Simulating an Artificial Ecosystem

Eleanor Camp | Mathematical Modeling | Advent 2024

Problem

- Simulate an artificial ecosystem in a 100x100 meter plot
- Enough sunlight is provided for 100 new plants to grow every season, randomly distributed
- Develop an eater that can move, eat, and reproduce
  - Eaters have a maximum energy level, gain energy from eating plants have energy costs associated with moving and reproducing
  - Eaters have 4 genes that code for movement and reproduction

Methods:

Simulation Setup

The simulation was conducted in a 100x100-meter virtual environment. The ecosystem consisted of two entities:

- **Plants**: Stationary resources that regrow periodically.
- **Eaters**: Mobile organisms with genetic traits (intelligence, strength, reproduction interest, and fitness).

At the start of the simulation, 100 plants and 50 eaters were placed randomly in the environment

Seasonal Simulation

Each simulation season followed these steps:

2. **Plant Growth**:
   Plants regrow in random locations after being consumed.
3. **Eater Actions:**
   - Eaters move randomly based on their movement gene.
   - They consume nearby plants to gain energy.
   - Interaction between male and female eaters allows reproduction if the male's fitness meets or exceeds the female's threshold.
   - Aging reduces eater fitness and increases the likelihood of death.
4. **Population Updates**:
   Dead eaters are removed, and offspring replace them if conditions allow.

Seasons were simulated sequentially for 10 iterations or until the eater population became extinct.

Data Collection

At the end of each season the following stats are collected:

- **Population Metrics**: The number of eaters and plants remaining.
- **Gene Frequencies**: The distribution of genetic traits across the eater population.
- **Energy Levels**: Energy distribution within the eater population.

Data was stored using a DataContainer class for analysis.

Key Tools

- **Programming Language**: Python.
- **Libraries:** NumPy
- **Visualization**: Population trends and genetic data were visualized with Matplotlib.

Results:

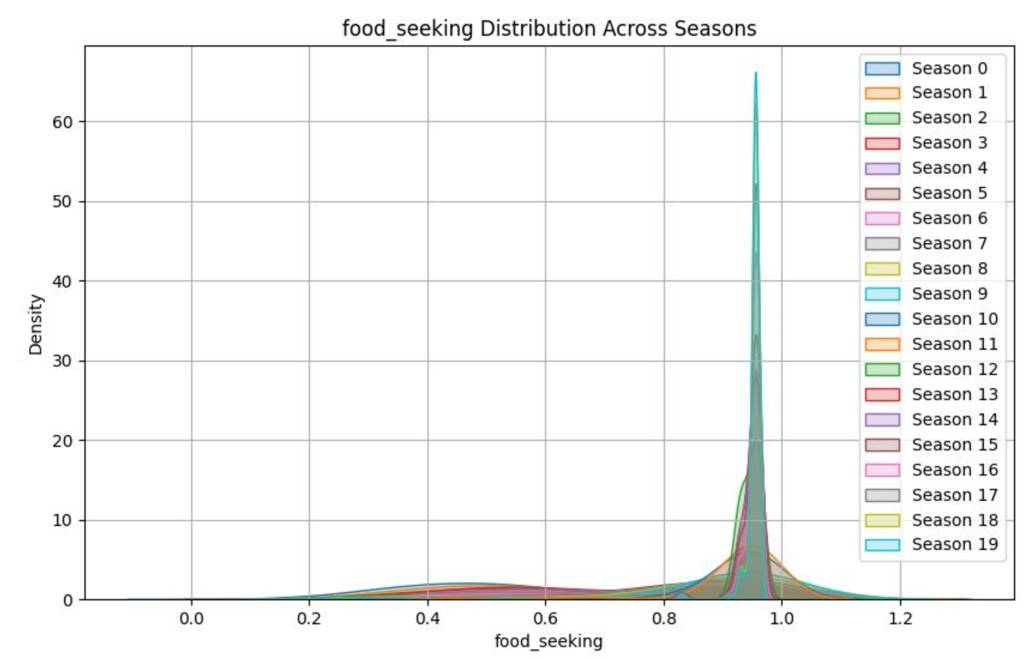

Figure 1. Kernel Density Estimate (KDE) of the distribution of the food_distribution variable across 20 simulated seasons

