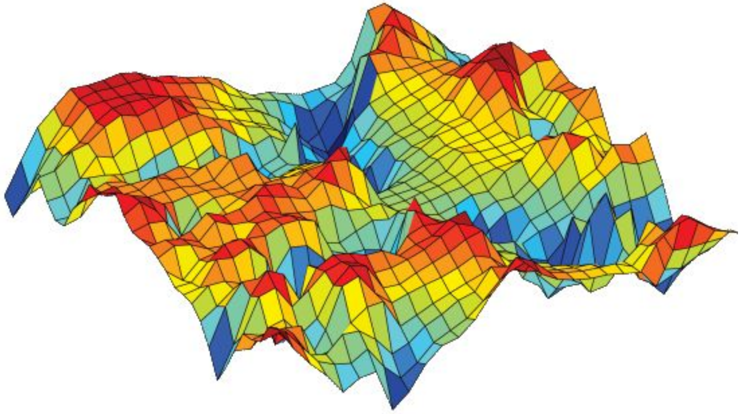
# Cluster Analysis Using a Self-Organizing Map



Yuan Shen Li
Il Shan Ng

## *The Problem*

Given a set of data points $\{x_1, x_2, \ldots x_N\}$, each of which has $F$ features, partition the points into $K$ disjoint clusters.

## *Clustering Algorithm of Choice*

Self-organizing map, which performs hard clustering

# *Self-Organizing Map (SOM)*

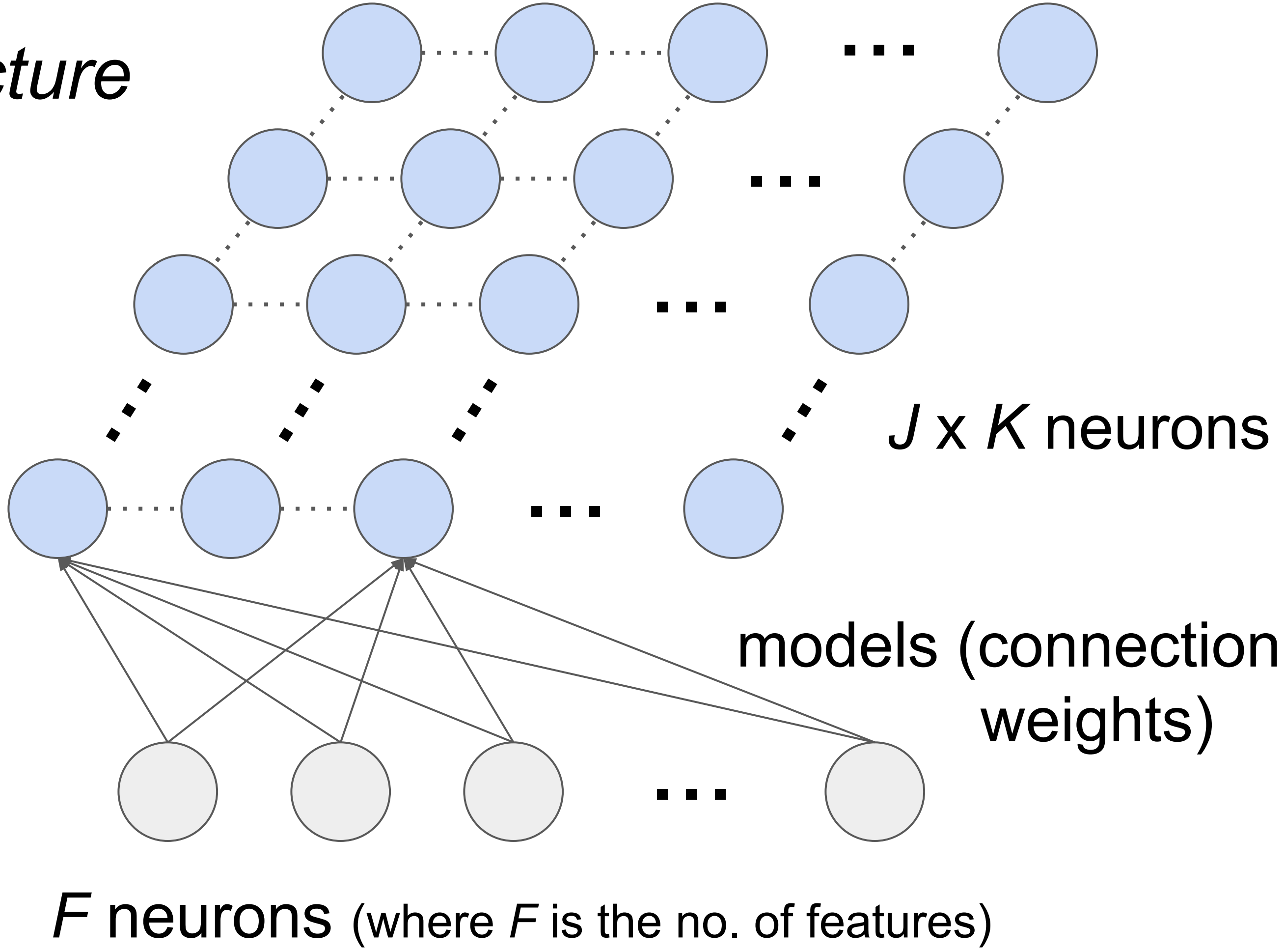
An artificial neural network that

- selectively tunes each neuron to a specific input pattern (competitive learning)
- thereby transforming an input of arbitrary dimension into a discrete, low-dimensional map (typically 1 or 2D)

# *SOM Architecture*

output layer
(2D topological
space)

input layer
(*F*-dim feature
space)



