mathematics

MDPI

Editorial

Where Are We Going with Statistical Computing? From Mathematical Statistics to Collaborative Data Science

Dominique Makowski [1] and Philip D. Waggoner [2,*]

[1] School of Psychology, University of Sussex, Brighton BN1 9QH, UK; d.makowski@sussex.ac.uk
[2] Department of Data Science, YouGov & Columbia University, New York, NY 10027, USA
* Correspondence: pdw2119@columbia.edu

Abstract: The field of statistical computing is rapidly developing and evolving. Shifting away from the formerly siloed landscape of mathematics, statistics, and computer science, recent advancements in statistical computing are largely characterized by a fusing of these worlds; namely, programming, software development, and applied statistics are merging in new and exciting ways. There are numerous drivers behind this advancement, including open movement (encompassing development, science, and access), the advent of data science as a field, and collaborative problem-solving, as well as practice-altering advances in subfields such as artificial intelligence, machine learning, and Bayesian estimation. In this paper, we trace this shift in how modern statistical computing is performed, and that which has recently emerged from it. This discussion points to a future of boundless potential for the field.

Keywords: statistical computing; programming; open science; open source; future; collaboration

1. Introduction

Statistical computing has rapidly developed in recent years. With the rise of data science as an academic field, the advancement of the “open” movement (encompassing source code, science, replication and reproducibility, technology, and development), and the increasingly distributed world both in terms of collaboration and computing, statistical computing is officially in a brave new world.

The aim of this paper is twofold. First, we want to underscore the evolution of statistical computing from the perspective of development. How people develop, how they work together, and how statistical computing is pushed forward as a field are all questions based on new developments. The first section, then, is focused on the “how” of the development of statistical computing.

Second, we are interested in the question of “what” as it relates to advances in statistical computing. More practically, given that we have a sense of how people develop and deepen statistical computing and related techniques, the second section pivots to address the developments in precisely what is being developed in this modern landscape. In the second section, we will cover broad realms of techniques, technologies, and applications relating to modern advancements in statistical computing.

In sum, we believe the current moment of statistical computing and all of its associated developments open a door to an exciting world of open, democratized development and work that has never before existed at such a large scale. To maximize this incredible potential, we must embrace these advances and adopt these new ways of developing into our workflows, team formations, and collaborative efforts.

2. The “How”: New Ways of Working

In the past few years, the process of developing statistical computing tools and techniques has drastically changed. This can be seen in many realms, including team-based

check for updates

Citation: Makowski, D.; Waggoner, P.D. Where Are We Going with Statistical Computing? From Mathematical Statistics to Collaborative Data Science. *Mathematics* **2023**, *11*, 1821. https://doi.org/10.3390/math11081821

Received: 2 March 2023
Accepted: 5 April 2023
Published: 12 April 2023

development, decentralized collaborations, and more broadly through the open science movement as it relates to data science as a maturing field in its own rite. As such, the "how" as it relates to the development of statistical computing is multifaceted and constantly in flux. Only when we understand and adopt these new developments will we be able to maximize the great potential of modern applications and advancements of statistical computing.

Before continuing, it is important to begin this first section with a caveat. Though much of what follows in this paper outlining our thoughts on the current state of the field addresses concepts and topics relating to many adjacent fields (e.g., software engineering), we are focused primarily on statistical computing; broadly defined as the notion of writing programs of all shapes and sizes to solve statistical tasks. While the perspectives and points we raise throughout can and often do apply widely to fields beyond statistical computing, not everything may. Additionally, as a result, we limit the assumptions and implications of what we discuss to the world of statistical computing. We leave it to the reader to expand and adopt elsewhere as they would like.

2.1. Open Development

The last decades have seen a shift from professional software created by for-profit companies to free software. Under the former model, mathematical advancements and statistical decisions used to be made fairly independently of code implementation, creating a relatively well-defined boundary between statisticians (belonging to the field of mathematics) and programmers (belonging to computer science) tasked with software implementation. Nowadays, the boundaries are much more blurred, with code implementation becoming an integral part of statistical computing. With this model change from proprietary to free and open-source software came strong benefits, such as financial savings for institutions and individuals. However, it raised the question of a sustainable development model: how to incentivize and reward software developers when their product is free?

