

Assignment 3 : Reinforcement Learning and Self-organizing Maps

Computational Intelligence

Spring 2021

Munawwar Anwar Adam
Khubaib Naeem Kasbati

Table of Contents

1	Question 1: Value Iteration	2
1.1	Problem Description and Solution	2
1.2	Results	2
2	Question 2: Self Organising Maps	4
2.1	Introduction	4
2.2	Problem Description and Solution	4
2.3	Results	5
3	Analysis	13

1 Question 1: Value Iteration

1.1 Problem Description and Solution

Our goal is to create a value iteration agent that takes the best possible path to a reward and stays away from obstacles, while minimizing movement.

In order to minimize movement, we set values which are neither reward nor obstacles immediate payoff as -1 to discourage additional movement. Reward was given the value of +100 while Obstacle was given the value -100 . Using the value iteration algorithm, optimum values were found.

1.2 Results

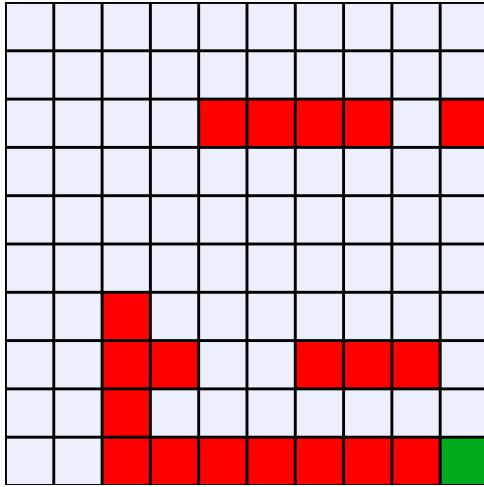
We tested our output on multiple grids, in order to ensure optimal results.

In Grid 1, we can see that the obstacles are avoided and the shortest path is being taken.

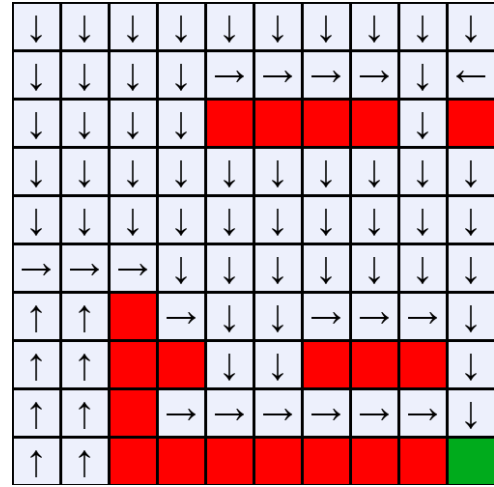
In Grid 2, we set 4 rewards, one for each corner. Over here, we can see that the agent's decision depends on which quadrant it is in and it approaches the reward that is closest to it, showing it has learned to minimize movement.

In Grid 3, we can see a similar setup, however, movement near the top left is restricted due to obstacles. It is seen that our Value iteration agent has learned that there is an obstacle and decides to approach another reward.

In Grid 4, we have placed several obstacles and one goal near the obstacles. Our value iteration agent has learned this and is optimally going towards the goal. We can see that on the right side of the obstacle cluster, the agent sticks near the obstacles in order to prevent unnecessary movement showing optimal behaviour.

