Extending the Butterfly Model

Isaiah Hollars

September 25, 2022

1 Introduction

We wish to extend the butterfly model to answer the question: how well do the butterflies clump and how random should their flights be in order to maximize this clumping?

A good place to start is by answering **exercise 4** from the book. Here, we just alter the q-value (which describes the probability of uphill flight) and observe which q-values seem best for clumping. Upon trying lots of values, the q-values around 0.40 looked to be giving the best clumping results. See figure 1.

2 Methods

In order to answer the main question, we need to formulate a way to measure clumping. We implemented two different metrics that the book authors recommend. The first metric we used is clumping-radius. Clumping radius reports the number of butterflies in a radius of 5 tiles averaged over all butterflies. The second metric we used is called clumping-patch. This metric just counts the number of patches that have at least one butterfly on them. Both of these metrics get calculated at the end of a 1000 tick simulation. See figure 2.

Now that we have metrics to measure the amount of clumping, we can conduct a experimental investigation. To accomplish this, we leverage the Behavior Space tool in NetLogo. This tool allows us to run hundreds of experiments back to back. In general, our experiments swept out the q-values from 0-1 in 0.01 increments. We also did 5 separate runs at each data value for a total of 500 data points. We ran this experiment on both realistic landscapes and on the artificial landscape. The experiments in NetLogo report the both clumping-radius and clumping-patch for a given q-value. It should be noted that we are looking to maximize the clumping-radius, since that refers to a measure of the number of butterflies. On the other hand, we seek to minimize the clumping-patch which is smaller when the butterflies are on clumped up and take up less patches. The data for landscape 1, landscape 2, and the artificial landscape are in figures 3, 4, and 5 respectively.

Figure 1: observation for question 4

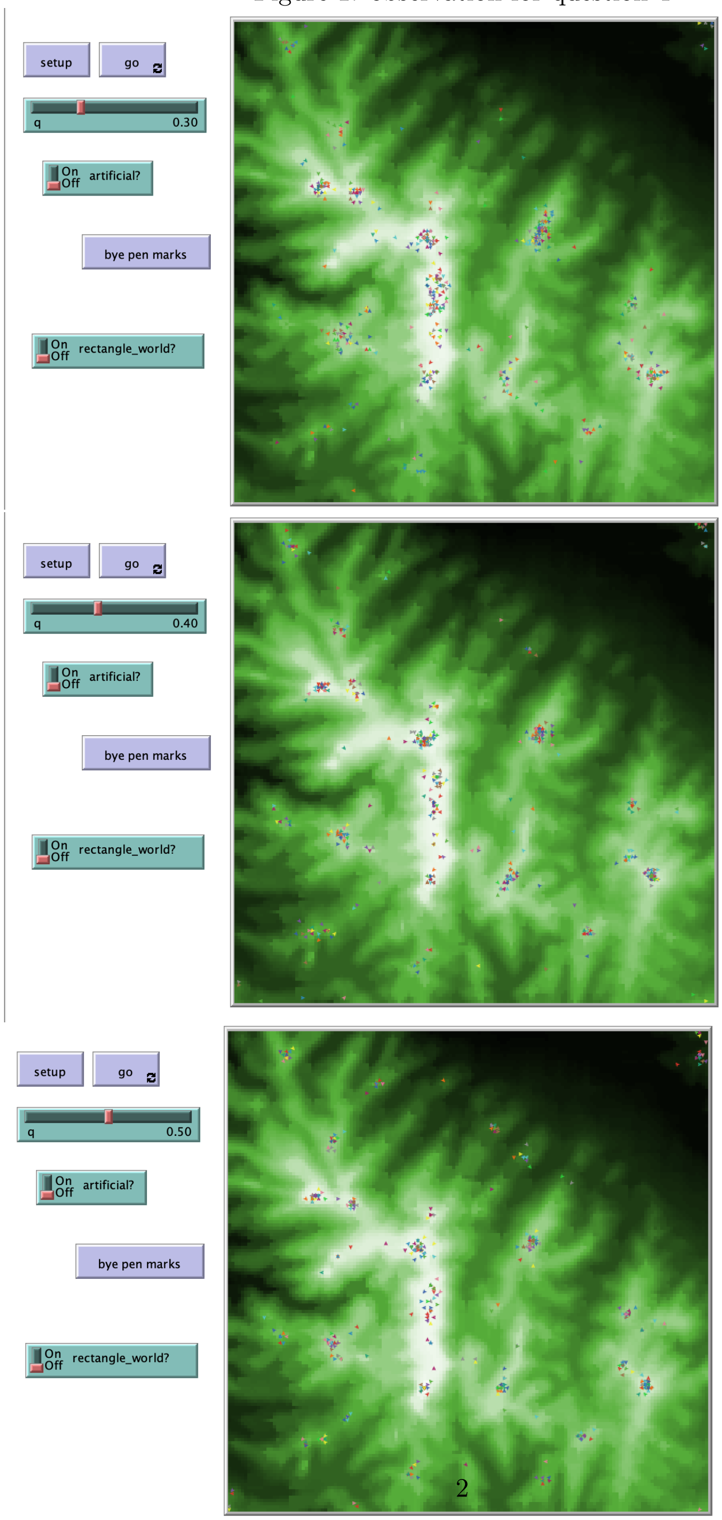
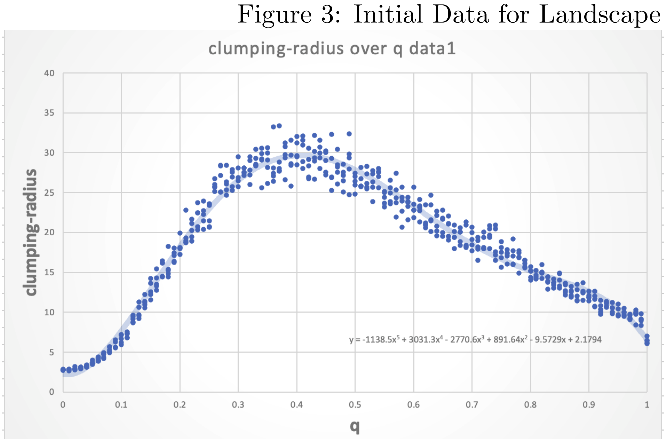


