Lab 2, Act 1

From guess-and-check to systematic search

NRES 746 – name: _____ group: _____

Picking up where we left off

Last week (Act 3) you tried a few guesses for the (a, b) parameters of the Ricker function by hand, recorded each one's SSR in a table, and tried to find a combination that best fit the tadpole data. That approach works, but it only samples a few scattered points out of a huge space of possible parameter combinations – there could easily be a much better combination nobody in the room happened to try.

Today we replace guess-and-check with **grid search**: an algorithm that computes SSR at every combination of a and b across a fine grid, and picks out the best one automatically. This is a preview of something you'll do again in a few weeks – once we replace "sum of squared residuals" with "likelihood".

Before you do anything else: close your laptop and put your phone away. Part 1 and the group check-in that follows are pen-and-paper only.

Part 1: Plan the search (individual, ~15 min)

1a. Propose a search range. Based on last week's guesses, propose a reasonable range to search over for a and for b (order-of-magnitude is fine – you're not committing to anything precise just yet):

a from _____ to _____ b from _____ to _____

Roughly how many (a, b) combinations would that range contain if you checked every 0.05 for a and every 0.01 for b (a rough estimate is fine)? _____

That's a lot of predictions and SSR calculations to run by hand. This is a job for a computer, not a pencil.

1b. Design the algorithm. Working from your range in 1a, write out (in plain English) the algorithm you'd need to check every combination of a and b and find the best one. Don't write R code; describe the procedure the way you'd explain it to someone who has never programmed. Think about: what you need to keep track of as you go, what gets recomputed at every single combination, and how you identify the winner once you've checked them all.

Group check-in: finalize your pseudocode, then choose your path (group, ~10 min)

Compare your pseudocode with your group and agree on a single, correct version.

Your group must now pick **one** of the two paths below for turning this pseudocode into working code. Both are equally valid; discuss and decide as a group before opening any laptops.

- **Path A (AI-assisted):** write a prompt describing your pseudocode and have an AI assistant generate the R code.
- **Path B (self-coded):** write the R code yourselves, using R's documentation and each other, but no AI tools.

Our group is using: Path A (AI-assisted) / Path B (self-coded) *(circle one)*

Laptops open now. Complete only the section below that matches your group's chosen path.

Part 2A: AI-assisted path (laptops, ~15-20 min)

Use the official search range for the rest of today's lab: $a =$ `seq(0.1, 3, by = 0.05)`, $b =$ `seq(0.02, 0.40, by = 0.01)`.

Write a prompt to an AI assistant describing your group's pseudocode and asking it to write R code that: (1) runs the grid search over the ranges above, (2) reports the best-fit (a, b) and its SSR, and (3) produces a heatmap and/or contour plot of the SSR surface across the grid.

Notes on the prompt you used and any changes you had to make to get the AI to do the right thing (don't provide the code it returned – we're interested in your prompt and the results, not the AI's implementation. Also, no need to re-write your pseudocode above- if that worked as a prompt, just say that and move on!):

Results. Best-fit $a =$ _____ $b =$ _____ SSR = _____

Look at the heatmap/contour plot. Is the minimum comfortably inside your search range, or pinned against an edge? (If it's at an edge, that's a sign the range needs to extend further.)

Reflection. This is open ended, but take a minute to reflect on what you just did. How much faster was the AI than you would have been (working alone or with your group)? Do you think you 'missed out' on some important learning by choosing the AI path? Can you think of ways of using AI to boost both productivity and learning?

Part 2B: Self-coded path (laptops, ~15-20 min)

Use the official search range for the rest of today's lab: $a =$ `seq(0.1, 3, by = 0.05)`, $b =$ `seq(0.02, 0.40, by = 0.01)`.

Translate your group's pseudocode into working R code yourselves (R's built-in help, documentation, stackOverflow and other human-generated websites, and your neighbors are fair game; AI tools are not). Run the grid search, find the best-fit (a, b) and its SSR, and produce a heatmap or contour plot of the SSR surface (`image()` or `contour()` both work well).

Results. Best-fit $a =$ _____ $b =$ _____ SSR = _____

Look at the heatmap/contour plot. Is the minimum comfortably inside your search range, or pinned against an edge? (If it's at an edge, that's a sign the range needs to extend further.)

Reflection. What part of turning your pseudocode into working R gave your group the most trouble? How did you resolve it (R documentation, trial and error, asking a neighbor, the TA)?

Quickly plot your best-fit curve over the data with `curve()`. Does it peak where you expected, around 12 mm? If instead it looks like it's just settling near the cluster of low points rather than rising to meet the one high point, that's not a bug in your code – with only five aggregated data points, three of them at or near zero, minimizing raw SSR doesn't always chase a peak. Hang onto that observation; it'll matter in Act 2.

Group discussion and revision (~5 min)

Compare your results with a different group (maybe one that took the *other* path) and compare best-fit results. Since a grid search is deterministic, you should land on the same (or very nearly the same) (a, b) regardless of path – if you didn't, discuss why (different search range, a bug, floating-point differences in how the grid landed on the true optimum).

Feel free to revise any of your answers above based on this discussion (it's fine to cross out and rewrite – I like to see your thinking evolve).

Optionally: please take a minute to record any changes you made and any new insights gained during the group discussion. You can also use this space to reflect on any areas of confusion ("muddiest points") or topics you want to review or dust off going forward.