Interestingly, concomitant technological developments were able to provide a solution. The rise of open-access online software development platforms, such as GitHub and SourceForge, enabled developers to post their code or piece of software publicly and let other users re-use and contribute to it. In such a setting, benefits are widely distributed. Developers can share and collaborate on code and projects in a way previously unimaginable. Hosts of repositories or pieces of software, which are implicitly extending the offer to contribute given the open hosting of the software, benefit from the pooled resources of experts and niche specialists who can contribute to aspects of the project. The result is a collaborative piece of software that has received the attention (and critical assessment) of numerous and diverse domain experts. The second-order benefit of this arrangement is that this development is accomplished in a free, transparent way. The contributor is rewarded with a proof of participation and a demonstration of skills. The host, then, is rewarded with a better result than would have been impossible without this level of access and collaboration. Moreover, the development of crowdfunding/sponsorship initiatives allows monetary forms of contribution, opening, in principle, the door to full open-development careers. While this model naturally carries some risks of fueling precarious freelance developer positions, it is nonetheless a disruptive business model for professional software development whose full impact is yet to be seen.

Beyond the benefits to developers and software hosts/project leaders, open development contributes more broadly to science in the form of greater reproducibility and transparency [1]. This wave, which has taken on a life of its own in the form of "open science", is addressed further in the following section. However, at present, it is useful to point out the link between open and widespread collaboration that is native to open development, and the benefits flowing to all involved and beyond to science as a whole. This recent wave of open development, then, can be thought of as a tangible expression of an advancement that touches many fields from software development, statistics, and data science, to medicine, engineering, and the social sciences.

2.2. Open Science

As elaborated in the previous section, recent years have witnessed the formation and expansion of the open science movement in virtually every corner of science, research, and development [2]. We can characterize the open science movement as a dedication to openly and ethically designing research studies, and then carrying them out accordingly, making code and replication data freely available. Implicit is the desire to democratize the research enterprise, where any interested scholar is encouraged to test, critique, and even challenge the merits, claims, and inferences of a study. This brand of scientific advancement has its roots in the earliest days of scientific research with the publishing of the first scientific journals as far back as the 17th century during the Scientific [3]. In the modern conception of open science, which is a function of the earliest versions of sharing scientific information and research findings, there is a need to lay bare all aspects of the study, from the design, materials, methods, data, and code. In so doing, the development of a network of like-minded scholars building more directly on each other's work takes [4].

Like open development, there are many benefits that emerge from open science. Most notably, open science represents a move away from closed and often isolated research practices, where access to findings and processes are closely guarded secrets. By shifting toward an open scientific approach, more widespread sharing of ideas is possible, which benefits the careers and reputations of the researchers who advance the ideas in the first place [5].

Beyond the career benefits of open science, the move away from *closed* science to *open* science represents a shift in how scientific ideas are shared, and as a result how the contours of the modern scientific landscape are evolving. When ideas and findings are more widely shared, the opportunities for more and diverse voices to enter the conversation are concurrently widened.

Importantly, though, with every development comes the potential for negative effects and downsides. Open science is not immune to this. For example, efforts to move toward open science have at times resulted in reinforced inequality, especially in STEM professions [6]. Further, through a process of "platform capitalism", some have suggested the flaws in the scientific process that open science seeks to remedy are instead "re-engineered", or simply shifted and reinforced [7]. As a result, this line of logic suggests that open science simply covers the existing flaws without changing or fixing anything. Still, while biases, divisions, and inequalities have and do exist in the realm of open science, the broader push to make research processes more transparent and democratized is still, at its core, a very beneficial shift in how scientific research is accomplished.

2.3. Open Access

Closely related to open science is the open data sub-movement; that is, making the material from projects openly and freely available to the public. One of the prime values of openly sharing data is for transparency and replication purposes. There is an increased demand for and expectation of making study data available, which allows for results to be replicated, and high standards of research to be maintained. In fact, many journals are beginning to require data to be made publicly available if the paper is accepted for publication. Common outlets for open data storage and hosting include the Open Science Foundation [8] and the Harvard Dataverse.

Beyond data warehousing, open access and its impacts on open science are very practically seen in the launching of many new journals, such as the *Journal of Open Source Software* (JOSS), the *R Journal*, or *SoftwareX*. These journals are characterized by a renewed approach to traditional publishing, including ease of submission, transparency of reviewing process, and accessibility.