Figure 2: code snippet question 5

```
to-report clumping-patch
  ; how many patches have butterflies on them?
  ; count the patches where the count of the turtles on the patch > 0
  report count patches with [count turtles-here > 0 ]
end

to-report clumping-radius
  ; report the average number of butterflies in a fixed radius
  ; that fixed radius is defined by the slider radius-for-clumping-metric
  report mean [count turtles in-radius radius-for-clumping-metric ] of turtles
end
```

Figure 3: Initial Data for Landscape 1



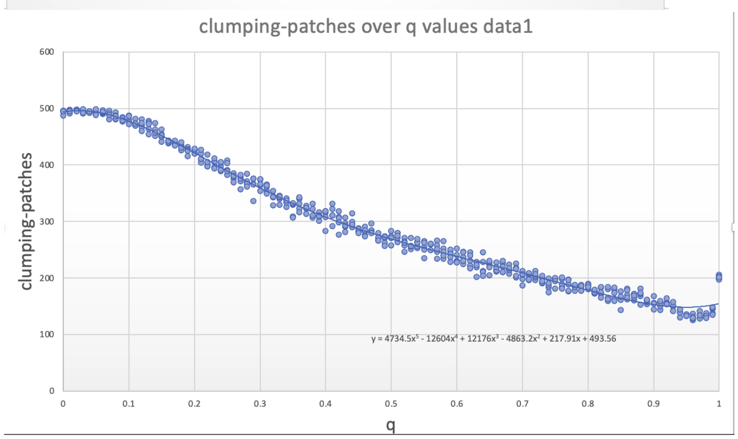


Figure 4: Initial Data for Landscape 2

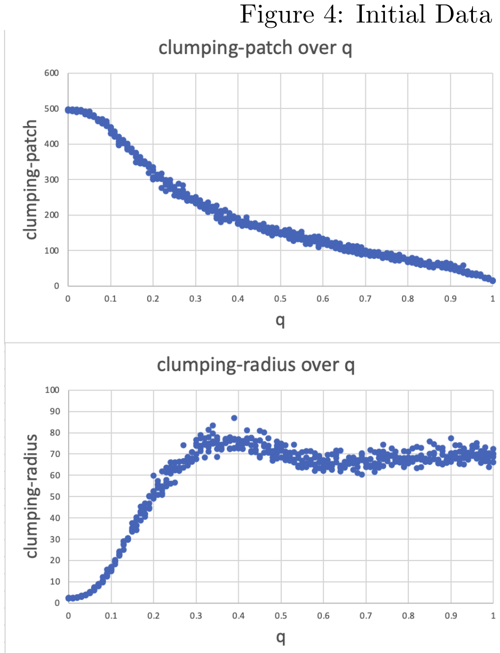


Figure 5: Initial Data for Artificial Landscape

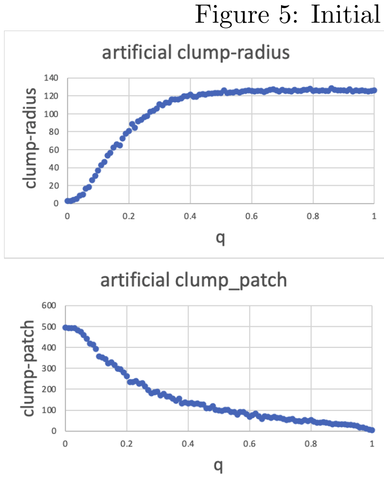
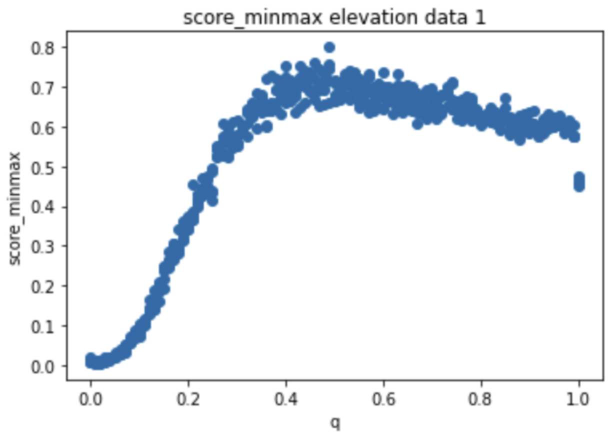
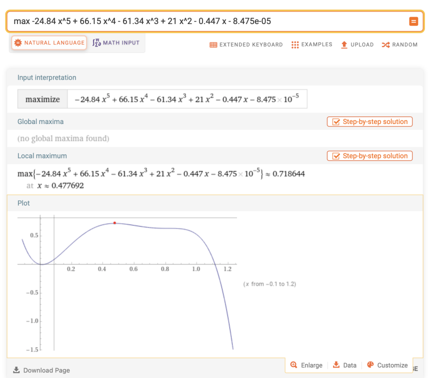


Figure 6: Landscape 1 min-max combined metric

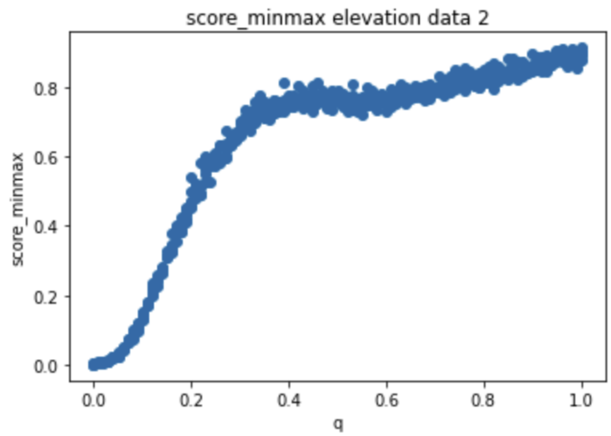




We then employ feature engineering so that we can combine our 2 metrics into one. We used both min-max normalization and z-score normalization on our metrics. Both of these normalization strategies ended up producing very similar results, so we will just stick with the min-max for the sake of this report. See this wiki for a brief overview of different types of feature scaling, including min-max. The min-max implemented scaled both metrics into the range of 0-1. Once we scale clumping-radius and clumping-patch with min-max, we can combine the two by taking an average. This new metric is called score. Before we did this, we technically had to invert clumping-patch before we combined it (so that a value closer to 1 is good for both metrics). We do this procedure for both landscape 1 and landscape 2 (the artificial landscape is not too interesting). See figures 6 and 7. We also do a 5th degree polynomial regression to try and nail down the maxima of this min-max score.

