Error Tight: Exercises for Lab Groups to Prevent Research Mistakes

Julia F. Strand

Department of Psychology, Carleton College

Abstract

Scientists, being human, make mistakes. We transcribe things incorrectly, we make errors in our code, and we intend to do things and then forget. The consequences of errors in research may be as minor as wasted time and annoyance, but may be as severe as losing months of work or having to retract an article. The purpose of this tutorial is to help lab groups identify places in their research workflow where errors may occur and identify ways to avoid them. To do this, this article applies concepts from human factors research on how to create lab cultures and workflows that are intended to minimize errors. This article does not provide a one-size-fits-all set of guidelines for specific practices to use (e.g., one platform on which to backup data); instead, it gives examples of ways that mistakes can occur in research along with recommendations for systems that avoid and detect them. This tutorial is intended to be used as a discussion prompt prior to a lab meeting to help researchers reflect on their own processes and implement safeguards to avoid future errors.

Translational Abstract

Everyone makes mistakes. In science, mistakes can occur in many ways: Researchers may transcribe things incorrectly, make typos when writing code to analyze data, forget to do something they intended to, and so forth. These mistakes may simply waste time or require redoing work, but in more serious cases, they can ruin an experiment or lead to false conclusions. However, learning how to avoid errors in research isn't a standard part of training. This tutorial is intended to help lab groups identify places in the research process where errors may occur and identify ways to avoid them. To do so, this article draws on lessons from high-risk fields such as aviation, surgery, and construction, all of which have developed explicit, practical strategies to reduce mistakes on the job. This tutorial is intended to be used as a discussion prompt before a lab meeting to help researchers reflect on their own processes and implement safeguards to avoid future errors.

Keywords: error detection, independent verification, mistakes

No one is immune from making mistakes. In research, mistakes might include analyzing raw data instead of cleaned data, reversing variable labels, transcribing information incorrectly, or inadvertently saving over a file. The consequences of these kinds of mistakes can range from minor annoyances like wasted time and resources to major issues such as retraction of an article (Kovacs et al., 2021). Mistakes can happen under any circumstances, but the incentive structure of science—which rewards rapid, prolific publication rather than slow, methodological, and systematic work—may increase the frequency of their occurrence.

Estimates of error frequency are difficult to obtain because many go undetected or unreported. One way of estimating error rates is to conduct reanalyses of published work; a recent assessment of statistical reporting in psychology journals over the last 30 years showed that 49.6% of articles had at least one statistical inconsistency (e.g., inaccurate p-values given the degrees of freedom and test statistic reported; Nuijten et al., 2016). However, this approach can detect only one kind of error—inaccurate statistical reporting—and does not enable us to distinguish between true mistakes (e.g., copy-pasting the wrong p-value) and intentional misreporting (e.g., rounding a p-value to be slightly lower than it actually was).

Another method for assessing error prevalence is through researcher surveys. In a survey of 486 psychology researchers, 79% reported that they had made mistakes with "very low" or "low" frequency (Kovacs et al., 2021). However, given the stigma associated with admitting being wrong in science (Fetterman & Sassenberg, 2015), self-reported error rates may underestimate true error rates. In addition, many errors are likely to go undetected, further deflating self-reported error rates. Even if errors occur relatively infrequently, their consequences can be severe: In Kovacs et al.'s (2021) survey, when asked about the most serious mistake made, 22% involved major or extreme consequences such as "strongly affecting the central conclusion of the article," and "damaged professional reputation."

Although some changes to the practice of science can be contentious (e.g., requirements to preregister), the wonderful thing about mistakes is that we can all agree they are a problem! So

what can we do to make it less likely we will make mistakes, and more likely we will catch the mistakes we *do* make? The first step is understanding why errors occur.

Why Do Errors Happen?

We can conceive of the root of errors in two different ways: the person approach and the systems approach (Reason, 2000). In the person approach—or, as Dekker calls it in *The Field Guide to Understanding Human Error* (Dekker, 2017), the "bad apple theory of human error"—errors are attributed to an individual's negligence, forgetfulness, or inattention. The systems approach, on the other hand, thinks of errors as consequences, not causes; that is, errors are "the inevitable by-product of people doing the best they can in systems that themselves contain multiple subtle vulnerabilities" (Dekker, 2017, p. 4). The person approach may be appealing because it is usually possible to identify someone who is responsible. It also provides easy resolution when errors occur: Simply direct blame toward whomever made the mistake. However, the person approach can do little to systematically reduce the likelihood of future errors, as it does not target the root cause of mistakes. Thus, preventing future errors requires taking a systems approach and conceiving of mistakes as shortcomings in our workflows, rather than failures of individuals (Rouder et al., 2019).

