General Article

Advances in Methods and Practices in Psychological Science
2019, Vol. 2(1) 3–11
© The Author(s) 2019
Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/2515245918801915
www.psychologicalscience.org/AMPPS

Minimizing Mistakes in Psychological Science

Jeffrey N. Rouder[1], **Julia M. Haaf**[2], **and Hope K. Snyder**[2]
[1]Department of Cognitive Sciences, University of California, Irvine, and [2]Department of Psychological Sciences, University of Missouri

Abstract

Developing and implementing best practices in organizing a lab is challenging, especially in the face of new cultural norms, such as the open-science movement. Part of this challenge in today's landscape is using new technologies, including cloud storage and computer automation. In this article, we discuss a few practices designed to increase the reliability of scientific labs, focusing on ways to minimize common, ordinary mistakes. We borrow principles from the theory of high-reliability organizations, which has been used to characterize operational practices in high-risk environments, such as aviation and health care. Guided by these principles, we focus on five strategies: (a) implementing a lab culture focused on learning from mistakes, (b) using computer automation in data and metadata collection whenever possible, (c) standardizing organizational strategies, (d) using coded rather than menu-driven analyses, and (e) developing expanded documents that record how analyses were performed.

Keywords

reliable science, open science, high-reliability organizations, data management

Received 2/10/18; Revision accepted 7/26/18

If you have been a member of a psychology lab, then perhaps you are familiar with things not going as well as planned. You may have experienced a programming error, equipment failure, or, more likely, some rather mundane human mistake. Our mistakes include failing to properly randomize an experiment, overwriting a file by typing in the wrong name, forgetting to record an important code, putting relevant information in the wrong directory, analyzing the wrong data set, mislabeling figures, and mistyping values of test statistics when transcribing from output to manuscripts. Finding these mistakes and preventing them from affecting publications is frustrating and time-consuming.

Lab practices, especially statistical practices, have come under scrutiny in the past several years. Psychological scientists perhaps sit at the confluence of three troubling trends: First, some findings that were once thought to be rock solid have failed to be replicated in preregistered research (Ebersole et al., 2016; Hagger et al., 2016; Harris, Coburn, Rohrer, & Pashler, 2013; Open Science Collaboration, 2015; Wagenmakers et al., 2016). Second, the field has been beset by a number of high-profile cases in which researchers fraudulently made up their data (Bhattacharjee, 2013). Third, seemingly improbable findings have been published in top-tier journals. The most famous of these were reported by Bem (2011), but several other claimed phenomena seem implausible as well (e.g., see Primestein, n.d.).

In response to this confluence, there have been many diagnoses and proposed solutions. Some stem from a global perspective according to which the problem is that incentives reward superficial success at the expense of the accumulation of knowledge (Finkel, Eastwick, & Reis, 2015; Nosek & Bar-Anan, 2012; Nosek, Spies, & Motyl, 2012). Researchers operating under such incentives may cut corners, especially in statistical testing (John, Loewenstein, & Prelec, 2012; Simmons, Nelson, & Simonsohn, 2011; Yu, Sprenger, Thomas, & Dougherty, 2014). Among the specific recommendations for solving this problem are to value replication experiments (Nosek et al., 2015; Roediger, 2012; Wagenmakers, Wetzels, Borsboom, van der Maas, & Kievit, 2012), to be open with data and methods so that other researchers may check

Corresponding Author:
Jeffrey N. Rouder, Department of Cognitive Sciences, Social and Behavioral Science Gateway, University of California, Irvine, CA 92697
E-mail: jrouder@uci.edu

your work (Rouder, 2016; Vanpaemel, Vermorgen, Deriemaecker, & Storms, 2015; Wicherts, Borsboom, Kats, & Molenaar, 2006), and to adopt statistical approaches that require more thought and care (Benjamin et al., 2018; Erdfelder, 2010; Gigerenzer, 1998; Rouder, Morey, Verhagen, Province, & Wagenmakers, 2016).

