of real-world datasets is recommended for assignments conducted at the end of the semester to promote the acquisition of practical data science skills.

The evaluation consisted of an assignment at the end of each topic (50%), and a final comprehensive exercise (50%). These projects allow students to freely choose their language and observe their natural language preferences and learning patterns. This curriculum design allows students to choose a language according to their own interests and aptitudes while learning the basics of both languages systematically and in depth.

3.4 Data Collection

This study employs a Mixed Methods Research method to collect quantitative and qualitative data (Guest & Fleming, 2014), as this approach allows for more comprehensive evidence and can address research questions that a single-method approach cannot adequately answer [25].

1. Submission of assignment file
   - Method: Assignment management system using Google Classroom
   - Content: Languages, code quality, and solution approaches used for each topic assignment
   - Frequency: At the end of each topic
   - Objective: Assessment of language selection tendencies and problem-solving skills
2. Self-assessment questionnaire
   - Methods: A questionnaire using the Visual Analog Scale (VAS) [26] was used, which has been validated as an effective method for capturing subtle gradations in respondent assessments (Shirahama et al., 2016). This approach allows for more precise measurements than traditional discrete scales while maintaining reliability in digital implementation.
   - Content: Self-assessment of language proficiency, motivation to learn, self-efficacy, etc. (0-100 scale)
   - Frequency: At the end of each topic and at the end of the semester
   - Objective: To analyze learners' subjective evaluation and psychological factors
3. Semi-structured interviews
   - Method: 30-minute interview in person or online
   - Content: Reasons for the selection of learning approaches, difficulties, comparative perspectives on languages, etc.
   - Frequency: Ten randomly selected from each group after the end of the semester.
   - Objective: Deep understanding of the learner's experiences and perceptions

The collected data will also be classified a posteriori into the following groups.

1. Dual language learning group: Students who used Python and R to the same degree and demonstrated a certain level of proficiency in both languages.
2. Single language acquisition group: students who primarily use one of the languages and show high proficiency in that language

3.5 Data Analysis

This study employs a mixed analysis approach that combines quantitative and qualitative analyses.

1. Descriptive statistics

- Calculate basic statistics (mean, standard deviation, median, etc.) to characterize each group
- Visualize distribution of language use ratios, test scores, and VAS scores (histograms, box plots, etc.)

2. Statistical hypothesis testing
    - Independent samples t-test: Test for differences in language proficiency, problem-solving ability, etc. between the two groups
    - Repeated two-way analysis of variance (ANOVA): analyzes changes in learning outcomes for both groups over time
    - Mann-Whitney U test: non-parametric test for comparing groups when normality cannot be assumed

3. Multivariate analysis
    - Multiple regression analysis: analyzes the relationship between language acquisition patterns (independent variable) and learning outcomes (dependent variable)
    - Principal Component Analysis (PCA): Extracts major learning behavior patterns from learning log data

4. Qualitative analysis
    - Thematic analysis [27]: a rigorous qualitative method involving familiarization with data, systematic coding, theme development, and refinement to extract meaningful patterns from interview data
    - Content analysis: Systematic categorization of reasons for selection of learning approaches and impressions from free-text responses

5. Mixed Analysis Methods
    - Integration of quantitative and qualitative results: following Fetters et al.'s (2013) [28] framework, results are integrated through [specific approach: narrative/data transformation/joint display] to achieve comprehensive interpretation

All statistical analyses were performed using R, with a significance level set at $\alpha = 0.05$. Effect sizes were reported using Cohen's d or Hedges' g.

3.6 Ethical considerations

This research will be conducted in accordance with the ethical principles of the Declaration of Helsinki [29] and the research ethics guidelines of Shimonoseki City University. Specifically, the following measures will be taken

1. Informed consent: The purpose of the research, methods, anticipated risks and benefits, and the use and protection of data were explained, and written consent was obtained.
2. Data anonymization: Personal identifiers are removed, and randomly generated ID numbers are assigned. The correspondence sheets were strictly controlled and destroyed after the study was completed.
3. Data Security: Collected data are encrypted and stored on a password-protected server. Access privileges were restricted to key members of the research team.
4. Freedom of participation and right to withdrawal: Ensure that participation in the research is completely voluntary and can be withdrawn at any time and without cause. Clearly, no disadvantages resulted from refusal or withdrawal.
5. Educational Equity: Ensure that all students have the same quality of educational opportunities regardless of their participation in the study.
6. Ethics Committee Approval: Approval was obtained from the Ethics Committee of Shimonoseki City University before the study began.

These ethical considerations will protect the rights and welfare of participants while simultaneously ensuring the conduct of scientifically valuable research.

4. Expected effects, etc.

The parallel learning approach for Python and R languages proposed in this study is expected to have multifaceted effects on data science education. Below, we discuss the anticipated benefits in terms of key aspects.

4.1 Effects of language acquisition

The parallel learning group (PLG) is expected to show more balanced proficiency in both languages compared to the single language focus group (SLFG). Specifically, the following effects are expected

1. Concept transfer between languages: We explore potential parallels with Cummins' (1979) [30] developmental interdependence hypothesis from bilingual education, suggesting that just as second language competence can build upon first language development, PLGs may develop transfer abilities based on strong foundational understanding in one programming language before intensive exposure to the second, although the cognitive mechanisms may differ significantly.
2. Improved metalinguistic cognition: Drawing parallels with Bialystok's (2001) [31] research on bilingual cognitive development, we hypothesized that PLGs may develop enhanced metalinguistic awareness in the context of programming languages, although the relationship between natural and programming language learning requires further investigation. (While Bialystok's work [31] focuses on natural language bilingualism, the cognitive advantages she identifies - particularly in selective attention and abstract structure understanding - may have relevant applications to programming language learning, though this theoretical extension requires empirical validation.)
3. Flexible language selection capability: Drawing inspiration from research on natural bilingualism (Grosjean, 2010) [32], we hypothesized that PLGs may develop similar capabilities in choosing appropriate programming languages for specific tasks, although this parallel requires empirical validation.