Fields in which errors have immediate and dire consequences such as medicine (Kohn et al., 2014; Leape, 2009), aviation (Helmreich, 2000), and nuclear power (Heo & Park, 2010) have already adopted a systems approach (see Frese & Keith, 2015 for a review) and recognize that errors are inevitable, even among highly trained professionals. One of the pillars of DevOps culture (Kim et al., 2021)—an approach used widely in software engineering—is the principle of continuous learning. This advocates for correcting mistakes without blame, identifying the root of the mistake, and sharing what was learned from the mistake throughout the institution. For example, when developers at Google identify an error, they conduct "blameless postmortems" focused on identifying why the mistake happened and with the assumption that "everyone involved in an incident had good intentions and did the right thing with the information they had" (Lunney & Lueder, 2017). The *name, blame, and shame* approach that is often applied in cases of scientific misconduct does little to reduce the likelihood of unintentional errors (Nath et al., 2006).

Thus, psychologists and other scientists may be able to learn from disciplines in which errors are severe and costly enough that significant resources have been devoted to understanding how to avoid them (see Aboumatar et al., 2021). This article is not intended to summarize the extensive literature on best-practices for data management; interested readers should consult the recommended readings for more in-depth tutorials on that topic. Instead, the article aims to bring the conceptual approach many other disciplines take regarding error prevention to psychological researchers. A guiding principle underlying this work is that we cannot simply hope that mistakes will not happen; we must assume mistakes *will* occur and create systems to catch them. Next, the article presents guidelines for fostering a lab that embraces *safety culture* (see below), followed by recommendations for standardizing lab practices with error prevention in mind.

Best Practices for Error Prevention

Safety Culture in the Lab

The term *safety culture* (or "climate of safety") has been used by human factors researchers since the Chernobyl nuclear plant disaster in 1986 (Pidgeon, 1991) to describe a set of practices, norms, and beliefs that are intended to minimize danger within an organization (Guldenmund, 2000; Pidgeon & O'Leary, 1994). This framework for fostering a culture to reduce aversive events can easily be applied to research labs. Pidgeon and O'Leary (1994) argue that safety culture is promoted by four facets.

First, responsibility for safety *should not lie solely at the operational level*; senior management must identify safety as a core value and demonstrate a commitment to it. In the airline industry, this means that management makes it clear that they would prefer delayed flights over potentially unsafe flights and, therefore, incentivizes practices that promote safety rather than efficiency. In the research lab, this means that senior lab personnel must be actively involved in crafting systems to help their trainees (and themselves!) avoid making errors (see below), and be willing to accept slower, more methodical progress.

The second component for building safety culture proposed by Pidgeon and O'Leary (1994) is *shared concern about hazards within an organization.* That is, the burden of thinking about safety should not be carried by just one part of the organization. In a typical research project in which all contributors (e.g., authors) are invested in the work, shared concern will likely occur naturally. However, when people are involved in the work but not invested in it (e.g., individuals responsible for data entry without much intellectual engagement or the expectation of authorship), they may feel less concerned about ensuring the accuracy of their work. Thus, shared concern may be facilitated by ensuring that everyone involved in the project is invested in its accuracy and understands how their work contributes to its success.

The third component for building safety culture is *establishing and conveying realistic norms and rules about hazards*. Senior lab personnel can explicitly convey to trainees that there is an expectation that all work in the lab will follow standard procedures intended to prevent errors. Just as rules about safety are explicitly posted on a construction site (e.g., "Hard hats must be worn beyond this point"), labs can also explicitly convey rules about implementing practices to reduce errors (e.g., "All code must be independently reviewed by at least one other contributor").

Finally, Pidgeon and O'Leary (1994) advocate for *ongoing reflection about current practices*. Although some concerns about safety can be predicted ahead of time, new ones will always arise, so it is important to make discussing safety a regular habit. Normalizing conversations surrounding risks and errors in the lab will help identify new threats, and may also make lab members more willing to admit when they have found errors. Indeed, research in auditing firms indicates that people are more likely to report errors in firms that have an open climate around errors rather than those that take a more punitive approach to errors (Gold et al., 2014). To help build a culture that reflects on error prevention in a research lab, senior members may share stories about their own mistakes or near-misses. Talking about mistakes can also be part of the process of onboarding new students so everyone in the lab understands the lab philosophy

surrounding errors and their responsibility to say something when they occur. For example, part of the process of training new students in my lab involves reading and discussing this document, and my lab handbook includes the statement: "When mistakes happen (or nearly happen) in the lab, it's a great opportunity for us to figure out how to make our systems work better. Tell Julia about it right away and we'll use what you found to improve the work we do."

Safety culture is a guiding principle for many industries, including mining (Pillay et al., 2010), construction (Wamuziri, 2006), offshore drilling (Cox & Cheyne, 2000), medical care (Singer & Vogus, 2013), and many others. Although errors in psychological research are not likely to be immediately life-threatening in the way that mistakes in these disciplines can be, psychology researchers can benefit from the lessons derived from these higher-risk fields to minimize errors. In addition to this conceptual approach of fostering a culture that accepts the reality of errors occurring, reducing errors also requires modifying lab procedures.