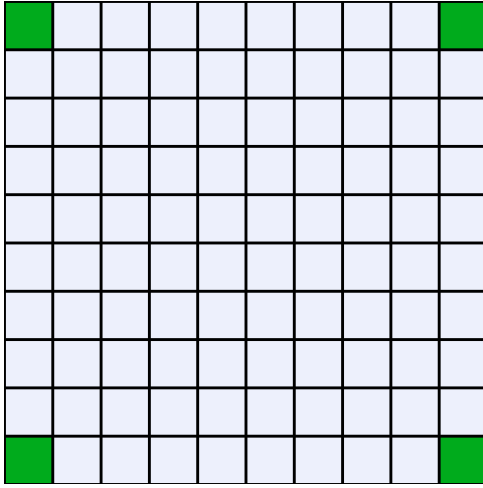


(a) Grid without Policy

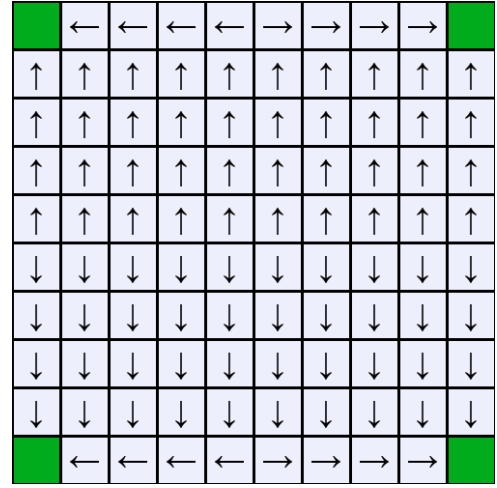


(b) Grid with Policy

Figure 1: Grid 1

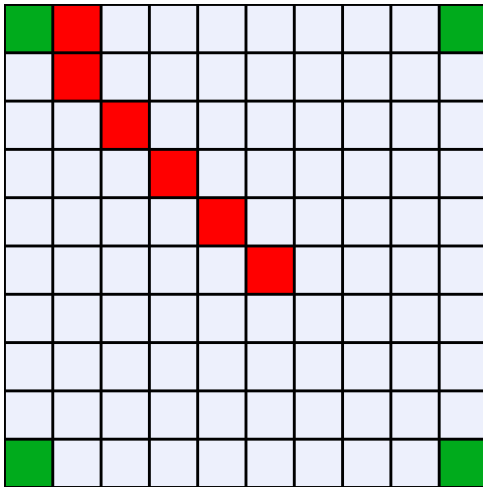


(a) Grid without Policy

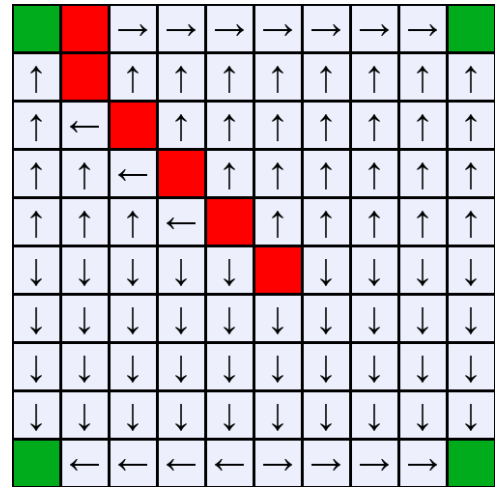


(b) Grid with Policy

Figure 2: Grid 2

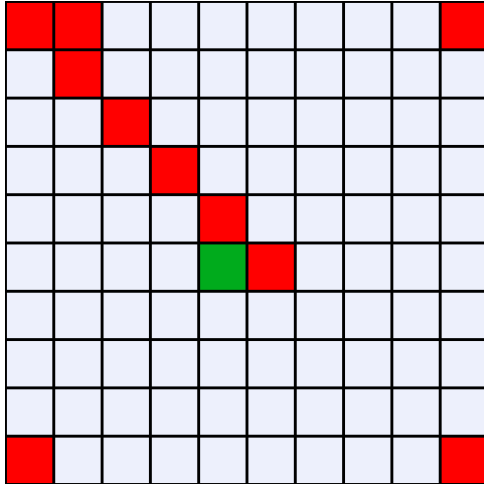


(a) Grid without Policy

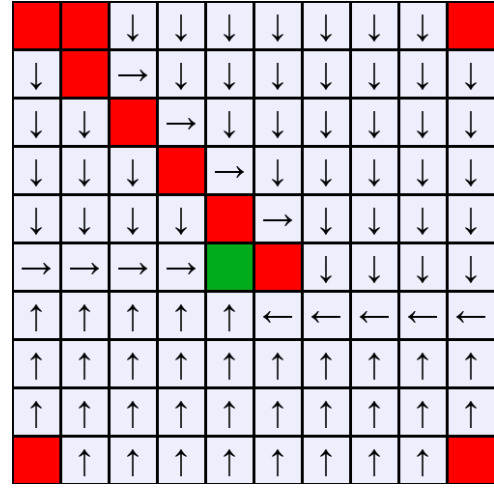


(b) Grid with Policy

Figure 3: Grid 3



(a) Grid without Policy



(b) Grid with Policy

Figure 4: Grid 4

2 Question 2: Self Organising Maps

2.1 Introduction

Self-Organizing Map (SOM) is an unsupervised neural network model. It is for clustering, dimension reduction, and feature detection. SOM is used for data visualization by projecting higher dimensional data into lower dimensional space leveraging topological similarity properties. While learning the model, SOM utilizes competitive learning techniques.

2.2 Problem Description and Solution

Our goal is to visualize the dataset “Total World Population by Gender from 1950 to 2100”. The features that we will be considering to cluster this data are **Total Population, Population Density, Population Female** and **Population Male**.

The Kohonen network consists of 10×10 nodes where each node is connected to the input layer which is representing a four dimension input vector. The four dimensions of the input correspond to the four features that we are considering. The values of the rest of the parameters is as follows :

- Number of Iterations : 20,000
- Initial Learning Rate : 0.01
- Initial Radius : 5

The exponential decay function for finding the neighborhood size is as follows:

$$\sigma(t) = \sigma_0 \times e^{-\left(\frac{t}{\lambda}\right)}$$

where σ_0 is the initial radius, t is the time step and λ is the time constant. The equation for updating weights is as follows:

$$W_{t+1} = W_t + \eta_t L_t (X_t - W_t)$$

where t is the time step, $L(t)$ is the learning rate and $\eta(t)$ is the influence rate. The equation of the learning rate is as follows:

$$L(t) = L_0 e^{-\left(\frac{t}{\lambda}\right)}$$

where t is the current time step, L_0 is the initial learning rate and λ is the time constant. The following equation is the expression of the influence rate:

$$\eta(t) = e^{\frac{-distance}{2\sigma^2(t)}}$$

2.3 Results

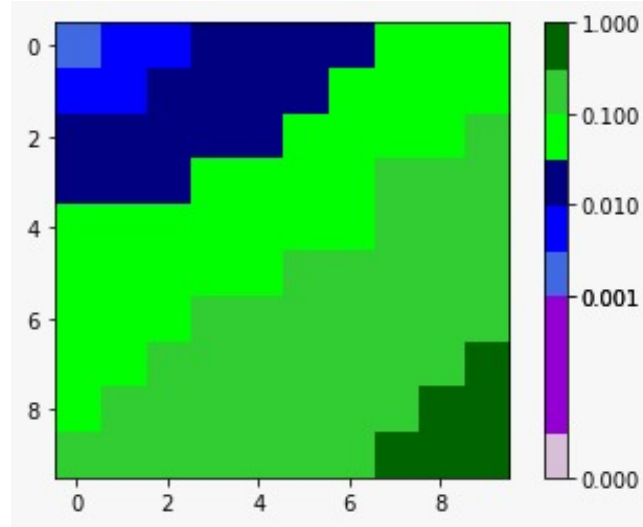


Figure 5: Visualisation of the clusters.

In the end, we obtained a clustering with five different clusters. As seen in figure 5, there are 5 different clusters with corresponding to a different color. The representative color of each country is then used to color that country on the map. In the following figures, we can see how the world's total population has progressed decade wise from 1950 to 2100.