This article is about none of these issues. It is about mundane, commonly made, obvious mistakes—the type that everyone can agree are detrimental. They include boneheaded moves, such as reporting statistics based on incomplete data, using the wrong version of a figure, failing to properly randomize an experiment, and misplacing important codes. Nobody appears to be immune from making such mistakes. Obviously, nobody wants to make them, but is it worthwhile to address them proactively? Here is why we think ordinary mistakes should be taken seriously:

First, we think these mistakes are common in the literature. Perhaps the easiest mistake to detect is a malformed statement of a statistical test. A statement of a statistical test is malformed if the combination of test statistic and degrees of freedom does not match the *p* value. Nuijten, Hartgerink, van Assen, Epskamp, and Wicherts (2016) set out to document the frequency of this one type of error by searching the Web for statistical tests in published reports. They found that about half the publications in 30 years of literature contained at least one malformed statement of a statistical test. Although the specifics of this result have been controversial (Schmidt, 2016), the findings do show that this type of preventable mistake enters the literature frequently. What about other types of mistakes? It is hard to know how often people make mistakes that get codified in the literature because they are difficult, if not impossible, for anyone other than the authors to catch. If Nuijten et al.'s findings serve as a proxy for these other types of mistakes, then there is reason to suspect that they too are common.

A second reason to take ordinary mistakes seriously is that they may act as a difficult-to-detect, fieldwide bias for certain results. This argument comes from Gould (1981/1996), who noted that simple mistakes tend to go in researchers' preferred direction. In his famed monograph *The Mismeasure of Man*, Gould traced how scientists concluded that women were less intelligent than men and that colonized people were less intelligent than Europeans, documenting how questionable research practices were used in reinforcing preconceived notions about race, gender, and intelligence. In this context, Gould discussed the role of simple errors. One might think that if simple errors just occur randomly, they should be as likely to go against the researchers' preferred direction as to go with it. However, Gould argued that simple errors go in the preferred direction more often than against it. One mechanism underlying this bias is selective checking. Researchers tend to check their work vigorously for mistakes when results are against their stated hypotheses and beliefs. They do not check as vigorously when the results are in the anticipated direction. Therefore, uncaught mistakes tend to reinforce confirmation bias.

Gould's (1981/1996) hypothesis strikes us as reasonable. Imagine a researcher who fails to properly randomize an experiment. Say the researcher is studying Stroop effects and is unaware that incongruent trials have been overrepresented. When there are many incongruent trials, Stroop effects tend to be attenuated (Logan & Zbrodoff, 1979). Upon failing to find the anticipated priming effect, the researcher will likely recheck the code and find the randomization mistake. Suppose instead, however, that there was an overrepresentation of congruent trials. In such cases, Stroop effects are exaggerated (Logan & Zbrodoff, 1979). The researcher, having established the Stroop effect, is less likely to recheck the code.

Because ordinary mistakes may be common and may serve as a source of confirmation bias, we next discuss the principles used to avoid mistakes in high-risk environments and their relevance to the psychology lab.

High-Reliability Organizations' Principles for Avoiding Mistakes

A starting point for us in improving lab practices is to consider practices in high-risk fields, where mistakes, failures, and accidents can have devastating consequences. Examples of such fields include aviation, the military, the nuclear-power industry, and health care. Fortunately, there is a subdiscipline of management devoted to studying and improving organizations that serve in such high-risk environments. Organizations that mitigate risks through ongoing processes are sometimes known as high-reliability organizations, and the principles they follow are known as high-reliability-organization principles (Weick, Sutcliffe, & Obstfeld, 2008).

Should your lab be a high-reliability organization? Fortunately, mistakes in psychology labs do not have life-or-death consequences. Nonetheless, errors threaten the validity of the literature. The good news is that the principles of high-reliability organizations transfer well to the academic lab setting. In this section, we review these five principles and describe how they lead to the construction of a better lab.

Principle 1: sensitivity to operations

Researchers in experimental psychology are in the knowledge-production business. They often focus on the *what* of this business. What are the experiments? What are the data? What are the theories? What inferences about the theories do the data allow? Attention is on outcomes rather than processes. Sensitivity to operations, however, means focusing on processes—in the case of the psychology lab, the *how* of knowledge production. How can researchers ensure that experiments are properly randomized? How should they document who did what where? How can they ensure the integrity of the knowledge produced? In practice, sensitivity to operations means studying the mechanistic processes by which a lab produces knowledge.

Principle 2: preoccupation with failure

High-reliability organizations are preoccupied with failures. In scrutinizing their operations, they look for points of possible failure. They are constantly trying to envision how things could go wrong and to take safeguards before they do. One element of this preoccupation is taking near-miss events as seriously as consequential mistakes. In aviation, for example, runway incursions that have no effect on operations and those that materially threaten safety are scrutinized in much the same way. In a lab setting, preoccupation with failure means looking for ways to proactively anticipate and avoid mistakes, and taking small mistakes seriously.

