Reproducibility and plant pathology: where do we stand and a way forward

Adam H. Sparks1,2,\*, Emerson Del Ponte3, Kaique dos Santos Alves3, Zachary Foster4, and Niklaus J. Grünwald5

1 Department of Primary Industries and Regional Development, Farming Systems Innovation, Perth WA 6000, Australia  
2 University of Southern Queensland, Centre for Crop Health, Toowoomba Qld 4350, Australia  
3 Departmento de Fitopatologia, Universidade Federal de Viçosa, Brazil  
4 Department of Botany and Plant Pathology, Oregon State University, Corvallis OR 97331, USA  
5 Horticultural Crops Research Unit, USDA Agricultural Research Service, Corvallis OR 97331, USA

\* Corresponding author: [Adam H. Sparks <[Adam.Sparks@dpird.wa.gov.au](mailto:Adam.Sparks@dpird.wa.gov.au)>](mailto:Adam.Sparks@dpird.wa.gov.au)

**Keywords: open science, open data, reproducible research, FAIR data**

# Abstract

Open research practices have been highlighted extensively during the last ten years in many fields of study as essential standards needed to promote transparency and reproducibility of scientific results. Scientific claims can only be evaluated based on how protocols, materials, equipment and methods were described; data were collected and prepared, and; analyses conducted. Openly sharing protocols, data and computational code is central for current scholarly dissemination and communication, but in many fields, including plant pathology, adoption of these practices has been slow. We randomly selected 300 articles published from 2012 to 2018 across 21 journals representative of the pathology discipline and assigned them scores reflecting their openness and reproducibility. We found that most of the articles were not very open, failing to share data or code in a meaningful way. We propose that using open source tools for producing open and reproducible work and analysis is advantageous, benefiting not just readers, but the authors as well and provide ideas and tools to promote open, reproducible research practices among plant pathologists.

# Introduction

Modern plant pathological research has many facets given the array of disciplines and sub-disciplines currently involved. Collectively, they contribute to increasing our basic and applied knowledge of several aspects of pathogen biology and disease development to ultimately improve plant disease management. Scientific research in the field varies from the purely observational or descriptive nature to inferential, based on experimental or simulation-derived data sets. Whatever the case, research findings are verifiable based on how much of the research materials, processes and outcomes are made available beyond what is reported in the scientific article and the ability of others to make use of the methods and results. These research findings include biological materials (host and pathogen genotypes), nucleic/protein sequences, experimental and simulated raw data annotations, drawings and photographs and statistical analysis code among other materials and data generated during the course of the research.

Recently, open science has been highlighted with many donors expecting data to be available (Government of Canada 2016; Noorden 2017; Australian Research Council 2018), journals in the field promoting the sharing of data (Del Ponte 2020), and other scientists interested in sharing and collaborating more widely (Wald 2010).

Reproducibility is one component under the umbrella of open science. By proactively practising open science, the work increases the chance to become more reproducible through the availability of data and code. That is, open science leads to reproducibility and replicability.

For us to easily discuss the topic, we first must define what we mean so that we may clearly communicate. Many of the terms used in this area have varying definitions that may or may not agree with each other. For instance, reproducible research is recently highlighted by many authors (Preeyanon et al. 2018; Wallach et al. 2018; Baker 2016; Iqbal et al. 2016; Editors at Nature 2016; Patil et al. 2016; Weissgerber et al. 2016; Brunsdon 2015; Sweedler 2015; Fitzjohn et al. 2014; Ioannidis 2014; Fidler and Gordon 2013; Stodden et al. 2013) as an important issue.

In biological sciences it is not always possible to use identical test material or perhaps the the time or resources are not available for full reproduction, *e.g.*, field trials that span years and locations or complex glasshouse experiments. Therefore, we will follow Peng’s (2009) definition that provides clear guidelines a minimum standard of “reproducible research”:

“The replication of scientific findings using independent investigators, methods, data, equipment, and protocols has long been, and will continue to be, the standard by which scientific claims are evaluated. However, in many fields of study there are examples of scientific investigations that cannot be fully replicated because of a lack of time or resources. In such a situation, there is a need for a minimum standard that can fill the void between full replication and nothing. One candidate for this minimum standard is ‘reproducible research,’ which requires that data sets and computer code be made available to others for verifying published results and conducting alternative analyses.” Peng, R. (2009). *Reproducible research and Biostatistics*. Biostatistics, 10 (3): 405-408.

Therefore our definition of reproducibility will be that the computer code and data are made freely available to others for verification and conducting alternate analyses or for use in instructional purposes. And that the software used are also easily obtained and preferably open source to avoid licensing or other issues related to accessibility for end-users related to costs or non-standard file formats, etc.

## Where do we stand in regards to reproducibility?

In an effort to understand where as a discipline we stand as plant pathologists in regards to open science and reproducible research, we surveyed a broad selection of articles to represent a broad swathe of publications to evaluate our status. We randomly selected 300 articles published from 2012 to 2018 across 21 journals (Table 1) and assigned them scores reflecting their openness and reproducibility. The journals represented both fundamental and applied areas as well as pathogen or other group-specific publications (See appendix for details on the methodology).

Forty-eight articles scored 0% out of a possible 100% for reproducibility. The mean value was 31.2% and the median value was 33.3%. One article scored 100% or was classed as fully reproducible. No journals surveyed were completely closed, all offered at least an option for open access but some were completely open.The majority of the articles were classed as fundamental, 160, with the remaining 138 classed as applied.

