BERKELEY INSTITUTE
FOR DATA SCIENCE

Berkeley Institute for Data Science
UNIVERSITY OF CALIFORNIA, BERKELEY
190 Doe Library
Berkeley, CA 94720
http://bids.berkeley.edu

Federico Levi
Email: federico.levi@nature.com

E-mail: stefanv@berkeley.edu
Phone: +1 510-701-4153

June 5, 2020

Dear Federico,

Thank you again for the careful review and helpful suggestions from you and the referees. We are delighted that you offered to publish our review article, if accepted, under a CC-BY 4.0 license at no charge.

Per your request, we revised our review to address the referees and to ensure that it is in Nature style throughout. Specifically:

1. We kept the main text below 3700 words.
2. We changed the title to “Array Programming with NumPy”.
3. In the abstract, we removed all references, made clear it is a review article, ended with a summary sentence, and stayed under the 200 word limit.
4. We made the methods section into a separate Supplementary Methods PDF and referenced it from the main text.
5. We annotated the four most significant references and moved URLs from the references to the text.
6. We created all images specifically for this publication and no display items in our manuscript are from third parties.
7. Per your email, we uploaded the final version of our review without signing the original copyright assignment form. We will wait for you to send another one with the right license.
8. We uploaded SVG versions of our main figures. There is one figure in the supplement in PDF format.
9. We uploaded a .tex document ordered as main text followed by figure legends.

Referee 1

Given how widely our philosophical outlook differs from that of referee 1, we cannot incorporate their feedback without substantially altering the gist of our paper. The aim

of our review is not to argue that NumPy is the best array programming library. We believe it is, but respect the choice of those who prefer languages like Matlab and R.

We agree that an abuse of vector syntax can produce dense, hard-to-read code. However, we did not find evidence to support the statement that "vectorization is known to be hard for new programmers". Furthermore, we have taught statistical & scientific computing extensively, and in our experience vector syntax is no harder to teach than many other programming concepts.

While not a focus of our review, we did briefly mention that there are other powerful scientific computing environments:

> The design of this new tool was informed by other powerful interactive programming languages for scientific computing such as Basis, Yorick, R, and APL, as well as commercial languages and environments like IDL and MATLAB.

We also mentioned new technologies on the horizon:

> New generation languages, interpreters, and compilers, such as Rust, Julia, and LLVM, will invent and determine the viability of new concepts and data structures.

They suggest that code should not have to be rewritten for new hardware; that would be ideal, and we would be delighted if modern compilers eventually allowed us to do so.

Referee 2

We addressed the two recommendations made by referee 2: we added more background on linear algebra libraries and random number generation in the supplemental material. We didn't mention masked arrays as that is a feature that will likely change (and perhaps even move out of NumPy) in the near future.

Linear Algebra. We expanded the list of linear algebra functionality provided by NumPy, referenced LAPACK, and added a sentence about open source software that preceded LAPACK.

Random Numbers. Added text mentioning the three new bit generators we recently added support for, including a link to empirical results testing their statistical soundness and performance.

Referee 3

Referee 3 made some good points that we agree with, but they requested no changes. They mentioned that a unified API would benefit the community; we agree, and are actively pursuing that goal.

Thank you for considering our resubmission; it's been a pleasure working with you.

Best regards,

Stéfan van der Walt

Jarrod Millman

Ralf Gommers