Principles 3 and 4: resilience in the face of failure and reluctance to simplify

Principles 3 and 4 both apply to failures, either small or catastrophic. Resilience refers to a mature attitude toward failures—acknowledging that, although they are to be minimized, they will occur from time to time. This maturity means that the organization has the processes in place to learn from failures so that they will not be repeated. Reluctance to simplify means that in diagnosing the cause of failures, simple answers, such as operator error, are not considered satisfying. The goal is to go to the root of the problem, accepting that the organization is responsible for anticipating routine human and machine failures. Resilience and reluctance to simplify may be implemented in experimental psychology labs as well. The key is to avoid considering failures to be failures of meticulousness. In a resilient lab, when things go wrong—and they will—people talk about them, document them, and learn from them.

Principle 5: deference to expertise

Deference to expertise is a principle designed to address hierarchies in organizations. Administrators may be higher in the organizational structure, but decisions about operations need to reflect deference to people who execute these operations on a daily basis. In health care, hospital administrators must defer to the expertise of nurses and doctors who execute the daily operations. In aviation, the mechanics who work on planes each day have a unique vantage on safety in the maintenance of planes. Labs, too, have a hierarchy. Deference to expertise means that each lab member, be it an undergraduate research assistant, a lab manager, a graduate student, a postdoctoral fellow, or a principal investigator (PI), has certain expertise. Undergraduates are helpful for understanding where human mistakes can happen in executing the experiments; graduate students can comment on errors when programming experiments and performing analyses. If a mistake is made in executing an experiment, then given their expertise, undergraduates may have the best insight into why the mistake occurred and what may be done to avoid it in the future. Listening to undergraduates in this regard is a form of deference to expertise.

From Principles to Practices

We adopted high-reliability-organization principles in 2014 and have been following them since. The five principles do not lead to any specific set of practices per se. Instead, they serve as guiding principles for formulating actions in response to real-world circumstances. The resulting practices reflect the pressures faced by the organization, the operations in place, and the foresight of the people within the organization, among other factors.

In our case, given the nature of the mistakes we were making, following high-reliability-organization principles has led to the following five practices: (a) adopting a lab culture focused on learning from mistakes, (b) implementing radical computer automation, (c) standardizing organizational strategies across lab members, (d) ensuring that statistical analyses are coded, and (e) developing expanded manuscripts in which documentation of analyses is woven into the manuscript files. These five strategies reflect the problems we faced and our overall comfort with computers when devising solutions.

We report our journey from the high-reliability-organization principles to these five practices because we think it can help other researchers minimize and mitigate mistakes. We provide no evidence of our practices' effectiveness other than our experiences. Moreover, other researchers who follow the same principles may come up with additional or alternative behavioral recommendations. We also describe our practices at a fairly general level without specific recommendations on implementation, if only because no one set of specific recommendations is best for all labs. Klein et al.

(in press) have described the cyberlandscape for sharing data in far more detail.

A lab culture focused on learning from mistakes

In our current lab culture, we discuss problems and mistakes readily and often. Mistakes are *socialized*; that is, they are interpreted as reflecting a failure of systems rather than a failure of people. This was not always the case, and the following story helps set up the contrast between a lab that learns from mistakes and one that does not.

Michael Pratte, a former graduate student and current assistant professor, tells the following story: Back when our experiments were programmed in C and executed in DOS, he mistakenly typecast a variable as an integer rather than as a float. As a result, the code was not warning participants when they responded too quickly. We routinely provide this warning to discourage participants from responding very quickly, as they may do to shorten the duration of the experimental session. Pratte recounts feeling sick to his stomach when this mistake was discovered, because he knew that the mistake might have affected several months' worth of data collection. He believed at the time that this mistake was his alone.

Why did this mistake happen? It would be easy enough to blame Pratte for miscasting the variable, but that blame would do nothing to improve the reliability of the lab. Instead, errors and failures need to be brought out into the open, where they may be examined. Otherwise, it is difficult to learn from them.

How was this idea put into action in Pratte's case? The PI, the first author of this article, had set up the lab so that experiments were programmed in C because he knew C well. But, C is a notoriously difficult programming language for newcomers, and newcomers tend to make this type of mistake. The core problem was the choice of C as opposed to a more user-friendly language. In response to Pratte's mistake, the lab moved on from C. Experiments are now programmed in the more forgiving Psychophysics Toolbox (Brainard, 1997).