Lab Protocols

Given that the most common cause of self-reported research errors is poor project preparation or management (Kovacs et al., 2021), a substantial proportion of errors may be avoided through programmatic changes to a research workflow.

Record Keeping

Keeping detailed records is part of the scientific process. However, in many labs, the only written record of the work is the article that is ultimately produced. Keeping a written record of the *process* of the work in addition to the final product it produces is useful for documenting the decisions that were made and reducing the likelihood of errors. In my lab, we keep these records in two forms: a Project Log and a Participant Log. The Project Log consists of a shared Google Doc that everyone on the team contributes to, but could be implemented in another form such as an electronic lab notebook (Nishida et al., 2020). Although the specific content within the Project Log will vary across labs, it may be helpful to include decisions made and the rationale for them (e.g., "we're going to run this study online rather than in the lab because it needs to be run between-subjects and we'll have trouble recruiting enough participants in the lab"), concrete steps in the research process (e.g., "VB wrote the code for analysis"), explanations of work that contributed to the article (e.g., "NDH drafted the introduction of the article"), and notations of when the work was checked by another lab member (e.g., "KS checked that the stimuli were properly labeled").

Having a detailed log of the process helps facilitate checking whether the work was done correctly. For example, knowing the intended volume of auditory stimuli will enable someone checking

Figure 1
A Snippet of a Sample Project Log From Our Lab

Sample Project Log

- May 3
  - JS remade stimuli at signal-to-noise ratio of -1. (Split off and leveled audio to -28dB total root mean square in Adobe Audition, then add in modulated babble leveled to -27dB). Combined audio and video in Adobe Premiere elements, exported as .mov files
    - JV checked all stimuli to make sure they're labeled correctly
- May 6
  - JS realized we can split up the passages for the experiment on subjective ratings into shorter chunks to get more observations per participant. So we'll have participants listen to a ~30 second chunk, do the subjective effort questions, and then repeat.
  - JS did power analysis for (see manuscript): we need 100 participants.
- May 10
  - JS split all the passages up into 20-30s chunks
    - JV checked the chunks to make sure they're the right length and content
- June 14
  - JV programmed the experiment in Gorilla.sc
    - Added subjective effort questions
    - Added filler questions after each ~30 sec chunk
    - Pseudorandomized order of presentation
    - Converted .mov files to .mp4s using Ffmpeg and uploaded files
    - Set a 2 Mbps minimum connection speed for participants as suggested by Gorilla Support to help with timing for loading video files
- June 18
  - JS checked the Gorilla.sc program. Decided to remove the temporal demand question in the subjective measure: doesn't apply to this task (we did this and justified why in Strand et al., 2018).
  - JV removed the temporal demand question from the Gorilla program, finalized the preregistration

Note. Acronyms are the initials of the team member who completed the task.

the work to verify that the actual volume matched the final volume (see Sample Project Log in Figure 1). A routine part of logging work can be flagging things that need to be checked by someone else. For example, in the May 3 entry of the sample project log in Figure 1, JS originally wrote "Who can check these stimuli to make sure they're labeled correctly?" and after checking them, JV replaced that line with "JV checked all stimuli . . . ." Thus, the log includes both a reminder to check work and information verifying that it was checked.

An added benefit of using a Project Log is that having a clear record of contributions over the course of the entire project can facilitate decisions about authorship at a later date. Knowing exactly how each individual contributed to the project may be useful for determining author order and can also help groups using the CRediT (Contributor Roles Taxonomy; Allen et al., 2014) system for tracking different forms of contributions to scientific scholarly output (see Holcombe et al., 2020 for an introduction to a web app and R package called Tenzing that is designed to facilitate the use of CRediT standards).

Many research groups also include a Participant Log for every project: a spreadsheet that includes each participant's ID, the date and time they were run and by which experimenter, and a place to provide notes about anything unusual that happened during data collection (e.g., the fire alarm went off and they had to stop early). This facilitates making decisions about excluding participants prior to looking at their data and can help to clarify missing or mislabeled data. This record is also useful if issues are discovered later that affect some but not all of the data (e.g., a particular experimenter was giving instructions incorrectly, one of the testing computers had a timing issue or was presenting auditory stimuli at the wrong level, etc.). Although information in the Participant Log is typically only used within a lab while a study is being run, if there are situations in which any of the information contained in the log may be useful postpublication, researchers should consider storing the Participant Log with the rest of the study data and making it accessible to others (subject to safeguards to protect participant identity).

Consistency Across Teammates

Labs often allocate work such that the same task—running participants, setting up equipment, transcribing participant responses—is done by multiple people. Using a written protocol for these tasks helps to ensure consistency in how these tasks are completed and avoid errors due to misunderstandings or misremembering (Gawande, 2010). This can be implemented via detailed, written protocols for how participants should be run that include the verbatim instructions experimenters should give, the order in which tasks should be completed, reminders about where to save data and what to name files, and other notes about administration. Some groups have even recorded videos of mock sessions of data collection to help ensure consistency across research sites (R. Klein et al., 2014).