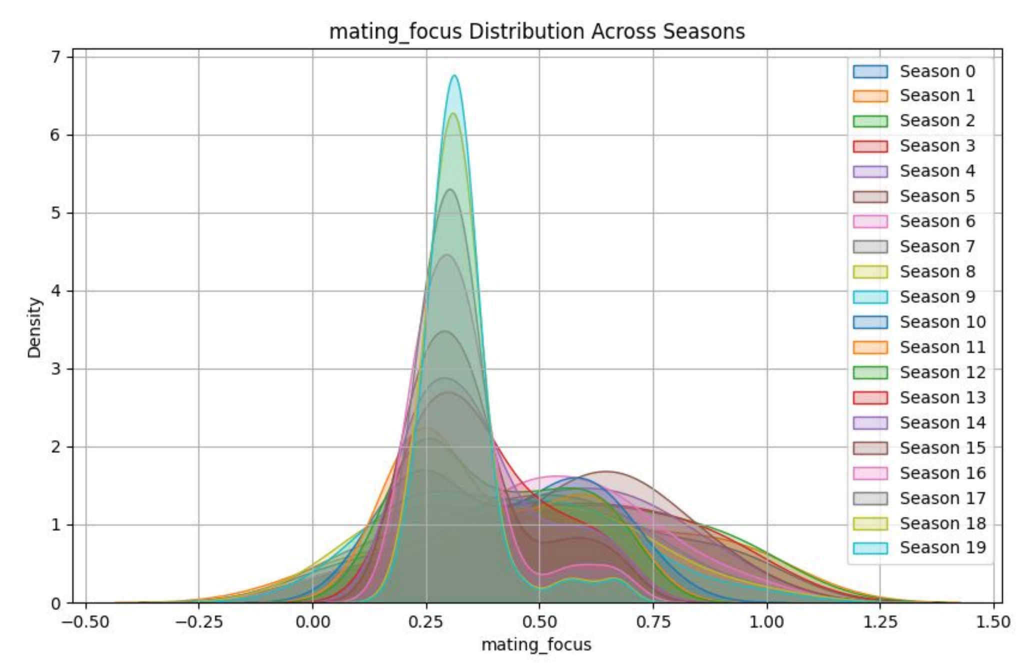

Figure 2. KDE of the distribution of the mating_focus variable across 20 simulated seasons

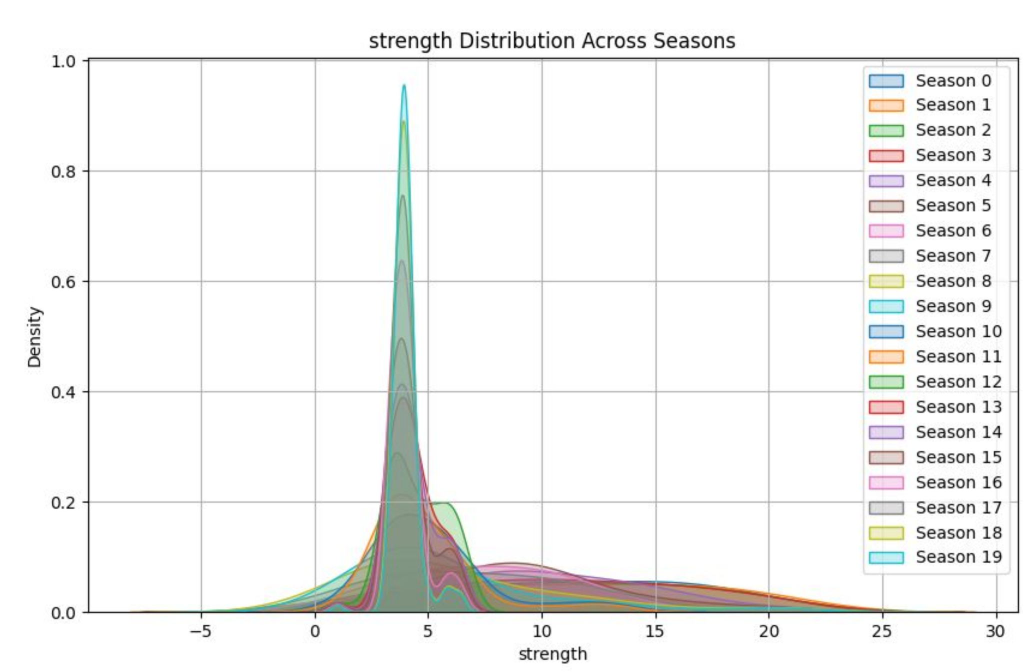

Figure 3. KDE of the distribution of the strength variable across 20 simulated seasons