One way of learning from mistakes is to record all of them. When we make a mistake, we open an adverse-event record, collaboratively, at a lab meeting. Our adverse-events form is simple: One box is for a statement of the problem and the mistake it led to, another is for a set of possible solutions, a third is for the resolution (which solution was chosen and why), and a fourth is for noting whether the resolution results in formal policy changes in the lab. We fill these boxes out together in a lab meeting, and they are logged within our database.

Labs do not need to wait for mistakes to have these discussions. High-reliability organizations conduct *after-event reviews*, at which processes are reviewed on a routine basis. For example, after a manuscript is submitted for publication, the lab may engage in a review of the process although there has been no precipitating mistake. Our lab, however, has not taken this course; we find the adverse-event approach sufficient.

Our recommendation is that all adverse events be socialized. Explicit statements of lab values should include some sense that mistakes, when they occur, are as much a failure of foresight of the lab as they are a failure of any individual.

Radical computer automation

All labs keep records about their experiments. The question is whether these records are sufficiently detailed to minimize the frequency of mistakes and mitigate them when they occur. One way of knowing if the records are sufficient is to perform *stress tests*. Consider the following scenarios:

- A graduate student has just discovered that the keyboard in Room 3 is sticky, and it must be hit multiple times for a single keystroke to be recorded. You have no idea how long this has been the case, but are sure the keyboard was fine last year. Can you identify all the data that have been obtained in Room 3 this year so that you can inspect them? (This is a true story from our lab. We had about a 3-week period with a bad keyboard. Fortunately, we were able to identify all data that were affected.)
- You have returned to a project after a long hiatus. You notice that the data were previously cleaned by a graduate student who, unfortunately, dropped out in his first year. Is there a system in which these cleaning decisions were recorded, or are they gone with the student? If the decisions were not documented, can you find the raw data? (This is a true story from our lab as well.)
- The new graduate student just changed the refresh rates on one of the computers to run her psychophysics experiment. She did so through the control panel and outside the experimental software. This is both possible and common in Windows and Mac OS. Unfortunately, the change affects the timing of other experiments run on the same computer. Do you have a record of which sessions were run at which timings? (This is a true story from our lab, too.)

To pass these stress tests, a lab needs detailed and complete records. The problem we faced before 2014 was that of incomplete records. People simply forgot to record all the information they should have. To address these mistakes, we undertook a fairly large-scale effort to radically automate metadata collection. The key for us was to adopt two new technologies: scripting and database management. As part of each experimental session, the computer launches a simple script asking the participant and experimenter to log in, and then collects demographics on the participant. The computer records the parameters for the session, including the experimenter, the room, the participant, the institutional-review-board (IRB) protocol for the session, the screen resolutions, the random-number-generator seed, and so forth. These parameters are entered into a relational database where they may be queried with readily available tools.

In service of the communal goal of improving the trustworthiness of the literature, we think it should become a fieldwide imperative to adopt greater computer automation. Some labs may be able to implement computer automation on their own, as was the case with our lab. Our main tool is a relational database, MySQL, which is run on a lab server. We have prewritten little scripts that insert the metadata into the database, and these are called by the experiments written in Psychophysics Toolbox. Of course, different labs may adopt different solutions in search of radical automation. Those that choose to do it on their own will find that the Web offers much help for learning database management, shell scripting, and programming. Software Carpentry, a nonprofit organization for improving research computing (https://software-carpentry.org), may be quite helpful; they provide a large collection of Web-based lessons in several useful technologies. The other approach is to use outside-the-lab expertise. The good news is that the needed expertise surely exists at your university—it might be in your department or college and available for free or at reasonable rates.

Standardization

The work of a lab may be organized in many ways. It is our experience that if each lab member is free to choose his or her own organizational strategy, each will choose a different approach. These differences are fine so long as each lab member tends to his or her own work. The differing strategies, however, become fodder for mistakes as soon as work is shared. From our experience, it is best if all lab members use the same organization.

Perhaps the best example of standardization is the Open Science Framework's (OSF's) storage system. The basic organizational unit in OSF is a *project*. Underneath projects are data files, manuscripts, and other components. Although projects differ somewhat, the basic structure helps researchers find elements with little guiding documentation.