For studies that require that lab members transcribe verbal responses from participants or code behaviors from videos, it may also be helpful to include protocols that give detailed information about how to transcribe or code responses for each task (e.g., a note to hide the column with the experimental condition when scoring to limit bias in the transcription process). Even if many transcription choices are straightforward, if coders are not given explicit instructions about what to do in unusual circumstances (e.g., the participant skips several trials, provides two words instead of one, the response is unintelligible, etc.), different people may code those responses differently. Using these standardized processes helps to avoid researcher misunderstanding or miscommunication about how the work is intended to be completed.

Errors can also be avoided by standardizing practices related to data storage and organization. For example, someone might be forgiven for thinking a file called "project_data_final.csv" was the appropriate data to be analyzed, despite the fact that they should have used "project_data_final_FINAL.csv" (I would not recommend this naming convention). To help avoid these kinds of errors, protocols can include explicit instructions for file naming conventions, standardized practices for commonly used variables, and instructions about the file types that should be used. Project TIER (Project TIER, 2022) provides a set of standards for documenting research and is likely to be a useful starting point for those looking to standardize how data are stored and organized (see also Sandve et al., 2013).

Checking Work

If we start with the assumption that mistakes will happen even when people are trying to avoid them, we must come up with methods of checking our work to find those mistakes. Among the most dominant paradigms in many safety-related disciplines (Larouzee & Le Coze, 2020) is James Reason's "swiss cheese" model of human failures. According to this model, in a complex system, each layer of protection against errors provides some defense but is imperfect. Although any given process may have holes in it, as long as the weaknesses of one layer are caught before the next, errors will not persist all the way through the project (Reason, 2000).

In scientific research, the first layer of protection against scientific errors is the approach of the individual researcher. This includes practices like "go slowly" and "be careful." For many, this is the extent of error prevention practices. As with any layer, however, it is imperfect. Thus, individuals can implement a second layer of protection by altering their workflow based on an understanding of how and why errors occur. When writing analysis code, for example, researchers can regularly write in tests to ensure that some of the assumptions about the data are actually true. This may include thinking through the number of participants or observations there *should* be at a given point in the analysis and including a line of code to check that the assumed number matches how many there *actually* are. It also includes visualizing all the raw data to identify obvious errors (e.g., ensure that proportions are bounded by 0 and 1). This additional scrutiny certainly catches some errors, but is not sufficient to catch all mistakes, as many errors occur in unexpected places.

The third layer of protection happens at the level of the lab or research group, and relies on multiple people verifying each step of the research process. In industries that rely heavily on coding, it would be considered poor practice to "publish" code that a single person had written and no one had verified in-house, but this is common practice in psychology. Additional scrutiny can be achieved by asking someone who did not write the code to thoroughly check every line to verify it. Given that it may be difficult

to thoroughly check data you believe are correct, insulating the "checker" from the hypotheses or outcomes (so that they are unaware of whether the results are expected or unexpected) may be helpful. Another strategy is telling the "checker" that there is an error somewhere in the code (you can even plant one, provided you come up with a system to make sure you remove it later!) to encourage them to look closely.[1] Alternatively, error-proofing code can be achieved by having two people write code independently to see if they arrive at the same conclusion. Success at this level is enhanced by having established a safety culture within the group so that the lab is mutually invested in the accuracy of everyone's work.

An additional layer of defense against errors can happen during the peer-review process, when reviewers identify issues that have managed to sneak past both the individual and the research group. Error detection at this level is facilitated by giving reviewers access to data and code associated with the experiment as many errors may not be detectable from the article alone. However, it is worth noting that carefully checking code is not a task many reviewers engage in, so authors should not rely on the peer-review process to identify errors.

The final layer of scrutiny is the scientific community, postpublication. Ideally, everyone wants to avoid or catch mistakes before publication. However, if that cannot be achieved, it is better to catch problems once they are published than to let them remain uncorrected in the literature. Thus, after publication, the availability of data and code in a publicly accessible repository such as the Open Science Framework further increases the likelihood that any mistakes will be found eventually. The thought of making your mistakes easier for others to find may be daunting, but finding them early facilitates scientific progress and ensures that future scientists do not waste time and resources building on spurious findings (Bishop, 2018).

In their work on building high-reliability organizations, Weick et al. (1999) advocate for approaching work with the expectation that things will go wrong and therefore actively seeking out problems (what they refer to as a "preoccupation with failure"). Researchers are more likely to go looking for problems or mistakes in their work when the data are not in line with their expectations. The danger of this "selective checking" is that we are only critical of a subset of our results: those we do not expect (see Bakker & Wicherts, 2011). Developing systems of looking for mistakes (Rouder et al., 2019)—and being open to finding them!—ensures that all results (not just surprising ones) are checked. Incorporating error hunting into every project makes it clear that checking for errors is not an indication of a lack of trust, it is simply part of the lab workflow.

