CIS501 – Lecture 15

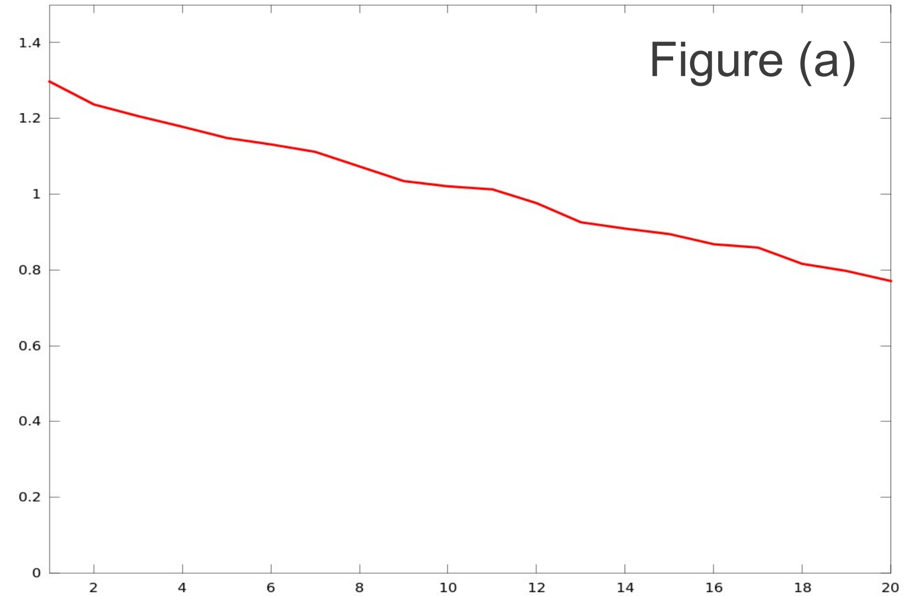
Woon Wei Lee
Fall 2013, 10:00-11:15pm,
Sundays and Wednesdays

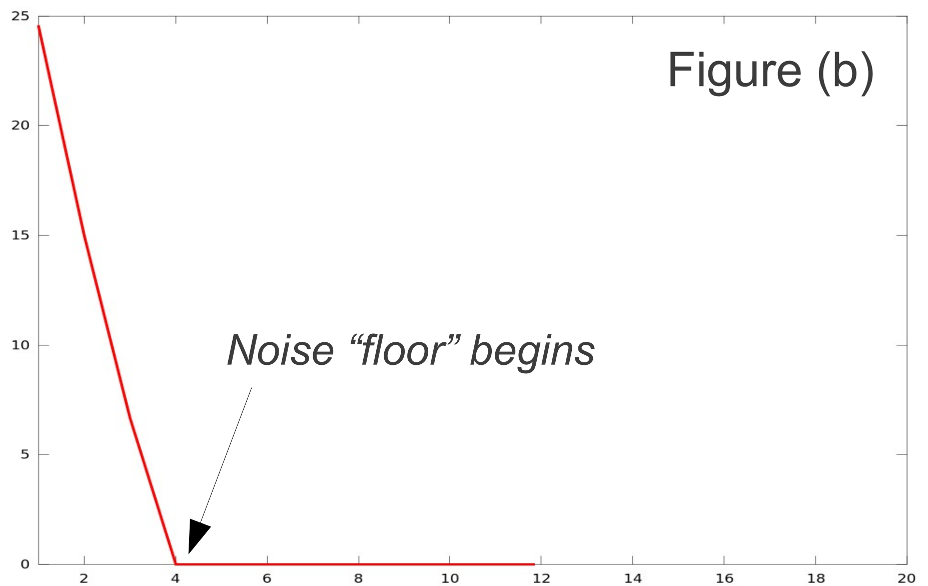
Masdar INSTITUTE

For today:

- Self-Organizing maps
- Unsupervised learning wrap-up
- Presentations
  - Azhar Ahmed
  - Khawla Masood Al Dhaheri

PCA (Cont'd)





i.e. The principle components are given by the *eigenvectors* of the covariance matrices

- For an *n*-dimensional dataset, there will be *n* such eigenvectors.
- Eigenvectors are mutually orthonormal
- Matrix of eigenvectors is hence a *rotation matrix*