Most articles did not make any computational methods available in any fashion with two, 0.007%, classing as ‘3,’ which was the highest score available (Figure 2i). Two hundred eighty-seven articles, 96%, were classed as ‘0’ and nine articles, 0.03%, appeared to not use any computational methods.

More articles did a good job of using software that was reasonably available to anyone with 206 scoring ‘2’ or ‘3’ (Figure 2ii).

Software citations were also reasonably well reported in the literature with 120 scoring ‘2’ where the main software was cited with version numbers and 57 scoring ‘3’ where full citations including R packages or the SAS PROC used were cited where appropriate (Figure 2iii).

However, data availability was mostly not available with 243 or 81.5% scoring 0’ where the data was not available or mentioned in the article. However, more articles, 29 or 9.7%, scored “3” than “1” or “2” combined, with two articles not producing shareable data (Figure 2iv).

The most frequently cited software was SAS. The remainder were mostly specialised software except for Microsoft Excel, tied for 9th (Table 2).

Unfortunately using our scoring method, most articles scored poorly for reproducibility or openness. Most journals in our discipline fail to mention the availability of the computational methods to support the article as published.

The software availability score was better as was the citing of software and versions used. SAS remains firmly entrenched in the discipline as the choice of software for statistical analysis followed by SPSS. It is interesting to note however, that with the popularity of Python that it does not even appear in our survey results whereas R was the fourth most commonly used software.

## Are there good examples in the field?

While the articles that were surveyed did not score well, we are aware of and would like to highlight that there are several good examples from the plant pathology discipline that can be used as exemplars.

There are different ways to make research more open and reproducible, which can be thought of as levels of reproducibility. The first level is including tables of raw data or code with the paper as a supplemental file. This is suitable for studies that may have a small data set or simple analysis or for demonstrations purposes as Madden *et al.* (2015) demonstrate in their discussion regarding the use of values in statistical analyses where they supply an *e-****X****tra*\* with reproducible examples for readers to refer to. This allows the reader to quickly view the extra materials and a DOI is assigned as a part of the article itself.

At the second level, Lehner *et al.* (2017) used GitHub to host a code repository of their research compendium website with data and a reproducible report that explains in detail all steps of the analysis and the R code for conducting a meta-analysis for assessing heterogeneity in the relationship between white mold incidence and soybean yield and between incidence and soybean tied (<https://emdelponte.github.io/paper-white-mold-meta-analysis/>). The website clearly demonstrates the analysis for readers and uses R so that anyone can easily replicate the work. Likewise, Duku *et al.* (2016) provide scripts for models, data and code for graphs, via a website hosted by GitHub, <http://adamhsparks.github.io/MICCORDEA/>, that can be used to replicate their study modelling the effects of climate change on rice bacterial blight and rice leaf blast in Tanzania.

The third level is the use of proper data repositories such as FigShare, Zenodo or OSF.io among others, allow for the deposition and updating of code, figures, data preprints or any other materials that support the article itself while providing a DOI and citation for a standalone citation. Sparks *et al.* (2011, 2014) used FigShare to provide models, data and code, (Sparks 2016) necessary to replicate model development and the subsequent study on the effects of climate change on potato late blight.

The fourth level includes packaging the analysis as a software package, *e.g.*, R, Julia or Python package at a minimum, with the use of continuous integration and possibly a software container like Docker as Khaliq *et al.* (Khaliq et al. 2020) have done. Their research compendium provides a Docker instance, DOI and full R package that enable readers to fully replicate their analysis of *Ascochyta rabiei* conidia dispersal in chickpea using the data collected and stepping through other points where weather data were investigated and various models were fit before deciding on the best fit and recreate any figures as published in the article (Khaliq et al. n.d.). When this level is employed, tools such as Binder, <https://mybinder.org>, can be used that allow readers and reviewers to launch an interactive session in their web browser and interact with the data and rerun the analysis in an RStudio instance or Jupyter notebooks as Mioroni *et al.* used (2019). Kamvar *et al.* (2015) took a slightly different approach by including all files necessary for the analysis and most output files in a repository (Kamvar et al. 2014) that also included an installable R package that was used for the original analysis. While there are many other methods, these two approaches illustrate some of the best-practices where the data and other files were deposited in repositories with DOIs and reproducibility issues were addressed by using R packages to handle dependencies and other versioning issues making the work more portable.

## Best practices to enhance reproducible research

A general work flow for producing academic research involves clearly defining a research question, obtaining data for testing the hypothesis, summarizing/analyzing and presenting data and results, and writing the manuscript. Here we defined three levels of reproducibility which are also related to the evolution of computational methods and reproducible practices (Figure 1).

The first level of reproducibility involves openly available research materials such as isolates or strains and/or nucleic acid sequences in public collections and citations for methods used. A second level involves providing raw data and code as binary files (PDF or other non-text files) in supplemental materials, which do not allow prompt access to the data and running the code because of use of expensive commercial software or a paywall. The highest level includes efforts to annotate structured raw data (FAIR – Findable, Accessible, Interoperable and Re-usable) (Wilkinson et al. 2016) and fully document the analysis using open source code which are deposited in public repositories and can be run by anyone following the download of data and code. The first level, as reported, is an essential step that is not substituted by the other practices and eventually researchers fail to provide sufficient description or correct citations. In the next section we present standards and tools that can be used to ensure reproducibility.

