I am applying to graduate school programs right now. One kind of programs I am applying for is Data science. When I checked the curriculum, there is often a course called Data Science in Python (or R). I think for data scientist (and maybe all scientists who want to apply their knowledge in computing), they do not want to and need to know how memory is allocated, and how each type of data is restricted and how garbage collector collects unused RAM. As a result, Python as a readable and user-friendly language is more suitable for them to design high–level structure.

The other kind of programs I am applying for is Bioinformatics. Python is also widely used in this field. Bioinformatics software pays particular attention to the challenges of organizing, searching, and manipulating enormous quantities of biological data1. Python’s simple syntax makes itself available to biologists who have less computer science background.

There is so many bioinformatics software which is written in Python. They are involved in data manipulation and analysis. For instance, Biopython is a set of freely available tools for biological computation like gene sequencing data mapping and filtering2.

Also, due to Python’s relatively slow speed, it is sometimes used as the glue between languages like C and C++. In this case, how the algorithms are implemented can be a headache for Python users. Python is not the most used language in Bioinformatics compared to R. Because R is more specifically designed for list and matrices manipulation and visualization. Python instead has more flexible functions, which make it accessible in so many fields.

Python is intensively used in Machine learning, artificial intelligence. The reasons I found for this phenomenon is that Python has an extensive selection of libraries and frameworks, which is similar to the modules we learned in lectures. It also reduces the cognitive overhead on developers, freeing up their mental resources so that they can concentrate on problem-solving and achieving project goals. Finally, the simple syntax makes it easier to collaborate or transfer projects between developers3.

I am glad to add Python to my toolbox. Because life is short, Python helps us care more about real problems we are facing, instead of producing new problems from the machine people built.

Reference

1. Michell L Model, Bioinformatics Programming Using Python, preface xii, http://oez.es/Bioinformatics%20Programming%20Using%20Python.pdf
2. Biostars, Question: Which Bioinformatics Tools Are Written In Python, https://www.biostars.org/p/50749/
3. Why Is Python So Good for AI, Machine Learning and Deep Learning https://www.netguru.co/blog/why-is-python-so-good-for-ai-machine-learning-and-deep-learning