Making Your Lab More Error-Tight

A recurring theme when reading about scientific errors is that mistakes happen in unexpected places and in unexpected ways. Reading examples of others' mistakes may therefore be useful to identify places where mistakes may happen in your process. Table 1 catalogs errors that researchers have made or nearly made at every stage of the research process: designing and programming experiments, collecting data, storing data, analyzing data, and reporting results. The rightmost column contains references to resources you can use to implement the approaches if you are not familiar with them.

This tutorial is meant to be discussed by research groups in a lab meeting. I recommend reading the article prior to the meeting, and then using the steps below to structure your discussion about how these issues apply to your own research.

Step 1

Make a list of the stages in a typical research project in your lab (e.g., what happens during the design phase, the data collection phase, etc.). Be sure to list every step even if it seems error-proof. For example, you may note that during each experimental session, participants must be given instructions, run on the most up-to-date version of the experiment, and assigned to the appropriate participant group.

Step 2

Brainstorm ways that errors might happen at each stage. These might be inspired by the examples given in Table 1, but it may also help to talk about ways that each phase was challenging to learn, or things that were unclear to trainees when they were first learning each stage. It is also likely to be useful to discuss ways that things have *almost* gone wrong previously: Identifying places where mistakes were nearly made is a great way of finding potential weak spots in a workflow. In the previous example, the experimenter could give the instructions incompletely or incorrectly, run the wrong version of the experiment, or assign a participant to the wrong group.

Step 3

Identify specific steps that could be used to reduce the likelihood of mistakes occurring at each stage (see the "How to avoid" column). To avoid the errors described in Step 2, you may decide to write a protocol that specifies exactly the instructions that should be given, ensure that the folder that contains the experiment does not contain anything else that it may be confused with (e.g., other experiments), and ask experimenters to double check the participant group before they begin.

It may be useful to write down any proposed changes to your workflow in a document that everyone has access to (e.g., final data files for analysis will be named . . .; the process for getting someone to independently check analysis code is . . .), such as a lab manual (Aly, 2018; Mehr, 2020). Keep in mind that if making all these changes seems overwhelming, it is perfectly reasonable to identify and implement a few changes that are manageable at first.

Step 4

Unfortunately, mistakes can happen, even in labs that implement all these practices. Therefore, it is worthwhile to discuss what to do in the event that someone finds an error. For example,

---

[1] A joke among computer programmers is "Ask a programmer to review 10 lines of code, they'll find 10 issues. Ask them to do 500 lines and they'll say it looks good" (Özil, 2013). Consider implementing checking at regular intervals rather than at the end of a project.

Table 1

Types of Errors That Can Be Made at Each Stage of the Research Process and How to Avoid Them

Stage	What can go wrong	Example	How to avoid
Designing/ programming	Errors in stimulus presentation software	Using mislabeled stimuli (Grave, 2021); programming an influential difference in the timing of two conditions (Strand, 2020); program was intended to randomly assign people to conditions but only assigned them to one condition.	Independent checking; leaving time to pilot and analyze pilot data prior to beginning the experiment so errors in programming are caught early; saving as much information as possible to recreate a trial if necessary
	Forgetting what you decided to do and why, or what you hypothesized and why	"Did we predict an interaction here?"; "Why did we choose Method A over Method B?"	Keeping records of decisions in a Project Log; formally preregistering your work (B. A. Nosek et al., 2018)
Collecting data	Equipment malfunction/ changes	Eyetracker becomes improperly calibrated; keyboard is sticky; screen resolution changes (Rouder et al., 2019); presenting stimuli at the wrong volume	Separate "running" computers from "coding/working" computers; keeping records of what equipment is used for each participant (to know which data to exclude) in a Participant Log
	Instructions are given to participants inconsistently	Telling some participants "complete both tasks to the best of your ability" and some "complete both tasks, but this task is the most important"	Using data collection protocols with clear scripts (or instructing experimenters to only read what is written on the instruction screen); keeping a Participant Log that includes which experimenters ran which participants
	Errors in manual coding	Incorrectly transcribing participant responses (Werner, 2018)	Giving explicit written instructions about how to do tasks; double-code pilot data to ensure consistency
	Experimenter forgets something during data collection	Forgetting to hit "record" prior to starting the participant on the task	Using data collection protocols with checklists for each step (Gawande, 2010)
Storing data	Data loss	Accidentally deleting files/writing over files	Using systems with version control like Git (Blischak et al., 2016; Chacon & Straub, 2014) or cloud storage; storing files in online repositories like the Open Science Framework to avoid overwriting and clearly delineate the active copy (see O. Klein et al., 2018 for a comparison of data sharing platforms); maintaining backups of all materials
	Using the wrong version of the data; poor documentation (not knowing what files to use/code to run/etc.)	Analyzing raw rather than cleaned data	Clear naming standards (Gorgolewski et al., 2016); using consistent file structure (e.g., only one file named project_data. csv is ever stored in the "data analysis" folder); maintaining a Project Log
	Variables in the data are mislabeled/ambiguous	Running the analysis on the wrong accuracy column in a dataset that contained two columns for accuracy—raw score and proportion correct; flipping variables (Miller, 2006); using mislabeled physical materials (Gewin, 2015); unintentionally replacing missing values (Aboumatar et al., 2021)	Setting up a lab style guide with clear and consistent naming standards (Arslan, 2019), including codebooks or metadata (e.g., each dataset is accompanied by a document that describes what each of the column headers means), manually checking for out-of-range values.
	Unwanted changes to data	Excel converting numbers to dates (Ziemann et al., 2016)	Using software without the known issues (Ziemann et al., 2016); following best practices for data organization in spreadsheets (Broman & Woo, 2018); implementing in-house independent checking
Analyzing data	Coding errors	Creating composite scores without reverse coding the necessary items; failing to exclude participants that should have been, treating a variable as an integer rather than a factor; scripting/coding error (Mann, 2013; Poldrack, 2013; Poldrack et al., 2020), reversing variable codes (Aboumatar et al., 2021)	Cleaning and analyzing data using a scripting language such as R in which every step is documented (Helping Organizations Migrate to the R Language, 2016); employing in-house independent checking; copiloting (Veldkamp et al., 2014); using a "Red Team" (Lakens, 2020); unit testing (Testing Your Code, 2013; Unit Testing for R, n.d.); having two coders work collaboratively to write code ("pair programming"; J. T. Nosek, 1998)