A well-organized lab should have a specified organizational structure. Particular attention should be given to the following: standardization of experimental metadata, standardization of folder-naming conventions, and standardization in versioning. Standardization in metadata means that all records of all experiments should look similar. The lab should have a standard format for recording information on, for example, participants, sessions, and IRB protocols. Of course, variables differ across experiments, but standardization of the naming conventions across experiments is always helpful. Likewise, we find it helpful to have a standardized naming convention across directories and files so that future understanding of projects is seamless.

One source of mistakes is the clutter presented by retaining multiple versions of work products. Members of the same lab should follow a common approach to versioning. A simple approach is to put set strings, such as dates or version numbers, directly into file names. However, this file-name approach is not ideal in many ways, and we find that people tend to make mistakes when using it. Moreover, this approach defeats versioning on most cloud storage systems, such as Google Drive, Box, and Dropbox. These systems have automatic versioning. Box, for example, automatically assigns version numbers to documents, and changing the file name defeats this feature. We use Git for versioning because it gracefully deals with all our versioning needs. Vuorre and Curley (2018) have written a Tutorial on Git, and other lessons and books are readily available online (see Chacon & Straub, 2014). Mistakes from the clutter of multiple versions are easily avoided by standardizing the versioning strategy ahead of time.

In our lab, we organize our research output by projects. A project comprises conceptually related research, and we tend to use a rather small scale to define a project. A project lives in two places: in the file system and in the database.

In the file system, we use the following conventions. Every project has the following five directories: "dev," "share," "papers," "presentations," and "grants." The "dev" directory is for private code development; the "share" directory is for any code we wish to share. The "papers" directory includes a subdirectory for each submission (e.g., there are currently three directories, "sub," "sub2," and "rev1," for the three main versions of this article). When appropriate, there is also a "private" subdirectory for all communication with editors and reviewers. The "private" subdirectory and the "dev" and "grants" directories are not included in our public branch of a project. All projects follow this form, which makes it easy to find things.

In the database, we record the title of the project, a description of it, the lead investigator, and when it was last modified. We also have log entries for each project, and as people work on a project, they can write what they did in such an entry. Also recorded are the outputs of a project—the publications and talks associated with it. Finally, projects in the database are associated with one or more repositories—those places on the file system where the files are located.

Experiments are separate from projects in our system. They have similar conventions for storing the IRB protocols, naming columns in spreadsheets, and so forth.

Coded analysis

We found in practice that researchers who use Excel, a menu-driven system, to analyze their data occasionally cannot re-create a graph. To provide for the greatest reliability, data analysis should instead be coded. The problem with menu-driven systems is that choices need to be made while navigating the menus. These menu choices may be made quickly, and are often made without any record. Excel is unreliable because although the outputs and formulas may be saved, there are many steps (e.g., the copying of cells) that are not documented. Some analysis programs, such as SPSS, have both a menu-driven interface and code-based representations. These programs are reliable to the degree that researchers remember to save their code.

There are also code-based systems without menus. These systems, which include R (R Core Team, 2017), MATLAB (The MathWorks, Natick, MA), and SAS, run simplified computer languages that are tailored for data analysis. The inputs are the code, which is usually stored as a matter of routine. One of the nicest features of coded analysis is that the code may be shared. In many cases, the code itself is so transparent that other researchers need no further documentation to understand and replicate the analyses.

Expanded manuscripts

One common error is inaccurate reporting of analytic results. Over the years, we have made such tragic mistakes as including the wrong figure in a publication and analyzing raw rather than cleaned data. One way of minimizing these errors is to expand manuscripts to include the provenance of analyses. A trustworthy manuscript also includes a healthy trail indicating the code that produced the analysis, the version of the code that was used, the version of the data that was used, when the analysis was conducted, and who conducted it. One simple approach could be to use comment functions available in most word processors and typesetting systems. All analyses can be extensively documented in the comments, and the comments, though not published, should remain with the document.

We take a more integrated and reliable approach to expanding documents. We use R Markdown, a new composite of two very powerful platforms. One is R (R Core Team, 2017), which was mentioned previously. The other is pandoc Markdown syntax (MacFarlane, 2013), a simple typesetting system that renders pdf, Word, or HTML documents. This syntax is used to typeset the text and equations, and it is easy to learn. From a layout and typesetting perspective, it may not be as powerful as Word, but it has all the features researchers need to do reliable science. R Markdown documents can utilize different layouts, and one of the developed layout options (provided by papaja) follows the American Psychological Association's guidelines (Aust & Barth, 2018).