4.2 Improve problem-solving skills

The concurrent learning approach is expected to contribute to improved problem-solving skills in the following ways

1. Promoting multidimensional thinking: Drawing from Guilford's (1967) [33] broader theory of human intelligence and cognitive abilities, we hypothesized that exposure to multiple programming languages may enhance students' capacity for diverse problem-solving approaches.
2. Enhanced algorithmic thinking: Drawing from Papert's (1980) constructionist theory [34], which emphasizes learning through the meaningful construction of mental models, we propose that experience with multiple programming languages may help students build their own understanding of abstract algorithmic concepts through concrete implementations in different contexts.
3. Improved debugging skills: Drawing from Vessey's (1985) [35] research on debugging expertise, which demonstrates how experts develop superior program chunking abilities and systematic debugging strategies, we hypothesized that experience with multiple programming languages may enhance students' ability to develop more sophisticated debugging approaches. However, this potential relationship between multilanguage experience and debugging expertise requires empirical validation.

4.3 Effectiveness of the learning environment

The proposed integrated learning environment (WSL, Jupyter Notebook, RStudio Server, VS Code) is expected to be effective in

1. Acquisition of practical skills: Drawing from Lave and Wenger's (1991) [36] theory of situated learning and legitimate peripheral participation, we propose that using industry-standard tools within a collaborative learning community allows students to gradually develop practical skills through authentic participation in data science practices, moving from peripheral to more central

participation in the field.

2. Seamless language switching: Integrated environments allow students to easily move between languages and facilitate comparative learning. While acknowledging the fundamental differences between natural and programming languages, we consider how Krashen's (1982) concept of 'comprehensible input' might inform our understanding of how learners process multiple programming syntax systems.
3. Facilitating self-regulated learning: Flexible environmental settings allow students to customize the environment to their own learning style, which is expected to lead to effective learning based on Zimmerman's (2002) [38] theory of self-regulated learning.

4.4 Learning Effectiveness by Specialty

This study focuses on two specialties, business data science and healthcare data science, and is expected to have the following effects

1. Improved domain-specific skills: Using datasets and tasks specific to each discipline, students are expected to simultaneously acquire practical domain knowledge and data science skills. This aligns with research showing that learning is fundamentally influenced by the context in which it takes place (Bransford et al., 2000) [39].
2. Acquiring a cross-disciplinary perspective: By learning both languages, students can compare approaches and methods from different disciplines and acquire a cross-disciplinary perspective.
3. Understanding discipline-specific language preferences: It is expected that more suitable languages and methods will be identified for each discipline, which may provide useful insights into language selection in future research and practice.

These anticipated effects form the hypotheses of this study and provide a basis for future empirical testing. As the study progresses, a comparison of these predicted and actual results will help to clarify the effectiveness and challenges of the concurrent learning approach in data science education.

While these expected effects provide a theoretical framework for our research, it is essential to validate them through an empirical investigation. To this end, we conducted a preliminary experiment in 2024 to test our hypotheses and refine our methodology before its full implementation in 2025. This preliminary study focused on examining students' responses to parallel language learning and testing the effectiveness of our BYOD environment.

5. Preliminary Experiment

In this study, a preliminary experiment was conducted in the required course "Introduction to Data Science Programming" (15 weeks) in 2024 to verify the educational effectiveness and feasibility of Python-R parallel learning, which will be implemented in 2025. The purpose of this preliminary experiment was to confirm the effectiveness of the proposed parallel learning approach and identify practical issues.

5.1 Survey Subjects and Timeframe

The survey was conducted with 88 students enrolled in the "Introduction to Data Science Programming" (a required course) offered in the second semester (15 weeks) of the 2024 academic year. The survey was conducted after the 11th lecture and evaluations based on what they had learned up to that point were obtained. There were 62 valid responses were received (response rate: 70.5%).

5.2 Curriculum Structure

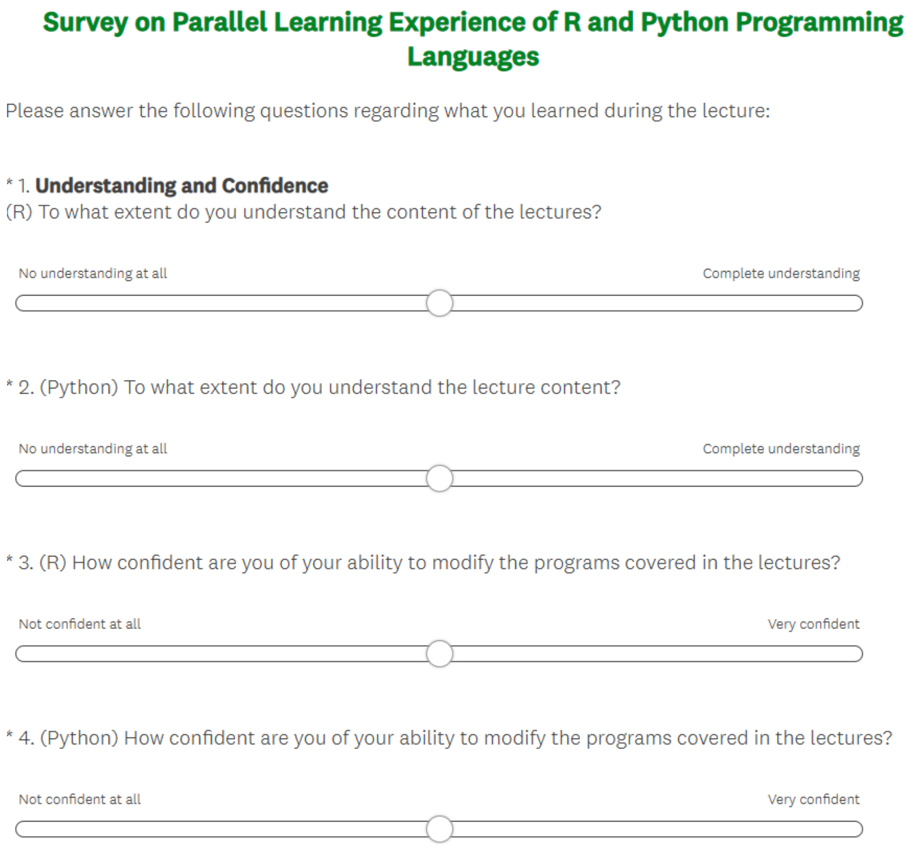
Students were provided with WSL2 (Windows Subsystem for Linux 2), Ubuntu 22.04, and JupyterLab environments on Windows 11 and were given hands-on training in a Bring Your Own Device (BYOD) format using their own laptop computers. The curriculum was designed to cover basic data science elements, such as functions, collections, data frames, and statistical methods, and to allow students to learn R and Python in parallel.