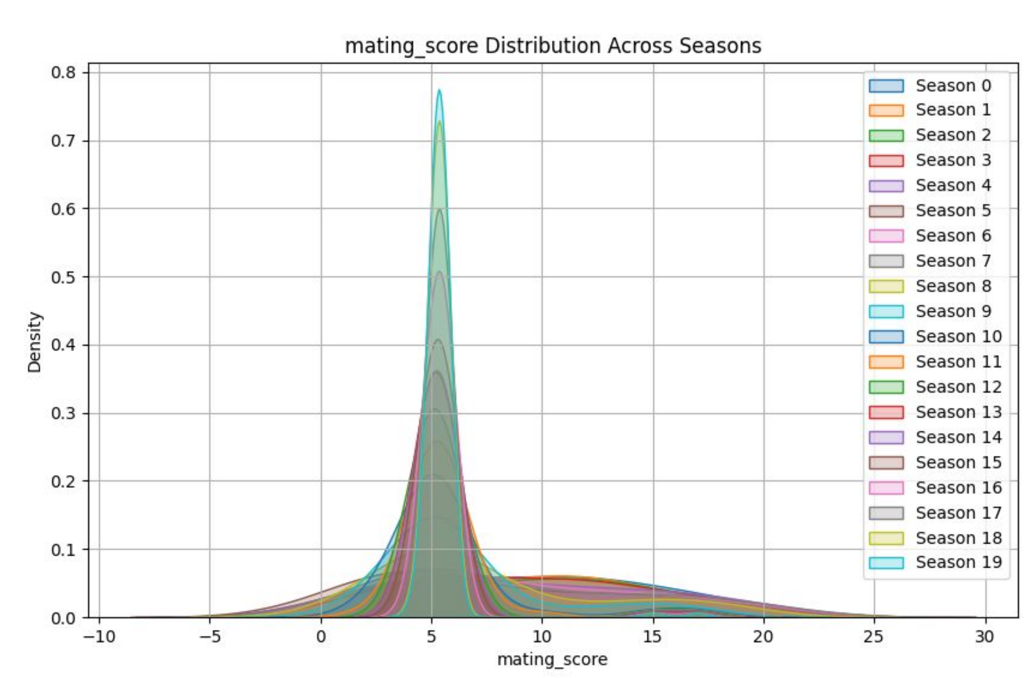

Figure 4. KDE of the distribution of the mating_score variable across 20 simulated seasons

