Vignette

Overview

This document aims to translate the Python code of our antigen spread and response simulation into concepts familiar to immunologists. We focus on the choices and models used in each step of the simulation process, with emphasis on the flexibility to adapt these models based on the latest scientific literature.

Simulation Structure

The simulation constructs a landscape (grid) and populates it with antigens. Over time, these antigens spread and elicit responses, which are captured and visualized.

Components and Models

1. `Antigen.py` : Modeling Antigen Behavior

- **Antigen Attributes**: Includes properties like spread, memory, response, and name. Each attribute plays a role in how an antigen behaves within the simulation.

- Each antigen is modeled as a point in antigenic landscape space, more specifically a Gaussian distribution centered at it's position in landscape space, with the height of the Gaussian at a specific (x,y) location representing the immune response

    - The memory will range between the following values
      1 - mild
      2 - intense
      3 - extreme
      4 - permanent (e.g. chicken pox)

    - The spread is used as a measure of how heavy the tails of the Gaussian, and specifics on how they are computed are shown in the `Landscape.py` section

- **Baseline Response Calculation**:
  - Formula: $\dfrac{c}{1 + e^{\frac{(a-mc)}{mc/4}}}$

    c is the initial immune response levels

    a is the age

    m is the memory level

  - Purpose: Calculates the baseline level of an antigen's response.

  - Explanation: This function models the decay rate of an antigen's response over time. The decay is non-linear, reflecting the complex dynamics of immune response.

  - Customization: This formula can be altered to match updated immunological research. Adjustments can be made to the rate of decay or the shape of the decay curve to better simulate real-world observations

2. `Landscape.py` : Constructing the Simulation Space

- **Grid Setup**: Initializes a grid where each cell can contain an antigen.

- **Time Progression**: Manages the flow of time within the simulation, affecting the state of antigens.

- **Antigen Interaction**: Handles how antigens interact with each other, which is crucial for understanding the spread and response dynamics.

- **3D Visualization**: Uses matplotlib to create 3D plots, showing the distribution and intensity of antigens over the landscape.

- **Modification Potential**: The landscape's size and the rules governing antigen interactions can be adjusted to reflect different environments or scales.

- The way we calculate the Z-scores are as follows $\exp\left(-\dfrac{(X-\bar{x})^2 + (Y-\bar{y})^2}{2 \cdot \text{spread}}\right)$.

- We also calculate what happens when a new antigen the space of another antigen's spread (of a certain threshold because tails go on infinitely for Gaussians) which can be found in the `update_antigen_response` function. We blunt the response based on the age the simulated individual was exposed to the antigen and based on the memory attribute mentioned in the Antigen section)

Specifically this is done by updating the response as the current immune response value times (1 + multiplier * memory), where the multiplier < 1 and dependent on the age the individual was exposed.

- As stated earlier for in the Antigen explanation, these formulas can be altered to match updated immunological research.

3. `Storage.py` : Data Storage and Snapshotting

- **Purpose**: Provides a mechanism for storing the state of the landscape at various time points.

- **Usage**: Enables analysis of the landscape's evolution and can be used for retrospective studies or model validation.

4. `Main.py` : Orchestrating the Simulation

- **Initialization**: Sets up the landscape and antigens, and defines the simulation duration.

- **Running the Simulation**: Executes the simulation cycle, with outputs including landscape states and visualizations.

- **Flexibility**: The initial parameters (e.g., number of antigens, landscape size, how often the simulated individual is exposed to antigens) can be varied to explore different scenarios or hypotheses.

- **Units**: The units of time here are in months.

Customization and Adaptation

- **Formula Adjustments**: Mathematical models like the baseline response calculation can be refined to align with emerging immunological research.

- **Parameter Tweaking**: Initial conditions such as antigen spread, memory, and landscape dimensions can be adjusted to simulate different biological or environmental conditions.