5.3 Survey Methodology

5.3.1 Quantitative evaluation by Visual Analog Scale (VAS)

A Visual Analog Scale (VAS) was employed for the evaluation, which is a measurement method that enables continuous subjective evaluation, and respondents responded by placing a mark at an arbitrary position on a horizontal line.

In this study, we used VAS to quantitatively analyze learners' subjective evaluations of parallel learning in R and Python from four perspectives: understanding, confidence, motivation, and potential. It has the advantage of being able to detect minute differences that are difficult to capture by using a conventional Likert scale. A scale of to 0-100 was used for each evaluation index to measure the learner’s perception more precisely. In this analysis, the left and right ends of the line were set as 0 and 100, respectively, and the results were quantified as integer values. Figure 1 shows the questionnaire form used.



Survey on Parallel Learning Experience of R and Python Programming Languages

Please answer the following questions regarding what you learned during the lecture:

* 1. **Understanding and Confidence**
(R) To what extent do you understand the content of the lectures?

No understanding at all — Complete understanding

* 2. (Python) To what extent do you understand the lecture content?

No understanding at all — Complete understanding

* 3. (R) How confident are you of your ability to modify the programs covered in the lectures?

Not confident at all — Very confident

* 4. (Python) How confident are you of your ability to modify the programs covered in the lectures?

Not confident at all — Very confident

Figure 1 Online survey form (in part, VAS)

The following four items were evaluated for the R language and Python:

1. “Understanding” (Q1: R, Q2: Python)
- "To what extent do you understand the content of the programs covered in the lecture?"
- 0: " No understanding at all" - 100: " Complete understanding "

2. “Confidence” (Q3: R, Q4: Python)
- How confident are you in your ability to modify the programs covered in the lecture?"
- 0: " Not confident at all " to 100: "Very confident "

3. “Motivation” (Q5: R, Q6: Python)
- How much do you enjoy learning?"
- 0: "Not enjoyable at all" to 100: "Extremely enjoyable "

4. “Potential” (Q7: R, Q8: Python)
- How useful do you think it will be for your future career?"
- 0: "Not useful at all" to 100: "Very useful"

The following questions were asked to measure language preference (Q9).

- "When comparing the R language and Python, which do you prefer more?"
- 0: "Strongly prefer R" to 100: "Strongly prefer Python"

Questions were also asked about programming experience (Q10) and the desired major course of study (Q11) as background information. ("Extensive programming experience" was selected by 0 respondents)

Programming experience (Q10)

- No prior programming experience
- Limited experience
- Extensive programming experience

Preferred Course of Study (Q11)

- Business
- Health care

In the free description field, there were entries for mathematics, AI, teaching, and sensibility that were undecided.

Free-text Qualitative Evaluation

The questions were in the form of free-answer questions.

Q12: "Please describe your preferences for Python and R, including the specific aspects that you like or find challenging in each language."

Additional reasons for adopting the VAS include the following.

1. Continuous evaluation is possible and can capture minute differences
2. Quantifies respondents' subjective evaluations and facilitates statistical analysis.
3. Visually intuitive answers are possible
4. Suitable for comparative evaluation of both languages