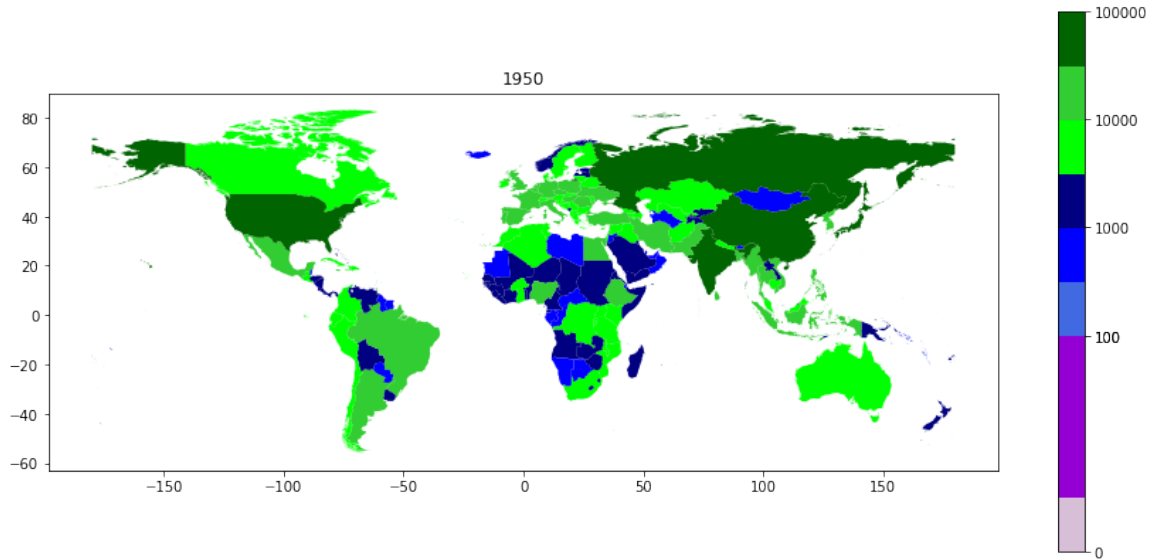


Figure 6: Worldwide Population in 1950

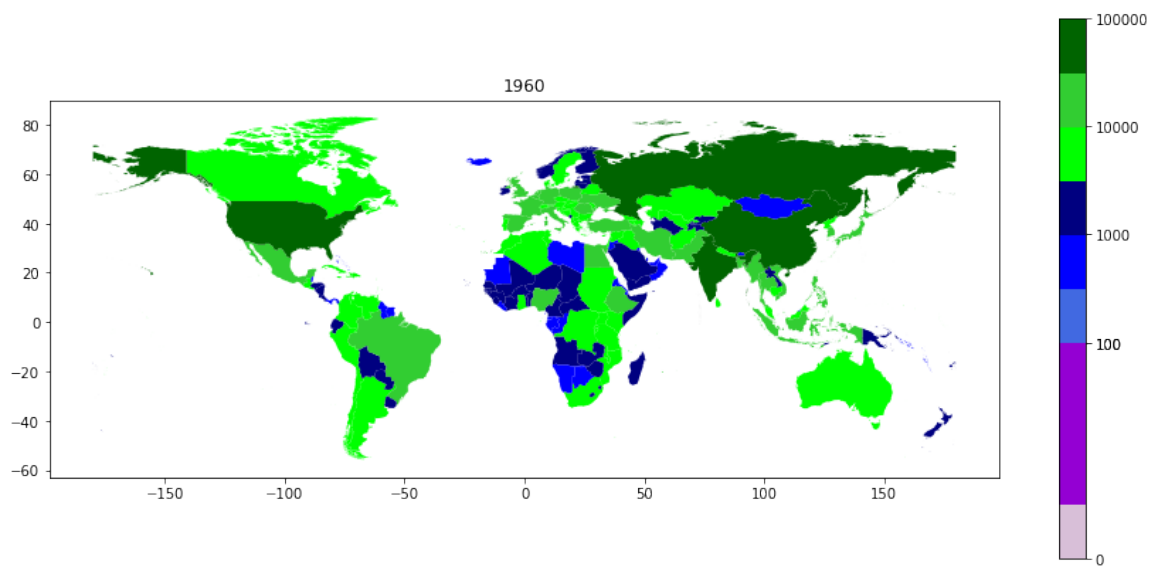


Figure 7: Worldwide Population in 1960

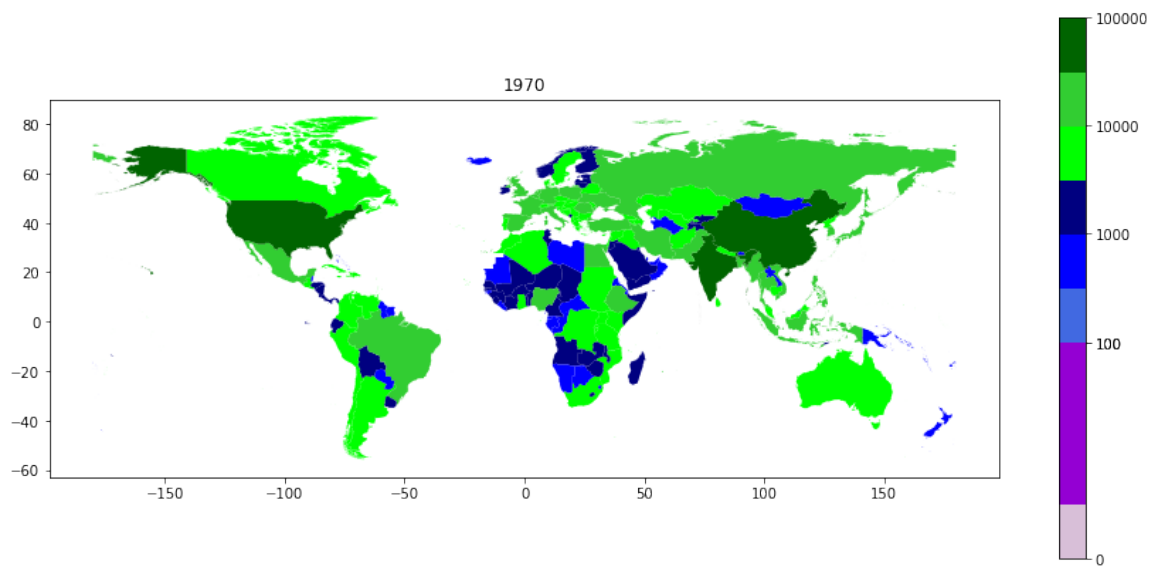


Figure 8: Worldwide Population in 1970

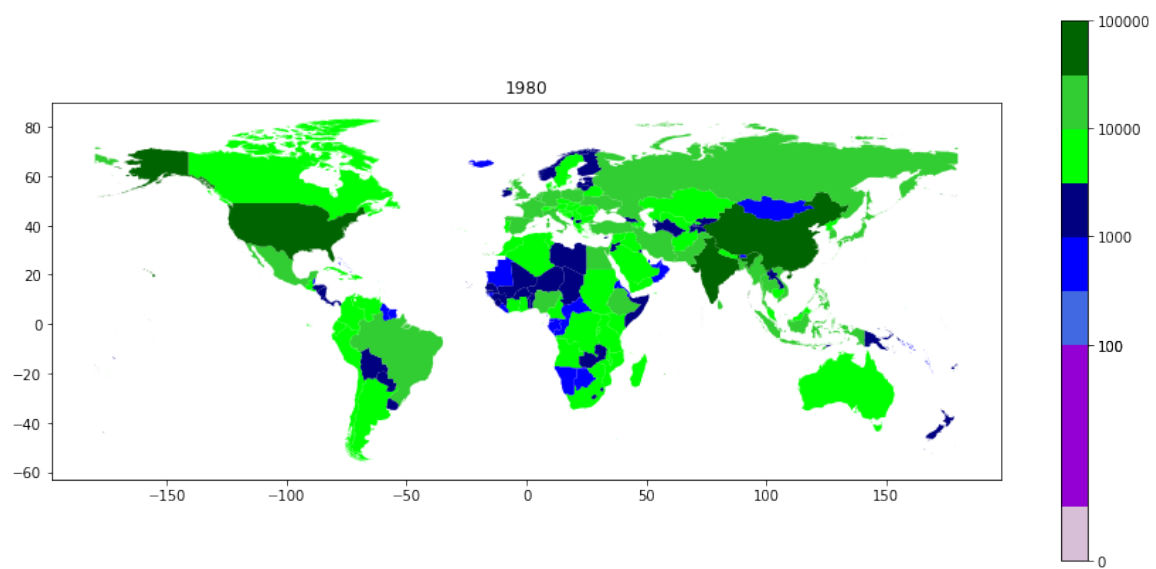


