Feedback for Charae Gilbert, Emma Husby, Evan Leu, Nick Rezaee, and Matthew Rhea , Collatz Exploration Project

The introduction is effective, but please try to write clear academic English even in a mathematical context. For example:

- Sentence 1: Was the conjecture named after Collatz in 1937? Or was his research in 1937? The word "is" should be past tense. Avoid the passive voice. *Who* named the conjecture?
- Sentence 2: The clause "he does not claim to the finding of the ocnjecture" violates English grammar and spelling. You say that "his publishing revealed" – what did Collatz write? You might say "his article" or "his research" but the gerund "publishing" is awkward here.
- Sentence 3: The sentence begins "The conjecture states..." and goes on to describe the process, or Collatz function. The conjecture is the next sentence – that the described process eventually reaches one.

I'm not trying to be nitpicky, but this is writing in an academic context, and basic conventions should be followed.

For specific claims, like "the conjecture has been proven for the first 5.764×10^{18} numbers," a source and date should be given (since such records change all the time). Also, did Collatz actually publish his conjecture? I haven't looked into this, but most treatments say something fuzzier, like he "introduced" the conjecture.

The `Collatz` function is pretty good, for creating the desired list of numbers. Note that the repeated `if n==1 or ctx == MAX_NUM:` part can (and should) be written only once, at the end of the while loop, and outside the scope of the if/else statements. To me it's a bit odd to have `testNegative` as a separate parameter to the function – why not just put the line `testNegative = (n < 0)` at the beginning of the function?

I like how you've used your main Collatz function in defining halting-steps and dropping-steps functions. That's a convenient thing about using lists – you can reuse them for other study. On the other hand, from a memory/speed perpsective, it's a bit wasteful to compute and store the entire sequence of numbers, from n down to 1, if you only care about how long it takes to find a number less than n. So personally, I would write separate code for the dropping-steps function at least.

Why did you write a separate function for testing negative input, by the way? The Collatz function seems to work for positive or negative input, as it is (though zero would give it trouble!).

The other functions – for plotting and time-analysis – look very good to me. There are some improvements possible (e.g., it's bad to repeatedly append to a numpy array), but you're doing a good job with techniques that were beyond what we had learned at the time. The wrapper approach to time-analysis should work well.

The visualizations are very nice. I think the log-scale is effective at bringing out some patterns in the highest-reached-number data. The trajectory pictures are nice, but a bit odd as scatter-plots. For time-series data like this, I would recommend a line-plot instead, so the viewer can se the up-down patters more easily. But since I haven't looked at both kinds of plots, maybe the scatter has its advantages – the plots for different parities *do* show some interesting phenomena.

I think the analysis of these plots is pretty clear, but could be a bit deeper and more precise (in Sections 5.1.1 and 5.1.2). It would be easier if those paragraphs were placed right next to the graphs, so I didn't have to flip back and forth.

For the negative cases, your hard work (maybe it was tedious?) paid off as you found all three known cycles. The new function effectively determines which cycle a number ends up at. The analysis here

really goes deep, and has shows some good instincts. For example, I think it's interesting and natural to look at which of the three cycles are obtained and how often. By looking at a range of small numbers (between -100 and -1), and then an equal-sized range of large numbers (between -100000 and -99900), you have data that's easy to compare... and the pictures are on the same page. This complements the overall even-distribution, as you mention at the end of your analysis.

The conclusion is a bit weaker than the analysis you've carried out. I think it's a common writing mistake to write a conclusion that feels like a vague memorial of the paper ("This paper has been very interesting and important"). It's better to write a few striking sentences, referring to specific analyses you've carried out. The suggestion of future work is a good idea, but was a bit unclear to me.

Grade breakdown.

Coding tasks: 9/10
Quality of exposition: 4/5
Depth of exploration: 5/5
Total: 18/20.