The key feature of an R Markdown document is that it may contain special boxes that are executed when the document is formatted. We use this feature to place R-code chunks into the R Markdown document, so that the chunks are executed in R when the document is formatted. The process of formatting the text and executing the R code at the same time is called *knitting*.

The submitted manuscript for this article, which is available at https://github.com/PerceptionAndCognitionLab/lab-transparent, provides an example of knitting. The R Markdown file p.Rmd in the papers/sub2 directory contains numerous R chunks, including the following, which assigns values −1, 0, 1, and 2 to the variable dat; takes its sample mean; and does a one-sample *t* test to see if the true mean is different from zero:

```
dat <- c(-1,0,1,2)
# the data are -1, 0, 1, 2

sampMean <- mean(dat)
# takes the mean of the data

tResults=t.test(dat)
# performs t-test

tOut=apa_print(tResults)$statistic
# apa-formatted string of t-test
```

The outputs are stored in the variables sampMean and tOut. We can reference the former within the text using `` `round(sampMean,2)` ``. When this document is knitted, the value of sampMean is rounded to two digits and printed; it is 0.50. A similar approach can be taken with the *t* test; for example, `` `r tOut` `` in this document yields "$t(3) = 0.77$, $p = .495$." Note that we never type the actual value of the statistics, and this approach prevents transcription errors. Moreover, if the data change—say, they are updated to include new participants—the code updates the values when it is run again. And if a researcher chooses different settings, say in cleaning the data, again, a simple run of the manuscript updates the values to reflect these new settings. We put our settings in a separate chunk for transparency.

This example chunk is too simple to be of much service. In a real-world application, one needs to retrieve data from a cloud, clean the data, perform analyses, and draw figures and tables. However, as users improve their R skills, these tasks become routine.

The knitted approach with R Markdown is growing in popularity. Here we highlight two innovative packages that we think are broadly useful. The first, which we mentioned previously, is Frederik Aust's package for writing APA-compliant manuscripts in R Markdown. This package, called papaja, does most of the formatting work for the author. In our example chunk, the function `apa_print()` comes from the papaja package, and it takes common test statistics computed in R and formats them in APA style. A review of papaja is provided in Aust and Barth (2018). The package also provides APA-compliant LaTeX tables. The second package is apaTables, an innovative R package developed by David Stanley. This package prints matrices and tables in R in an APA-compliant format. It too not only is convenient, but also eliminates transcription errors. (See Stanley & Spence, 2018, for an introduction and guide to apaTables.)

Conclusions: Minimizing Mistakes and Moving Toward a More Open Science

In this article, we have used the high-reliability-organization principles to make recommendations for minimizing mistakes. The recommendations are to adopt a lab culture focused on mistakes, use radical computer automation, standardize organizational strategies, use coded analyses, and expand manuscripts to include documentation of analyses. We have yet to find a researcher who argues against these practices. Instead, the more common response concerns the time commitment. Is it worth the time to implement these recommendations? This question is pertinent in the neoliberalized university, where administrators focus on bean counting of publications, citations, and grant revenues. And it is especially pertinent for young scholars eyeing their first appointment or a tenure clock.

We think there are a few different questions rolled up here. The easiest to answer is whether it is worth the time to read about and implement high-reliability-organization principles. The answer is assuredly "yes." The principles are simple, and the reading takes under an hour. Implementing them at the most general level is a matter of mind-set and focus. Shifting toward sensitivity to operations, being preoccupied with failure, being resilient in the face of failure, and deferring to expertise is always worth the time.

The real time commitment, however, comes in changing how things are done to avoid mistakes. We have listed our five behavioral approaches. Some are not time intensive; for example, standardization involves little to no time cost. However, others, such as adopting radical computer automation and expanded documents, may require learning new skills. And that does take time. For us, time spent on making the lab more reliable is an up-front investment. Once we implement a change, it seems that many subsequent activities become easier and more convenient. We feel that our specific recommendations, especially those about automation, are great time-savers over the course of years or decades. Researchers for whom these technologies are new need not pick them all up at once. They may be implemented in steps. Perhaps you may learn about R Markdown this year, about relational databases next year, and so on.

One of the hidden benefits of making the lab more reliable is that it opens the door to open science. We define open science as working to preserve the ability of other people to reach their own opinions about our data and analyses. Because other people can reach their own opinions, opening up our work is a scary proposition that involves some intellectual risk and professional vulnerability. After all, we would be grateful but also mortified—and especially the latter—if someone found a critical error in our work. One way we try to manage this risk and vulnerability is to do the best we can to avoid mistakes. Having a reliable pipeline gives us the confidence to be public and open. And being open reinforces the need to be reliable.

