ERROR TIGHT

EXERCISES FOR LAB GROUPS TO PREVENT RESEARCH MISTAKES

By Julia Strand

"humans, even diligent, meticulous, and highly trained professionals, make mistakes."
- Nath, Marcus, & Druss (2006)

No one is immune from making mistakes. In research, mistakes might include things like 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 a paper.[1] Mistakes can happen under any circumstances, but their occurrence may be amplified by the incentive structure of science which rewards rapid, prolific publication rather than slow, methodological, and systematic work.

Although some changes to the process of doing science can be contentious (e.g., requirements to share data), the wonderful thing about *mistakes* is that we can all agree it would be great if we made fewer of them. So how can we set up our labs and our research workflows to make it less likely we'll make mistakes and more likely we'll catch the mistakes we make?

One clear path is to treat mistakes as what they are: shortcomings in our existing systems and workflows rather than failures of individuals.[2] Avoiding mistakes therefore requires that we put systems in place to prevent errors and catch the errors that manage to slip through.[3] The "name, blame, and shame" approach that is often applied in cases of scientific misconduct can do little to reduce the likelihood of unintentional errors.[4]

The purpose of this project is to provide hands-on exercises for lab groups to identify places in their research workflow where errors may occur and pinpoint ways to address them. The appropriate approach for a given lab will vary depending on the kind of research they do, their tools, the nature of the data they work with, and many other factors. Therefore, this project does not provide a set of one-size-fits-all guidelines, but rather is intended to be an exercise in self-reflection for researchers and provide resources for solutions that are well-suited to them.

Two key themes that stand out in the suggested solutions below are *standardization* and *looking for problems*.

- **Standardization:** Many errors can be avoided by standardizing digital organization. For example, someone might be forgiven for thinking a file called "project_data_final.csv" was the final, cleaned data to be analyzed, despite the fact that they should have used "project_data_final_FINAL.csv." The standardization recommendations given below apply to keeping records (e.g., which participant was run in which condition, on which computer, by which research assistant, etc.) and organizing files and materials (e.g., how final data files are named, how commonly used variables are labeled, etc.). Standardization can help prevent errors and facilitate independent checking of work.

- **Looking for problems:** Another general class of recommendations has to do with creating systems and protocols to check for issues, even when there aren't reasons to expect errors may be present. Researchers may be 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 don't expect. Developing a culture and systems of looking for mistakes (and being open to finding them![2]) ensures that all results (not just surprising ones) are checked. Implementing protocols for looking for errors has the added benefit that it conveys to students that mistakes are a normal part of the research process. This may lead to students being more willing to admit when they have found mistakes. Further, it makes it clear that checking for errors in any particular element of the project isn't an indication of a lack of trust, it's simply part of the process.

Making Your Lab More Error Tight

"You must learn from the mistakes of others. You can't possibly live long enough to make them all yourself."
- Samuel Levenson

A recurring theme when reading about the errors others have made is that mistakes happen in unexpected places and in unexpected ways. Therefore, reading examples of ways that others have made mistakes may be fruitful in stoking your creativity about where mistakes may happen in your process.

Step 1:

Before meeting as a group, read the table below. The "How to avoid" column contains references to resources you can use to implement the approaches if you aren't familiar with them.

Stage	What can go wrong	Example	How to avoid
Designing/ programming	Errors in stimulus presentation software	Programming in an influential difference in the timing of two conditions,[5] writing a program that is intended to randomly assign people to conditions but only assigns to one condition	Independent checking*, build in time to pilot and analyze the data as you plan to identify any issues, save as much information within a data script as possible to recreate a trial if necessary

	Forgetting what you decided to do in a study and why, or what you hypothesized and why	"Did we predict an interaction here?" "Why did we choose method A over method B?"	Preregistration,[6] maintain a collaborative project record**
Collecting data	Equipment malfunction/changes	Eyetracker becomes improperly calibrated, keyboard is sticky, screen resolution changes[2]	Separate "running" computers from "coding/working" computers, keep records of what equipment is used for each participant (to know which data to exclude), maintain a collaborative lab project log.
	Instructions are given to participants inconsistently	Some participants are told "complete both tasks to the best of your ability" and some are told "complete both tasks, but this task is the most important"	Data collection protocols with clear scripts (or instructing experimenters to only read what is written on the instruction screen), records to keep track of which experimenters ran which participants (in case issues are identified after the fact)
	Transcription errors (anything coded manually)	Experimenter incorrectly transcribing participant responses[7]	Explicit written instructions, pair coding (in which two people code the data together at the start to ensure consistency), select a subset of data to double-code
	Experimenter forgets something during data collection	The experimenter forgets to hit "record" prior to starting the participant on the task	Data collection protocols with checklists for each step[8]
Storing data	Data loss	Accidentally deleting files/writing over files	Use systems with version control like Git [9,10] or Dropbox, store files in online repositories like the Open Science Framework (to