- **Visualization Changes**: The output plots can be customized to highlight specific aspects of the simulation, aiding in data interpretation and presentation.
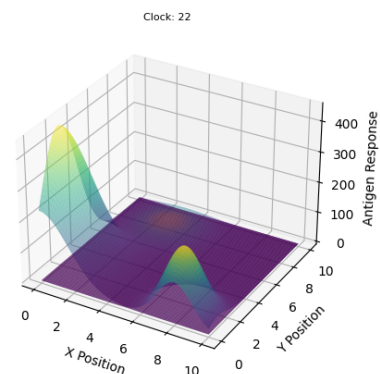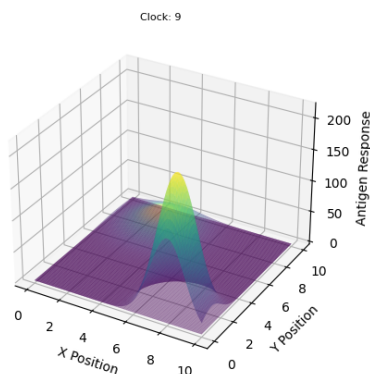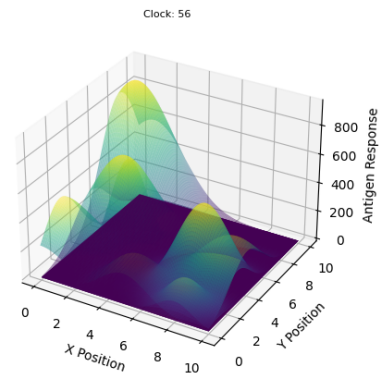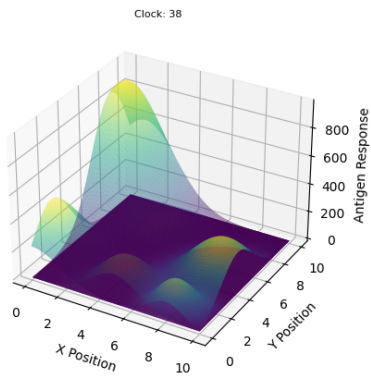
Example
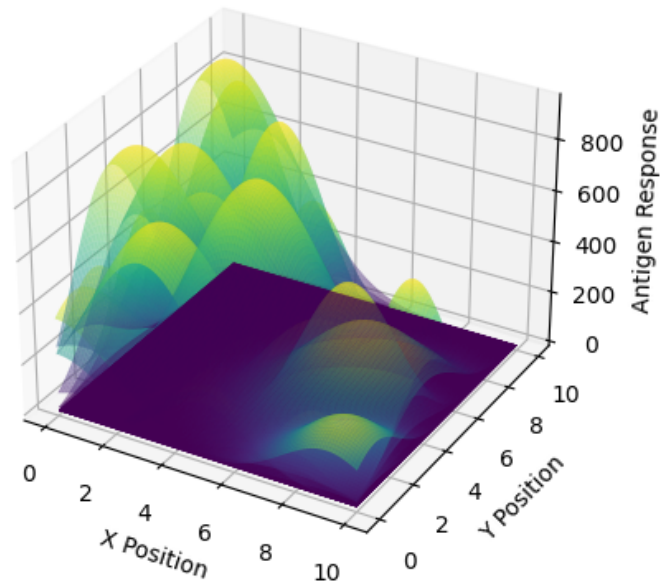
Here is an example output that shows the overall landscape when ran with the random seed at 42. We ran it with a 10 by 10 landscape, with 144 months and exposure to 24 antigens.

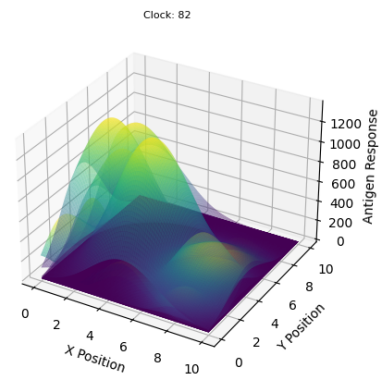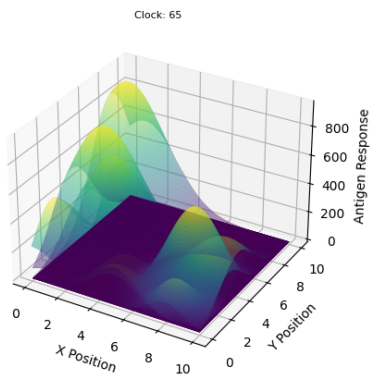Here are some midway points of the landscape:
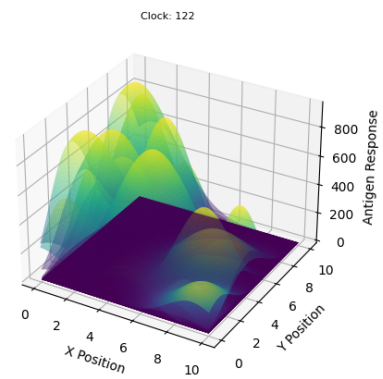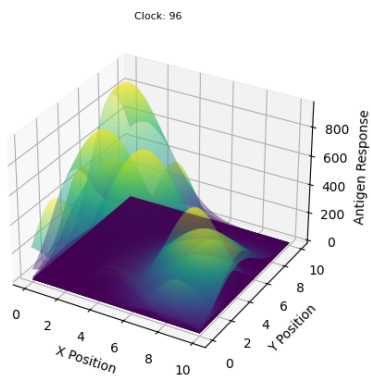
9 months, then 22 months in

38 months, then 56 months

Clock: 38

Clock: 56

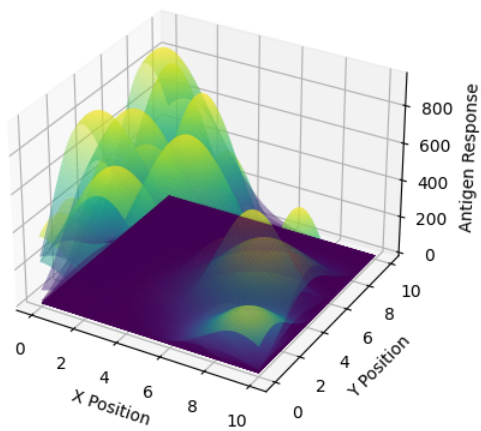# 65 months then 82 months

Clock: 65

Clock: 82

96 months then 122 months

Clock: 96

Clock: 122

Final 144 months as seen earlier

Conclusion

This simulation serves as a flexible tool for visualizing and understanding antigen behavior in a controlled environment. By modifying its parameters and models, it can be adapted to reflect current scientific understanding and hypotheses in immunology.

This document is meant to serve as a bridge between the computational models and immunological concepts, providing clarity on how the simulation reflects and can adapt to the dynamic field of immunology.