(table continues)

Table 1 (*continued*)

Stage	What can go wrong	Example	How to avoid
	Statistical errors	Failing to include random slopes in an analysis that warranted them (Rohrer et al., 2021)	Implementing in-house independent checking; code copiloting (Veldkamp et al., 2014); using a "Red Team" (Lakens, 2020)
Reporting/ writing	Copy/paste errors	While transcribing values from the statistical output to the manuscript file, copy/pasting the wrong value	Using R Markdown (Aust & Barth, 2020; Xie et al., 2020) or another system to avoid having to cut/paste; in-house independent checking
	Incorporating incorrect elements	Inserting the wrong figure into a manuscript (Rouder et al., 2019)	Using R Markdown (Aust & Barth, 2020; Getting Started with R Markdown, n.d.; Xie et al., 2020) or another system to link the data and figures with the paper; independent checking the output against the manuscript
	Citation errors	Citing the wrong article; failing to include a citation for an article	Using a reference manager to manage citations rather than typing references by hand; independent checking to ensure the article cited actually supports the claim being made and all cited articles appear in the reference section

you might set as a lab policy that after identifying an error, the first step is to ask someone to verify that a problem has occurred (to avoid alerting the whole lab in the event of a false alarm). It is also useful to discuss who to tell first, how to evaluate if the problem affects published papers or works in progress, and so on. For principal investigators, this can be an important opportunity to explicitly tell your trainees that they will *not* be punished or penalized for reporting an error. It may also be useful to remind students that sharing stories of near-misses are also informative, because it is possible to incorporate changes to your workflow based on those as well.

Step 5

After implementing some of the changes, plan a follow-up meeting where you can discuss what worked well and what needs improvement, and refine your process as needed.

Conclusions

Although entirely eliminating errors from research seems like a laudable goal, it is important to consider that the strategies described above require researchers' time and effort that could otherwise be invested elsewhere. To evaluate the value of these error mitigation practices, it may therefore be necessary to weigh the potential benefits (i.e., What are the consequences of avoiding errors?) against the costs (i.e., How much time and effort are necessary to implement these steps?). For some disciplines, this cost-benefit analysis is clear. In accounting, where errors are financially costly (Stefaniak & Robertson, 2010), or in surgery (Haynes et al., 2009) and aviation (Degani & Wiener, 1991), in which mistakes can be fatal, systems explicitly designed to avoid errors are standard practice, even if they decrease efficiency.

In psychological research, major mistakes may threaten researchers' careers, hamper progress in the field, and undermine public faith in science. Thus, the clear benefit of implementing error mitigation strategies is avoiding these adverse outcomes. However, many of the methods described in this article have benefits beyond error prevention as well. For example, practices like preregistration and sharing data increase research transparency and facilitate a more cumulative science, maintaining a digital organization system saves time and energy searching for content, and writing commented code facilitates reuse.

The costs of implementing error-prevention practices can range from very low (adopting a consistent file naming convention) to very high (having three team members independently write the same analysis code to ensure they arrive at the same outcome), so researchers must decide the approach that seems most reasonable given the context in which the decision is made. Critically, all the changes suggested above can be incorporated piecemeal; it is possible to add any component individually rather than implementing them all at once, so the individual costs need not be paid in one lump sum. Further, the costs in our discipline are already reduced because high-risk disciplines have done the hard work of identifying effective strategies for reducing errors. Thus, psychology would benefit from adopting these strategies; we must approach our work with the understanding that humans will make mistakes and preventing those mistakes requires reexamining both lab culture and research workflow.