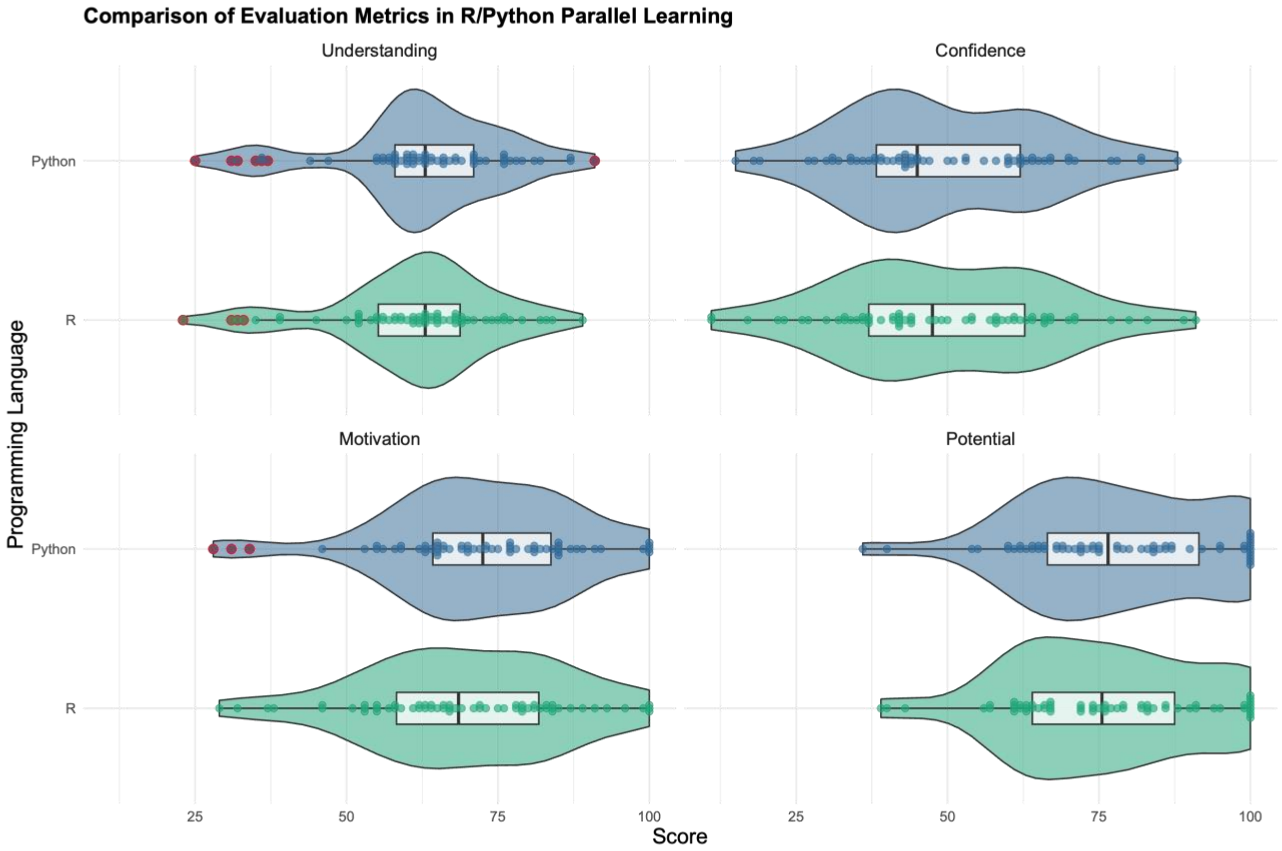


Figure 2 Distribution of Four Metric VAS Score

Table 1 Descriptive Statistics for VAS Scores

Metric	Language	*n*	Mean	SD	Median	Min	Max
Understanding	R	62	61.16	13.45	63.0	23	89
	Python	62	62.39	14.00	63.0	25	91
Confidence	R	62	49.47	18.17	47.5	11	91
	Python	62	49.65	16.63	45.0	15	88
Motivation	R	62	69.73	16.82	68.5	29	100
	Python	62	72.39	15.84	72.5	28	100
Potential	R	62	76.00	15.85	75.5	39	100
	Python	62	78.27	15.83	76.5	36	100

Wilcoxon's signed-rank test was used to compare the R and Python languages in this study. The test results are listed in Table 2.

Table 2 Results of Wilcoxon Signed-Rank Test between R and Python VAS Scores

Basic Test Statistics

Metric	W-stat	p	p (adj)	Effect size r	Effect Int.
Understanding	512.0	0.012	0.049	0.318	Medium
Confidence	701.5	0.904	1.000	0.015	Negligible
Motivation to learn	369.5	0.004	0.014	0.370	Medium
Potential	382.0	0.008	0.033	0.335	Medium

Detailed Effect Analysis

Metric	Confidence Interval		H-L est.	Significance
	(Lower)	(Upper)		
Understanding	-3.000	-0.500	-1.500071	TRUE
Confidence	-1.500	1.000	-0.000058	FALSE
Motivation to learn	-5.000	-1.000	-2.999991	TRUE
Potential	-3.500	-0.500	-2.000036	TRUE

Using this data, quantitative (descriptive and inferential statistics) and qualitative (text mining) analyses were conducted to examine the educational effects of learning R and Python in parallel from multiple perspectives.

First, the VAS score analysis was explained. Figure 2 shows a graph that visualizes the distribution of the four metrics (Understanding, Confidence, Motivation, and Potential) by superimposing a box-and-whisker plot on a violin plot and bee swarm plot. The outliers are indicated in red. Table 1 summarizes the descriptive statistics of the VAS scores.

To simultaneously analyze the four metrics ("Understanding," "Confidence," "Motivation," and "Potential"), this study employed adjusted values using the Bonferroni method. This ensures the appropriate control of the first type of error and statistical reliability of the conclusions. Note that the adjusted value is calculated as four times the original value, and if it exceeds one, it is rounded to one. The results of the analysis of these four metrics are described below.

1. Comparative analysis of "Understanding"

The results of the Wilcoxon signed-rank test for "Understanding" confirmed a statistically significant difference between R and Python (W = 512.00, p = 0.012, r = 0.318). The effect size was medium, with a Hodges-Lehmann estimator of -1.50 (95% CI: [-3.00, -0.50]). Descriptive statistics showed that Python (mean = 62.39, SD = 14.00, median = 63.00) had slightly higher understanding than R (mean = 61.16, SD = 13.45, median = 63.00). From the distribution plots in Figure 2, a higher concentration of scores was observed in Python, although the variation in understanding was similar in both the languages.

2. Comparative analysis of "Confidence"

No statistically significant difference was detected for "Confidence" (W = 701.50, p = 0.904, r = 0.015). Effect sizes were negligible, with a Hodges-Lehmann estimator of 0.00 (95% CI: [-1.50, 1.00]). For descriptive statistics, R (mean = 49.47, SD = 18.17, median = 47.50) and Python (mean = 49.65, SD = 16.63, median = 45.00) were nearly equivalent. From the distribution chart in Figure 2, we can observe that the distribution pattern of confidence in both languages is similar and tends to be concentrated around the median value.

3. Comparative analysis of "Motivation"

A significant difference was detected for "Motivation" (W = 369.50, p = 0.004, r = 0.370). The effect size was medium, with a Hodges-Lehmann estimate of -3.00 (95% CI: [-5.00, -1.00]), with Python (mean = 72.39, SD = 15.84, median = 72.50) showing significantly higher learning motivation than R (mean = 69.73, SD = 16.82, median = 68.50), indicating a significantly higher motivation than R (mean = 69.73, SD = 16.82, median = 68.50). The distribution plots in Figure 2 confirm the bias toward the higher-score range in Python.

4. Comparative analysis of "Potential"

Significant differences were also identified in perceptions of "Potential" (W = 382.00, p = 0.008, r = 0.335). The effect size was medium, with a Hodges-Lehmann estimate of -2.00 (95% CI: [-3.50, -0.50]), with Python (mean = 78.27, SD = 15.83, median = 76.50) receiving slightly higher ratings of future potential than R (mean = 76.00, SD = 15.85, median = 75.50). The distribution in Figure 2 shows that both languages have a slightly higher evaluation of future potential than the other languages. The distribution in Figure 2 shows a bias toward higher scores in both languages but is more pronounced in Python.

The results of these analyses showed significantly more positive evaluations of Python, particularly in terms of motivation and perceived future potential. However, the differences in understanding were small and no differences were observed in confidence, suggesting that parallel learning does not cause a significant bias in basic language acquisition.