*SOM Training Algorithm*

1. Initialization step

   initialize models (connection weights) to small random values $m_{jkf}$

2. Competitive step

3. Cooperative step

4. Adaptive step

# *Competitive Step*

- Feed one data point $x_i$ into the input layer
- Each neuron in the output layer computes its *discriminant* for this data point

$$d_{jk}(x_i) = \sum_{f=1}^{F} (x_{if} - m_{jkf})$$

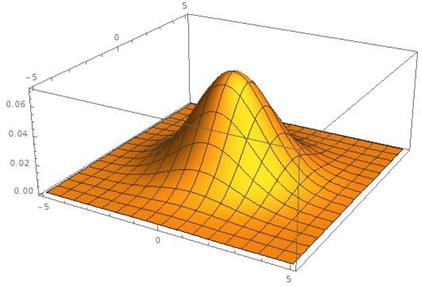
- Neuron with the lowest discriminant "wins" the data point and gets excited

# *Cooperative Step*

- Neurons that are close to the winning neuron also get excited

- Define a gaussian *topological neighborhood* centered on the winning neuron

$$T_j = \exp\left[\frac{-D_j^2}{2\sigma^2}\right]$$




where $D_j$ is the topological distance between neuron $j$ and the winning neuron

# *Adaptive Step*

- Update the models (connection weights) using

$$\Delta m_{jkf} = \alpha(t) \cdot T_j \cdot (x_{if} - m_{jkf})$$

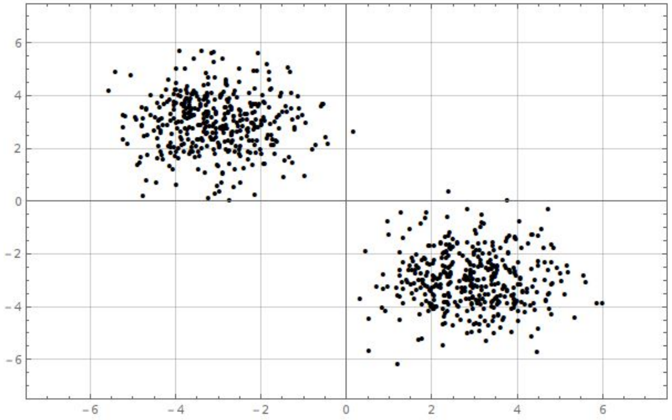
  where α is a learning rate that decays over time

- Winning neuron learns the most, and neighboring neurons learn proportionately to their proximity to the winning neuron