Resources

For groups who wish to read more, I recommend:

- Bishop, D. V. M. (2018). Fallibility in science: Responding to errors in the work of oneself and others. *Advances in Methods and Practices in Psychological Science, 1*(3), 432–438.
- Kovacs, M., Hoekstra, R., & Aczel, B. (2021). The role of human fallibility in psychological research: A survey of mistakes in data management. *Advances in Methods and Practices in Psychological Science, 4*(4), 25152459211045930.
- Klein, O., Hardwicke, T. E., Aust, F., Breuer, J., Danielsson, H., Hofelich Mohr, A., Ijzerman, H., Nilsonne, G., Vanpaemel, W., & Frank, M. C. (2018). A practical

guide for transparency in psychological science. *Collabra: Psychology*, *41*(1), 20. https://doi.org/10.1525/collabra.158
- Rouder, J. N., Haaf, J. M., & Snyder, H. K. (2019). Minimizing mistakes in psychological science. *Advances in Methods and Practices in Psychological Science*, *2*(1), 3–11.
- Sandve, G. K., Nekrutenko, A., Taylor, J., & Hovig, E. (2013). Ten simple rules for reproducible computational research. *PLoS Computational Biology*, *9*(10), e1003285.
- Project TIER. (2022). https://www.projecttier.org

References

Aboumatar, H., Thompson, C., Garcia-Morales, E., Gurses, A. P., Naqibuddin, M., Saunders, J., Kim, S. W., & AWise, R. (2021). Perspective on reducing errors in research. *Contemporary Clinical Trials Communications*, *23*, 100838. https://doi.org/10.1016/j.conctc.2021.100838

Allen, L., Scott, J., Brand, A., Hlava, M., & Altman, M. (2014). Publishing: Credit where credit is due. *Nature*, *508*(7496), 312–313. https://doi.org/10.1038/508312a

Aly, M. (2018). *The key to a happy lab life is in the manual*. Nature Publishing Group UK. https://doi.org/10.1038/d41586-018-06167-w

Arslan, R. C. (2019). How to automatically document data with the codebook package to facilitate data reuse. *Advances in Methods and Practices in Psychological Science*, *2*(2), 169–187. https://doi.org/10.1177/2515245919838783

Aust, F., & Barth, M. (2020). *papaja: Reproducible APA manuscripts with R Markdown*. http://frederikaust.com/papaja_man/

Bakker, M., & Wicherts, J. M. (2011). The (mis)reporting of statistical results in psychology journals. *Behavior Research Methods*, *43*(3), 666–678. https://doi.org/10.3758/s13428-011-0089-5

Bishop, D. V. M. (2018). Fallibility in science: Responding to errors in the work of oneself and others. *Advances in Methods and Practices in Psychological Science*, *1*(3), 432–438. https://doi.org/10.1177/2515245918776632

Blischak, J. D., Davenport, E. R., & Wilson, G. (2016). A quick introduction to version control with Git and GitHub. *PLoS Computational Biology*, *12*(1), e1004668. https://doi.org/10.1371/journal.pcbi.1004668

Broman, K. W., & Woo, K. H. (2018). Data organization in spreadsheets. *The American Statistician*, *72*(1), 2–10. https://doi.org/10.1080/00031305.2017.1375989

Chacon, S., & Straub, B. (2014). *Pro Git*. https://www.git-scm.com/book/en/v2. https://doi.org/10.1007/978-1-4842-0076-6

Cox, S. J., & Cheyne, A. J. T. (2000). Assessing safety culture in offshore environments. *Safety Science*, *34*(1–3), 111–129. https://doi.org/10.1016/S0925-7535(00)00009-6

Degani, A., & Wiener, E. L. (1991). *Human factors of flight-deck checklists: The normal checklist*. https://ti.arc.nasa.gov/m/profile/adegani/Flight-Deck_Checklists.pdf

Dekker, S. (2017). *The field guide to understanding "human error."* CRC Press.

Fetterman, A. K., & Sassenberg, K. (2015). The reputational consequences of failed replications and wrongness admission among scientists. *PLoS ONE*, *10*(12), e0143723. https://doi.org/10.1371/journal.pone.0143723

Frese, M., & Keith, N. (2015). Action errors, error management, and learning in organizations. *Annual Review of Psychology*, *66*(1), 661–687. https://doi.org/10.1146/annurev-psych-010814-015205

Getting Started with R Markdown. (n.d.). Retrieved 2021 from https://ourcodingclub.github.io/tutorials/rmarkdown/

Gewin, V. (2015). *Rice researchers redress retraction*. https://doi.org/10.1038/nature.2015.18055

Gold, A., Gronewold, U., & Salterio, S. E. (2014). Error management in audit firms: Error climate, type, and originator. *The Accounting Review*, *89*(1), 303–330. https://doi.org/10.2308/accr-50592

