**Computational Exercise 3.1 – Simulating the Evolution and Stability of the G-matrix**

© 2017 Adam G. Jones & Stevan J. Arnold

The goal of this exercise is to become familiar with a G-matrix simulation program and build some intuition regarding the evolution of the G-matrix. For this Exercise you will work in a small group (3 or 4 people). Ensure that your group has at least one person with a Windows-based computer.

Begin by downloading the simulator from the following Github site:

<https://github.com/JonesLabTAMU/Gmatrix>

You will be able to download the simulator executable as well as a pdf file containing instructions. The program will run on any machine running a modern version of Windows (anything since Windows 98). However, it will not run on a Mac or Linux machine unless you are using Wine to emulate a Windows environment (and even then you may observe strange behavior).

The basic idea of the simulator is to build intuition related to the evolution and stability of the G-matrix. While the simulator can generate numeric results, we’ll focus primarily on the graphical representation of the G-matrix. For this exercise, we will investigate two important phenomena related to the evolution of the bivariate mean and the G-matrix. The first situation is the approach to a stationary optimum that is displaced from the population mean, and the second is evolution in response to a continuously moving optimum. Perform the following steps:

**1. Investigate the Approach to a New Optimum.**

**\*Each time you run the simulation, look for errors in the message bar at the bottom of the main window. If you get an error that starts with “Error choosing…”, then the population could not sustain itself, the simulation terminated, and the results are not reliable. You will have to change your parameter values to be more amenable to population survival. The most common problem is that the optimum is moving too fast or is too far from the population mean.**

**Step 1:** To see how the G-matrix constrains the approach to a new optimum, change the following parameter values from their defaults (leave everything else alone for now):

No. Females in Population: 512

Strength of selection (w2): 99 99 [change both values]

No. Complete Replications: 1

Str. sel’n after peak shift (w2): 99 99

Trait optima shift (real number) 10 0

Move peak at generation: 200

Var. in Optimum Position (real no.): 0 0

In the lower left corner **select the button next to “Move Peak Once”**. Press “Okay” to save your parameter changes. Run the simulation and see what happens. Did the population mean follow a straight or curved path to the optimum?

Curved path to the optimum. Moves rapidly along eigenvector 1 (major axis of G-matrix) then slowly turns to move mostly along eigenvector 2.

**Step 2:** Change the mutational correlation to see how different values affect the approach to the optimum. Try runs with the value of “Mut. Corr. Traits 1, 2 (hundredths)” set to 90, 75, 50, 25 and 0. What effect does a smaller mutational correlation have on the approach to the optimum?

Lower mutational correlation – moves along axis 2 sooner, less speed difference between axes ~ a more direct path. And there may be less rotation stability with low mutational correlation (more wobble). The wobble is more severe when the bivariate trait mean (G-matrix center) is closer to the optimum.

**Step 3:** Move the location of the new optimum. Set the mutational correlation back to 75. First try moving the optimum up and to the right, by setting “Trait optima shift (real number)” to 7.5 and 7.5 [if the values are too large, the population will not be able to replace itself and the simulation will stop with an error]. What path does the population take toward the optimum? What do you think will happen if you set the position of the optimum to 7.5 and -7.5 (i.e., down and to the right)? Try it and see if your intuition was correct. For a positive mutational correlation of 0.75 like you used here, how does the approach to the optimum differ for the three cases investigated here (i.e., up and to the right; to the right; and down and to the right)?

The optimum shift of (7.5, 7.5) is roughly parallel the major eigenvector of the G-matrix: moves rapidly and directly. (7.5, -7.5) parallel minor eigenvector. Moves directly but slowly.

**2. Examine and Understand the Flying Kite Effect.**

**Step 1:** Close the simulator program and restart it to get back to default parameter values. Experience evolution in response to a moving optimum by using the following parameter values:

No. Females in Population: 512

Strength of selection (w2): 99 99 [change both values]

No. Complete Replications: 1

Mut. Corr. Traits 1, 2 (hundredths) 0

Str. sel’n after peak shift (w2): 99 99

Trait optima shift (real number) 0.02 0

Var. in Optimum Position (real no.): 0 0

Run the simulation and see how the population mean tracks the moving optimum.

G-matrix is circular. G-matrix directly tracks shifting optimum but with lots of rotation/ spin. Highly unstable wrt angle.

Rotate G-matrix: reverse the sign of the mutational correlation

**Step 2:** Change “Mut. Corr. Traits 1, 2 (hundredths)” to 75, which results in a mutational correlation of 0.75. How does this non-zero mutational correlation change the response to the moving optimum?

Almost no rotation, mostly follows along major eigenvector/ Gmatrix axis. Moves away from optimum (kite effect) on secondary axis.

**Step 3:** Investigate how the selection parameters affect the flying kite. Leave the mutational correlation at 0.75 and try altering the selection parameters. What effect do you think a positive correlational selection will have on the flying kite’s behavior? Set “Sel. Corr. Traits 1, 2 (hundredths)” and “Sel. Corr. after sel’n (hundredths)” to 75, for example, to see. What effect might negative correlational selection have? What about stronger or weaker selection (try changing the four strength of selection parameters to values like 9 (very strong selection) or 999 (very weak selection) to see what happens. Think about why the G-matrix and response to selection is behaving this way.

Pos corr selection – G-matrix follows optimum more closely (less kite effect)

Negative – kite effect gets worse over time

Strong selection : bivariate trait mean closely tracks optimum. G-matrix is very stable

Weak selection: barely tracks optimum, mostly along primary axis, not much rotation

**3. Challenge Round.**

**Challenge 1:** Start with the parameter values from problem 1, step 1, above. That is, you want a single shift of the optimum. Now set “Mut. Corr. Traits 1, 2 (hundredths)” to zero and run the simulation. The population mean should move directly toward the bivariate optimum. Without changing the mutational correlation (i.e., leave “Mut. Corr. Traits 1, 2 (hundredths)” with a value of zero), come up with a set of parameter values under which the population mean takes a curved path to the bivariate optimum. Feel free especially to move the position of the optimum (“Trait optima shift (real number)”), change the mutational variances, and alter the selection parameters.

**From Problem 1, Step 1:**

No. Females in Population: 512

Strength of selection (w2): 99 99 [change both values]

No. Complete Replications: 1

Str. sel’n after peak shift (w2): 99 99

Trait optima shift (real number) 10 0

Move peak at generation: 200

Var. in Optimum Position (real no.): 0 0

Mut. Corr Traits 1,2 (hundredths) 75

Cause of curved trajectories: Gmax biases direction of evolution initially. Beta points toward the optimum the whole time, but as you move along Gmax, Beta moves to be more parallel to Gmin. So path curves to parallel Gmin.

Challenge 1:

* Trait optima shift must not parallel the major or minor axis of the G-matrix - (10,0) works

**Challenge 2:** If time permits, investigate one other question that arose during your simulation runs.

**At the end of the hands-on period, we will have a short discussion of each group’s discoveries, so be prepared to discuss how the G-matrix evolved and why.**