5.3.2 Sentiment analysis of free-text comments

For the sentiment analysis of free-text comments, we employed a method combining morphological analysis and a sentiment word dictionary. First, morphological analysis was performed using RMeCab (Version 1.0) to extract nouns, verbs, and adjectives. To extract vocabulary containing sentimental expressions, vocabulary was classified according to the following criteria.

- Positive vocabulary: "fun," "interesting," "good," "like," "easy to understand," "simple," "convenient," "useful," "efficient," "attractive," "friendly," "easy to use," etc.
- Negative vocabulary: "difficult," "complicated," "hard to understand," "difficult," "confusing," "poor at," "troublesome," "demanding," "anxious," "dislike," "hard to use," etc.

Sentiment scores were calculated using the following formula:

$$S = \frac{p - n}{p + n + 1}$$

p: total number of positive expressions, *n*: total number of negative expressions

where p is the total number of positive expressions and n is the total number of negative expressions. Adding 1 to the denominator prevents zero division and normalizes the score to the range [-1, 1]. This sentiment score indicates that the closer the value is to -1, the more negative the sentiment expressions are, and the closer it is to 1, the more positive the sentiment expressions are.

Table 3 shows the descriptive statistics of the sentiment scores by language-preference group. The Python weak preference group had a neutral mean sentiment score of 0.000 and lowest standard deviation of 0.243. In contrast, the R weak preference group had a mean sentiment score of 0.271 and the Python strong preference group had a mean sentiment score of 0.389, indicating relatively positive sentimental expressions.

Table 3 Sentiment Scores by Language Preference Group

Language Preference Group	n	Mean	SD	Median	Positive Ratio
Strong R Preference	20	0.000	0.243	0.000	11.1 %
Weak R Preference	12	0.271	0.295	0.250	50.0 %
Weak Python Preference	18	0.000	0.243	0.000	61.5 %
Strong Python Preference	12	0.389	0.347	0.500	66.7 %

As shown in Figure 3, the distribution of sentiment scores was visualized using a combination of violin, box-and-whisker, and bee swarm plots. The analysis showed that the R strong preference group and Python weak preference group tended to concentrate their emotion scores on approximately 0. In contrast, the R weak preference group and Python strong preference group showed a relatively wide distribution of emotion scores and a tendency to use more positive emotional expressions. In particular, the Python strong-preference group showed the highest distribution of emotion scores compared with the other groups.

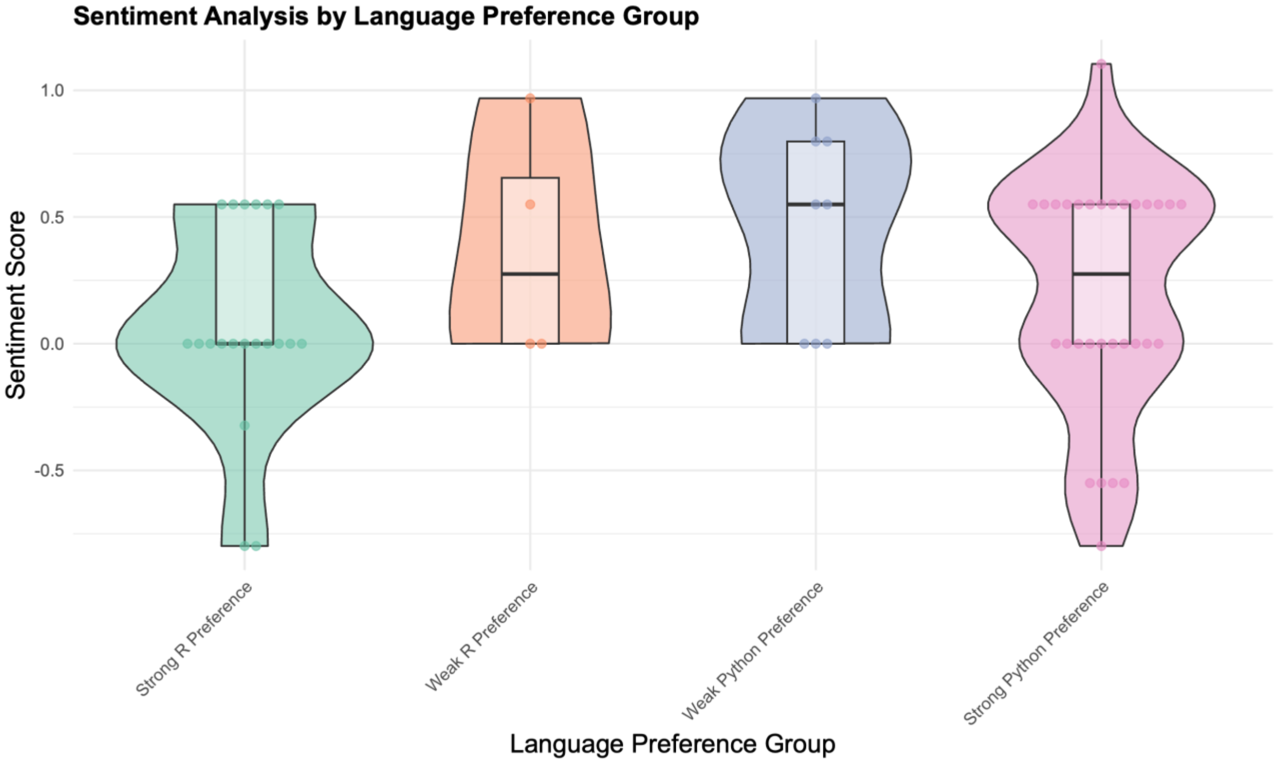


Figure 3. Distribution of sentimental expression by language preference group

To analyze the differences in emotional expressions between the language preference groups in more detail, we focused our analysis on the ratio of positive emotional expressions. As shown in Figure 4, the positive ratio for each group was visualized as a bar graph.

