Work session and stand-ups

Eleutherodactylus Research Studio · Fall 2026

Tom Devitt

2026-10-07

Bug 3, revisited: where the line goes

Inline code runs **where it sits**, top to bottom, like a chunk. It can only use objects made **above** it.

✗ Out of order: sentence before chunk

````
We measured `r nrow(by_male)` males.

```{r}
by_male <- measurements |>
  summarise(n_calls = n(), .by = male)
```
````

The sentence asks for `by_male` **before** the chunk below it has made it.

✓ Correct order: chunk, then sentence

````
```{r}
by_male <- measurements |>
  summarise(n_calls = n(), .by = male)
```

We measured `r nrow(by_male)` males.
````

The chunk makes `by_male` first, so the sentence can use it.

Bug 3: what each order renders

✗ Out-of-order render: it fails

In the **Background Jobs** tab:

```
Error in eval(parse_only(code), envir = envir) :
  object 'by_male' not found
Quitting from team3-notebook.qmd:45-45
```

- **No new HTML.** The render stops at line 45
- Any HTML you already had is the **old** one. Check its time stamp before you trust it

✓ Correct-order render: it works

In the rendered HTML:

> **Step 5. Summaries**
>
> We measured 7 males.

- The number is computed, not typed: add a male and re-render, and it changes to 8

45–45 is one line: your sentence, not a chunk. A one-line range in the error means inline code. Move the sentence below the chunk that makes the object.

In your team notebook

`by_male` and `by_species` are made in **Step 5**. Every sentence with a number from them goes **after** the Step 5 chunk:

```
We measured `r sum(by_species$n_calls)` calls from
`r sum(by_species$n_males)` males of `r nrow(by_species)` species.
```

Two more things that look like a bug but aren't:

- **The sentence only updates when you click Render.** Running chunks or Run All doesn't change it, in the editor or in the last HTML
- **Render starts a fresh R session.** A value you changed in the Console doesn't count. Change it in the notebook

Four observers, one dataset

You split the recordings and pool the numbers. That only works if every observer would have put the boundaries in the same places. **Two rules**, written in your notebook before anyone measures:

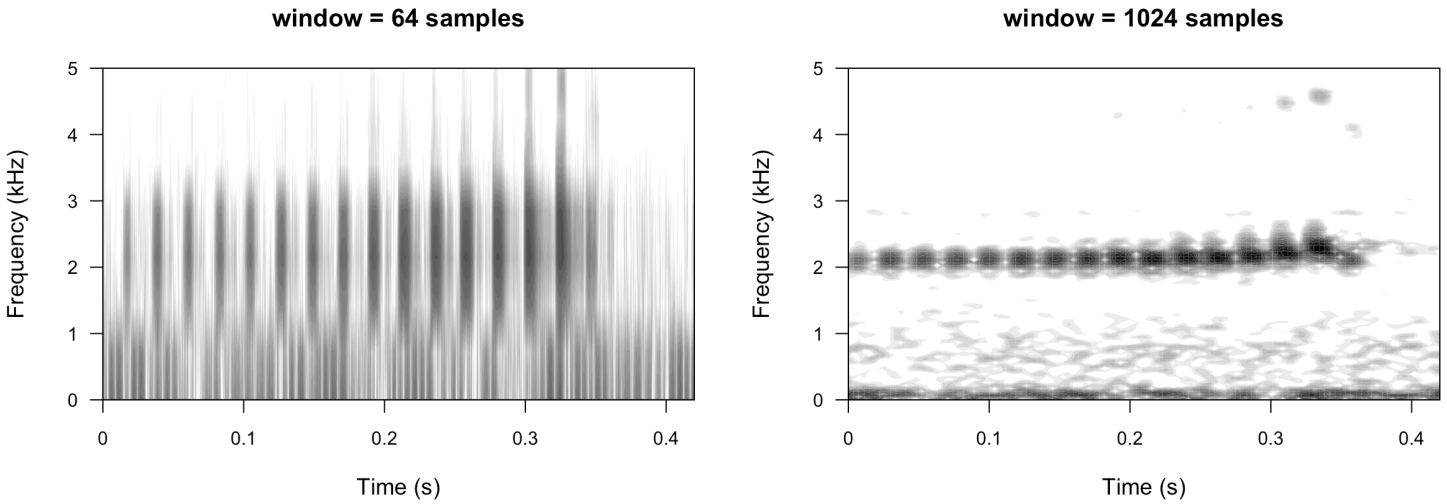
1. Which calls count (the window rule)