When making your science more open and reproducible, methods, software used, this includes items such as R, Julia or Python packages that were directly used in the analysis or production of the paper, etc. should be cited properly. Deposit and annotate biological materials with an herbarium or other repository. Provide a full description of all equipment used, *e.g.*, a Spectrum Technologies Watchdog 2700 weather station was used to record wind speed, direction, rainfall, temperature and relative humidity at one-hour intervals. This allows end-users to identify what was used and identify the methods used more accurately. Just as importantly, this acknowledges the contributions of others whose works were instrumental in your research. This also helps ensure that you as a researcher can reconstruct what you have done since you will have good notes and documentation and be able to identify if something changes, *e.g.*, a package version, what effect it had on your research.

The use of programming or scripting languages such as R, Julia, SAS or Python enables you as a scientist to keep very detailed records of what was computationally performed. This is as opposed to using software such as spreadsheet programs like Excel, Google Sheets, Numbers, Calc or others which can be used for simple statistical analyses and visualization (but should not be) or other point-and-click software packages that do not enable you to keep an accurate record of the steps taken to import, format, visualise and analyse data.

Prefer text files for saving small sets of data. Data that are saved in binary formats such as PDF files are difficult to reuse because they are not easily machine readable. In many cases, data sets are small enough and curated in spreadsheets, which should be saved as a plain text file, *e.g.*, comma separated (CSV) files. This also helps ensure that the data are reusable. Larger data sets may warrant the use of a proper database like MariaDB or PostgreSQL which provide users with several benefits but two important ones to mention here are data redundancy, no duplicated records; and data consistency, all of your records are sure to be recorded in the same way for every observation. While databases may offer many advantages, the trade-off is that they are more complex to set up and administer, especially for a small data set that might be only a few rows and columns in a database.

Ideally, once the data are complete, best practices for keeping your data as you perform your work include treating the raw data as read-only and using file permissions to prevent changes to the raw data files. It should be noted, that the use of a database management system also allows for both of these at the expense of added complexity. Saving files in proprietary formats such as .xls(x) can also lead to issues in the future when opening using newer (or older) software versions. Unexpected changes to values in the data (Ziemann et al. 2016) may also occur when using proprietary formats.

If steps are followed to make the data FAIR then it will be readable by humans and machines alike this will help support discoveries and support further research. In turn, sharing data will lead to new citations for your work as others discover and use it. To make your data the most widely discoverable and usable, ensure that it has a persistent identifier. A digital object identifier (DOI) is the most common (<https://www.doi.org/>) but the Handle.Net Registry (HNR) (<https://handle.net/>) is also an option. There are different options for generating a DOI for your data and other materials. FigShare, Zenodo and OSF all offer long-term archival along with a service to generate a DOI for your materials. The use of a persistent identifier works to ensure that even if the data are moved, they can still be located using that unique identifier. For more on FAIR data, visit Go-Fair <https://www.go-fair.org/fair-principles/>.

Once you have determined how to best manage your source code for analysis and the data sets the next step is to consider how to share your data. Providers such as FigShare, Dataverse, OSF.io (the Open Science Framework) and Zenodo allow for you to deposit your data and generate a DOI for sharing your project once you are finished with it.

Other providers exist that allow for you to not only track changes but also to share the data openly, these include GitHub, Gitlab and Bitbucket. GitHub is arguably the most popular and widely used software development platform currently. Data that is encoded in CSV or other plain text formats can easily be deposited in a repository along with code for analysis to enable changes to be tracked and other users to download and replicate the work.

We would advise against the practice of depositing data on a laboratory website or a site such as GitHub only are not an optimal way to preserve and share your work over the long term. Doing either of these leaves the work in an unstable state where future users may be unable to access the work as they are fraught with link-rot and other issues. It is a best practice to always ensure that you have deposited the data with a provider such as Zenodo, FigShare or OSF and generated a DOI for the materials to help ensure continued accessibility. Many of these providers provide rather easy ways to link the project with a software development repository to help ensure that the data are available in perpetuity. If readers are uncertain, we suggest to also consult with local librarians about possible resources. Most universities some other research-focused workplaces provide a facility for staff to deposit papers and other academic materials, but this may extend to software development repositories as well.

In this work we have evaluated the state of reproducibility in the plant pathology literature and presented suggestions for areas of improvement. One of the most important messages that we would like to share is that there is more to the work than just the paper. Sharing materials detailing the analysis that was performed and documenting the data provide citable products and enhance the manuscript providing the reader with a richer set of information with which to understand the work that was performed and often leads to greater impact than for works that do not share data and or code. With mandates for openly sharing data becoming more common with funding agencies, we can and should embrace this and move the discipline forward and have a greater impact with our work.

# Appendix

## Article selection

We randomly selected 300 articles published from 2012 to 2018 across 21 journals (Table 1) and assigned them scores reflecting their openness and reproducibility. The journals represented both fundamental and applied areas as well as pathogen or other group-specific publications. A list of randomly selected pages was assigned to a randomized list of the 21 journals where the page number fell within an article for the given journal. In cases where an article was not suitable, *e.g.*, a review or otherwise not related to plant pathology, the next article was selected until a suitable article was found. Notes regarding the selection of articles can be found in the file, “Reproducibility\_in\_plant\_pathology\_notes.ods,” available in this paper’s repository in `/inst/extdata’. The pages list was numbered from page one and went to page 150. This was done since some journals restart their numbering with each issue and also ensures that the journal is more likely to have a page number corresponding to the randomly generated value. This also assumes that there is no effect or bias on reproducibility based on the time of year that an article was published since most journals start with page number one at the beginning of the year.