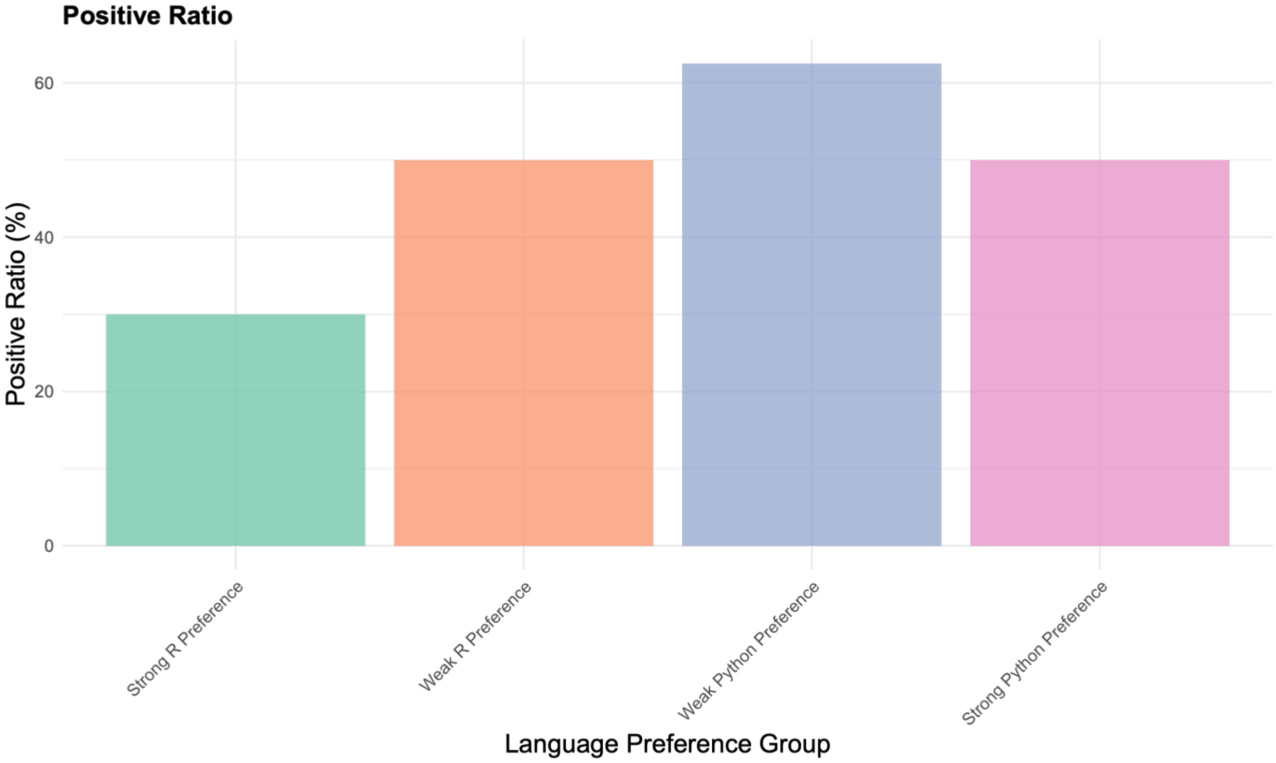


Figure 4: Positive Ratios by Language Preference Group

The analysis showed that The Python strong-preference group had the highest positive ratio (66.7 %), followed by the Python weak-preference group (61.5 %). In contrast, the R weak preference group showed a neutral value of 50.0%, whereas the R strong preference group had the lowest positive ratio (11.1 %). These results suggest that

Python-preferring learners tended to use more positive expressions.

The results of this analysis indicate that students' affective responses to parallel learning reflect complex evaluations based on their understanding of language characteristics and learning experiences rather than simple likes and dislikes. In particular, the tendency of students who express a strong preference to make more objective evaluations suggests that parallel learning may promote a deeper understanding of language characteristics.

Finally, we analyzed the characteristic patterns of sentimental expressions from the learners' free descriptions. In positive contexts, expressions such as "like," "easy," and "easy to understand" were commonly used, especially positive evaluations regarding the "ease of understanding" of the language. On the other hand, in negative contexts, expressions such as "difficult," "complicated," and "confusing" confusing were observed, with particular reference to the difficulties in dealing with errors.

Furthermore, distinctive expression patterns were identified for each language preference group: the R language preference group often referred to the intuitive understandability of statistical processing, while the Python preference group was characterized by references to the abundance of libraries and practicality. In the neutral group, many objective expressions that compared and evaluated the features of both the languages were observed. These expression patterns may reflect differences in perceptions of each group's language preferences.

It should be noted that the results described above are only preliminary experiments, and that the findings were obtained under limited conditions of educational practice for first-term students. It is expected that more effective educational methods will be established in the future through further practice and accumulation of research.

These preliminary results largely support our theoretical expectations outlined in Section 4, particularly regarding the development of metalinguistic competence and effectiveness of the BYOD environment. However, they also highlight several areas requiring attention in full implementation, such as the need for more structured support for students who show strong language preferences. These insights will be invaluable in refining our approach to the 2025 implementation.

6. Discussion

This study was designed to test the effectiveness of a parallel learning approach for Python and R languages in data science education. Below, we discuss key aspects of this study and provide future perspectives.

6.1 Effectiveness of the concurrent learning approach

In developing our theoretical framework, we carefully considered parallels from natural language acquisition research, while acknowledging the fundamental differences between natural and programming languages. Programming languages, which are formal systems with precise syntax and semantics, are likely to engage different cognitive processes than natural languages. However, insights from multilingual education research provide useful metaphors and hypotheses for investigating how learners might effectively acquire multiple programming languages. The following analysis explores these potential parallels, while maintaining appropriate theoretical caution.

The concurrent learning approach proposed in this study has the potential to provide a new perspective on data science education. While acknowledging the fundamental differences between natural and programming languages, we draw inspiration from theories of multilingual acquisition in natural language learning (cf. Hufeisen and Marx, 2007) [40] to explore potential parallels in how learners might develop competencies across multiple programming languages. The following benefits are expected.

1. Improved meta-linguistic competence: Learning multiple languages in a comparative manner may deepen the understanding of the abstract structure of languages and improve adaptability to new languages and frameworks.
2. Diversification of problem-solving approaches: Understanding the characteristics of different languages enables multifaceted approaches to problems and fosters creative problem-solving skills.
3. Readiness for practical work: It is expected that mastery of both languages, which are widely used in industry, will facilitate adaptation to the work environment after graduation.