- *The first ten consecutive calls with no handling noise and no overlapping frogs.* The first clean run, not the best-looking calls, so two observers land on the same calls
- **Don't skip a call inside the run.** Intervals come from consecutive starts (`lead(start_abs) − start_abs`). A skipped call doubles one interval, and nothing flags it. If a call can't be measured, end the run there

2. Where a call starts and ends (the boundary rule)

- One sentence that names the panel, the threshold, and the tail: *"start = first pulse at least twice the background in the oscillogram; end = end of the last pulse; echo tail excluded"*

Read times from the waveform



- At `wl = 512` and 44.1 kHz each spectrogram column spans **12 ms**. The spectrogram smears every edge sideways; the oscillogram doesn't
- Use the spectrogram to confirm that a blip is the frog, then read the time off the waveform
- **Same display for everyone:** `wl = 512, ovlp = 50`, same `flim`, same zoom width. Zoom with `tlim =`, so the axis stays in window time and there are no offsets to add

Calibrate before you split

New: `10-observer-calibration.qmd` on the share.

1. **Today, as a team (15 min):** pick one recording, write both rules
2. **Each of you, alone:** measure the *same* ten calls into `calibration/selections_<initials>.csv`. No comparing first
3. **Steward:** render. You get each observer's habits (early starts, long ends), plus **repeatability**: the share of the variation that is the frog and not the observer
4. **Together:** look at the worst calls, fix the rules, re-measure if needed

Then **split by male, not by file**, spread each species' males across observers, and never let two people measure the same file in the team folder. The updated template's Step 3 stops if they do.

```
file.copy("/stor/work/FRI_Bio206_TD_Spring2026/10-observer-calibration.qmd", "~/")
```

Today

- **First 15 minutes:** stand-ups, team by team. Errors you couldn't fix go first
- **Next 15:** calibration setup. Choose the recording, write both rules
- **The rest:** work. Calibration selections first, then your own recordings into `selections_*.csv`
- I'll circulate. Bring me the exact error message, not a description of it

Stand-up: three things, two minutes

One person per team:

1. **Does it render from scratch?** Delete outputs, Restart R, Render: yes / no
2. **The error you couldn't fix**, if there is one
3. **What's still done by hand**

Before Monday

Steward

- [ ] Notebook renders from scratch through **Step 5**
- [ ] Monday's fixes in: jitter seed in Step 6, a **Decisions** section, and `sessionInfo()` at the end
- [ ] Any typed number in your text replaced with inline code, placed after Step 5
- [ ] From the updated `team-measurement-template.qmd` on the share, paste the new Step 1 text and the two Step 3 checks into your notebook. Your copy doesn't update itself

Everyone

- [ ] Your ten calibration calls in `calibration/selections_<initials>.csv`, measured alone
- [ ] `selections_<initials>.csv` named exactly, and your Decisions line sent to the steward

Coming up

Due Sun Oct 11, 11:59 PM: the calibration notebook, rendered (steward submits on Canvas for the team).

Monday Oct 12: summary statistics and uncertainty, **led by the peer mentors** (I'm away). Slides are posted in advance. Bring your `by_male` and `by_species` tables and your calibration numbers.

Credits

- *Prepared with Claude Code (Claude Opus 5.5, Anthropic): deck drafting, including the observer-consistency slides*