Predator-Prey Simulation Project Report

Your Name

December 3, 2024

Contents

Abstract

This report presents a detailed overview of a Predator-Prey Simulation project implemented using Python. The simulation models the dynamics of a predator-prey ecosystem using a grid-based environment, allowing for interactive visualization and analysis of population changes over time. The project includes both an interactive simulation with a graphical interface and a script for generating phase diagrams based on multiple simulation runs.

1 Introduction

The predator-prey relationship is a fundamental concept in ecology, illustrating how populations of two species interact over time. This project aims to simulate these dynamics using computational methods to provide insights into how initial conditions affect long-term outcomes such as extinction or coexistence.

2 Simulation Overview

2.1 Interactive Simulation

The interactive simulation allows users to:

- Adjust initial populations of prey and predators.
- Control simulation speed.
- Observe real-time interactions on a grid.
- View live graphs of population changes.

2.2 Phase Diagram Generation

A separate script runs multiple simulations across varying initial conditions to generate a phase diagram. This diagram visually represents regions where different outcomes occur, such as:

- All prey dying.
- All predators dying.
- Long-term coexistence.

3 Implementation Details

3.1 Programming Language and Libraries

The project is implemented in Python 3 and utilizes the following libraries:

- **Pygame**: For the graphical user interface and visualization.

- **NumPy**: For numerical computations.
- **Matplotlib**: For plotting graphs and phase diagrams.
- **Tqdm**: For displaying progress bars in the terminal.

3.2 Code Structure

The codebase consists of two main scripts:

1. `predator_prey_simulation.py`: Contains the interactive simulation.
2. `predator_prey_phase_diagram.py`: Generates the phase diagram by running multiple simulations.

3.3 Key Components

- **Agent Classes**: Defines the behaviors of prey and predator entities.
- **UI Elements**: Custom classes for buttons and sliders in the simulation interface.
- **Simulation Logic**: Contains the rules for movement, reproduction, and energy dynamics.
- **Data Visualization**: Live plotting of population changes and generation of phase diagrams.

4 Simulation Results

4.1 Interactive Simulation Output

Upon running the interactive simulation, users can observe:

- Predators and prey moving on the grid.
- Predators displaying their energy levels.
- A brief "+Energy Gained" message when predators consume prey.
- A live graph updating population counts over time.

simulation_screenshot.png

Figure 1: Screenshot of the Interactive Simulation Interface

(Replace "simulation_screenshot.png" with the filename of your actual screenshot.)

4.2 Phase Diagram Analysis

The phase diagram generated illustrates the outcomes based on different initial populations:

phase_diagram.png

Figure 2: Phase Diagram of Predator-Prey Simulation

(Replace "phase_diagram.png" with the filename of your actual phase diagram image.)

4.2.1 Interpretation of Results

The diagram displays:

- **Red Region**: Scenarios where all prey died.
- **Blue Region**: Scenarios where all predators died.
- **Green Region**: Scenarios with long-term coexistence.

Contour lines highlight the boundaries between different outcome regions, indicating critical thresholds in initial population values.

5 Conclusion

The Predator-Prey Simulation project successfully demonstrates the impact of initial conditions on ecosystem dynamics. The interactive simulation provides valuable insights into predator-prey interactions, while the phase diagram offers a comprehensive overview of possible outcomes. This project serves as a foundation for further exploration and enhancement of ecological models.

References

- Lotka, A. J. (1925). *Elements of Physical Biology.* Williams and Wilkins.
- Volterra, V. (1926). Variations and fluctuations of the number of individuals in animal species living together. *Journal fur die Reine und Angewandte Mathematik*, 163, 119–134.