We have been practicing open science for about 2 years. It is our view that we have gained a not-so-obvious benefit, as follows: There are many little decisions that people must make in performing research. To the extent that these little decisions tend to go in a preferred direction, they may be thought of as subtle biases. These decisions are often made quickly, sometimes without much thought, and sometimes without awareness that a decision has been made. Being open has increased our awareness of these little decisions. Lab members bring them to the forefront early in the research process, when they may be critically examined. One example is that a student brought up outlier detection very early in the process, knowing not only that she would have to report her approach, but also that other people could try different approaches with the same data. Addressing these decisions head on, transparently, and early in the process is an example of how practicing open science improves our own science.

Action Editor

Alex O. Holcombe served as action editor for this article.

Author Contributions

J. N. Rouder adapted the database and computer-automation solutions reported in this article. J. M. Haaf adapted the R Markdown expanded-document solutions. H. K. Snyder has served as a critical user of these approaches and has suggested several improvements. The three authors wrote the manuscript jointly.

Declaration of Conflicting Interests

The author(s) declared that there were no conflicts of interest with respect to the authorship or the publication of this article.

References

Aust, F., & Barth, M. (2018). papaja: Prepare APA journal articles with R Markdown [Computer software]. Retrieved from https://github.com/crsh/papaja

Bem, D. J. (2011). Feeling the future: Experimental evidence for anomalous retroactive influences on cognition and affect. *Journal of Personality and Social Psychology*, *100*, 407–425. doi:10.1037/a0021524

Benjamin, D. J., Berger, J. O., Johannesson, M., Nosek, B. A., Wagenmakers, E.-J., Berk, R., . . . Johnson, V. E. (2018). Redefine statistical significance. *Nature Human Behaviour*, *2*, 6–10.

Bhattacharjee, Y. (2013, April 26). The mind of a con man. *New York Times*. Retrieved from http://www.nytimes.com/2013/04/28/magazine/diederik-stapels-audacious-academic-fraud.html?pagewanted=all

Brainard, D. H. (1997). The Psychophysics Toolbox. *Spatial Vision*, *10*, 433–436.

Chacon, S., & Straub, B. (2014). *Pro Git*. Retrieved from https://www.git-scm.com/book/en/v2

Ebersole, C. R., Atherton, O. E., Belanger, A. L., Skulborstad, H. M., Allen, J. M., Banks, J. B., . . . Nosek, B. A. (2016). Many Labs 3: Evaluating participant pool quality across the academic semester via replication. *Journal of Experimental Social Psychology*, *67*, 68–82.

Erdfelder, E. (2010). A note on statistical analysis. *Experimental Psychology*, *57*, 1–4. doi:10.1027/1618-3169/a000001

Finkel, E. J., Eastwick, P. W., & Reis, H. T. (2015). Best research practices in psychology: Illustrating epistemological and pragmatic considerations with the case of relationship science. *Journal of Personality and Social Psychology*, *108*, 275–297.

Gigerenzer, G. (1998). We need statistical thinking, not statistical rituals. *Behavioral & Brain Sciences*, *21*, 199–200. doi:10.1017/S0140525X98281167

Gould, S. J. (1996). *The mismeasure of man*. New York, NY: W. W. Norton. (Original work published 1981)

Hagger, M. S., Chatzisarantis, N. L. D., Alberts, H., Anggono, C. O., Batailler, C., Birt, A. R., . . . Zwienenberg, M. (2016). A multilab preregistered replication of the ego-depletion effect. *Perspectives on Psychological Science*, *11*, 546–573.

Harris, C. R., Coburn, N., Rohrer, D., & Pashler, H. (2013). Two failures to replicate high-performance-goal priming effects. *PLOS ONE*, *8*(8), Article e72467. doi:10.1371/journal.pone

John, L. K., Loewenstein, G., & Prelec, D. (2012). Measuring the prevalence of questionable research practices with incentives for truth telling. *Psychological Science*, *23*, 524–532.

Klein, O., Hardwicke, T. E., Aust, F., Breuer, J., Danielsson, H., Hofelich Mohr, A., . . . Frank, M. C. (in press). A practical guide for transparency in psychological science. *Collabra: Psychology*.