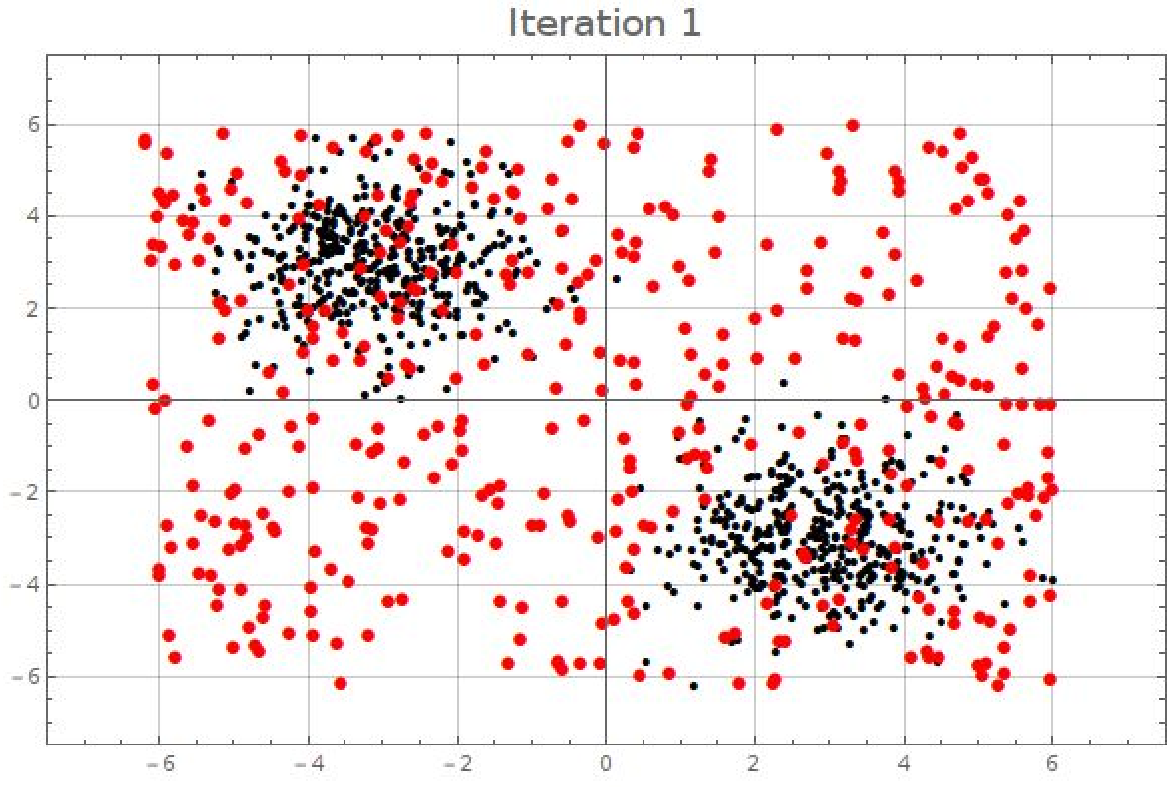
# *Testing on Dummy Data*

- Create 2D dummy data by simulating from two bivariate normal distributions

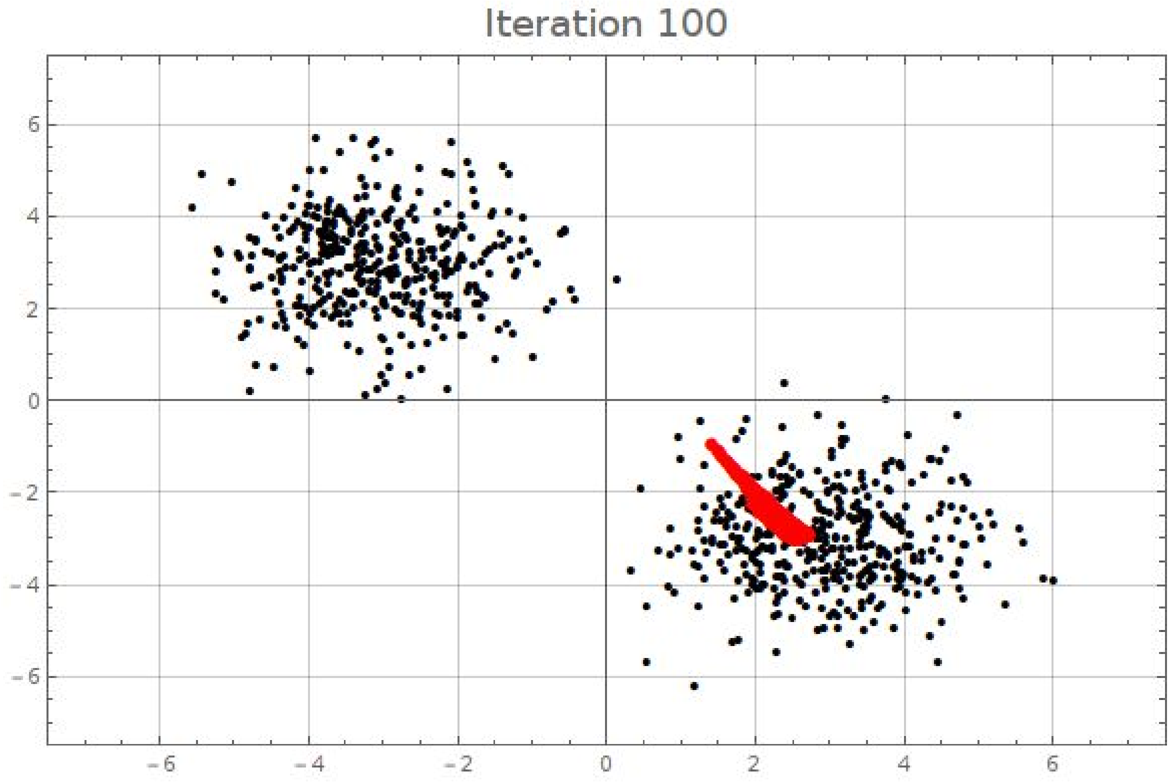
- True number of clusters is 2



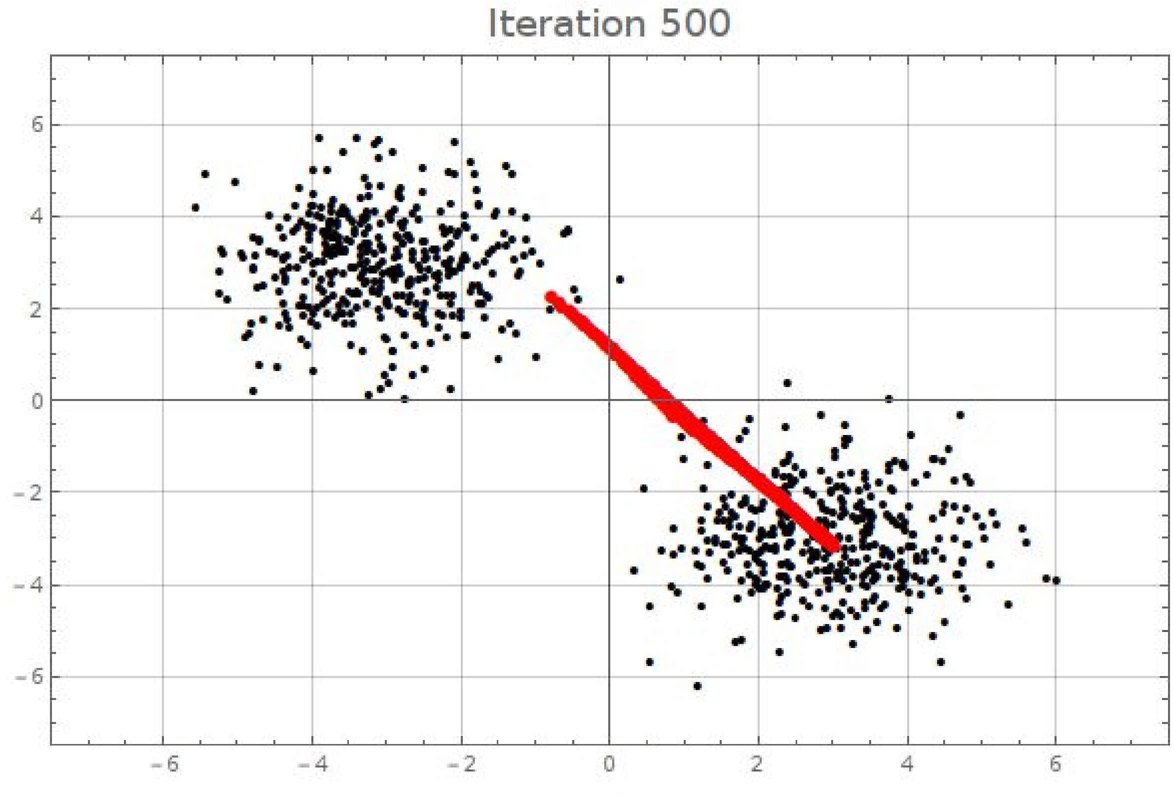
# *Visualization of SOM Training* (initialization)