Upon observing that the results from landscape 1 and landscape 2 were quite different, we decided to do one more combination. In figure 8, we average the min-max scores we got from the different landscapes to create one metric.

Figure 7: Landscape 2 min-max combined metric



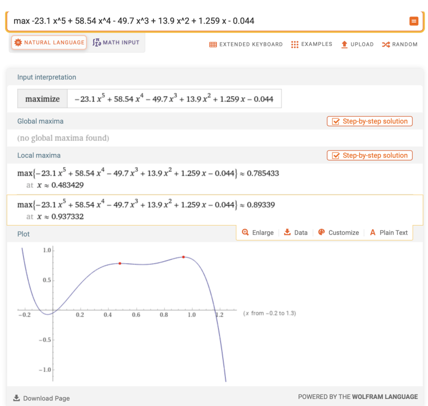
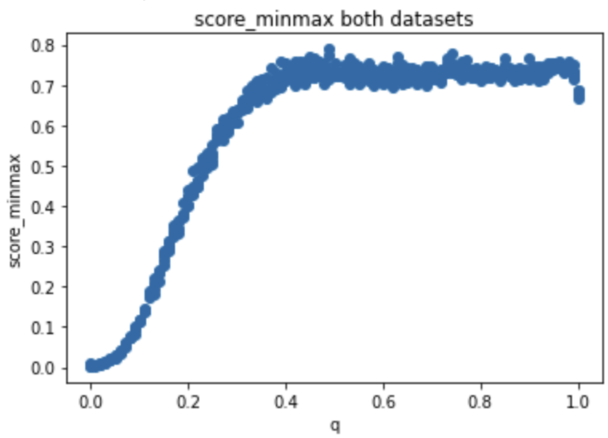
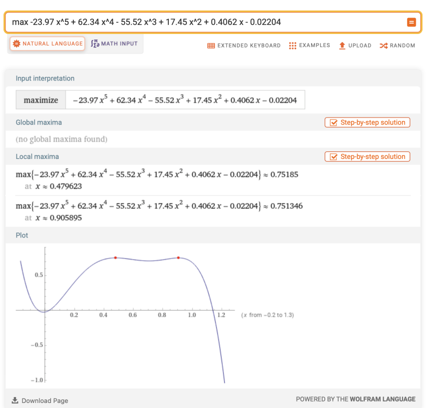


Figure 8: Both Landscapes min-max combined metric





3 Analysis and Answers

With all these plots at our disposal, we can now answer many of the questions we might have had. Firstly, the artificial landscape was not interesting because a higher q-value was always better. The artificial landscape has very few local maxima, so the butterflies are not punished for probability-1 uphill travel like they should be. By looking at figure 6, we see that on Landscape 1, butterflies did best at a q-value around 0.48. We also see a notable decline after this point. The analysis for landscape 2 (figure 7) has a global max at $q = 1$ when $0 \leq q \leq 1$. It is interesting to note that the polynomial fit to the curve also has a local maximum at $q = 0.48$. Landscape 2 was also fairly smooth and didn't contain many local maxima similar to the artificial landscape. When we combine these the min-max scores from the landscapes in figure 8, we don't see anything super surprising. The score is maximized around $q = 0.48$ and at q-values close to 1.

4 Conclusion

A simulated butterfly made to chose the optimal q-value will be hit with trade-offs. The lower q-values allow them to explore more and hopefully end up on the largest hill in the long run. The downfall is that they aren't clumped as tightly on the hill tops. Large $q-$-values can cause butterflies not to explore enough, but it makes sure to clump them tight with their own hill-mates.

The fact that $q = 0.48$ was the best choice for landscape 1 and was also a local maximum q-value for landscape 2 is telling. Moving forward, it might be a better idea to try a larger radius for clumping-radius. In theory, this should favor lower q-values. Even though q-values close to 1 performed well on average, they seem wrong; i.e., they don't seem to replicate the behavior of real butterflies. Since real butterflies don't behave like this, this result is likely due to the way we have measured clumping or in how we have designed the landscapes. The q-value of 0.48 is much more interesting, because it was a non-obvious emergent outcome of the model, which also resembles the randomish flight of butterflies. An optimal q-value of 0.48 makes you feel good about your model, while an optimal q-value of 1.0 makes you cringe.

5 Appendix