Logan, G. D., & Zbrodoff, N. J. (1979). When it helps to be misled: Facilitative effects of increasing the frequency of conflicting stimuli in a Stroop-like task. *Memory & Cognition*, *7*, 166–174.

MacFarlane, J. (2013). Pandoc: A universal document converter [Computer software]. Retrieved from http://pandoc.org

Nosek, B. A., Alter, G., Banks, G. C., Borsboom, D., Bowman, S. D., Breckler, S. J., . . . Yarkoni, T. (2015). Promoting an open research culture. *Science*, *348*, 1422–1425.

Nosek, B. A., & Bar-Anan, Y. (2012). Scientific utopia: I. Opening scientific communication. *Psychological Inquiry*, *23*, 217–243.

Nosek, B. A., Spies, J. R., & Motyl, M. (2012). Scientific utopia: II. Restructuring incentives and practices to promote truth over publishability. *Perspectives on Psychological Science*, *7*, 615–631.

Nuijten, M. B., Hartgerink, C. H., van Assen, M. A., Epskamp, S., & Wicherts, J. M. (2016). The prevalence of statistical reporting errors in psychology (1985–2013). *Behavior Research Methods*, *48*, 1205–1226.

Open Science Collaboration. (2015). Estimating the reproducibility of psychological science. *Science*, *349*, aac4716. doi:10.1126/science.aac4716

Primestein, D. J. (n.d.). *What is psi-chology?* Retrieved from http://www.psi-chology.com

R Core Team. (2017). R: A language and environment for statistical computing (Version 3.4) [Computer software]. Retrieved from https://www.R-project.org/

Roediger, H. L., III. (2012). Psychology's woes and a partial cure: The value of replication. *Observer*, *25*(2), *9*, 27–29.

Rouder, J. N. (2016). The what, why, and how of born-open data. *Behavior Research Methods*, *48*, 1062–1069. doi:10.3758/s13428-015-0630-z

Rouder, J. N., Morey, R. D., Verhagen, J., Province, J. M., & Wagenmakers, E.-J. (2016). Is there a free lunch in inference? *Topics in Cognitive Science*, *8*, 520–547.

Schmidt, T. (2016). Sources of false positives and false negatives in the STATCHECK algorithm: Reply to Nuijten et al. (2016). *arXiv*. Retrieved from https://arxiv.org/abs/1610.01010

Simmons, J. P., Nelson, L. D., & Simonsohn, U. (2011). False-positive psychology: Undisclosed flexibility in data collection and analysis allows presenting anything as significant. *Psychological Science*, *22*, 1359–1366. doi:10.1177/0956797611417632

Stanley, D. J., & Spence, J. R. (2018). Reproducible tables in psychology using the apaTables package. *Advances in Methods and Practices in Psychological Science*, *1*, 415–431.

Vanpaemel, W., Vermorgen, M., Deriemaecker, L., & Storms, G. (2015). Are we wasting a good crisis? The availability of psychological research data after the storm. *Collabra: Psychology*, *1*, Article 3. doi:10 .1525/collabra.13

Vuorre, M., & Curley, J. P. (2018). Curating research assets: A Tutorial on the Git version control system. *Advances in Methods and Practices in Psychological Science*, *1*, 219–236.

Wagenmakers, E.-J., Beek, T., Dijkhoff, L., Gronau, Q. F., Acosta, A., Adams, R. B., Jr., . . . Zwaan, R. A. (2016). Registered Replication Report: Strack, Martin, & Stepper (1988). *Perspectives on Psychological Science*, *11*, 917–928.

Wagenmakers, E.-J., Wetzels, R., Borsboom, D., van der Maas, H. L. J., & Kievit, R. A. (2012). An agenda for purely confirmatory research. *Perspectives on Psychological Science*, *7*, 627–633. doi:10.1177/1745691612463078

Weick, K. E., Sutcliffe, K. M., & Obstfeld, D. (2008). Organizing for high reliability: Processes of collective mindfulness. *Research in Organizational Behavior*, *3*, 81–123.

Wicherts, J. M., Borsboom, D., Kats, J., & Molenaar, D. (2006). The poor availability of psychological research data for reanalysis. *American Psychologist*, *61*, 726–728.

Yu, E. C., Sprenger, A. M., Thomas, R. P., & Dougherty, M. R. (2014). When decision heuristics and science collide. *Psychonomic Bulletin & Review*, *21*, 268–282.