JOSS is an example worth highlighting, as it acts as a template for all of the themes addressed thus far in our paper on openness and collaboration. JOSS fully leverages the features of GitHub to use it as a platform where storage, submission, reviewing, and publishing takes place, reducing their maintenance cost and successfully enabling a diamond open access publishing model, with no cost for the author or the reader. Further, paper and software reviewers are welcomed in the same spirit as collaborators on a piece of software hosted on GitHub. This review and publication cycle is an excellent illustration of how multiple facets of open science symbiotically integrate, from open development to open access publishing. Additionally, of note, traditional and longstanding journals are also embracing openness in their publication through offering open access publication of articles, albeit often at a large cost to the researcher. While the ripples created by the open science wave are significant and notable, finding reasonable, widely-agreed-upon and fair solutions to old and new problems is still—to follow programmers' vernacular—a WIP (work in progress). Nonetheless, the followers of the open movement(s) seem well equipped and eager to take these challenges on. Continual advances in this realm are expected, and positive outcomes can realistically be hoped for.

In conclusion, statistical computing's future seems likely to be linked with broader ideological movements. Naturally, the most prescient is open science, driven by an implicit demand for transparency and democracy that also manifests across other fields—notably politics, economics, and other social science subfields. That being said, we also expect other currents and issues to further shape the development of statistical computing; for instance, that of "slow science", environmentalism, and social justice. It would not be surprising to witness the emergence of formalized trends, such as "slow computing" (influenced by economic ideas of "degrowth" and a focus on individual wellbeing), "green computing" (i.e., defined by sustainability and eco-friendliness), "inclusive computing" (with an emphasis and focus on social justice), and a deepening of "affective computing" and "social computing" (both with emphasis on the impact to and role of the individual in computational endeavors). For example, the latter is increasingly becoming formalized with the advent of the new IEEE open journal, the *Journal of Social Computing*. As a result, as so often occurs, we expect technological innovations to fuse with new and reenergized mindsets to affect the "how" of statistical computing as much as the "what", discussed in the following section.

3. The "What": New Techniques and Approaches

Parallel to the wave of open science, another revolution has taken the world by storm, which is directly related to statistical computing: data science. The field of data science, which has roots in multiple subjects, is now developing into a mature standalone discipline. This can be seen through the establishment of new journals, schools, and degree programs at all levels, from bachelors to doctoral. Further, many research institutes are appearing dedicated to advancing this burgeoning field at times within their discipline (e.g., the Harvard Ophthalmology Clinical Data Science Institute), generically in service of data science as its own field (e.g., the New York University Center for Data Science), or even in the context of new aspects of the field, such as justice and ethics (e.g., the University of Virginia's Center for Data Ethics and Justice).

Data science and open science, then, have exerted substantial influence on statistical computing through the very process of developing computational techniques. That is, to perform data science, statistics must be engaged in virtually all aspects of the process, which includes both development and application of statistics as well as computing for implementation of techniques and tools to serve the project's end. As techniques and tools are developed, they are increasingly developed in an open way to encourage wider engagement with the tools, as well as to encourage wider contributions from the broader "open" community. This can be most clearly seen through collaborative software development, as previously discussed.

With the advancement and development of data science and the open science movement, statistical computing is simultaneously reaping the benefits of these wider communities and advancing at a fast rate and in new, larger-scaled ways. While we briefly mention some of the new areas in the following sections, this list is by no means exhaustive, and many exciting innovations and development paths are taking place in parallel.

3.1. Artificial Intelligence

Leveraging the ever-increasing amount and availability of data produced and recorded through online interactions, artificial intelligence (AI), and more specifically machine learning (ML), are areas where enormous advances have been made. Applications are wonderfully diverse, from task-specific applications (e.g., [9–12]), to larger-scale ecosystems covering every part of a data modeling pipeline, from making sense out of messy data to building predictive models, all within a unified software interface such as H2O [13–15], scikit-learn [16], or tidymodels [17]. Despite the ease of use of these new technologies, the latter underscores a current point of tension in statistical computing, as the field is split around polyvalent, easy-to-use, fast-to-build tools and languages; and low-level languages or dedicated, sometimes model-class-specific, ecosystems used for production or for particular applications.