Gorgolewski, K. J., Auer, T., Calhoun, V. D., Craddock, R. C., Das, S., Duff, E. P., Flandin, G., Ghosh, S. S., Glatard, T., Halchenko, Y. O., Handwerker, D. A., Hanke, M., Keator, D., Li, X., Michael, Z., Maumet, C., Nichols, B. N., Nichols, T. E., Pellman, J., . . . Poldrack, R. A. (2016). The brain imaging data structure, a format for organizing and describing outputs of neuroimaging experiments. *Scientific Data*, *3*(1), 160044. https://doi.org/10.1038/sdata.2016.44

Grave, J. (2021). Scientists should be open about their mistakes. *Nature Human Behaviour*, *5*(12), 1593. https://doi.org/10.1038/s41562-021-01225-2

Guldenmund, F. W. (2000). The nature of safety culture: A review of theory and research. *Safety Science*, *34*(1–3), 215–257. https://doi.org/10.1016/S0925-7535(00)00014-X

Gawande, A. (2010). *The checklist manifesto*. http://www.usafp.org/wp-content/uploads/2014/06/Winter-15-The-Checklist-Manifesto.pdf

Haynes, A. B., Weiser, T. G., Berry, W. R., Lipsitz, S. R., Breizat, A.-H. S., Dellinger, E. P., Herbosa, T., Joseph, S., Kibatala, P. L., Lapitan, M. C. M., Merry, A. F., Moorthy, K., Reznick, R. K., Taylor, B., & Gawande, A. A. (2009). A surgical safety checklist to reduce morbidity and mortality in a global population. *The New England Journal of Medicine*, *360*(5), 491–499. https://doi.org/10.1056/NEJMsa0810119

Helmreich, R. L. (2000). On error management: Lessons from aviation. *BMJ*, *320*(7237), 781–785. https://doi.org/10.1136/bmj.320.7237.781

Helping Organizations Migrate to the R Language. (2016, September 29). http://r4stats.com/articles/migrate-to-r/

Heo, G., & Park, J. (2010). A framework for evaluating the effects of maintenance-related human errors in nuclear power plants. *Reliability Engineering & System Safety*, *95*(7), 797–805. https://doi.org/10.1016/j.ress.2010.03.001

Holcombe, A. O., Kovacs, M., Aust, F., & Aczel, B. (2020). Documenting contributions to scholarly articles using CRediT and tenzing. *PLoS ONE*, *15*(12), e0244611. https://doi.org/10.1371/journal.pone.0244611

Kim, G., Humble, J., Debois, P., Willis, J., & Forsgren, N. (2021). *The DevOps handbook: How to create world-class agility, reliability, & security in technology organizations* (2nd ed.). IT Revolution Press.

Klein, O., Hardwicke, T. E., Aust, F., Breuer, J., Danielsson, H., Hofelich Mohr, A., IJzerman, H., Nilsonne, G., Vanpaemel, W., & Frank, M. C. (2018). A practical guide for transparency in psychological science. *Collabra: Psychology*, *41*(1), 20. https://doi.org/10.1525/collabra.158

Klein, R., Ratliff, K., Vianello, M., Adams, R., Bahník, S., Bernstein, M., Bocian, K., Brandt, M., Brooks, B., Brumbaugh, C., Cemalcilar, Z., Chandler, J., Cheong, W., Davis, W., Devos, T., Eisner, M., Frankowska, N., Furrow, D., Galliani, E., . . . Nosek, B. (2014). Data from investigating variation in replicability: A "many labs" replication project. *Journal of Open Psychology Data*, *2*(1), e4. https://doi.org/10.5334/jopd.ad

Kohn, L. T., Corrigan, J. M., & Donaldson, M. S. (2014). *To err is human: Building a safer health system*. Institute of Medicine (U.S.) Committee on Quality of Health Care in America.

Kovacs, M., Hoekstra, R., & Aczel, B. (2021). The role of human fallibility in psychological research: A survey of mistakes in data management. *Advances in Methods and Practices in Psychological Science*, *4*(4), 25152459211045930. https://doi.org/10.1177/25152459211045930

Lakens, D. (2020). Pandemic researchers - recruit your own best critics. *Nature*, *581*(7807), 121. https://doi.org/10.1038/d41586-020-01392-8

Larouzee, J., & Le Coze, J.-C. (2020). Good and bad reasons: The Swiss cheese model and its critics. *Safety Science*, *126*, 104660. https://doi.org/10.1016/j.ssci.2020.104660

Leape, L. L. (2009). Errors in medicine. *International Journal of Clinical Chemistry*, *404*(1), 2–5. https://doi.org/10.1016/j.cca.2009.03.020

Livio, M. (2013). Lab life: Don't bristle at blunders. *Nature*, *497*(7449), 309–310. https://doi.org/10.1038/497309a

Lunney, J., & Lueder, S. (2017). *Postmortem culture: Learning from failure*. https://sre.google/sre-book/postmortem-culture/