## Scoring criteria

Four individuals were assigned to rate a randomised list of journal articles using scoring criteria devised for the purposes of this research (Table 3).

Each journal was classified as to whether they were completely open (TRUE), behind a paywall (FALSE) or a combination (BOTH). The five-year impact factor for 2018 for each journal was retrieved from InCites Journal Citation Reports, Clarivate Analytics and entered in a separate sheet in the Google sheets file. This was downloaded, saved as an Open Document Spreadsheet (ODS) file and is left-joined with the article notes using a left\_join() from the ‘dplyr’ package (Wickham et al. 2020) for analysis when the import\_notes() command is executed.

Articles were classified as fundamental or applied and a note was made whether the article’s primary focus was using or developing molecular techniques (TRUE/FALSE). This classification was based on the evaluator’s judgement of the article.

Where possible, the software used in the preparation of the publication was recorded in the notes if it was cited or otherwise specified in the article text.

Values for software packages were checked for spelling consistency and corrections were made manually where necessary. When working with these data in R, all strings of software character values were converted to fully upper-case to standardise the capitalisation and alleviate any issues with capitalisation used between evaluators. A custom R function, import\_notes(), was written to import the data, format the columns properly and calculate the overall reproducibility score in R (R Core Team 2021).

A total reproducibility score was calculated as the per cent of total possible for a given paper. However, some papers did not use specialized software therefore they were not scored for this category.

We recognise that the scoring methods were developed by the authors specifically for this exercise and are not entirely subjective. We have based our scoring decisions on our own experiences in the field of plant pathology. While many software packages may be widely used or “freely” available if one has an institutional license, these types of software often hinder partners and interested parties that do not have these luxuries. Thus, the emphasis was given to using open source alternatives where possible over commercially available, non-free software packages. The view of the authors is one that this is something to be encouraged as it allows the broadest possible audience to benefit from the work at the lowest possible cost.

## Statistical analysis

The statistical analysis was performed using R version 4.1.0 (R Core Team 2021) on an Apple MacBook Air (M1, 2020). Bayesian mixed-effect models were fit using the contributed package ‘brms’ (Bürkner 2018) to evaluate the fixed effects of the publication (journal title), article class, molecular focus, five-year journal impact factor, journal open access policy on the reproducibility index and year to test for changes over time in the reproducibility index.

Relatively flat priors were selected with a mean of 0 and a standard deviation of 10 for the fixed effects.

The random effects were year and assignee for all models excepting the model which tested for year effects. In this model, the random effects were the publication title and assignee.

Models were fit using the hurdle\_gamma() family function due to zero-inflated continuous data and the assumption that a zero value would only come from one process, the scoring. Each model was run for 3000 iterations in four chains.

The adapt\_delta() and max\_treedepth() values were adjusted as necessary on a case-by-case basis for each of the models to ensure that the chains mixed well.

Model finesses were evaluated using model summaries and diagnostic plots from ‘brms’ (Bürkner 2018) and posterior fits using pp\_check() from the contributed R package ‘bayesplot’ (Gabry et al. 2019).

Probability of direction or maximum probability of effect was plotted, pd(), and a test for practical equivalence, equivalence\_test(), was performed for each model using the contributed R package ‘bayestestR’ (Makowski et al. 2019).

# Authors’ contributions

A.H.S, E.DP., Z.F. and N.J.G. conceived of the presented idea and scoring system. N.J.G. provided five-year impact factors. A.H.S., E.DP., Z.F. and K.dSA. evaluated articles for scoring. A.H.S. designed the computational framework for analysis and created the research compendium with feedback from E.DP. and Z.F. A.H.S. wrote the manuscript in consultation with E.DP.

# Data availability statement

The raw data for this work are documented and available from DOI [https://doi.org/10.5281/zenodo.4941722](10.5281/zenodo.4941722).

# Code availability statement

All code used in the analyses and data visualization and associated materials have been made available as research compendium available from DOI [https://doi.org/10.5281/zenodo.1250664](10.5281/zenodo.1250664). A webpage version of the compendium is available from <https://openplantpathology.github.io/Reproducibility_in_Plant_Pathology/>.

# Acknowledgements

# References

Australian Research Council. 2018. ARC Open Access Policy. Available at: <https://www.arc.gov.au/policies-strategies/policy/arc-open-access-policy> [Accessed April 20, 2020].

Baker, M. 2016. Muddled meanings hamper efforts to fix reproducibility crisis. Nature News.

Brunsdon, C. 2015. Quantitative methods I: Reproducible research and quantitative geography. Prog. Hum. Geogr.

Bürkner, P.-C. 2018. Advanced Bayesian multilevel modeling with the R package brms. The R Journal. 10:395–411.

Del Ponte, E. M. 2020. Towards a more open and transparent plant pathology research. Tropical Plant Pathology. 45:361–362 Available at: <https://doi.org/10.1007/s40858-020-00388-z>.

Duku, C., Sparks, A. H., and Zwart, S. J. 2016. Spatial modelling of rice yield losses in Tanzania due to bacterial leaf blight and leaf blast in a changing climate. Climatic Change. 135:569–583.

Editors at Nature. 2016. Reality check on reproducibility. Nature. 533:437.