Figure 9: Worldwide Population in 1980

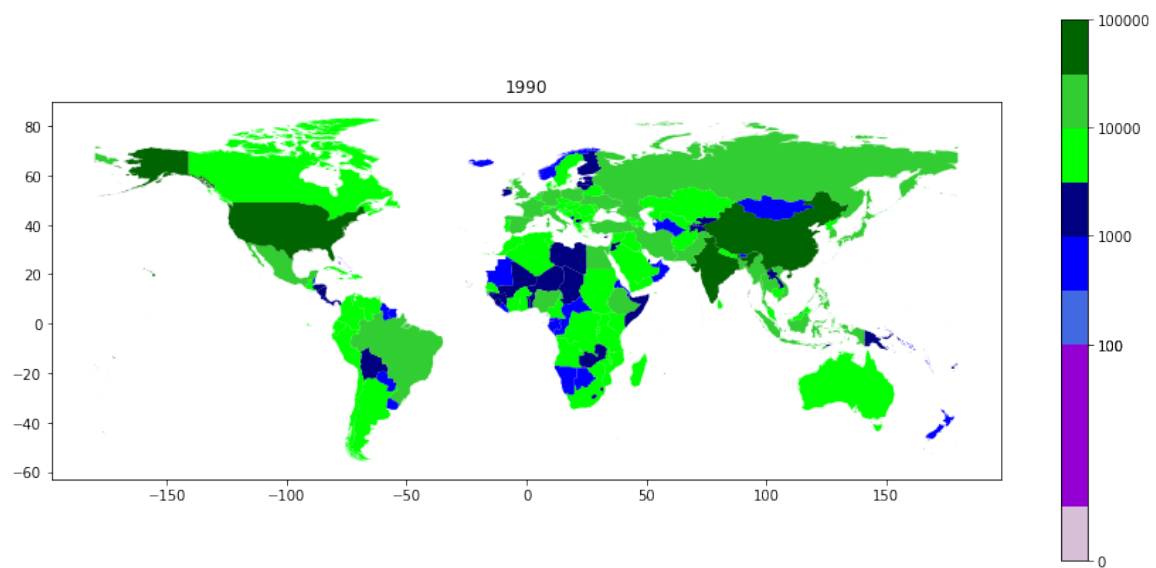


Figure 10: Worldwide Population in 1990

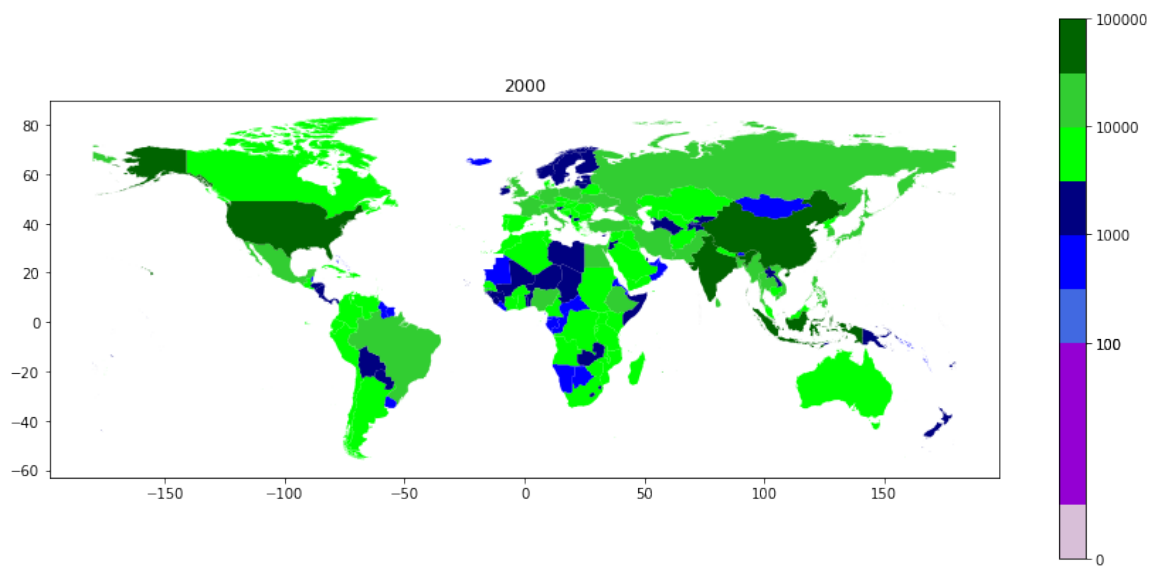


Figure 11: Worldwide Population in 2000

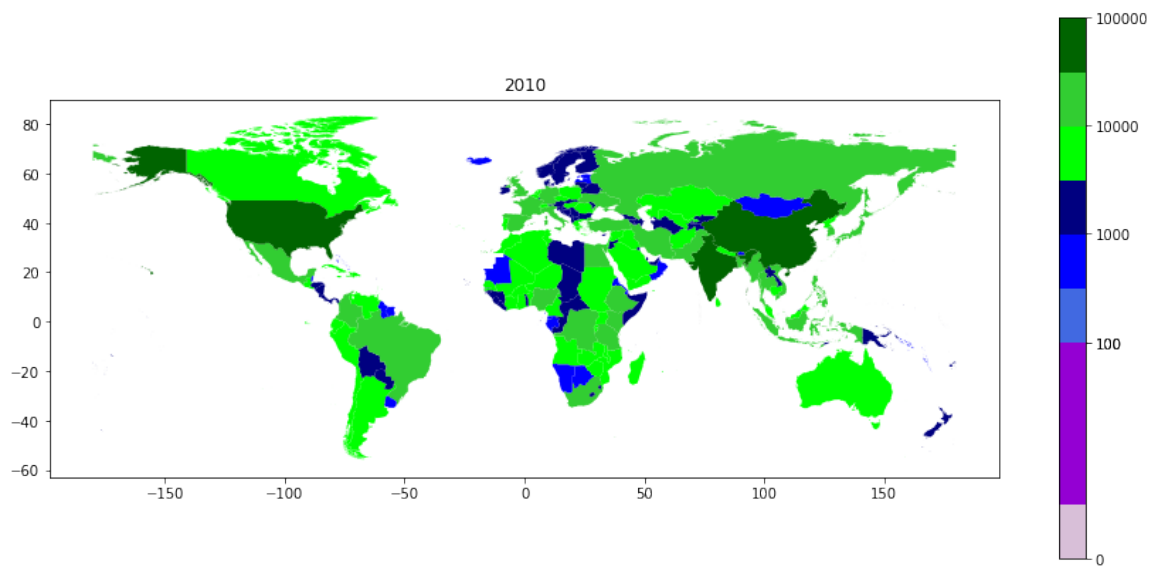


Figure 12: Worldwide Population in 2010

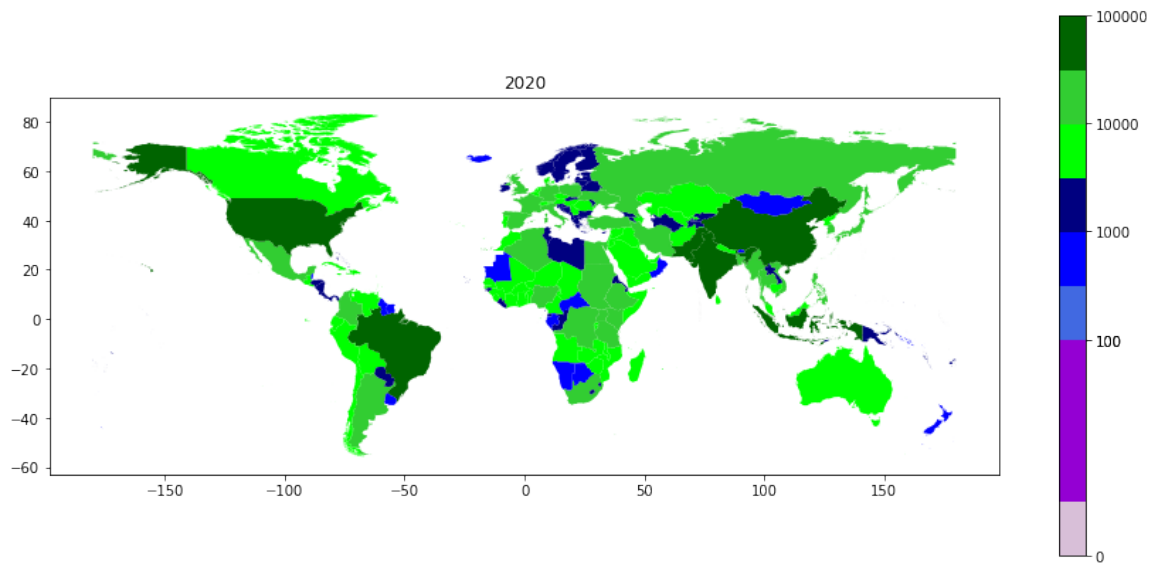


