2025 Volume: 01 Issue: 06 www.ijrpa.com ISSN 2456-9995 Review Article

International Journal Research Publication Analysis

Page: 01-09

EDUCATION IN PYTHON USING NUMPY AND PANDAS

Fatema Bohra*[1], Dr. Vishal Shrivastava[2], Er. Amit Kumar Tewari[3], Dr. Akhil Pandey[4]

[1]Artificial Intelligence and Data Science, Arya College of Engineering & I.T., Jaipur, India

[2]Professor, Computer Science and Engineering, Arya College of Engineering & I.T. Jaipur, India.

[3]Associate Professor, Computer Science and Engineering, Arya College of Engineering & I.T. Jaipur, India.

[4]Professor Computer Science and Engineering Arya College of Engineering & I.T. Jaipur, India.

Article Received: 02 October 2025
Article Revised: 22 October 2025
Published on: 12 November 2025

***Corresponding Author: Fatema Bohra**
Artificial Intelligence and Data Science, Arya College of Engineering & I.T., Jaipur, India. DOI: https://doi-doi.org/101555/ijrpa.8768

ABSTRACT

In today's digital world, learning programming isn't just for computer scientists — it's a valuable skill for everyone. Python has become one of the most popular programming languages because it's easy to understand and powerful enough to do big things. In this research, we explore how Python, especially through two popular libraries — **NumPy** and **Pandas** — is making education more practical and interactive. NumPy helps students understand numbers, arrays, and math operations, while Pandas allows them to analyze real-world data, like marksheets or survey results, just like they would in Excel — but with code. Together, these tools make learning fun, improve problem-solving skills, and prepare students for real-life applications in data science, analytics, and beyond.

1. INTRODUCTION

Python is now one of the top choices when it comes to learning how to code. Why? Because it's simple, clean, and doesn't confuse beginners with too much technical stuff. But Python is more than just basic programming — it's a gateway to **data science**, **AI**, and **automation**.

In schools and colleges, Python is often introduced early in programming courses. However, students usually find it more exciting when they move beyond just "printing Hello World" and start working with real data. That's where **NumPy** and **Pandas** come in.

- **NumPy** helps students handle numbers, arrays, and math operations — like matrices and averages — in a very clean and fast way. It replaces manual calculation with powerful coding.
- **Pandas** helps students play with tabular data — think of Excel files, CSVs, or marksheets — and allows them to clean, sort, and analyze the information quickly.

Together, these libraries help students connect theory with practice. Instead of just learning formulas, they can see them work on real data. Whether it's analyzing student scores, handling data for a science project, or exploring trends in weather, these tools make learning more meaningful. This paper explains how Python with NumPy and Pandas can make education smarter, more useful, and more fun.

2. Related Works

Over the past few years, Python has become a favorite programming language in the field of education and data science. Many researchers and educators have noticed that students understand concepts better when they use Python along with libraries like **NumPy** and **Pandas**.

Studies have shown that when students learn math, statistics, or data analysis using real-life datasets in Python, they find the subjects more interesting and easier to understand. For example:

- Some universities have started using **NumPy** to teach matrix operations and linear algebra because it simplifies complex calculations.
- **Pandas** has been introduced in data handling classes to help students clean and analyze data just like professionals do in companies.

Interactive platforms like **Jupyter Notebook** are also being used widely to let students code, see outputs instantly, and write notes in the same place — making it a great learning tool. Tools like **Matplotlib** are often added for visual learning, so students can create charts and graphs of their data, which helps them understand patterns better.

The success of these teaching methods proves that Python with NumPy and Pandas not only makes learning fun but also gives students skills that are directly useful in careers like data science, business analytics, and machine learning.

3. Proposed Methodology

In this research, we want to show how Python, with the help of **NumPy** and **Pandas**, can be used effectively in education to teach students how to handle real-world data and understand mathematical concepts through hands-on practice.

Here's how the process works step by step:

Step 1: Learn Basic Python

We start by teaching the basics of Python — things like variables, loops, and functions. This builds a strong foundation for what comes next.

Step 2: Use NumPy for Numerical Learning

Once students understand Python, we introduce them to **NumPy**. They learn how to:

- Create arrays (like rows of marks)
- Do calculations such as averages, sums, or matrix multiplications
- Understand math concepts in a simpler, code-based way

For example

python

CopyEdit

```
Import numpy as np
Marks = np.array([80, 85, 90])
Print ("Average Marks:", np.mean(marks))
```

Step 3: Use Pandas to Work with Real Data

Next, students use **Pandas** to load and explore datasets. For instance, they can read a CSV file that has names, marks, and subjects, and then:

- Filter out top scorers
- Calculate totals and averages
- Sort students based on performance

Python

CopyEdit

```
import pandas as pd
df = pd.read_csv("students.csv")
df['Average'] = df[['Math', 'Science', 'English']].mean(axis=1)
top_students = df[df['Average'] > 85]
```

```
print(top_students)
```

Step 4: Mini Project or Case Study

After learning the basics, students are given a small project like analyzing student marks or a survey dataset. They apply what they learned to:

- Clean the data
- Calculate stats
- Draw conclusions

Flowchart of the Learning Process

Plaintext

CopyEdit

Start

↓

Learn Python Basics (Syntax, Loops, Functions)

↓

Explore NumPy (Arrays, Math Operations)

↓

Explore Pandas (DataFrames, Filtering, Analysis)

↓

Mini Project with Real Dataset

↓

Review and Assessment

↓

End

4. Role of NumPy in Python

4.1 What is NumPy?

NumPy is a library for performing **numerical computations** efficiently. It introduces a powerful **N-dimensional array object (ndarray)** and provides a vast collection of mathematical functions to operate on these arrays.