Given the tension which often accompanies any realm experiencing rapid development and advancement, attempts have been made to unify research, exploration, and accessibility with production, application, and efficiency. A recent and successful example is the development of the Julia language [18], which is framed as solving the so-called "two-language problem": the fact that many scientific programs are prototyped in a slow but flexible language and then reimplemented in faster but less flexible languages for practical applications.

Another interesting aspect of AI-related developments is the direct impact on computing itself. Some ML and AI advances are even reciprocally benefitting programming capabilities in the form of development assistants (e.g., GitHub copilot, automated code review tools, and more recently the impressive and somewhat unexpected coding abilities of ChatGPT all of which carry the promise of increasing productivity and optimizing developers' work.) As a note, we are aware of the uncertainty, drawbacks, and fear at times relating to ChatGPT and similar technologies, especially in an academic setting [19]. However, for present purposes, we are focused instead on the advances of software and statistical computing, all of which carry both benefits and drawbacks. As a practical example, referring back to JOSS, as well as new software review outlets, such as *ROpenSci* [20,21], software review processes are substantially eased by the inclusion of automated bots, which is a sample area where this reciprocal impact is clear.

3.2. Bayesian Estimation

While machine learning is leveraged to realize incredible payoffs when it comes to building predictive models and pipelines, another area of development worth mentioning involves reforming the process of inference and uncertainty quantification: the Bayesian approach. In the modern expression of the Bayesian world, development arises in a terrain of reconsidering the value and approach to null-hypothesis significance testing (NHST). Not only does the Bayesian framework provide alternative methods for extracting meaning and making decisions about data (e.g., by providing alternative indices such as the Bayes Factor), it also changes the way we think about and quantify uncertainty as we estimate parameters while building complex models.

The development of Bayesian methods on the algorithmic side also parallels the growth of Bayesian-inspired models of how the brain works, which is revolutionizing neuroscience (see [22]). This line of research connects biological intelligence with AI, and efforts are thus being made to optimize Bayesian estimation processes (which are typically computationally expensive) to improve or extend AI capabilities. The bidirectional influence applies here too, as AI research, such as into convolutional neural networks (CNN), is helping scientists in many areas of research, from neuroscientists attempting to better understand the brain and test neurocognitive theories (see [23] for a recent example linking deep learning with psychological manifestations such as hallucinations), to political scientists uncovering election fraud (see [24] for a clever application of CNN to reveal systematic voting fraud in the 1988 Mexican presidential election), and social scientists with new applications of methods such as Bayesian kriging for geospatial modeling (see [25] for a computational exploration of Bayesian kriging in the big data era, published in this Special Issue of *Mathematics*).

Central to widespread Bayesian adoption is the (relatively) recent development of APIs to easily create and sample from Bayesian models within domain-general languages. Some prominent advances and examples include: brms [26] and rstanarm [27] for R; pymc3 [28] for Python; or Turing [29] for Julia.

3.3. Results Communication

The aforementioned developments in algorithms, techniques, libraries, and approaches are complemented by concurrent and notable progress in the area of results communication. Central to the development of this aspect of modern and recent advances in statistical computing is clear reporting, wide accessibility, and ease of translating technical concepts into aesthetically pleasing, well-formatted reports with minimal effort. This advancement can be clearly seen when comparing the former industry standard (LaTeX) with the modern industry standard for technical reporting (markdown, regardless of flavor, e.g., GitHub, R, Quarto, etc.). In our opinion, this is the final piece of the puzzle to achieve open, reproducible, and high-quality statistical computing, with wide accessibility and easy consumption of research findings and technical output.

Advancements in technical reporting of this variety come in several forms: data visualization, advanced tables, and machine-generated human-readable technical reporting. First, data visualization is now a major area of focus in statistical computing, data science, and ML. Blurring the boundaries between scientific visualizations and art, the advent of initiatives to promote beautiful and informative graphs (e.g., tidytuesdays on Twitter) and generative art (e.g., artworks of Thomas Lin Pedersen or Danielle Navarro), coupled with recent pushes from major journals to favor visual over tabular rendering of findings when possible, have pushed this formerly niche corner of statistical computing to be widely accepted and pursued, with higher quality now expected. The working implementation of the grammar of graphics in ggplot2 [30] has introduced a new API to plotting libraries and has inspired many counterparts in other languages (e.g., plotnine in Python or Gadfly in Julia). Recent developments, such as D3.js [31], plotly, and shiny [32] have further contributed to advancements of data visualization by introducing interactivity, offering the user the ability to "experiment by themselves" and explore the data as they see fit.

