

The files here correspond to the project described in the paper titled “An optimal-fitness framework for modeling perceptual compression”, authored by Victor Quintanar-Zilinskas. All references below to tables, figures, and paper sections correspond to parts of the paper.

All .m files were generated and run using Matlab, and are contained in the “code” zip file.

Date of last modification (expressed as “_YYYY_MMDD”) is appended to all uploaded files (including this one); for any given file, refer to the most recent version.

c_1.m, c_2.m, c_3ab.m, c_3cd.m, c_4a.m, and c_4b.m respectively generate Fig 1, Fig 2, Figs 3a 3b, Figs 3c and 3d, Fig4a, and Fig4b. Data depicted in Figs 3c and 3d are generated in **c_3cd.m** as described in “Methods: Expected reward distributions, E3”. Data depicted in Fig4 are generated in **c_4a.m and c_4b.m** as described in “Methods: Expected reward computation, E4”.

SuppEqSolutions.nb is a Wolfram Mathematica notebook that serves this project as a workbench for manipulating and simplifying mathematical expressions. Section labels correspond to subsections of the “S1 Appendix” paper section, and the expressions manipulated are presented in order of their appearance in S1.

SuppEqSolutions.pdf is provided for those without access to Wolfram Mathematica.

Simul.m

Overview

This script performs the simulations described in “Methods: Compression strategy competition, E3”. The data generated are saved in .mat files that are labeled with a letter and a number that respectively correspond to the simulation condition and to the simulation replicate. Simulation conditions A-G are as defined in Table 3; conditions H-M respectively correspond to the rows of Table 2.

Explanation of data output

This script simulates in parallel the evolution of three populations of a simple virtual organism, which evolve according to mostly similar rules, save with respect to the percentage of the population that survives (and, reproduces; thus, population size stays constant) from one generation to the next. Thus, its output, a 7-column matrix, reports the evolution of the three populations.

Every row represents a simulated generation. Thus, the values in columns 3, 5, and 7 (are zero) can be ignored; these columns correspond to information used within the simulation of individual generations, but this information need not be saved in a record of population changes from one generation to the next. The values in column 2, 4, and 6 report the number of RM individuals (out of 100; the rest are BH) for populations whose generation-to-generation survival rate are, respectively, 90%, 50%, and 20%.

The values in column 1 consist of one or more sequences of integers that start with a value in the 15-60 range and decrease to 1. Decreasing sequences come about as follows: every time the script is run, a fixed number of generations are simulated, and thus for a particular run of the script, the first column can be interpreted as a “generation countdown” indicator. Meanwhile, chains of decreasing sequences are formed as follows: each sequence corresponds to a script run, and when a run is completed, its results are appended to the record of previously-simulated generations of that particular triplet of populations.

Instructions

Before the script is run, the user should ensure that parameters “p” (p(HS) in Table 3) and “life” (Lifetime Decisions in Tables 2 and 3) are set to the desired values. Also, “envi” should be set to “1” or “2” according to whether E3A or E3B is being simulated.

When this script is run in Matlab, the user is asked “fresh? 0, 1, 2, 3, 4”

If a simulation of a triplet of populations is ongoing, input “0”

If a simulation that was paused and saved is being restarted, input “1” (check line 37 to ensure that the desired file is being reloaded, and check the folder to ensure that the file is present)

If a simulation is being initialized, with the initial frequency of RM being 80% (50%, 20%), input “2” (“3”, “4”)

When a run of the script is complete, its final output indicates the status of the populations being simulated. Typically, the simulation of the populations can be regarded as complete when the values in columns 2, 4, and 6 have all converged to either 0 or 100. Typically, when the simulation is deemed complete, the record thereof (the matrix “savethis”) is saved; otherwise, the simulation is continued.

SimulationData.zip contains .mat (described below, in “X#.mat”) files generated using Simul.m.

X#.mat (“X” represents a single letter)

The files of this format are contained within SimulationData.zip.

In brief: when viewing the data contained, odd columns can be ignored, file rows each represent a simulated generation, the numbers in the even columns represent the number of organisms carrying the allele for the RM perceptual strategy, and column 2 (4, 6) corresponds to a population survival rate of 90% (50%, 20%). The contents of the odd columns are described above, in “Simul.m: Explanation of data output”.

qlook.m extracts and organizes data from the .mat files and performs the statistical tests reported in Tables 2 and 4.

To use this program, put it in the same folder as “SimulationData.zip”, unzip the latter, and delete the “_YYYY_MMDD” from the name of the unzipped folder. Then, when the program is run, it will ask the user “which output?”

“1” returns the first and last rows of the matrices saved in the .mat files with prefix A-G, which were generated in E3B. Data corresponding to the same simulation condition are stacked vertically; thus, one 20x7 matrix is emitted for each of the 7 simulation conditions.

“2” produces a filtered version of “1”: from the end of every simulation, it displays each population’s count of RM organisms. Every triplet of columns corresponds to a simulation condition, and within each triplet the columns correspond to distinct survival rates (90%, 50%, and 20%) while the rows correspond to distinct simulation replicates.

“3” returns a matrix called x2d, which puts together the third columns of each triplet in the output generated by input “2”, and dOut, which contains the p-values of the statistical tests indicated in Table 4.

“4” is analogous to “1”, but for .mat files with prefix H-M, generated in E3A.

“5” returns matrix dOut; the columns (left to right) correspond to simulation conditions H-M, row 1 reports the number of RM fixations, and row 2 reports the probability (if $p(\text{RM fixation})=.5$) of a value as extreme as the result in row 1 (binomial test).