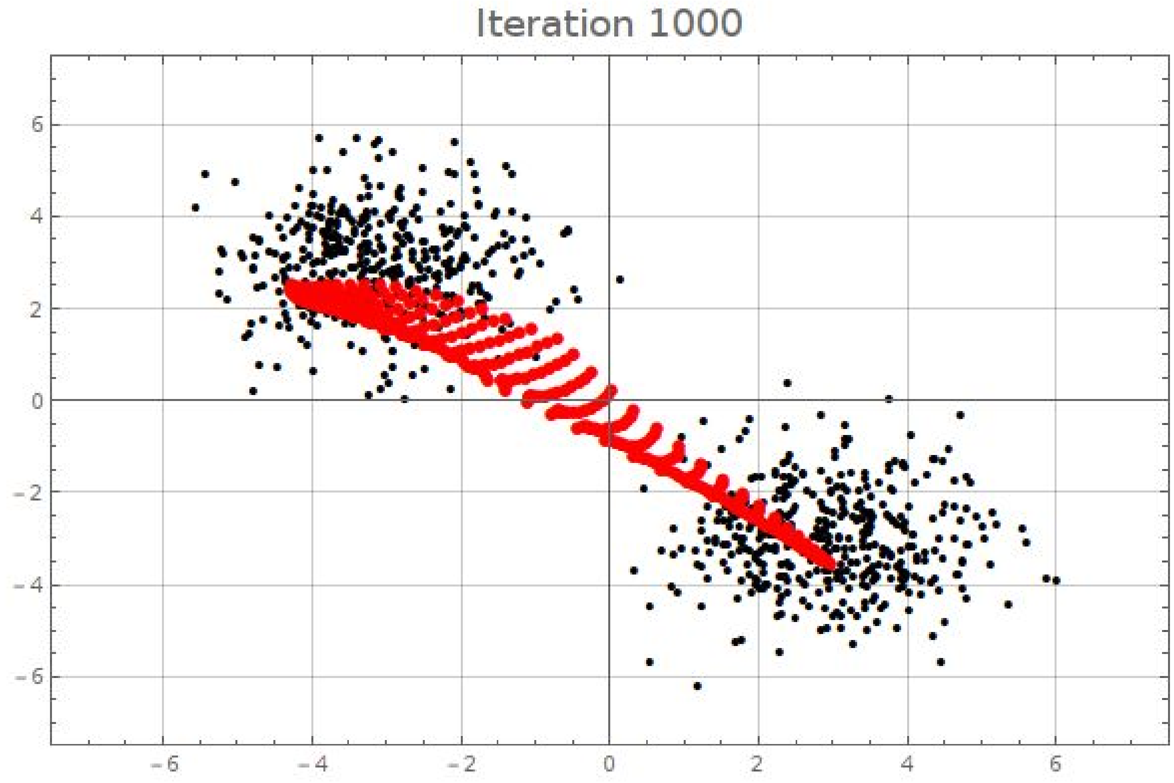
# *Visualization of SOM Training*



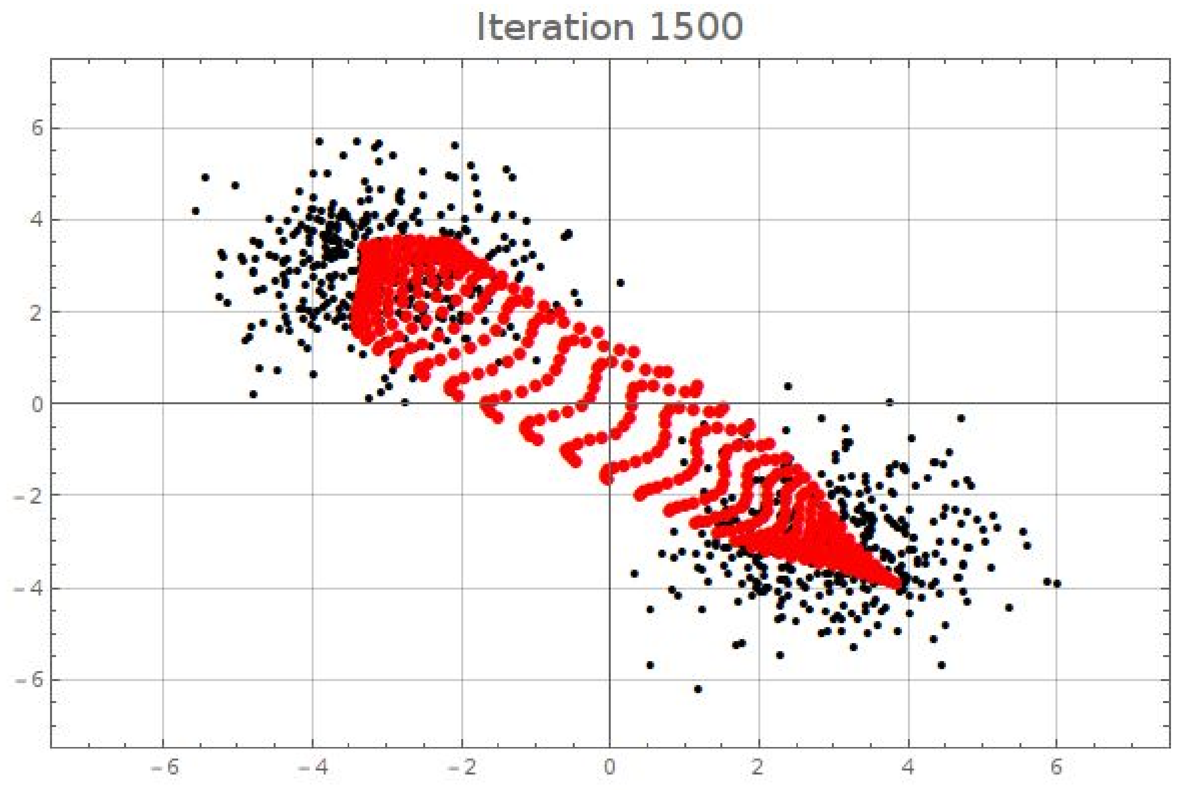
# *Visualization of SOM Training*



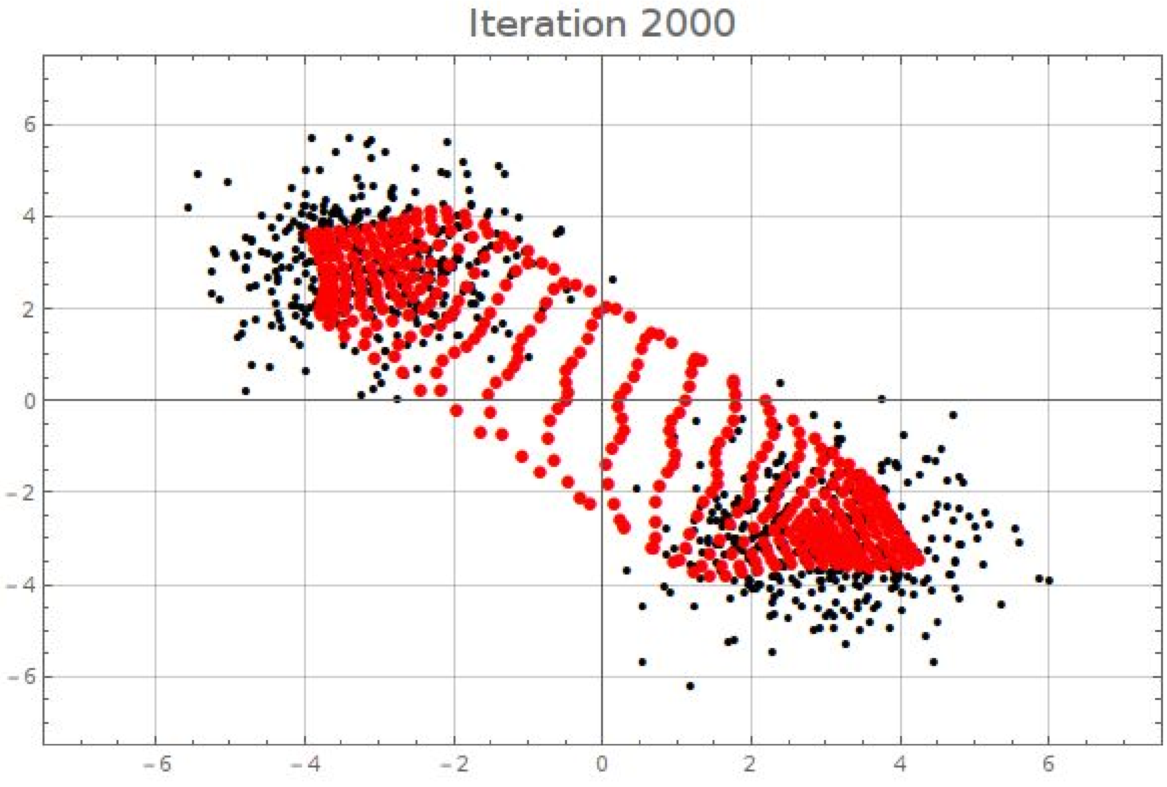
# Visualization of SOM Training



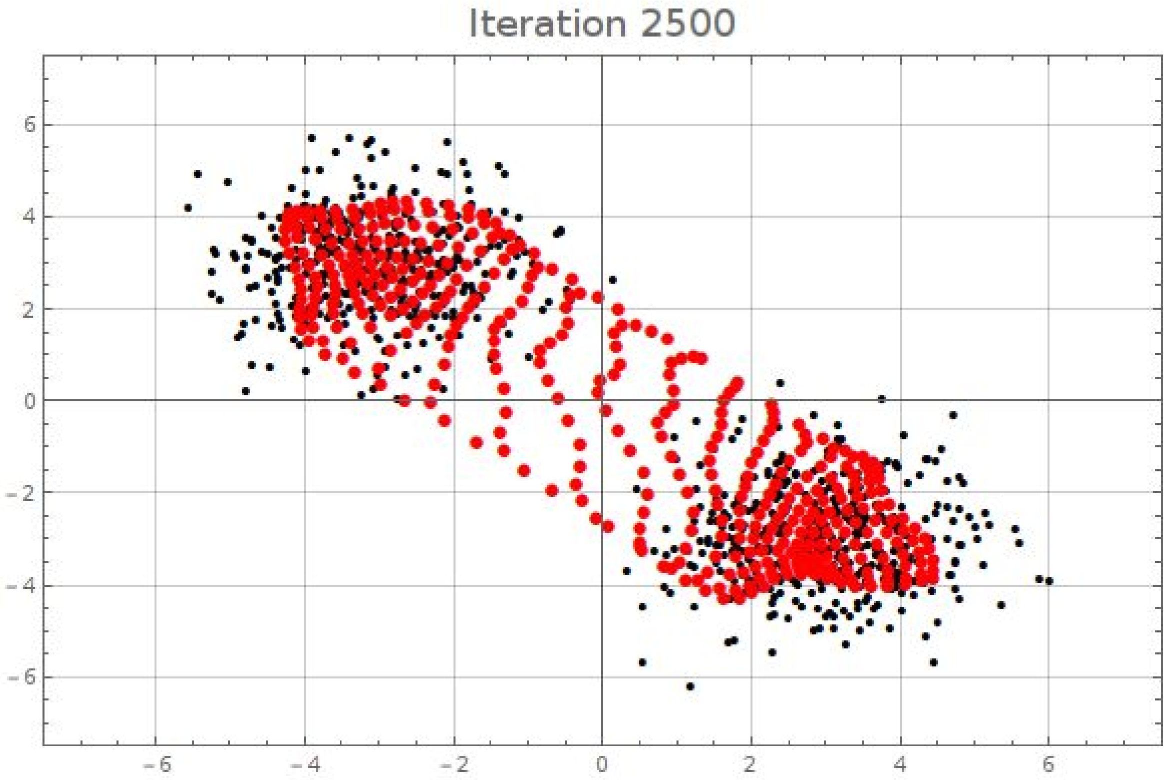