Besides figures, tools for advanced table creation allow the presenting of numerical results in an appealing and accurate way. Software with this scope exists in all the major statistical computing languages, such as gt [33], reactablefmtr [34], and knitr [35,36] in R, and in Python, PrettyTable, PrettyHTMLTable, and even pandas via DataFrame.to_html.

Figures and tables are specific ways of communicating results, but are individual parts of a broader process of creating technical reports papers and scientific papers. Report generation of this sort is facilitated by tools that allow for more transparency and reproducibility by automatizing parts of the standardized text (e.g., values in parentheses that provide details of a statistical test). Of note, the recently developed software has allowed for automating effect size labeling [37], describing statistical models (e.g., [38]), or clarifying the approach used for outliers treatment (see Theriault et al., published in this Special Issue of *Mathematics*). Another tool used for more accurate statistical reporting is "statcheck" [39], a tool developed to allow for checking existing documents for accurate reporting of tests, which is useful for reviewing others' or one's own work.

A final, but important refinement worth mentioning that ties in with several points made throughout aims at making results more readable, aesthetically pleasing, and ultimately understandable, all in a reproducible and easy-to-manage way. This is the fruit of recent document-generation frameworks that are able to combine code (which can be from multiple languages), figures, and text into well-formatted outputs. Recent examples of these software tools include Quarto and RMarkdown [40], which can be combined with Shiny for cloud-based reporting [35], and officer [41]. In Python, similar packages for reporting include pandas [42], Jinja2 [43], and WeasyPrint, to name a few. Further, similar libraries to those mentioned in R and Python include Weave [44] and Pluto [45] in Julia.

4. Concluding Remarks

Statistics was once exclusively seen as a component of mathematics, and its practitioners were required to be trained and familiar with mathematical concepts, formulas, and equations. Statisticians could develop entire theories and frameworks in isolation, and report their ideas in traditional scientific outlets. However, the path of statistics is now in the process of synchronizing with computer science, as good software development and efficient algorithm design become key to statistical advances.

To answer the question of where we are going with statistical computing posed in the title of this paper, we suggest that decentralized collaboration (embracing and pushing forward open science in all its aspects), cross-pollination of experts in multiple fields working in multiple languages, and a focus on users will increasingly characterize this field. Namely, to the latter group, the term "users" is an ever-widening concept, which includes many people with many purposes. For instance, users may include the lay user interested in writing better technical documents, the statistician–scientist interested in publishing reproducible, well-formatted statistical results, or the operational data scientist collaborating with internal stakeholders on developing new ways of computing and sharing findings across their team.