			avoid over-writing and clearly delineate the active copy)
	Using the wrong version of the data, poor documentation (not knowing what files to use/code to run/etc.)	"No, you were supposed to use *mydata_final_final.csv* for the analysis, not *mydata_final.csv*",	Clear naming standards,[11] consistent file structure, collaborative project record
	Variables in the data are mislabeled/ambiguous	A dataset contains two columns for accuracy—raw score and proportion correct—and the analysis is run on the wrong "acc" column, mislabeled physical materials[12]	Set up a lab style guide with clear and consistent naming standards,[13] include codebooks or metadata
	Software errors	Excel converting things to dates[14]	Using software without the known issues,[14] in-house independent checking
Analyzing data	Coding errors	Creating composite scores without reverse coding the necessary items, failing to exclude participants you should have, variable treated as an integer rather than a factor, scripting/coding error[15–17]	Use a scripting language in which every step is documented,[18] in-house independent checking, co-piloting,[19] "Red Team"[20], unit testing [21,22]
	Statistical errors	Failing to include random slopes in an analysis that warranted them[5]	In-house independent checking, code co-piloting[19] "Red Team"[20]
Reporting/ writing	Copy/paste errors	While transcribing values from the statistical output to the manuscript file, copy/pasting the wrong value	Use R Markdown[23–25] or another system to avoid having to cut/paste. In-house independent checking
	Incorporating incorrect	Inserting the wrong figure	Use R Markdown[23–25] or another

elements	into a manuscript[2]	system to make data and figures linked with the paper. Independent checking the output against the manuscript.
Citation errors	Citing the wrong paper	Use a reference manager to manage citations, independent checking to ensure the paper cited actually supports the claim being made

* One option for implementing **independent checking** is asking someone who didn't write the code to thoroughly check every line of code to verify it. Given that it may be difficult to thoroughly check data you believe are correct, insulating the "checker" from the results (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. Alternatively, independent checking can be achieved by having two people write code independently to see if they arrive at the same conclusion.
** Maintaining a **project record** may involve using electronic lab notebooks[26] or even a shared document that everyone on the team can contribute to (e.g., a Google Doc). Entries in the log include decisions made (e.g., "we're going to do this as a within-subjects study") and rationale for them (e.g., "because we don't think we can recruit enough participants to do a between-subjects study") as well as concrete steps in the research process (e.g., "AB wrote the code for analysis, YZ checked it"). Project records can also contain information about participants such as 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. An added benefit of project records is that having a clear record of contributions can facilitate decision-making about authorship at a later date.

Step 2:

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.

Step 3:

Brainstorm ways that errors might happen at each stage. These might be inspired by the examples given, but it may also help to talk about ways that each phase was challenging to learn, ways errors have nearly been made at each stage, things that were unclear to trainees when they were first learning each stage, etc.

Step 4:

Identify specific, concrete steps that could be used to reduce the likelihood that mistakes might occur at each stage (see “How to avoid” column above). It may be useful to write these down in a document 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…,). Note that if making all these changes seems overwhelming, it's perfectly reasonable to identify and implement a few changes that are manageable at first. Every bit helps.

Step 5:

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, you might set as a lab policy that a 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, etc. For PIs, this can be an important opportunity to explicitly tell your trainees that they will not be punished or penalized for reporting an error.

Step 6:

Make a plan to follow up after implementing some of the changes and refine as needed.

Additional Recommendations

Ideally, you want to avoid/catch mistakes before publication. However, even if you can't achieve that, it is better to catch problems once they're published than let them stay in the literature.

- Sharing data and code[27] (e.g., by posting it to an online repository such as the Open Science Framework) during review increases the likelihood that mistakes will be identified by peer-reviewers or editors, in time for you to correct them prior to publication. After publication, the availability of data and code increases the likelihood that any mistakes will be found eventually. Although the thought of making your mistakes easier for others to find may be daunting, if mistakes are present, it is better to find them than waste time and resources in the future by attempting to build on spurious findings.[28]
- Conducting direct replications[29] of your own work as part of follow-up research is also an effective way of verifying your results.

If you have suggestions/recommendations/examples of mistakes/solutions of your own, please share them at errortight.com!

Resources

For groups who wish to read more, I recommend:

- Aczel, B., Kovacs, M., & Hoekstra, R. (preprint). The role of human fallibility in psychological research: A survey of mistakes in data management. https://psyarxiv.com/xcykz/
- 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.
- Rohrer, J. M., et al. (2021). Putting the Self in Self-Correction: Findings From the Loss-of-Confidence Project. *Perspectives on Psychological Science: A Journal of the Association for Psychological Science*, 1745691620964106.
- 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.

Acknowledgements

I'm very grateful to all the people who have publicly shared their mistakes [7,30,31] and provided feedback and input on this project: Violet Brown, Naseem Dillman-Hasso, Daniel Lakens, Jeff Rouder, & Dan Simons.