# *Visualization of SOM Training*



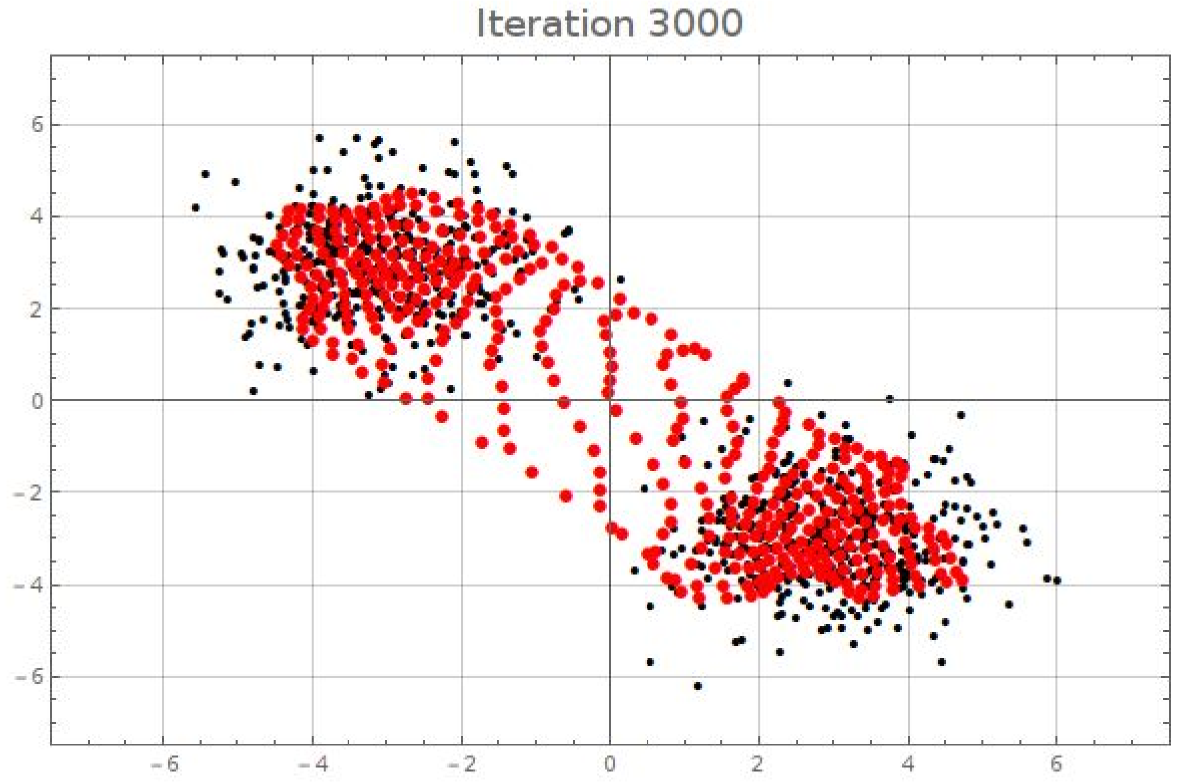
# Visualization of SOM Training



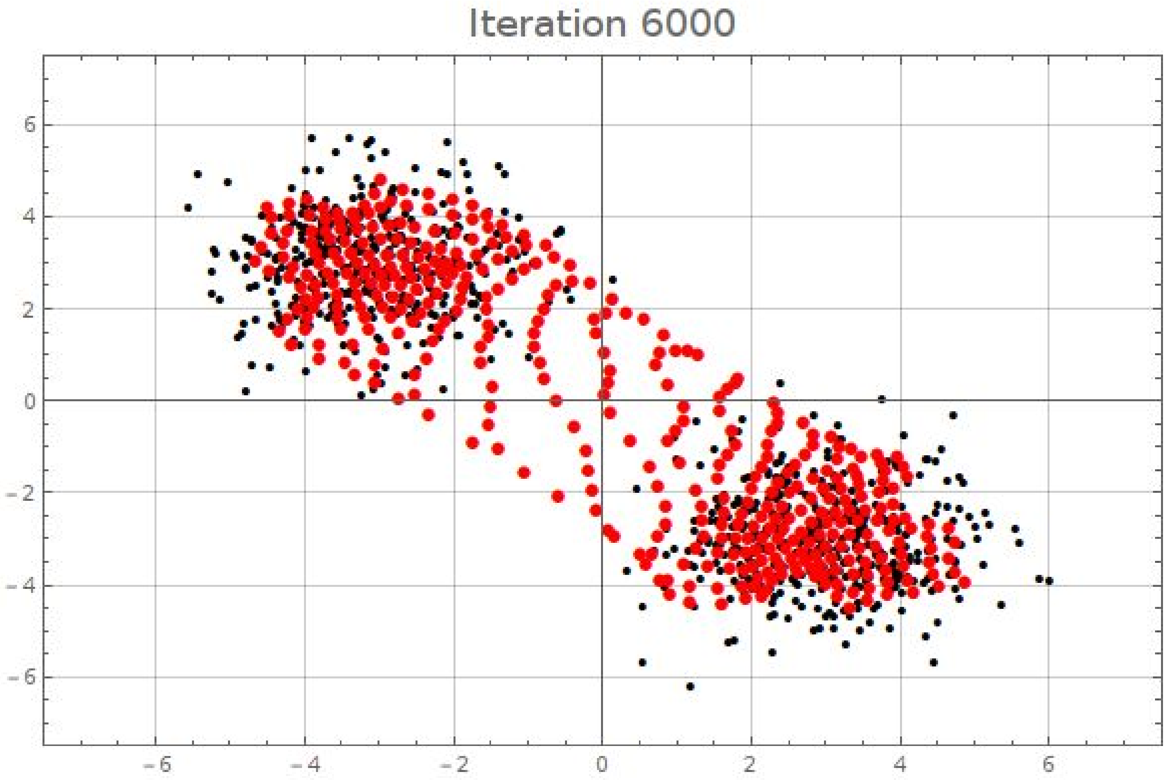
# *Visualization of SOM Training*



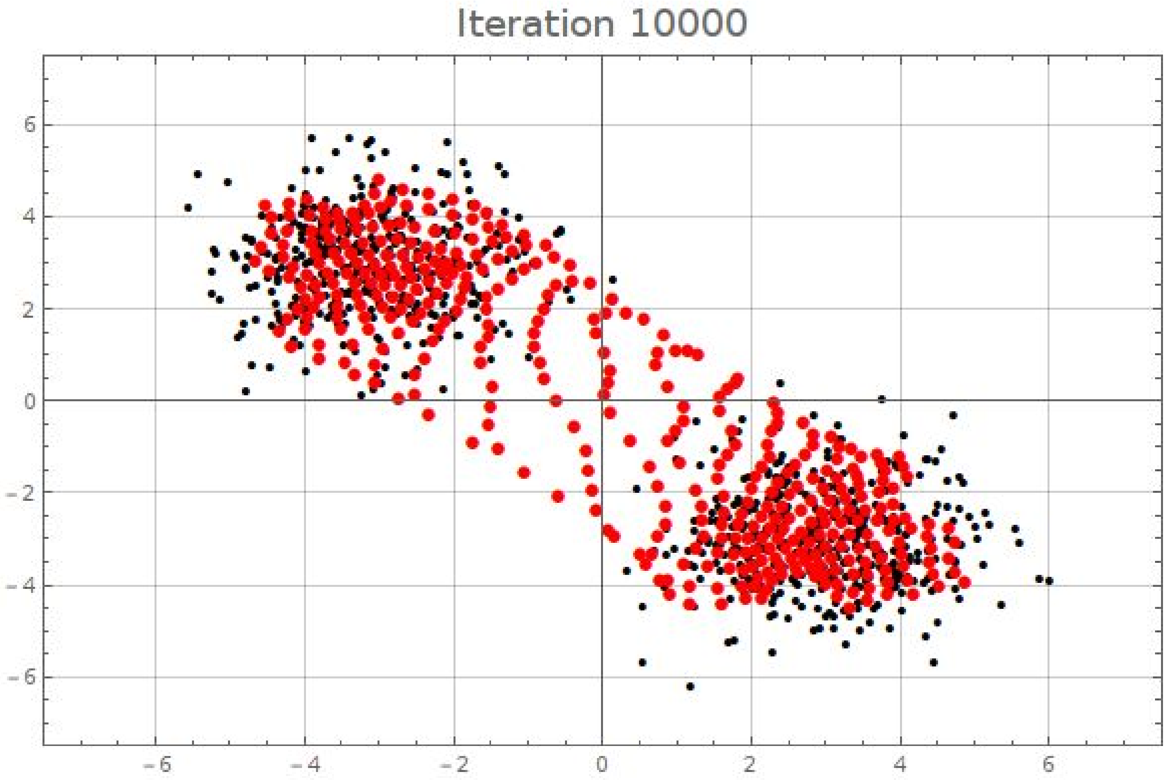
# *Visualization of SOM Training*



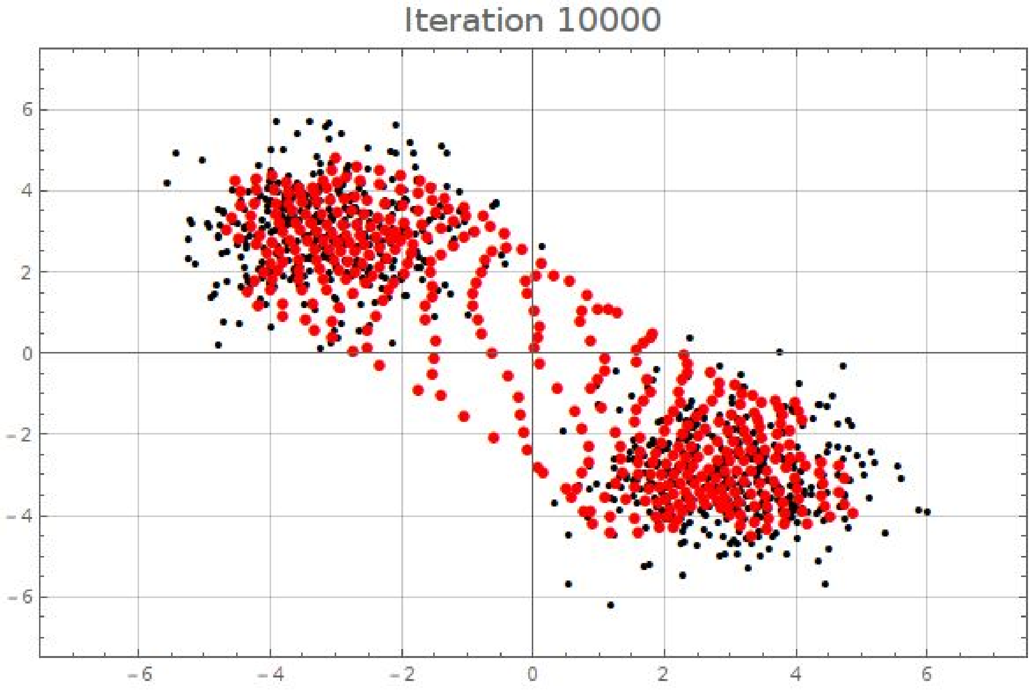
