Simulating Population Dynamics of Earthworms Using Markov Chains

Your Name

Simulating Population Dynamics of Earthworms Using Markov Chains

Introduction

Simulating population dynamics is crucial for understanding the adaptive capacity of earthworm populations under environmental pressures. This chapter employs a Markov chain approach to model the transitions between fitness categories and simulate fitness distributions over time. By incorporating environmental variables, the simulation captures the effects of habitat variations on earthworm populations.

Markov Chain Model

States and Transition Probabilities

Each earthworm population is categorized into fitness states, denoted as $S_1, S_2, \ldots, S_n$. The states represent discrete fitness levels, where higher states indicate greater fitness. Transitions between states are governed by a Markov chain with: 1. **State Transition Matrix (P)**:

$$P = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{bmatrix},$$

where p_{ij} is the probability of transitioning from state S_i to state S_j.

2. **Fitness Transitions**:
   - Transitions depend on environmental factors (e.g., soil pH, moisture, temperature).
   - Transition probabilities are modified by a fitness function:

   $$p_{ij}(t) = p_{ij,0} \cdot F_{\text{env}}(t),$$

   where $p_{ij,0}$ is the baseline transition probability, and $F_{\text{env}}(t)$ reflects environmental fitness.

Initial Population Distribution

The initial population is distributed across fitness states as:

$$\mathbf{N}(0) = \begin{bmatrix} N_1(0) \\ N_2(0) \\ \vdots \\ N_n(0) \end{bmatrix},$$

where $N_i(0)$ is the number of individuals in state S_i at time $t = 0$.

Population Dynamics

The population distribution at time t evolves as:

$$\mathbf{N}(t+1) = P \cdot \mathbf{N}(t).$$

Simulation Framework

Defining Environmental Fitness Function

The environmental fitness function $F_{\text{env}}(t)$ modifies transition probabilities:

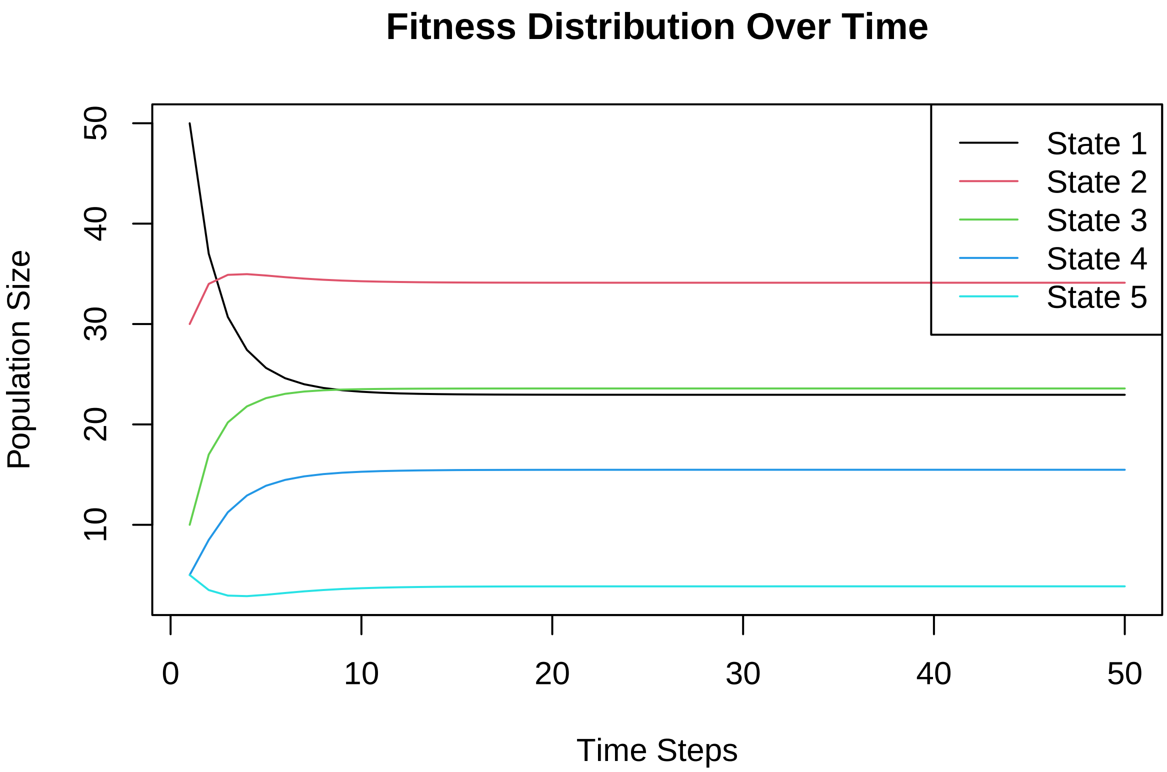
$$F_{\text{env}}(t) = F_{\text{pH}} \cdot F_M \cdot F_T,$$

where: - F_{pH}, F_M, and F_T represent fitness contributions from pH, moisture, and temperature, respectively.

Numerical Simulation

```
# Parameters
n_states <- 5  # Number of fitness states
n_steps <- 50  # Number of time steps
initial_population <- c(50, 30, 10, 5, 5)  # Initial distribution across states

# Baseline transition matrix
P <- matrix(c(
  0.6, 0.3, 0.1, 0.0, 0.0,
  0.2, 0.5, 0.2, 0.1, 0.0,
  0.1, 0.3, 0.4, 0.2, 0.0,
  0.0, 0.2, 0.3, 0.4, 0.1,
  0.0, 0.0, 0.1, 0.3, 0.6
), nrow = n_states, byrow = TRUE)

# Fitness modifiers (environmental effects)
environmental_fitness <- function(t) {
  # Example: Sinusoidal temperature effect
  temp_effect <- exp(-0.1 * (sin(2 * pi * t / n_steps) - 1)^2)
  moisture_effect <- exp(-0.05 * (runif(1, 0.8, 1.2) - 1)^2)
  pH_effect <- exp(-0.03 * (runif(1, 0.9, 1.1) - 1)^2)
  temp_effect * moisture_effect * pH_effect
}

# Simulate population dynamics
population <- matrix(0, nrow = n_steps, ncol = n_states)
population[1, ] <- initial_population

for (t in 2:n_steps) {
  fitness_mod <- environmental_fitness(t)
  modified_P <- P * fitness_mod
  modified_P <- sweep(modified_P, 1, rowSums(modified_P), FUN = "/")  # Normalize rows
  population[t, ] <- population[t - 1, ] %*% modified_P
}

# Visualization
matplot(1:n_steps, population, type = "l", lty = 1, col = 1:n_states,
        xlab = "Time Steps", ylab = "Population Size",
```

```
        main = "Fitness Distribution Over Time")
legend("topright", legend = paste("State", 1:n_states), col = 1:n_states, lty = 1)
```



Applications and Implications

Predicting Fitness Distributions

The model predicts how environmental changes affect the distribution of fitness categories over time, providing insights into population resilience.

Adapting to Environmental Stressors

Simulations reveal which fitness states are most vulnerable to specific environmental pressures, guiding conservation strategies.

Optimizing Habitat Management

By identifying optimal environmental conditions, the model aids in creating habitats that maximize population fitness.

Conclusion

Using Markov chains to simulate earthworm population dynamics under environmental pressures provides a powerful tool for ecological research and conservation. By capturing transitions between fitness states and incorporating environmental factors, this approach offers valuable predictions for population management. “‘