Figure 13: Worldwide Population in 2020

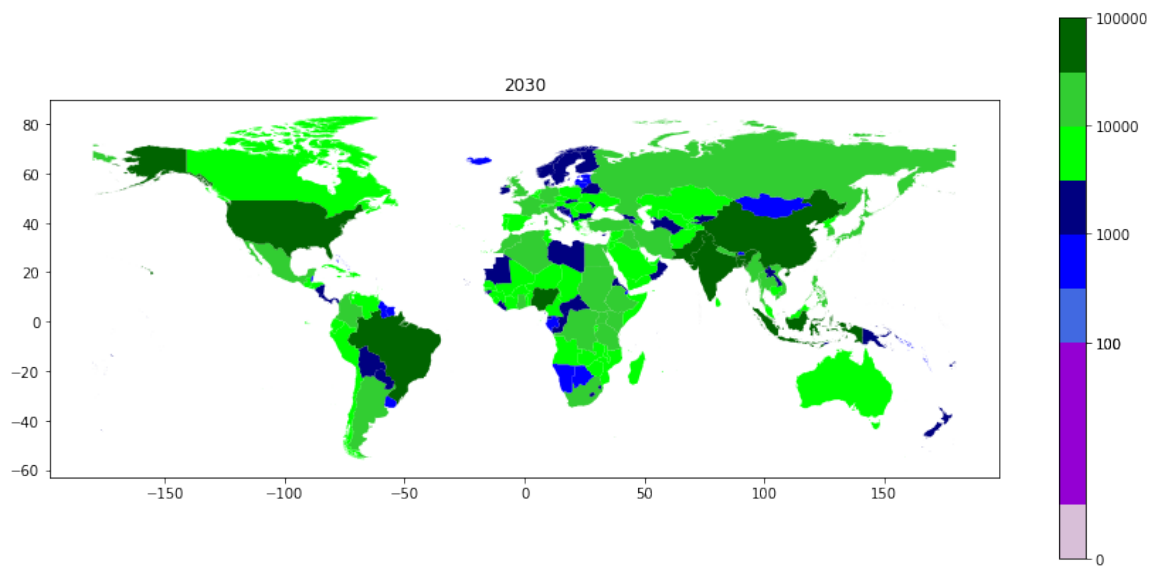


Figure 14: Worldwide Population in 2030

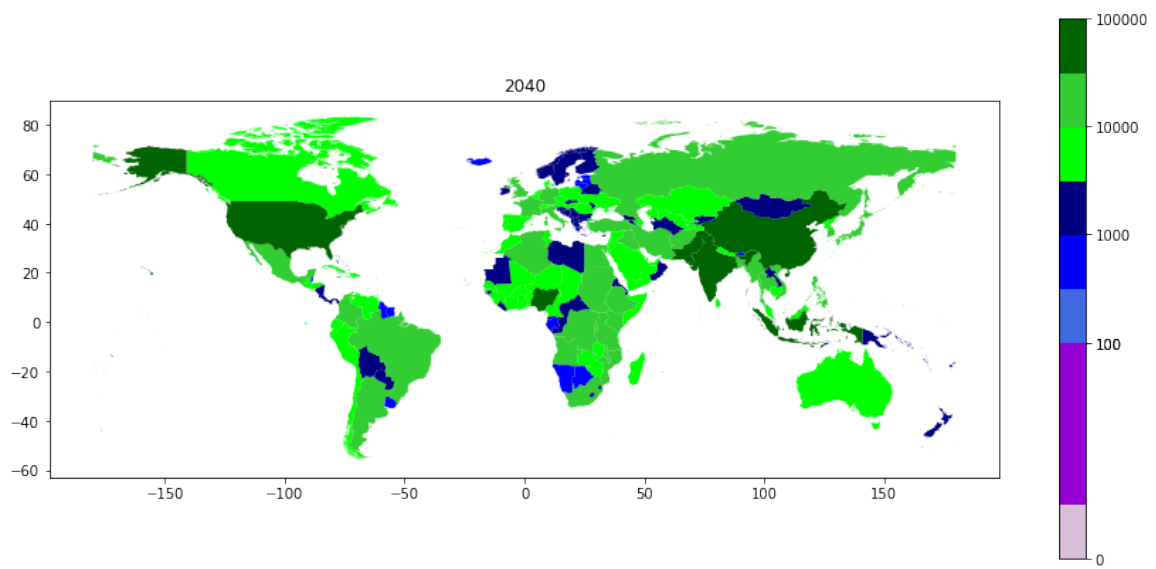


Figure 15: Worldwide Population in 2040

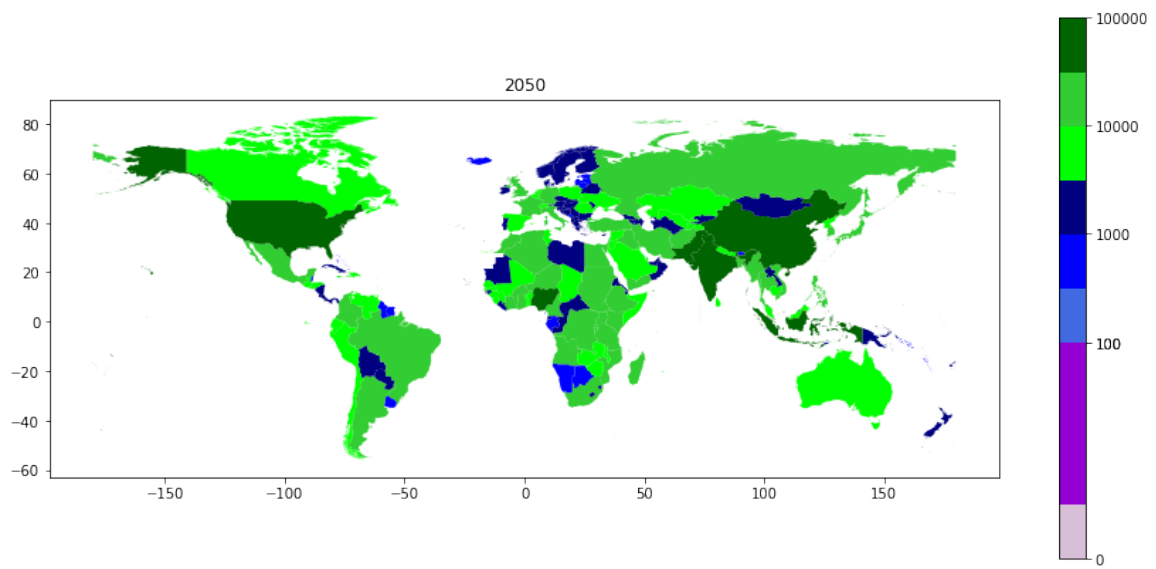


Figure 16: Worldwide Population in 2050