Fidler, F., and Gordon, A. 2013. Science is in a reproducibility crisis: How do we resolve it? Phys.org. Available at: <http://phys.org/news/2013-09-science-crisis.html?utm{\_}content=bufferde383{\{}{\&}{\}}utm{\_}source=buffer{\{}{\&}{\}}utm{\_}medium=twitter{\{}{\&}{\}}utm{\_}campaign=Buffer{\{}{\#}{\}}ajTabs>.

Fitzjohn, R., Pennell, M., Zanne, A., and Cornwell, W. 2014. Reproducible research is still a challenge. ROpensci.org. 2014 Available at: <http://ropensci.org/blog/2014/06/09/reproducibility/>.

Gabry, J., Simpson, D., Vehtari, A., Betancourt, M., and Gelman, A. 2019. Visualization in Bayesian workflow. J. R. Stat. Soc. A. 182:389–402.

Government of Canada. 2016. Tri-Agency Open Access Policy on Publications - Science.gc.ca. Tri-Agency Open Access Policy on Publications. Available at: <http://www.science.gc.ca/eic/site/063.nsf/eng/h_F6765465.html?OpenDocument> [Accessed April 12, 2019].

Ioannidis, J. P. A. 2014. How to make more published research true. PLoS Medicine. 11.

Iqbal, S. A., Wallach, J. D., Khoury, M. J., Schully, S. D., and Ioannidis, J. P. A. 2016. Reproducible research practices and transparency across the biomedical literature. PLoS Biology. 14:1–13.

Kamvar, Z. N., Larsen, M. M., Kanaskie, A. M., Hansen, E. M., and Grünwald, N. J. 2015. Spatial and temporal analysis of populations of the sudden oak death pathogen in Oregon forests. Phytopathology. 105:982–989.

Kamvar, Z. N., Larsen, M. M., Kanaskie, A. M., Hansen, E. M., and Grünwald, N. J. 2014. Sudden Oak Death in Oregon Forests: Spatial and temporal population dynamics of the sudden oak death epidemic in Oregon Forests.

Khaliq, I., Fanning, J., Melloy, P., Galloway, J., Moore, K., Burrell, D., et al. 2020. The role of conidia in the dispersal of *Ascochyta rabiei*. European Journal of Plant Pathology. 158:911–924.

Khaliq, I., Melloy, P., Burrell, D., Fanning, J., Galloway, J., Moore, K., et al. *ChickpeaAscoDispersal: A research compendium to accompany ‘the role of conidia in the dispersal of Ascochyta rabiei’*. Available at: <https://github.com/adamhsparks/ChickpeaAscoDispersal>.

Lehner, M. S., Pethybridge, S. J., Meyer, M. C., and Del Ponte, E. M. 2017. Meta-analytic modelling of the incidence–yield and incidence–sclerotial production relationships in soybean white mould epidemics. Plant Pathology. 66:460–468.

Madden, L. V., Shah, D. A., and Esker, P. D. 2015. Does the *p* value have a future in plant pathology? Phytopathology. 105:1400–1407.

Makowski, D., Ben-Shachar, M. S., and Lüdecke, D. 2019. bayestestR: Describing effects and their uncertainty, existence and significance within the Bayesian framework. Journal of Open Source Software. 4:1541.

Miorini, T. J. J., Kamvar, Z. N., Higgins, R. S., Raetano, C. G., Steadman, J. R., and Everhart, S. E. 2019. Differential aggressiveness of *Sclerotinia sclerotiorum* isolates from North and South America and partial host resistance in Brazilian soybean and dry bean cultivars. Tropical Plant Pathology. 44:73–81.

Miorini, T. J. J., Kamvar, Z. N., Higgins, R., Raetano, C. G., Steadman, J., and Everhart, S. E. 2018. Data and analysis for Variation in pathogen aggression and cultivar performance against \_sclerotinia sclerotiorum\_ in soybean and dry bean from the U.S. And Brazil. Available at: <https://osf.io/2x7fc/> [Accessed July 29, 2020].

Noorden, R. van. 2017. Gates Foundation demands open access. Nature. 541:270–270.

Patil, P., Peng, R. D., and Leek, J. 2016. A statistical definition for reproducibility and replicability. bioRxiv. Available at: <http://biorxiv.org/content/early/2016/07/29/066803>.

Peng, R. D. 2009. Reproducible research and biostatistics. Biostatistics. 10:405–408.

Preeyanon, L., Pyrkosz, A. B., and Titus Brown, C. 2018. Reproducible bioinformatics research for biologists. In Chapman; Hall/CRC, p. 185–217.

R Core Team. 2021. *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing. Available at: <https://www.R-project.org/>.

Sparks, A. 2016. Global-late-blight (meta)modelling. Available at: <https://figshare.com/articles/dataset/Global_Late_Blight_Modelling/1066124/6>.

Sparks, A. H., Forbes, G. A., Hijmans, R. J., and Garrett, K. A. 2011. A metamodeling framework for extending the application domain of process-based ecological models. Ecosphere. 2:art90.

Sparks, A. H., Forbes, G. A., Hijmans, R. J., and Garrett, K. A. 2014. Climate change may have limited effect on global risk of potato late blight. Global Change Biology. :3621–3631.

Stodden, V., Guo, P., and Ma, Z. 2013. Toward reproducible computational research: An empirical analysis of data and code policy adoption by journals. PLoS One. 8:2–9.