References

1. Aczel, B., Kovacs, M. & Hoekstra, R. The role of human fallibility in psychological research: A survey of mistakes in data management.
2. Rouder, J. N., Haaf, J. M. & Snyder, H. K. Minimizing Mistakes in Psychological Science. *Advances in Methods and Practices in Psychological Science* **2**, 3–11 (2019).
3. Bates, D. W. & Gawande, A. A. Error in medicine: what have we learned? *Ann. Intern. Med.* **132**, 763–767 (2000).
4. Nath, S. B., Marcus, S. C. & Druss, B. G. Retractions in the research literature: misconduct or mistakes? *Med. J. Aust.* **185**, 152–154 (2006).
5. Rohrer, J. M. *et al.* Putting the Self in Self-Correction: Findings From the Loss-of-Confidence Project. *Perspect. Psychol. Sci.* 1745691620964106 (2021).
6. Nosek, B. A., Ebersole, C. R., DeHaven, A. C. & Mellor, D. T. The preregistration revolution. *Proc.*

Natl. Acad. Sci. U. S. A. **115**, 2600–2606 (2018).
7. Werner, K. https://twitter.com/kaitlynmwerner/status/1021047716355493889 (2018).
8. Guwande, A. The checklist manifesto. *New York: Picadur* (2010).
9. Blischak, J. D., Davenport, E. R. & Wilson, G. A Quick Introduction to Version Control with Git and GitHub. *PLoS Comput. Biol.* **12**, e1004668 (2016).
10. Chacon, S. & Straub, B. Pro Git. https://www.git-scm.com/book/en/v2 (2014).
11. Gorgolewski, K. J. *et al.* The brain imaging data structure, a format for organizing and describing outputs of neuroimaging experiments. *Scientific Data* **3**, 160044 (2016).
12. Gewin, V. Rice researchers redress retraction. http://www.nature.com/news/rice-researchers-redress-retraction-1.18055 (2015) doi:10.1038/nature.2015.18055.
13. Arslan, R. C. How to Automatically Document Data With the codebook Package to Facilitate Data Reuse. *Advances in Methods and Practices in Psychological Science* **2**, 169–187 (2019).
14. Ziemann, M., Eren, Y. & El-Osta, A. Gene name errors are widespread in the scientific literature. *Genome Biol.* **17**, 177 (2016).
15. Mann, R. Prawns and Probability. http://prawnsandprobability.blogspot.com/2013/03/rethinking-retractions.html (2013).
16. Poldrack, R. Anatomy of a coding error. *http://www.russpoldrack.org/* http://www.russpoldrack.org/2013/02/anatomy-of-coding-error.html (2013).
17. Coding error postmortem. https://reproducibility.stanford.edu/coding-error-postmortem/.
18. Helping Organizations Migrate to the R language. http://r4stats.com/articles/migrate-to-r/ (2016).
19. Veldkamp, C. L. S., Nuijten, M. B., Dominguez-Alvarez, L., van Assen, M. A. L. M. & Wicherts, J. M. Statistical Reporting Errors and Collaboration on Statistical Analyses in Psychological Science. *PLoS One* **9**, e114876 (2014).
20. Lakens, D. Pandemic researchers - recruit your own best critics. *Nature* **581**, 121 (2020).
21. Unit Testing for R. https://testthat.r-lib.org/.
22. Testing your code. https://drclimate.wordpress.com/2013/10/10/testing-your-code/ (2013).
23. Aust, F. & Barth, M. papaja: Reproducible APA manuscripts with R Markdown. http://frederikaust.com/papaja_man/ (2020).
24. Xie, Y., Allaire, J. J. & Grolemund, G. R Markdown: The Definitive Guide. https://bookdown.org/yihui/rmarkdown/ (2020).
25. Getting Started with R Markdown. https://ourcodingclub.github.io/tutorials/rmarkdown/.
26. Nishida, E., Ishita, E., Watanabe, Y. & Tomiura, Y. Description of research data in laboratory notebooks: Challenges and opportunities. *Proc. Assoc. Inf. Sci. Technol.* **57**, (2020).
27. Klein, O. *et al.* A practical guide for transparency in psychological science. *psyarxiv.com › rtygmpsyarxiv.com › rtygm* (2018) doi:10.31234/osf.io/rtygm.
28. Bishop, D. V. M. Fallibility in Science: Responding to Errors in the Work of Oneself and Others. *Advances in Methods and Practices in Psychological Science* **1**, 432–438 (2018).
29. Simons, D. J. The value of direct replication. *Perspect. Psychol. Sci.* **9**, 76–80 (2014).
30. Livio, M. Lab life: don't bristle at blunders. *Nature* **497**, 309–310 (2013).
31. Ronald, P. Lab Life: The Anatomy of a Retraction. https://blogs.scientificamerican.com/food-matters/lab-life-the-anatomy-of-a-retraction/ (2013).