Observations and Improvement Recommendations

1. **Could the code be made more efficient? A literal interpretation of the instructions above is not necessary.**
    - Initial program was not filtering based on neighbourhood radius and was updating the entire board
    - Program could be made more efficient by updating weights within the neighbourhood radius. This also gives us well defined boundaries which might serve the purpose better

2. **Is the code best structured for later use by other developers and in anticipation of productionisation?**
    - No
    - Variable names are special characters
    - Tough to follow the code and mathematical calculations and their relevance in the original code.

3. **How would you approach productionising this application?**
    - Give meaningful/descriptive variable names
    - Create functions for distance, influence, learning rate and neighbour radius
    - **Optional**: Create a Class that trains and returns output weights based on given input. Each instance of the class is initialised for a given max iteration count and map dimensions

4. **Anything else you think is relevant.**
    - Only good up to 1000 iterations, 100x100 map and 100 rows of data. Anything beyond takes too long.
    - Experimentation with learning rate, radius and influence, time constant exponents needed
    - Parallelization?