Sweedler, J. V. 2015. Striving for reproducible science. Analytical Chemistry. 87:11603–11604.

Wald, C. 2010. Scientists embrace openness.

Wallach, J. D., Boyack, K. W., and Ioannidis, J. P. A. 2018. Reproducible research practices, transparency, and open access data in the biomedical literature, 20152017 ed. Ulrich Dirnagl. PLOS Biology. 16:e2006930.

Weissgerber, T. L., Garovic, V. D., Winham, S. J., Milic, N. M., and Prager, E. M. 2016. Transparent reporting for reproducible science. Journal of Neuroscience Research. 00:1–6.

Wickham, H., François, R., Henry, L., and Müller, K. 2020. *Dplyr: A grammar of data manipulation*. Available at: <https://CRAN.R-project.org/package=dplyr>.

Wilkinson, M. D., Dumontier, M., Aalbersberg, Ij. J., Appleton, G., Axton, M., Baak, A., et al. 2016. The FAIR Guiding Principles for scientific data management and stewardship. Scientific Data. 3:1–9 Available at: <https://www.nature.com/articles/sdata201618> [Accessed April 20, 2020].

Ziemann, M., Eren, Y., and El-Osta, A. 2016. Gene name errors are widespread in the scientific literature. Genome Biology. 17:177.

# Tables

Table 1: Journal titles selected for inclusion and the number (n) of articles from each journal that were evaluated.

| Journal | n |
| --- | --- |
| Australasian Plant Pathology | 8 |
| Canadian Journal of Plant Pathology | 13 |
| Crop Protection | 14 |
| European Journal of Plant Pathology | 11 |
| Forest Pathology | 14 |
| Journal of General Plant Pathology | 17 |
| Journal of Phytopathology | 14 |
| Journal of Plant Pathology | 15 |
| Molecular Plant Pathology | 18 |
| Molecular Plant-Microbe Interactions | 16 |
| Nematology | 11 |
| Physiological and Molecular Plant Pathology | 18 |
| Phytoparasitica | 19 |
| Phytopathologia Mediterranea | 9 |
| Phytopathology | 19 |
| Plant Disease | 15 |
| Plant Health Progress | 10 |
| Plant Pathology | 24 |
| Revista Mexicana de Fitopatología | 10 |
| Tropical Plant Pathology | 9 |
| Virology Journal (Plant Viruses Section) | 16 |

Table 2: The top ten sofware programs that were found to be used in the papers that were surveyed.

| Software Used | n |
| --- | --- |
| SAS | 53 |
| MEGA | 42 |
| SPSS | 26 |
| R | 19 |
| BLAST | 16 |
| STATISTICA | 11 |
| GENSTAT | 10 |
| CLUSTAL W | 9 |
| CLUSTAL X | 7 |
| EXCEL | 7 |
| MRBAYES | 7 |
| BIOEDIT | 5 |
| PAUP | 5 |

Table 3: Articles were scored in four areas for reproducibility on a scale of '0' to '3' by four independent assessors to rank articles in each of four areas, 'Computational methods availability', 'Data availability', 'Software availability' and 'Software citation'.

| Category | Index Value | Definition |
| --- | --- | --- |
| Computational methods availability | 0 | Not available or not mentioned in the publication |
|  | 1 | Used expensive proprietary software that only institutions would typically purchase |
|  | 2 | Proprietary software that most individuals can afford (e.g., Excel) |
|  | 3 | Entirely open source and free software (e.g., R, Julia, Python) |
|  | NA | No software was used in the research that can be determined |
| Data availability | 0 | Not available or not mentioned in the publication |
|  | 1 | Available upon request to the author |
|  | 2 | Online, but inconvenient or non-permanent (e.g., login needed, paywall, FTP server, personal lab website that may disappear, or may have already disappeared) |
|  | 3 | Freely available online to anonymous users for foreseeable future (e.g., archived using Zenodo, dataverse or university library or some other proper archiving system) |
| Software availability | 0 | Not available or not mentioned in the publication |
|  | 1 | Expensive proprietary software that only institutions would typically purchase |
|  | 2 | Used proprietary software that most individuals can afford (e.g., Excel) or SAS that is now available free for use with a university login |
|  | 3 | Used entirely open source and free software (e.g., R, Julia, Python) |
|  | NA | No software was used in the research that can be determined |
| Software citation | 0 | Not mentioned |
|  | 1 | Software mentioned by name only |
|  | 2 | Software cited with version number |
|  | 3 | All software components (SAS PROCs, R, Julia or Python packages, etc.) cited |
|  | NA | No software was used in the research that can be determined as the article is written |

# Figures

![Figure 1: An example open and reproducible research workflow. Starting with the question, determine the methodology, describe it, make it available and cite it. Data are used in analysis and any binary files or code are made available as supplements to the manuscript. Source code and raw data are made available in a public repository, preferably with version control for tracking changes through time and a DOI for final released products.](data:application/eps;base64,)

Figure 1: An example open and reproducible research workflow. Starting with the question, determine the methodology, describe it, make it available and cite it. Data are used in analysis and any binary files or code are made available as supplements to the manuscript. Source code and raw data are made available in a public repository, preferably with version control for tracking changes through time and a DOI for final released products.