However, the possibility of increased cognitive load and decreased learning efficiency owing to parallel learning must also be considered, and these points must be carefully examined.

6.2 Advantages and Challenges of a BYOD Environment

The BYOD (Bring Your Own Device) environment employed in this study is expected to have the following advantages and challenges

Advantages:

1. Continuity of learning: Personal devices allow for a seamless learning environment on- and off-campus.
2. Customizability: This enables the environment to be configured to meet the needs of individual students.
3. Cost efficiency: May reduce capital investment in part of the university.

Issue:

1. Environmental diversity: Differences in device performance and settings among students may affect their learning experience.
2. Security Risk: Security management of personal devices is challenging.
3. Technical Support: The burden of technical support for a wide variety of devices may increase.

To address these issues, it is necessary to develop appropriate guidelines and establish a technical support system, particularly to address the diverse learning spaces and technical challenges identified by Sølvberg and Rismark (2012) [41] in their study of mobile learning environments.

6.3 Optimize educational methods specific to your field of expertise

This study focuses on two specialized areas: business data science and healthcare data science. The following points are important for the optimization of educational methods specific to these disciplines.

1. Integration of Domain Knowledge: Curriculum that effectively integrates programming education with domain knowledge should be developed.
2. Use of real data: Efforts should be made to bridge the gap between theory and practice through exercises using actual datasets from each field.
3. Industry-academia collaboration: Collaboration between companies and medical institutions is important for designing curricula that reflect industry needs.

Through these optimizations, more effective professional education based on Kolb's (1984) [42] experiential learning cycle, which emphasizes concrete experience, reflective observation, abstract conceptualization, and active experimentation, is expected to be realized.

6.4 Limitations of the study and future challenges

The following limitations and issues exist in this study and require further investigation in future studies.

1. Sample size and generalizability: Since the study was conducted at a single university, the generalizability of the results should be done with caution. Further validation in multiple institutions is warranted.
2. Verification of long-term effects: Long-term effects that cannot be captured in a one-year observation period require follow-up studies.
3. Consideration of individual differences: A more detailed analysis of the impact of individual differences in learning styles and cognitive abilities on results is needed.
4. Validity of evaluation indicators: Further verification and improvement of evaluation indicators for programming and problem-solving skills are needed.
5. Role of educators: Further research on the skills and role of educators in effectively implementing a concurrent learning approach is needed.

It is hoped that addressing these issues will more clearly demonstrate the effectiveness of the concurrent learning approach in data science education, and lead to the establishment of a practical educational model. In the future, the findings of this study can be used as the basis for planning larger-scale, longer-term research projects.

Finally, this study proposes a new approach to data science education, and the results may have important implications for educational practices and policymaking. However, we must not forget that the essence of education lies in the growth of individual learners. We conclude this paper by emphasizing the importance of flexible educational approaches that respond to the needs and potential of each individual learner as well as technological and methodological advances.

References

[1] Munehiko Sasajima, Ken Ishibashi, Takehiro Yamamoto, Takayuki Yumoto, Hiroaki Ohshima, Tetsuya Fujie, Naoki Kato (2024), "On the Education of Lower Grades in the Faculty of Data Science through Problem-Solving Exercises Using Real Data Reflections on PBL Exercises through FY2023", Proceedings of the Annual Conference of JSAI. (in Japanese) DOI: 10.11517/pjsai.JSAI2024.0_3P1GS304

[2] Kumar, S., Aithal, S. (2020), "How Lucrative & Challenging the Boundaryless Opportunities for Data Scientists?", International Journal of Management, Technology, and Social Sciences, 5(2), pp.110-129. DOI: 10.5281/ZENODO.3966222

[3] Donoho, D. (2017), "50 years of data science", Journal of Computational and Graphical Statistics, 26(4), pp.745-766. DOI: 10.1080/10618600.2017.1384734

[4] Bialystok, E., Craik, F. I., Green, D. W., & Gollan, T. H. (2009), "Bilingual minds". Psychological science in the public interest, 10(3), pp.89-129. DOI: 10.1177/1529100610387084

[5] Dewey, J. (1938), "Experience and education", New York: Macmillan.

[6] Kolb, D. A. (1984), "Experiential Learning: Experience As The Source Of Learning And Development", Journal of Business Ethics.

[7] Lave, J., & Wenger, E. (1991), "Situated learning: Legitimate peripheral participation", Cambridge University Press. DOI: 10.1017/CBO9780511815355

[8] Toshiki Shimbaru (2023), "Facts about Mathematical, Data Science, and AI Education Programs (数理・データサイエンス・AI 教育プログラムの実状)", Japan Journal of Educational Technology., 47(2), pp.333-342. (in Japanese) DOI: 10.15077/jjet.46095

[9] Tomohiro Nakada, Claudia Gherghel (2024), "Report on Questionnaire Survey in Introduction to Data Science", Research Report of JSET Conferences, 2024(2), pp.2436-3286. (in Japanese) DOI: 10.15077/jsetstudy.2024.2_91

[10] Yusupova, S. B. et al. (2022), "The Advantage of Using e-Learning in Teaching Students Programming Languages", IEEE Region International Conference. DOI: 10.1109/SIBIRCON56155.2022.10017040

[11] Robins, A. (2019), "Novice programmers and introductory programming", In The Cambridge Handbook of Computing Education Research (S. Fincher & A. Robins, Eds.). Cambridge University Press, 327-376. DOI: 10.1017/9781108654555.013

[12] Perez, F., & Granger, B. E. (2015), "Project Jupyter: Computational narratives as the engine of collaborative data science", Retrieved September, 11(207), 108. https://blog.jupyter.org/project-jupyter-computational-narratives-as-the-engine-of-collaborative-data-science-2b5fb94c3c58

[13] Rule, A., Birmingham, A., Zuniga, C., Altintas, I., Huang, S. C., Knight, R. & Rose, P. W (2019), "Ten simple rules for writing and sharing computational analyses in Jupyter Notebooks", PLoS Computational Biology, 15(7), e1007007. DOI: 10.1371/journal.pcbi.1007007