# *Visualization of SOM Training*



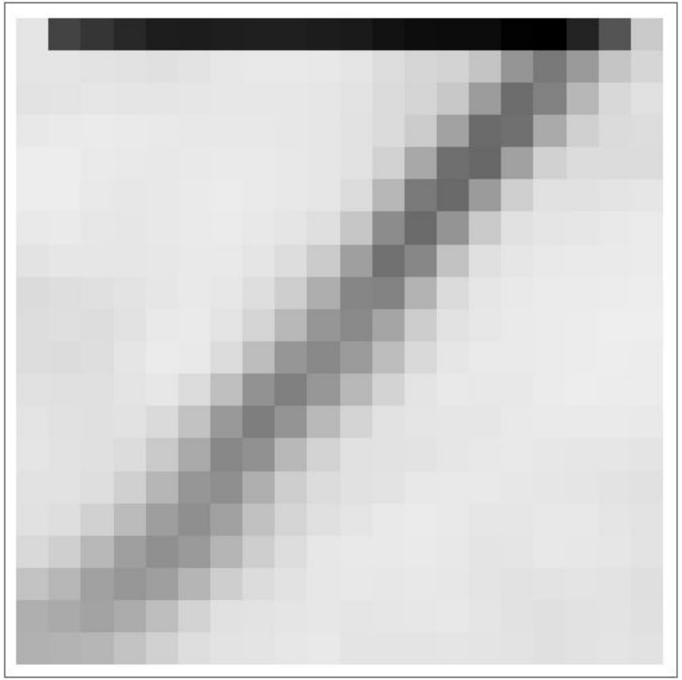
# *Visualization of SOM Training* (convergence!)



# *Interpretation of SOM Output*

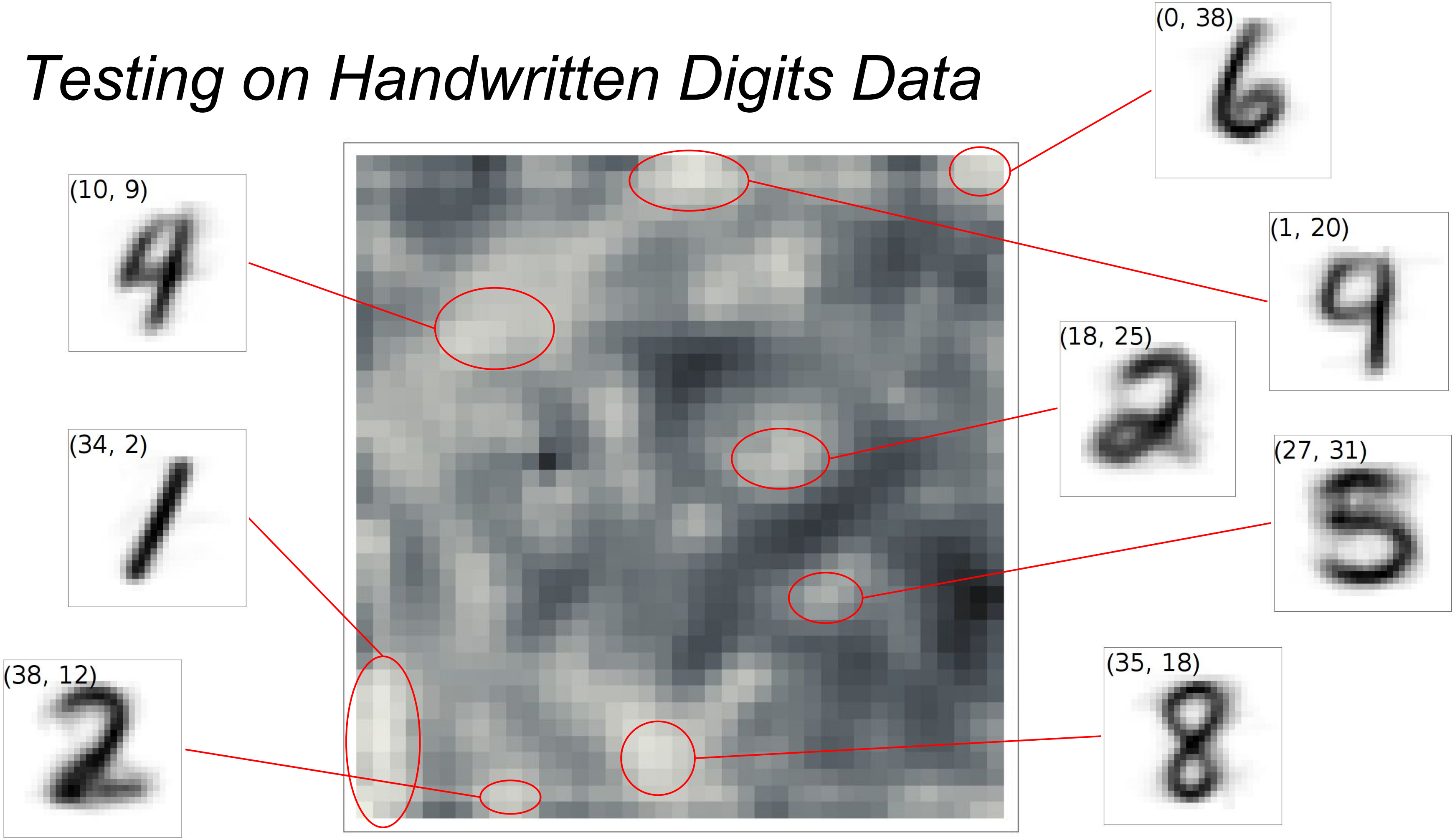



2D feature space



2D topological space

# *Testing on Handwritten Digits Data*



## *Things to do*

- Use SOM to cluster World Development Indicators data
  - cluster centers will represent “prototype” countries
- Fine-tune parameters (learning rate, sigma)
- Test other initialization methods (uniform)
- Test other topological neighborhood functions (step function)

# *References*


- Tan, Pang-Ning, Michael Steinbach, and Vipin Kumar. "Introduction to Data Mining." (2006)
- Kohonen, Teuvo. "Essentials of the self-organizing map." *Neural Networks* 37 (2013): 52-65
- Bullinaria, John A. "Self Organizing Maps: Fundamentals." http://www.cs.bham.ac.uk/~jxb/NN/l16.pdf