![Figure 2: Aggregated article scores for each of the four categories, (i) shows computational methods availability 0 - not available or not mentioned to 3 - freely available online to anonymous users. (ii) presents ‘Software availability,’ where ‘0’ was ‘Not available or not mentioned in the publication’; ‘1’ was ‘Expensive proprietary software that only institutions would typically purchase’; ‘2’ was ‘Used proprietary software that most individuals can afford (e.g., Excel) or SAS that is now available free for use with a university login’; and ‘3’ was ‘Used entirely open source and free software (e.g., R, Julia, Python)’; ‘NA’ indicates that no specialised software was necessary or used. (iii) displays ‘Software citated,’ where ‘0’ was ‘Not available or not mentioned in the publication’; ‘1’ was ‘Software mentioned by name only’; ‘2’ was ‘Software cited with version number’; and ‘3’ was ‘All software components (SAS PROCs, R, Julia or Python packages, etc.) cited’; ‘NA’ indicates that no specialised software was necessary or used. (iv) shows ‘Data availability,’ where ‘0’ was ‘Not available or not mentioned in the publication’; ‘1’ was ‘Available upon request to the author; ’2’ was ‘Online, but inconvenient or non-permanent (e.g., login needed, paywall, FTP server, personal lab website that may disappear, or may have already disappeared)’; and ‘3’ was ‘Freely available online to anonymous users for foreseeable future (e.g., archived using Zenodo, dataverse or university library or some other proper archiving system)’; ‘NA’ indicates that no data were generated, e.g. a methods paper.](data:application/eps;base64,)

Figure 2: Aggregated article scores for each of the four categories, (i) shows computational methods availability 0 - not available or not mentioned to 3 - freely available online to anonymous users. (ii) presents ‘Software availability,’ where ‘0’ was ‘Not available or not mentioned in the publication’; ‘1’ was ‘Expensive proprietary software that only institutions would typically purchase’; ‘2’ was ‘Used proprietary software that most individuals can afford (e.g., Excel) or SAS that is now available free for use with a university login’; and ‘3’ was ‘Used entirely open source and free software (e.g., R, Julia, Python)’; ‘NA’ indicates that no specialised software was necessary or used. (iii) displays ‘Software citated,’ where ‘0’ was ‘Not available or not mentioned in the publication’; ‘1’ was ‘Software mentioned by name only’; ‘2’ was ‘Software cited with version number’; and ‘3’ was ‘All software components (SAS PROCs, R, Julia or Python packages, etc.) cited’; ‘NA’ indicates that no specialised software was necessary or used. (iv) shows ‘Data availability,’ where ‘0’ was ‘Not available or not mentioned in the publication’; ‘1’ was ‘Available upon request to the author; ’2’ was ‘Online, but inconvenient or non-permanent (e.g., login needed, paywall, FTP server, personal lab website that may disappear, or may have already disappeared)’; and ‘3’ was ‘Freely available online to anonymous users for foreseeable future (e.g., archived using Zenodo, dataverse or university library or some other proper archiving system)’; ‘NA’ indicates that no data were generated, e.g. a methods paper.

# Colophon

This report was generated on using the following computational environment and dependencies:

#> ─ Session info ───────────────────────────────────────────────────────────────  
#> setting value   
#> version R version 4.1.0 (2021-05-18)  
#> os macOS Big Sur 11.4   
#> system aarch64, darwin20   
#> ui X11   
#> language (EN)   
#> collate en\_AU.UTF-8   
#> ctype en\_AU.UTF-8   
#> tz Australia/Perth   
#> date 2021-06-16   
#>   
#> ─ Packages ───────────────────────────────────────────────────────────────────  
#> package \* version date lib source   
#> assertthat 0.2.1 2019-03-21 [1] CRAN (R 4.1.0)  
#> base64enc 0.1-3 2015-07-28 [1] CRAN (R 4.1.0)  
#> bookdown 0.22 2021-04-22 [1] CRAN (R 4.1.0)  
#> cli 2.5.0 2021-04-26 [1] CRAN (R 4.1.0)  
#> colorspace 2.0-1 2021-05-04 [1] CRAN (R 4.1.0)  
#> crayon 1.4.1 2021-02-08 [1] CRAN (R 4.1.0)  
#> curl 4.3.1 2021-04-30 [1] CRAN (R 4.1.0)  
#> data.table 1.14.0 2021-02-21 [1] CRAN (R 4.1.0)  
#> DBI 1.1.1 2021-01-15 [1] CRAN (R 4.1.0)  
#> DiagrammeR \* 1.0.6.1 2020-05-08 [1] CRAN (R 4.1.0)  
#> DiagrammeRsvg \* 0.1 2016-02-04 [1] CRAN (R 4.1.0)  
#> digest 0.6.27 2020-10-24 [1] CRAN (R 4.1.0)  
#> dplyr \* 1.0.6 2021-05-05 [1] CRAN (R 4.1.0)  
#> ellipsis 0.3.2 2021-04-29 [1] CRAN (R 4.1.0)  
#> evaluate 0.14 2019-05-28 [1] CRAN (R 4.1.0)  
#> extrafont 0.17 2014-12-08 [1] CRAN (R 4.1.0)  
#> extrafontdb 1.0 2012-06-11 [1] CRAN (R 4.1.0)  
#> fansi 0.5.0 2021-05-25 [1] CRAN (R 4.1.0)  
#> farver 2.1.0 2021-02-28 [1] CRAN (R 4.1.0)  
#> flextable \* 0.6.6 2021-05-17 [1] CRAN (R 4.1.0)  
#> gdtools 0.2.3 2021-01-06 [1] CRAN (R 4.1.0)  
#> generics 0.1.0 2020-10-31 [1] CRAN (R 4.1.0)  
#> ggplot2 \* 3.3.3 2020-12-30 [1] CRAN (R 4.1.0)  
#> glue 1.4.2 2020-08-27 [1] CRAN (R 4.1.0)  
#> gtable 0.3.0 2019-03-25 [1] CRAN (R 4.1.0)  
#> hms 1.1.0 2021-05-17 [1] CRAN (R 4.1.0)  
#> htmltools 0.5.1.1 2021-01-22 [1] CRAN (R 4.1.0)  
#> htmlwidgets 1.5.3 2020-12-10 [1] CRAN (R 4.1.0)  
#> janitor \* 2.1.0 2021-01-05 [1] CRAN (R 4.1.0)  
#> jsonlite 1.7.2 2020-12-09 [1] CRAN (R 4.1.0)  
#> knitr 1.33 2021-04-24 [1] CRAN (R 4.1.0)  
#> labeling 0.4.2 2020-10-20 [1] CRAN (R 4.1.0)  
#> lifecycle 1.0.0 2021-02-15 [1] CRAN (R 4.1.0)  
#> lubridate 1.7.10 2021-02-26 [1] CRAN (R 4.1.0)  
#> magrittr 2.0.1 2020-11-17 [1] CRAN (R 4.1.0)  
#> munsell 0.5.0 2018-06-12 [1] CRAN (R 4.1.0)  
#> officer \* 0.3.18 2021-04-02 [1] CRAN (R 4.1.0)  
#> patchwork \* 1.1.1 2020-12-17 [1] CRAN (R 4.1.0)  
#> pillar 1.6.1 2021-05-16 [1] CRAN (R 4.1.0)  
#> pkgconfig 2.0.3 2019-09-22 [1] CRAN (R 4.1.0)  
#> purrr 0.3.4 2020-04-17 [1] CRAN (R 4.1.0)  
#> R6 2.5.0 2020-10-28 [1] CRAN (R 4.1.0)  
#> RColorBrewer 1.1-2 2014-12-07 [1] CRAN (R 4.1.0)  
#> Rcpp 1.0.6 2021-01-15 [1] CRAN (R 4.1.0)  
#> readODS 1.7.0 2020-07-10 [1] CRAN (R 4.1.0)  
#> readr 1.4.0 2020-10-05 [1] CRAN (R 4.1.0)  
#> Reproducibility.in.Plant.Pathology \* 0.0.1-2 2021-06-14 [1] local   
#> rlang 0.4.11 2021-04-30 [1] CRAN (R 4.1.0)  
#> rmarkdown 2.9 2021-06-15 [1] CRAN (R 4.1.0)  
#> rstudioapi 0.13 2020-11-12 [1] CRAN (R 4.1.0)  
#> rsvg \* 2.1.2 2021-05-03 [1] CRAN (R 4.1.0)  
#> Rttf2pt1 1.3.8 2020-01-10 [1] CRAN (R 4.1.0)  
#> scales 1.1.1 2020-05-11 [1] CRAN (R 4.1.0)  
#> sessioninfo 1.1.1 2018-11-05 [1] CRAN (R 4.1.0)  
#> snakecase 0.11.0 2019-05-25 [1] CRAN (R 4.1.0)  
#> stringi 1.6.2 2021-05-17 [1] CRAN (R 4.1.0)  
#> stringr 1.4.0 2019-02-10 [1] CRAN (R 4.1.0)  
#> systemfonts 1.0.2 2021-05-11 [1] CRAN (R 4.1.0)  
#> tibble 3.1.2 2021-05-16 [1] CRAN (R 4.1.0)  
#> tidyr \* 1.1.3 2021-03-03 [1] CRAN (R 4.1.0)  
#> tidyselect 1.1.1 2021-04-30 [1] CRAN (R 4.1.0)  
#> utf8 1.2.1 2021-03-12 [1] CRAN (R 4.1.0)  
#> uuid 0.1-4 2020-02-26 [1] CRAN (R 4.1.0)  
#> V8 3.4.2 2021-05-01 [1] CRAN (R 4.1.0)  
#> vctrs 0.3.8 2021-04-29 [1] CRAN (R 4.1.0)  
#> visNetwork 2.0.9 2019-12-06 [1] CRAN (R 4.1.0)  
#> withr 2.4.2 2021-04-18 [1] CRAN (R 4.1.0)  
#> xfun 0.24 2021-06-15 [1] CRAN (R 4.1.0)  
#> xml2 1.3.2 2020-04-23 [1] CRAN (R 4.1.0)  
#> yaml 2.2.1 2020-02-01 [1] CRAN (R 4.1.0)  
#> zip 2.2.0 2021-05-31 [1] CRAN (R 4.1.0)  
#>   
#> [1] /Users/adamsparks/Library/R/4.1/library  
#> [2] /Library/Frameworks/R.framework/Versions/4.1-arm64/Resources/library

The current Git commit details are:

#> Local: main /Users/adamsparks/OneDrive - Department of Primary Industries and Regional Development/Development/GitHub/OpenPlantPath/Reproducibility\_in\_Plant\_Pathology  
#> Remote: main @ origin (git@github.com:openplantpathology/Reproducibility\_in\_Plant\_Pathology.git)  
#> Head: [0c0d39f] 2021-06-16: Remove reproducibility score table as it duplicates figure 2