

Writing your first academic paper

If you are working in academics (and you are if you are working with [Jeff](#) then you are at least for the moment) you will need to write some papers. This guide is designed to help you get started on writing your first paper since the process can seem a little weird at first. This guide is focused on writing your first "first author" paper and not on collaborative papers which involve more interactions with other groups.

When to write your first academic paper?

As soon as possible! The purpose of graduate school is (in some order):

- Freedom
- Time to discover new knowledge
- Time to dive deep
- Opportunity for leadership
- Opportunity to make a name for yourself
 - R packages
 - Papers
 - Blogs
- Get a job

The first couple of years of graduate school are typically focused on (1) teaching you all the technical skills you need and (2) data dumping as much hard-won practical experience from more experienced people into your head as fast as possible.

After that one of your main focuses should be on establishing your own program of research and reputation. Especially for Ph.D. students it can not be emphasized enough *no one will care about your grades in graduate school but everyone will care what you produced*. See for example, Sherri's excellent [guide on CV's for academic positions](#).

Your advisor believes firmly that [R packages](#) and blog posts are just as important as papers, but the primary signal to most traditional academic communities still remains published peer-reviewed papers. So you should get started on writing them as soon as you can (definitely before you feel comfortable enough to try to write one).

Even if you aren't going to be in academics, papers are a great way to show off that you can (a) identify a useful project, (b) finish a project, and (c) write well. So the first thing you should be asking when you start a project is "what paper are we working on?"

What is an academic paper?

A scientific paper can be distilled into four parts:

1. A set of methodologies
2. A description of data
3. A set of results
4. A set of claims

When you (or anyone else) writes a paper the goal is to communicate clearly items 1-3 so that they can justify the set of claims you are making. Before you can even write down 4 you have to do 1-3. So that is where you start when writing a paper.

How do you start a paper?

The first thing you do is you decide on a problem to work on with your [advisor](#). This can be a problem that your advisor thought of or it can be a problem you are interested in, or a combination of both. Ideally your first project will have the following characteristics:

1. Concrete
2. Solves a scientific problem
3. Gives you an opportunity to learn something new
4. Something you feel ownership of
5. Something you want to work on

Points 4 and 5 can't be emphasized enough. I'll try to help you come up with a problem, but if you don't feel like it is *your* problem it will make writing the first paper a total slog. Hopefully we can find an option where you are just insanely curious to know the answer at the end, to the point where you *just have to figure it out* and kind of don't care what the answer is. That doesn't always happen, but that makes the grind of writing papers go down a lot easier.

Once you have a problem the next step is to actually do the research. I'll leave this for another guide, but the basic idea is that you want to follow the usual [data analytic process](#):

1. Define the question
2. Get/tidy the data
3. Explore the data
4. Build/borrow a model
5. Perform the analysis
6. Check/critique results
7. Write things up

The hardest part for the first paper is often knowing where to stop and start writing.

How do you know when to start writing?

Sometimes this is an easy question to answer. If you started with a very concrete question at the beginning then once you have done enough analysis to convince yourself that you have the answer to the question. If the answer to the question is interesting/surprising then it is time to stop and write.

If the answer isn't interesting/surprising but you started with a concrete question it is also time to stop and write. But things often get more tricky with this type of paper as most journals when reviewing papers filter for "interest" so sometimes a paper without a really "big" result will be harder to publish. **This is ok!!** Even though it may take longer to publish the paper, it is important to publish even results that aren't surprising/novel. I would much rather that you come to an answer you are comfortable with and we go through a little pain trying to get it published than you keep pushing until you get an "interesting" result, which may or may not be justifiable.

If you started with a question that wasn't so concrete then it gets a little trickier. The basic idea here is that you have convinced yourself you have a result that is worth reporting. Usually this takes the form of between 1 and 5 figures that show a coherent story that you could explain to someone in your field.

For the first paper you will get a lot of help from me in deciding when to stop. But in general one thing you should be working on in graduate school is your own internal timer that tells you, "ok we have done enough, time to write this up". I found this one of the hardest things to learn in graduate school, but if you are going to stay in academics it is a critical skill. There are rarely deadlines for paper writing (unless you are submitting to CS conferences) so it will eventually be up to you when to start writing. If you don't have a good clock, this can really slow down your ability to get things published and promoted in academics.

One good principle to keep in mind is "the perfect is the enemy of the very good" Another one is that a published paper in a respectable journal beats a paper you just never submit because you want to get it into the "best" journal.

How do you start writing?

1. Once you have a set of results and are ready to start writing up the paper the first thing is *not to write*. The first thing you should do is create a set of 1-4 publication-quality plots (see Chapter 10 [here](#)). Show these to Jeff to get confirmation on them before proceeding.
2. Start a project on [Overleaf](#) and invite Jeff to join.
3. Write up a story around the four plots in the simplest language you feel you can get away with, while still reporting all of the technical details that you can.
4. Go back and add references in only after you have finished the whole first draft.
5. Add in additional technical detail in the supplementary material if you need it.
6. Write up a reproducible version of your code that returns exactly the same numbers/figures in your paper with no input parameters needed.

What are the sections in a paper?