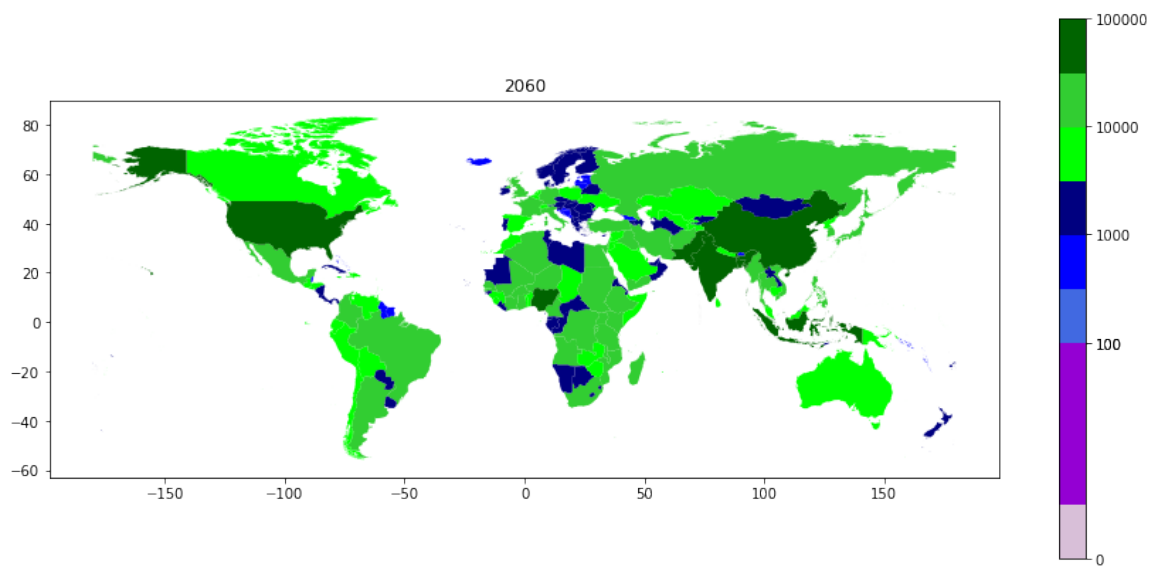


Figure 17: Worldwide Population in 2060

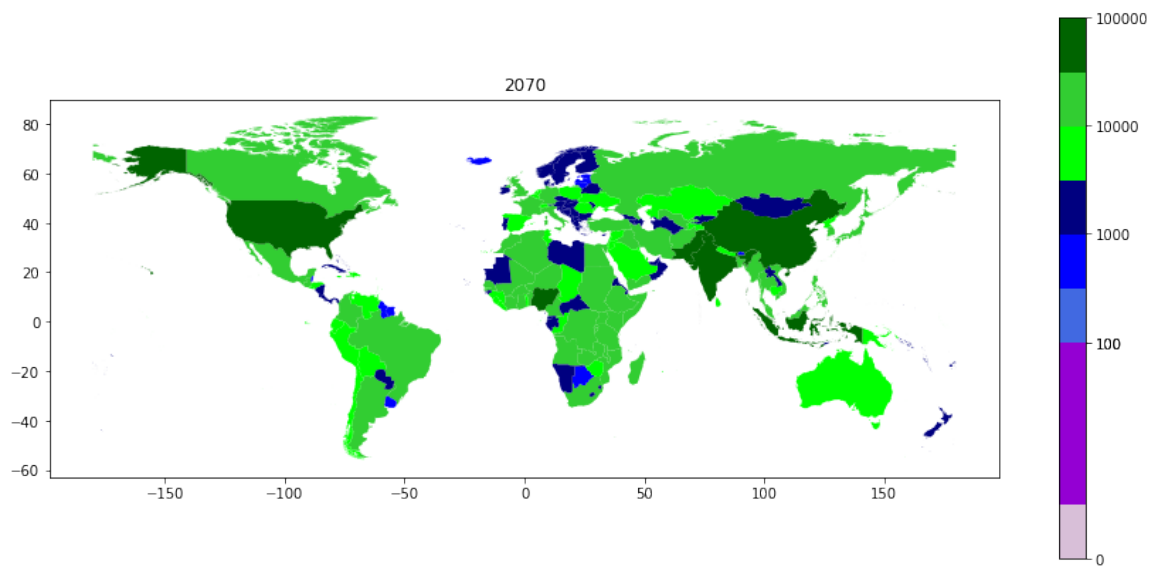


Figure 18: Worldwide Population in 2070

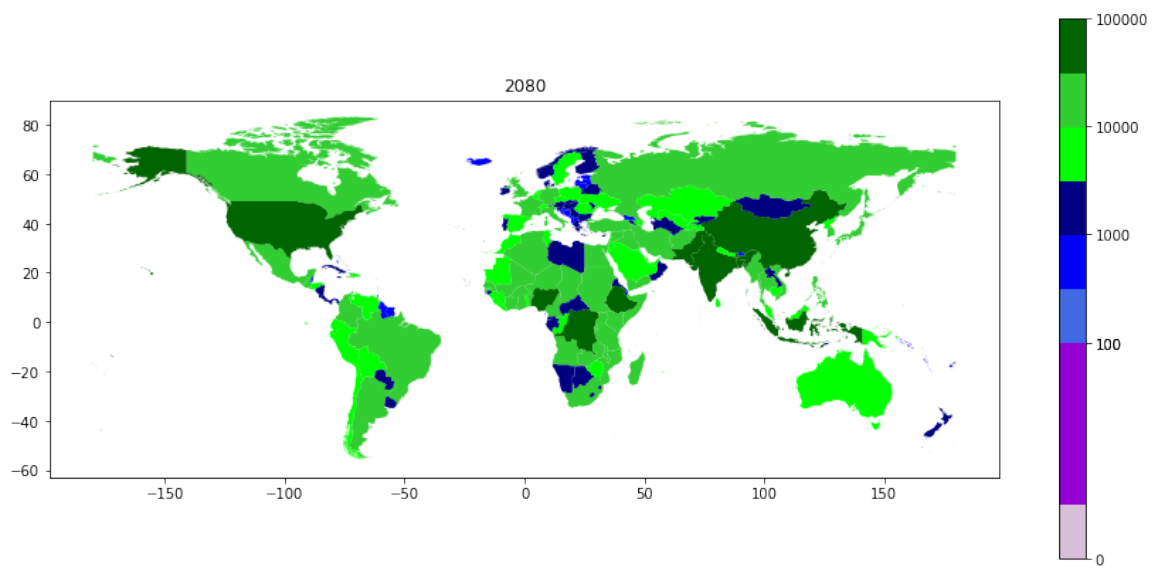


Figure 19: Worldwide Population in 2080

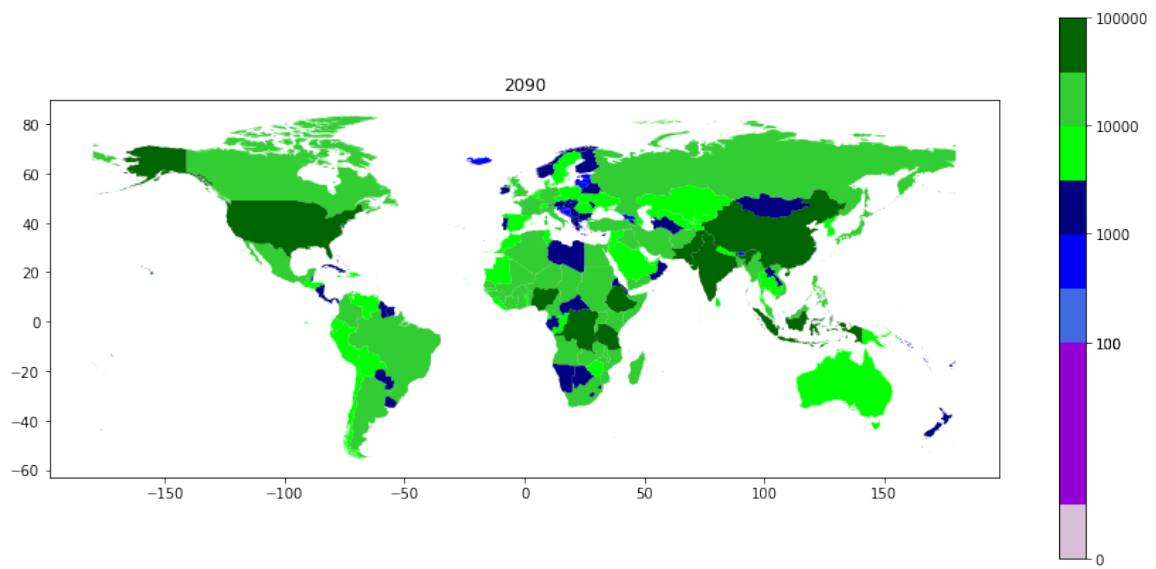