Whether contributing original ideas or enjoying the benefits emerging from the modern network of collaborative and open development, statistical computing is at the center of performing and sharing good science in reproducible ways. Additionally, most thrillingly, there is no end in sight to the development and evolution of statistical computing.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ram, K. Git can facilitate greater reproducibility and increased transparency in science. *Source Code Biol. Med.* **2013**, *8*, 1–8. [CrossRef] [PubMed]
2. National Academies of Sciences, Engineering, and Medicine. *Open Science by Design: Realizing a Vision for 21st Century Research*; National Academies of Sciences, Engineering, and Medicine: Washington, DC, USA, 2018.
3. David, P.A. The Historical Origins of 'Open Science': An essay on patronage, reputation and common agency contracting in the scientific revolution. *Capital. Soc.* **2008**, *3*, 5. [CrossRef]
4. Vicente-Saez, R.; Martinez-Fuentes, C. Open Science now: A systematic literature review for an integrated definition. *J. Bus. Res.* **2018**, *88*, 428–436. [CrossRef]
5. McKiernan, E.C.; Bourne, P.E.; Brown, C.T.; Buck, S.; Kenall, A.; Lin, J.; McDougall, D.; Nosek, B.A.; Ram, K.; Yarkoni, T.; et al. How open science helps researchers succeed. *elife* **2016**, *5*, e16800. [CrossRef] [PubMed]
6. Bahlai, C.; Bartlett, L.J.; Burgio, K.R.; Fournier, A.M.; Keiser, C.N.; Poisot, T.; Whitney, K.S. Open science isn't always open to all scientists. *Am. Sci.* **2019**, *107*, 78–82. [CrossRef]
7. Mirowski, P. The future (s) of open science. *Soc. Stud. Sci.* **2018**, *48*, 171–203. [CrossRef]
8. Foster, E.D.; Deardorff, A. Open science framework (OSF). *J. Med. Libr. Assoc. JMLA* **2017**, *105*, 203. [CrossRef]
9. Chen, T.; He, T.; Benesty, M.; Khotilovich, V.; Tang, Y.; Cho, H.; Chen, K.; Mitchell, R.; Cano, I.; Zhou, T. Xgboost: Extreme gradient boosting. *R Package Version 0.4-2* **2015**, *1*, 1–4.
10. Waggoner, P.D. A batch process for high dimensional imputation. *Comput. Stat.* **2023**, 1–22. [CrossRef]
11. Waggoner, P.D. *Modern Dimension Reduction*; Cambridge University Press: Cambridge, UK, 2021.
12. Wright, M.N.; Ziegler, A. Ranger: A fast implementation of random forests for high dimensional data in C++ and R. *arXiv* **2015**, arXiv:1508.04409. [CrossRef]
13. H_2Oai H_2O: RInterface for, H.2.O. R Package Version 3.38.0.2. Available online: https://github.com/h2oai/h2o-3 (accessed on 1 March 2023).
14. H_2Oai H_2O: Python Interface for H.2.O. Python Package Version 3.38.0.2. Available online: https://github.com/h2oai/h2o-3 (accessed on 1 March 2023).
15. H_2O.ai. H_2O: Scalable Machine Learning Platform. Version 3.38.0.2. Available online: https://github.com/h2oai/h2o-3 (accessed on 1 March 2023).

16. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Duchesnay, E.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
17. Kuhn, M.; Wickham, H. *Tidymodels: A Collection of Packages for Modeling and Machine Learning Using Tidyverse Principles*; Tidymodels: Boston, MA, USA, 2020.
18. Bezanson, J.; Edelman, A.; Karpinski, S.; Shah, V.B. Julia: A fresh approach to numerical computing. *SIAM Rev.* **2017**, *59*, 65–98. [CrossRef]
19. Thorp, H.H. ChatGPT is fun, but not an author. *Science* **2023**, *379*, 313. [CrossRef]
20. Boettiger, C.; Chamberlain, S.; Hart, E.; Ram, K. Building software, building community: Lessons from the rOpenSci project. *J. Open Res. Softw.* **2015**, *3*, e8. [CrossRef]
21. Ram, K. rOpenSci-open tools for open science. In *AGU Fall Meeting Abstracts*; Harvard University: Cambridge, MA, USA, 2013; Volume 2013, p. ED43E-04.
22. Friston, K.; FitzGerald, T.; Rigoli, F.; Schwartenbeck, P.; Pezzulo, G. Active inference and learning. *Neurosci. Biobehav. Rev.* **2016**, *68*, 862–879. [CrossRef]
23. Suzuki, K.; Seth, A.K.; Schwartzman, D.J. Modelling Phenomenological Differences in Aetiologically Distinct Visual Hallucinations Using Deep Neural Networks. *bioRxiv* **2023**.
24. Cantú, F. The fingerprints of fraud: Evidence from Mexico's 1988 presidential election. *Am. Political Sci. Rev.* **2019**, *113*, 710–726. [CrossRef]
25. Byers, J.S.; Gill, J. Applied Geospatial Bayesian Modeling in the Big Data Era: Challenges and Solutions. *Mathematics* **2022**, *10*, 4116. [CrossRef]
26. Bürkner, P.C. Brms: An R package for Bayesian multilevel models using Stan. *J. Stat. Softw.* **2017**, *80*, 1–28. [CrossRef]
27. Goodrich, B.; Gabry, J.; Ali, I.; Brilleman, S. Rstanarm: Bayesian Applied Regression Modeling via STAN, Version 2. 2020. Available online: https://mc-stan.org/rstanarm/ (accessed on 1 March 2023).
28. Salvatier, J.; Wiecki, T.V.; Fonnesbeck, C. Probabilistic programming in Python using PyMC3. *PeerJ Comput. Sci.* **2016**, *2*, e55. [CrossRef]
29. Ge, H.; Xu, K.; Ghahramani, Z. Turing: A language for flexible probabilistic inference. In Proceedings of the International Conference on Artificial Intelligence and Statistics, Lanzarote, Spain, 9–11 April 2018; PMLR: Lanzarote, Spain, 2018; pp. 1682–1690.
30. Wickham, H. Ggplot2. Computational Statistics. *Wiley Interdiscip. Rev.* **2011**, *3*, 180–185. [CrossRef]
31. Bostock, M. D3. Js-Data-Driven Documents. Available online: http://d3js.org (accessed on 1 March 2023).
32. Sievert, C. *Interactive Web-Based Data Visualization with R, Plotly, and Shiny*; CRC Press: Boca Raton, FL, USA, 2020.
33. Iannone, R.; Cheng, J.; Schloerke, B.; Hughes, E.; Seo, J. GT: Easily Create Presentation-Ready Display Tables. 2022. Available online: https://gt.rstudio.com/ (accessed on 1 March 2023).
34. Cuilla, K. Reactablefmtr: Streamlined Table Styling and Formatting for Reactable. 2022. Available online: https://kcuilla.github.io/reactablefmtr/ (accessed on 1 March 2023).
35. Xie, Y.; Allaire, J.J.; Grolemund, G. *R Markdown: The Definitive Guide*; Chapman and Hall/CRC: Boca Raton, FL, USA, 2018.
36. Xie, Y. Knitr: A comprehensive tool for reproducible research in R. In *Implementing Reproducible Research*; Chapman and Hall/CRC: Boca Raton, FL, USA, 2018; pp. 3–31.
37. Ben-Shachar, M.S.; Lüdecke, D.; Makowski, D. Effectsize: Estimation of effect size indices and standardized parameters. *J. Open Source Softw.* **2020**, *5*, 2815. [CrossRef]
38. Makowski, D.; Ben-Shachar, M.S.; Patil, I.; Lüdecke, D. Automated Results Reporting as a Practical Tool to Improve Reproducibility Methodological Best Practices Adoption, C.R.A.N. Available online: https://github.com/easystats/report (accessed on 1 March 2023).
39. Nuijten, M.B.; Polanin, J.R. "Statcheck": Automatically detect statistical reporting inconsistencies to increase reproducibility of meta-analyses. *Res. Synth. Methods* **2020**, *11*, 574–579. [CrossRef] [PubMed]
40. Allaire, J.; Xie, Y.; McPherson, J.; Luraschi, J.; Ushey, K.; Atkins, A.; Wickham, H.; Cheng, J.; Chang, W.; Iannone, R. Rmarkdown: Dynamic Documents for R, Version 1. 2018. Available online: https://cran.r-project.org/web/packages/rmarkdown/index.html (accessed on 1 March 2023).
41. Gohel, D. Officer: Manipulation of Microsoft Word and PowerPoint Documents. 2018. Available online: https://davidgohel.github.io/officer/ (accessed on 1 March 2023).
42. McKinney, W. Pandas: A foundational Python library for data analysis and statistics. *Python High Perform. Sci. Comput.* **2011**, *14*, 1–9.
43. Ronacher, A. Jinja2 Documentation. Welcome to Jinja2—Jinja2 Documentation (2.8-dev). 2008. Available online: https://www.devdoc.net/python/jinja-2.10.1-doc/ (accessed on 1 March 2023).

44. Matti, P. Weave.jl: Scientific Reports Using Julia. *J. Open Source Softw.* **2017**, *2*, 204. Available online: http://dx.doi.org/10.21105/joss.00204 (accessed on 1 March 2023).
45. van der Plas, F.; Dral, M.; Berg, P.; Huijzer, R.; Bochenski, N.; Mengali, A.; Lungwitz, B.; Burns, C.; Priyashan, H.; Ling, J.; et al. Fonsp/Pluto.jl v0.19.22, Version 0.19.22; Zenodo. 2023. Available online: https://zenodo.org/record/7576119#.ZDJtznYzZPY (accessed on 1 March 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.