Normally when you are writing your first paper it will be about a statistical method you have developed to analyze a particular set of data, although sometimes it may be an analysis paper.

Keep in mind that most people will read the title of your paper only, a small fraction of those people will read the abstract, a small fraction of those people will read the introduction, and a small fraction of those people will read your whole paper. So make sure you get to the point quickly!

The sections of a methods paper are:

1. **Title:** Should be very short, no colons if possible, and state the main result. Example, "A new method for sequencing data that shows how to cure cancer". Here you want to make sure people will read the paper without overselling your results - this is a delicate balance.
2. **Abstract:** In (ideally) 4-5 sentences explain (a) what problem you are solving, (b) why people should care, (c) how you solved the problem, (d) what are the results and (e) a link to the [software you developed](#) implementing the method.
3. **Introduction:** A more lengthy (1-3 pages) explanation of the problem you are solving, why people should care, and how you are solving it. Here you also review what other people have done in the area. The most critical thing is never underestimate how little people know or care about what you are working on. It is your job to explain to them why they should.
4. **Methods:** This is the meat of a methods paper. You should state and explain your statistical model, what the parameters are, how you chose them, and any strengths or weaknesses of your proposed approach.
5. **Comparisons:** Compare your proposed approach to the state of the art methods. I believe in doing this with simulations (where you know the right answer) and data you haven't simulated (where you don't know the right answer). If you can base your simulation on data, even better. I also believe in [simulating both the easy case \(where your method should be great\) and harder cases where your method might be terrible](#).
6. **An example analysis:** It is good to have one compelling "use case" laid out from the beginning to the end. Ideally this is the problem that motivated you to create the method in the first place.
7. **Conclusions:** Summarize what you did and explain why what you did is important one more time. Provide a link to the software implementing your method and the code you used to perform the analysis.
8. **Supplementary Information:** Sometimes, if there are a lot of technical computational, experimental or statistical details, you can include a supplement that has all of the details so folks can follow along. As far as possible, try to include the detail in the main text but explained clearly.

The length of the paper will depend a lot on which journal you are targeting. In general the shorter/more concise the better. But unless you are shooting for a really glossy journal (ask Jeff before doing this first) you should try to include the details in the paper itself. This means most papers will be in the 4-15 page range, but with a huge variance.

The main difference with an analysis paper is you present the analysis/results you are doing first and move all of the methods and comparisons to after (or to the supplement).

Working with Jeff

When writing your first paper the usual way we will do it will be:

1. You write a draft of the paper
2. Schedule a block of time to re-write the paper with Jeff (usually 2-3 hours).
3. Based on the re-write finish up the new draft
4. Edit the paper
5. Submit

The first draft should be as complete as you think you can make it. Ideally this is something you think you could submit. If you haven't written an academic paper before this is almost never ready to submit. You should not feel bad about this or let it slow you down. Your goal is to write a draft.

The purpose of the re-write with Jeff is 2-fold. First, we will get the draft in "publication ready" shape. Don't feel bad if this involves a massive re-write. My advisor completely (and I mean completely) reworked the text in my first couple of papers as I was learning. Which brings us to the second purpose of this joint writing session - to learn how a paper is written and what to be thinking about when writing.

The joint writing session will almost never be enough time to tidy up all the details, it will usually focus on big points. So then you'll need to edit the paper to get all the details/references ready to go.

Finally, you'll submit the paper! This usually involves meeting with Jeff again to go through the manuscript submission process in case there are any files missing. Unfortunately, most manuscript submission systems are a bit clunky and this can take up to an hour to get everything entered/submitted. Almost always we will submit the paper simultaneously to a journal and to a pre-print server like [biorxiv](https://www.biorxiv.org/).

The process is actually just getting started

When you submit a paper you will feel this huge sense of accomplishment and relief. You should!!

But the process is actually only getting started. To get a paper published will also require:

1. Finishing up and distributing software
2. Receiving and responding to referee comments (the target of a future tutorial)
3. Re-writing the paper until it gets accepted

Once your paper is accepted you get to celebrate, hooray!!!! But you still aren't done. Most of the impact from a paper (especially in methods) comes from what happens after the paper is published. You will need to maintain the software, respond to questions, fix any typos in the paper (which will still happen), and

continue to "support" the work you did. But you will have the satisfaction of an amazing paper making you want to do all that extra work...

Developing a style

Scientific papers are written in a particular style. The best way to learn how to read these papers is to go to some journals that regularly publish good papers like [Biostatistics](#) or [Genome Biology](#) and read a ton of papers.

Learn how to write papers in a very clear and simple style. Whenever you can write in plain English and make the approach you are using understandable and clear. This can (sometimes) make it harder to get your papers into journals. But simple, clear language leads to much higher use/reading/citation/impact of your work.

I learned by mimicking the styles of other people. Examples of great writers are: [John Storey](#), [Nancy Reid](#), [Shirley Liu](#), [Brad Efron](#), and [Robert May](#), but there are a lot of other people that you can learn from. As with any writing, the more you read, the better a writer you will be.

I also find the [Hemmingway App](#) a good way to keep my writing simple.

A couple of other guides that may be useful

1. Guide to [reviewing](#)
2. Guide to [writing software](#)
3. Guide to [sharing data](#)

[Refer](#)