4.2 Why NumPy?

Python's built-in lists are:

- Not optimized for numerical operations

- Slower and memory-inefficient when handling large data

NumPy solves this with:

- **Compact, fast arrays** that use less memory
- **Vectorized operations** (loop-free computation)
- **Broadcasting** (applying operations across different shapes

4.3 Features of NumPy

Feature	Description
ndarray	A fast, memory-efficient multi-dimensional array
Mathematical Functions	Operations like mean, median, dot product, etc.
Linear Algebra	Matrix multiplication, eigenvalues, inverse, etc.
Random Numbers	Simulations, shuffling, sampling
Fourier Transform	For signal processing
Broadcasting	Efficiently perform operations on different-shaped arrays

4.4 NumPy in Action

python

CopyEdit

```
import numpy as np
# Create array
arr = np.array([10, 20, 30, 40])

# Compute statistics
Print ("Mean:", np.mean(arr))
Print ("Standard Deviation:", np.std(arr))
Print ("Squared Elements:", np.square(arr))
```

4.5 Use Cases of NumPy

- Scientific computing
- Machine learning algorithms
- Physics simulations
- Image and signal processing
- Education: Helping students visualize matrix operations and vector calculations

5 Role of Pandas in Python

5.1 What is Pandas?

Pandas is a data manipulation and analysis library built on top of NumPy. It introduces two powerful data structures:

- **Series** (1D labeled array)
- **DataFrame** (2D labeled table like a spreadsheet)

It simplifies data handling, cleaning, transformation, and analysis — all essential skills in data science and education.

5.2 Why Pandas?

Pandas makes it:

- Easy to load, explore, and modify real-world datasets (CSV, Excel, SQL)
- Simple to filter, group, merge, and transform data
- Powerful for time-series analysis, missing value handling, and summarization

5.3 Features of Pandas

Feature	Description
DataFrames	Excel-like tables with rows and columns
Read/Write	CSV, Excel, JSON, SQL
Data Cleaning	Handle missing values, remove duplicates
Grouping	Summarize data using groupby()
Time Series	Indexing and resampling with dates
Joining/Merging	Combine multiple data tables

5.4 Pandas in Action

python

CopyEdit

```
import pandas as pd
# Load dataset
data = pd.read_csv("students.csv")
# View first rows
print(data.head())
# Average marks
print("Average Marks:", data['Marks'].mean())
# Filter top scorers
```

```
top = data[data['Marks'] > 80]
print(top)
```

5.5 Use Cases of Pandas

- Data preprocessing in machine learning pipelines
- Exploratory data analysis (EDA)
- Report generation and business analytics

5.6 Comparison Table: NumPy vs Pandas

Feature	NumPy	Pandas
Structure	N-dimensional array (ndarray)	Series and DataFrame
Data Type	Numerical data only	Mixed types (strings, dates, etc.)
Speed	Extremely fast for math	Fast for data manipulation
Use Case	Scientific computing	Data cleaning, exploration
Syntax	Closer to math (indexing)	Closer to spreadsheet users
Dependencies	Base for many other libraries	Built on top of NumPy

5.7 Importance in Education

How NumPy Helps in Teaching

- Teaches matrix manipulation, vector math
- Helps students write efficient code
- Builds foundation for ML algorithms and simulations

How Pandas Helps in Teaching

- Provides real-world data for practice
- Improves understanding of database-like operations
- Prepares students for data science jobs and research.

CONCLUSION

In today's technology-driven world, learning how to work with data is becoming just as important as learning how to read or write. This research showed how Python — especially with its libraries **NumPy** and **Pandas** — can make education more practical, interesting, and future-ready.

By using **NumPy**, students can understand mathematical operations like averages, arrays, and matrix calculations in a much easier and faster way. It helps turn difficult concepts into simple, hands-on learning experiences.

Pandas, on the other hand, allows students to work with real-life data — just like how professionals handle Excel files or online data. It helps them learn how to clean, filter, and analyze data, which are valuable skills in any field today — from business and science to social studies.

What makes Python, NumPy, and Pandas so powerful in education is that they:

- Help students move from just theory to real-world practice
- Build skills needed for careers in data science, analytics, AI, and more
- Make learning interactive and visual using tools like Jupyter Notebooks

Future Scope of Python

Python has grown exponentially over the past decade and is now one of the **most in-demand and versatile programming languages** in the world. Its simplicity, readability, and vast library ecosystem make it suitable for a wide range of domains — and its future looks even brighter.

REFERENCES

1. McKinney, W. (2012). *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython*. O'Reilly Media.
2. Oliphant, T. E. (2006). *A Guide to NumPy*. USA: Trelgol Publishing.
3. VanderPlas, J. (2016). *Python Data Science Handbook: Essential Tools for Working with Data*. O'Reilly Media.
4. Raschka, S., & Mirjalili, V. (2019). *Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow 2*. Packt Publishing.
5. Chollet, F. (2017). *Deep Learning with Python*. Manning Publications.
6. Pedregosa, F., et al. (2011). *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, 12, 2825-2830.
7. Abadi, M., et al. (2016). *TensorFlow: A System for Large-Scale Machine Learning*. Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), 265-283.

8. Paszke, A., et al. (2019). *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. Advances in Neural Information Processing Systems, 32, 8024–8035.
9. Hunter, J. D. (2007). *Matplotlib: A 2D Graphics Environment*. Computing in Science & Engineering, 9(3), 90–95.