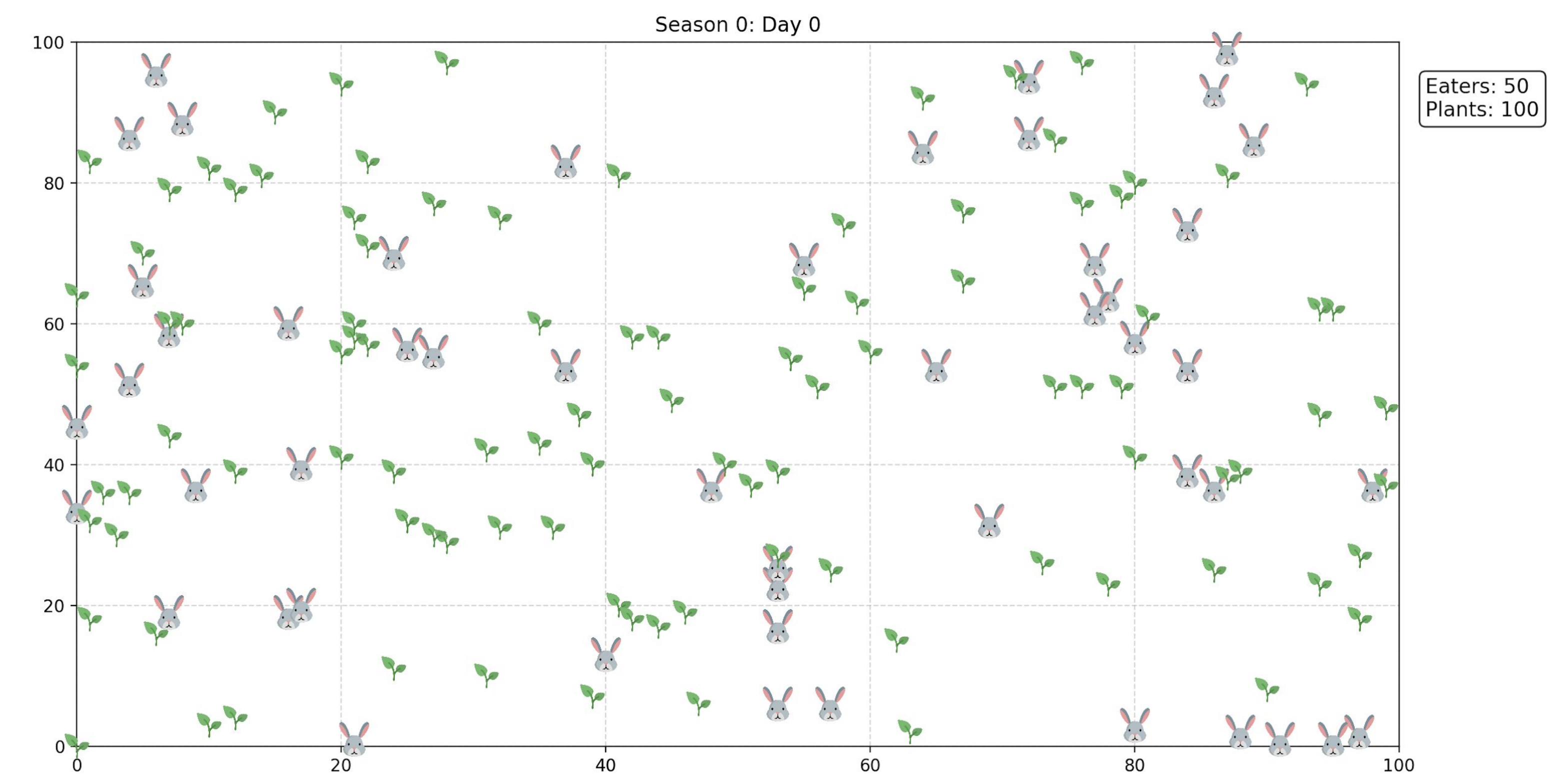

Figure 5. Initial setup of 50 random eaters and 100 random plants

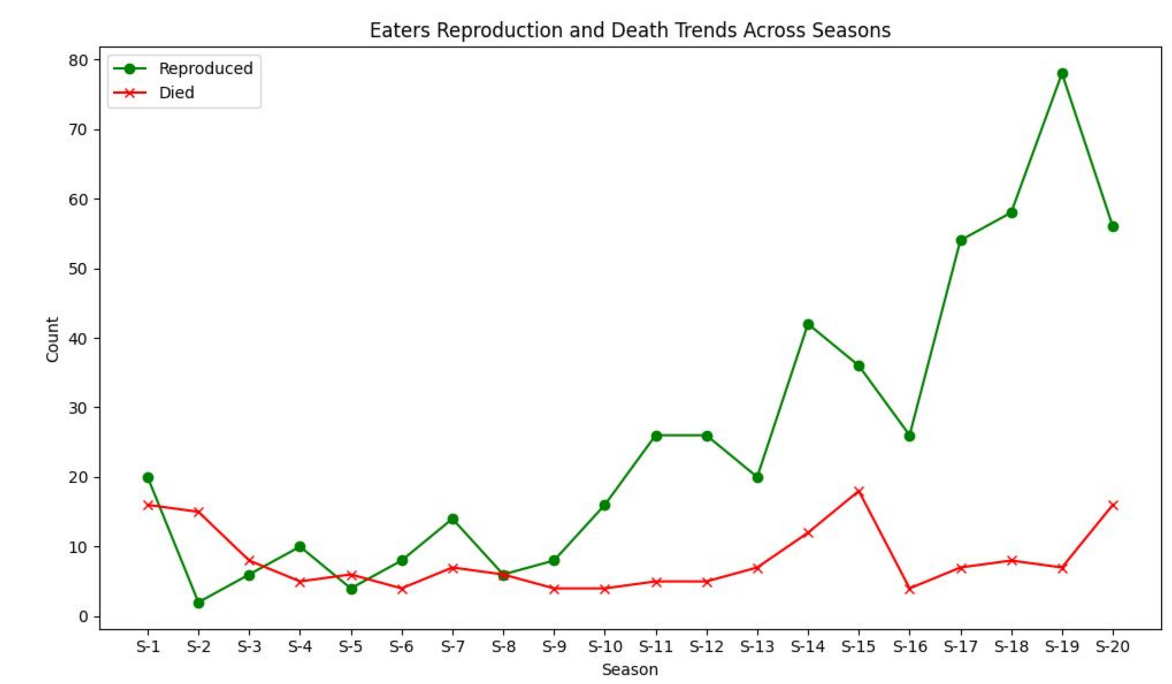

Figure 6. Line plot showing the number of eaters that reproduced and died in each season

Conclusions

- The food_seeking variable was the most important gene passed on to the children eaters, as it seemed to converge in Figure 1
- Eaters can be *too* focused on finding a mate, potentially causing issues like wasted energy (Figure 2)
- Having high values for strength or mating_score is not important in this ecosystem (Figure 3 and Figure 4)
- Many of these values change depending on starting values, such as the initial energy an eater is born with, or how many eaters are present from the start of the simulation

Future Work

Improved Genetic Algorithm – The current algorithm just takes the parent's mean value for each gene to pass onto the child. In the future it would be ideal to introduce more mutation and crossover potential
Environmental Factors – The environment remains the same throughout the simulation. Adding functionality for weather or temperature changes would make the simulation more effective
Social Structure – Adding some kind of social nature to the organisms would better reflect real world ecosystems

Citations

Shonkwiler, R. W., & Mendivil, F. (2009). Explorations in Monte Carlo Methods. Springer New York.