Figure 20: Worldwide Population in 2090

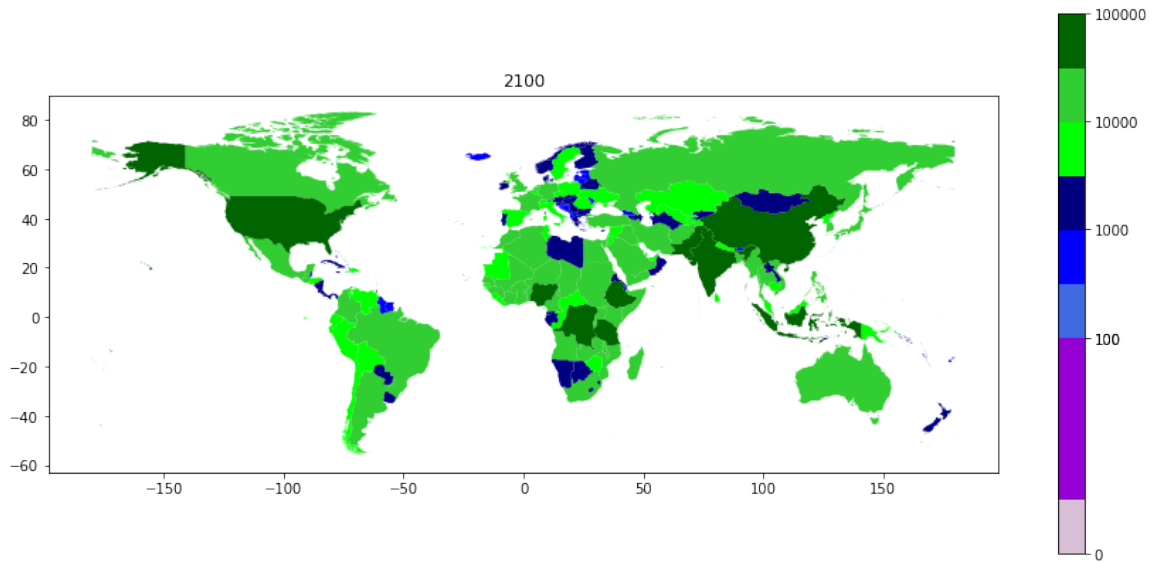


Figure 21: Worldwide Population in 2100

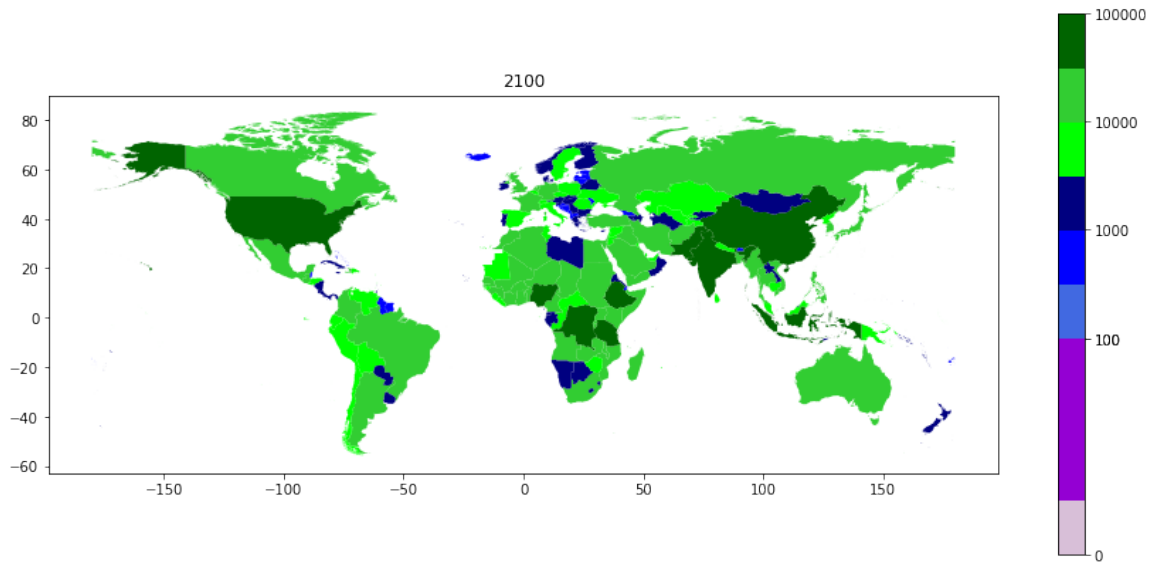


Figure 22: Worldwide Population in 2100

3 Analysis

In figure 6, we can see that there are blue patches which indicate a smaller world progression. However, as time progresses these patches start to turn into shades of green, which indicates a growing population. In addition to, we can see countries which have a dark green color such as China, USA, India, Pakistan. These are the countries which have the highest population in the world.