[14] V. F. Ochkov, A. Stevens and A. I. Tikhonov (2022), "Jupyter Notebook, JupyterLab – Integrated Environment for STEM Education", 2022 VI International Conference on Information Technologies in Engineering Education (Inforino), pp. 1-5, DOI: 10.1109/Inforino53888.2022.9782924

[15] Junko Kambe, Asuka Yamaguchi, Kazue Tamada, Toshiki Matsuda (2024), "A Preliminary Survey on Improving ICT Usage by New Students Using BYOD", Research Report of JSET Conferences, 2024(2), pp. 1-4. (in Japanese) DOI: 10.15077/jsetstudy.2024.2_1

[16] Shadish, W. R., Cook, T. D., & Campbell, D. T. (2002), "Experimental and quasi-experimental designs for generalized causal inference", Houghton Mifflin. https://api.semanticscholar.org/CorpusID:938218

[17] Leeks, S. (2020), "Windows Subsystem for Linux 2 (WSL 2): Tips, Tricks, and Techniques", Packt Publishing. https://packt.link/free-ebook/9781800562448

[18] Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B. E., Bussonnier, M., Frederic, J. & Jupyter development team (2016), "Jupyter Notebooks - a publishing format for reproducible computational workflows", International Conference on Electronic Publishing.
https://api.semanticscholar.org/CorpusID:36928206

[19] Gandrud, C. (2020), "Reproducible research with R and RStudio", Chapman and Hall/CRC.
https://christophergandrud.github.io/RepResR-RStudio/

[20] Alessandro Del Sole (2022), "Visual Studio Code Distilled: Evolved Code Editing for Windows, macOS, and Linux. 3rd ed.", Apress. DOI: 10.1007/978-1-4842-9484-0

[21] Nolan, D., & Lang, D. T. (2015), "Data Science in R: A Case Studies Approach to Computational Reasoning and Problem Solving", Chapman and Hall/CRC. DOI: 10.1201/b18325

[22] Biggs, J., Tang, C., & Kennedy G. (2023), "Teaching for quality learning at university 5th Edition", McGraw-hill education (UK).
https://www.mheducation.co.uk/teaching-for-quality-learning-at-university-5e-9780335250820-emea-group

[23] Hmelo-Silver, C. E. (2004), "Problem-based learning: What and how do students learn?", Educational psychology

review, 16(3), pp.235-266. DOI: 10.1023/B:EDPR.0000034022.16470.f3

[24] Vandergon, V. O., Reveles, J., Herr, N., Nguyen-Graf, D., Rivas, M., d'Alessio, M., & Foley, B. (2018), "Engaging Students in Conducting Data Analysis: The Whole-Class Data Advantage", In I. Management Association (Ed.), K-12 STEM Education: Breakthroughs in Research and Practice, pp. 970-996, IGI Global. DOI: 10.4018/978-1-5225-3832-5.ch045

[25] Guest, G., Fleming, P (2014), "Mixed Methods Research", In Public Health Research Methods (pp. 581-614), SAGE Publications, Inc. DOI: 10.4135/9781483398839.n19

[26] Shirahama, N., Watanabe, S., Ikegami, F., & Mori, Y. (2016), "Development of VAS App to Improve the Youth Mental Health Research Environment", ICIC Express Letters Part B: Applications, 7(9), pp.1939-1944. DOI: 10.24507/icicelb.07.09.193

[27] Braun, V., & Clarke, V. (2006), "Using thematic analysis in psychology", Qualitative research in psychology, 3(2), pp.77-101. DOI: 10.1191/1478088706qp063oa

[28] Fetters, M. D., Curry, L. A., & Creswell, J. W. (2013), "Achieving integration in mixed methods designs—principles and practices", Health services research, 48(6pt2), pp.2134-2156. DOI: 10.1111/1475-6773.12117

[29] World Medical Association (2013), "World Medical Association Declaration of Helsinki: ethical principles for medical research involving human subjects", JAMA, 310(20), pp.2191–2194. DOI: 10.1001/jama.2013.281053

[30] Cummins, J. (1979), "Linguistic interdependence and the educational development of bilingual children", Review of Educational Research, 49(2), pp.222-251. DOI: 10.2307/1169960

[31] Bialystok, E. (2001), "Bilingualism in development: Language, literacy, and cognition", Cambridge University Press. DOI: 10.1017/CBO9780511605963

[32] Grosjean, F. (2010), "Bilingual: Life and reality", Harvard University Press. DOI: 10.4159/9780674056459

[33] Guilford, J. P. (1967), "The nature of human intelligence", McGraw-Hill. DOI: 10.3102/00028312005002249

[34] Papert, S. (1980), "Mindstorms: Children, computers, and powerful ideas". Basic Books, Inc. DOI: 10.5555/1095592

[35] Vessey, I. (1985), "Expertise in debugging computer programs: A process analysis", International Journal of Man-Machine Studies, 23(5), pp.459-494. DOI: 10.1016/S0020-7373(85)80054-7

[36] Lave, J., & Wenger, E. (1991), "Situated learning: Legitimate peripheral participation", Cambridge University Press. DOI: 10.1017/CBO9780511815355

[37] Krashen, S. D. (1982), "Principles and practice in second language acquisition", Pergamon Press.

[38] Zimmerman, B. J. (2002), "Becoming a self-regulated learner: An overview", Theory Into Practice, 41(2), pp.64-70. DOI: 10.1207/s15430421tip4102_2

[39] Bransford, J. D., Brown, A. L., & Cocking, R. R. (2000), How people learn: Brain, mind, experience, and school: Expanded edition. National Academy Press.

[40] Hufeisen, B., & Marx, N. (2007), "How can DaFnE and EuroComGerm contribute to the concept of receptive multilingualism? Theoretical and practical considerations", In J. D. ten Thije & L. Zeevaert (Eds.), Receptive Multilingualism: Linguistic analyses, language policies and didactic concepts (pp. 307-321). John Benjamins Publishing Company. DOI: 10.1075/hsm.6.20huf

[41] Sølvberg, A. M., & Rismark, M. (2012), „Learning spaces in mobile learning environments", Active Learning in Higher Education, 13(1), pp.23-33. DOI: 10.1177/146978741142918

[42] Kolb, D. A. (1984), "Experiential learning: Experience as the source of learning and development" Prentice-Hall.