Project upon a subset of these eigenvectors → dimensionality reduction

- The λ value → the eigenvalues of the covariance matrix
- Sorting these and plotting gives the *singular spectrum* (**SS**)

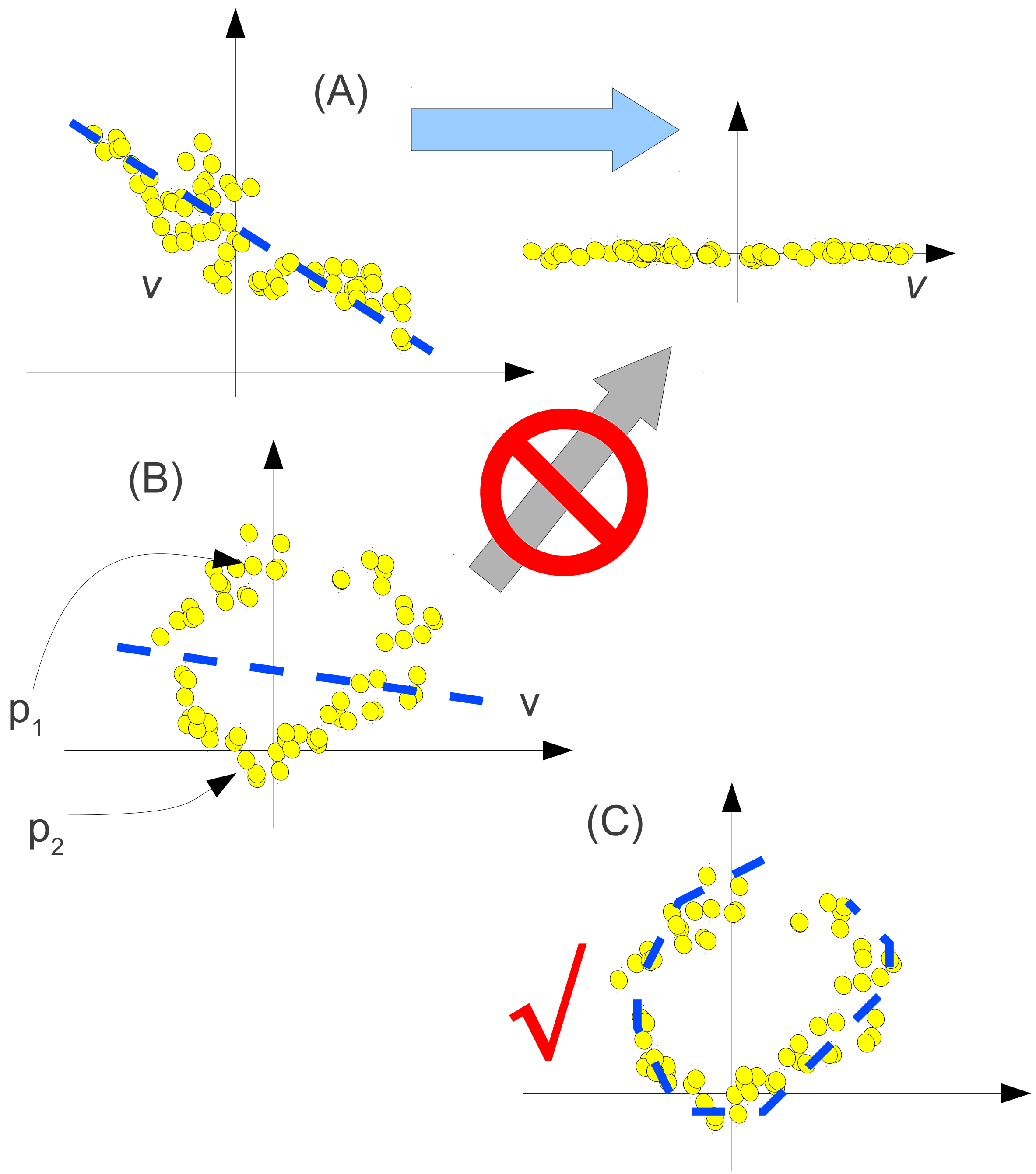
Figures (a) and (b):

(a) SS corresponding to 20 dimensional white noise

(b) SS for 3 dimensional white noise embedded in 20 dimensional space

- Note the noise "floor" in figure (b).

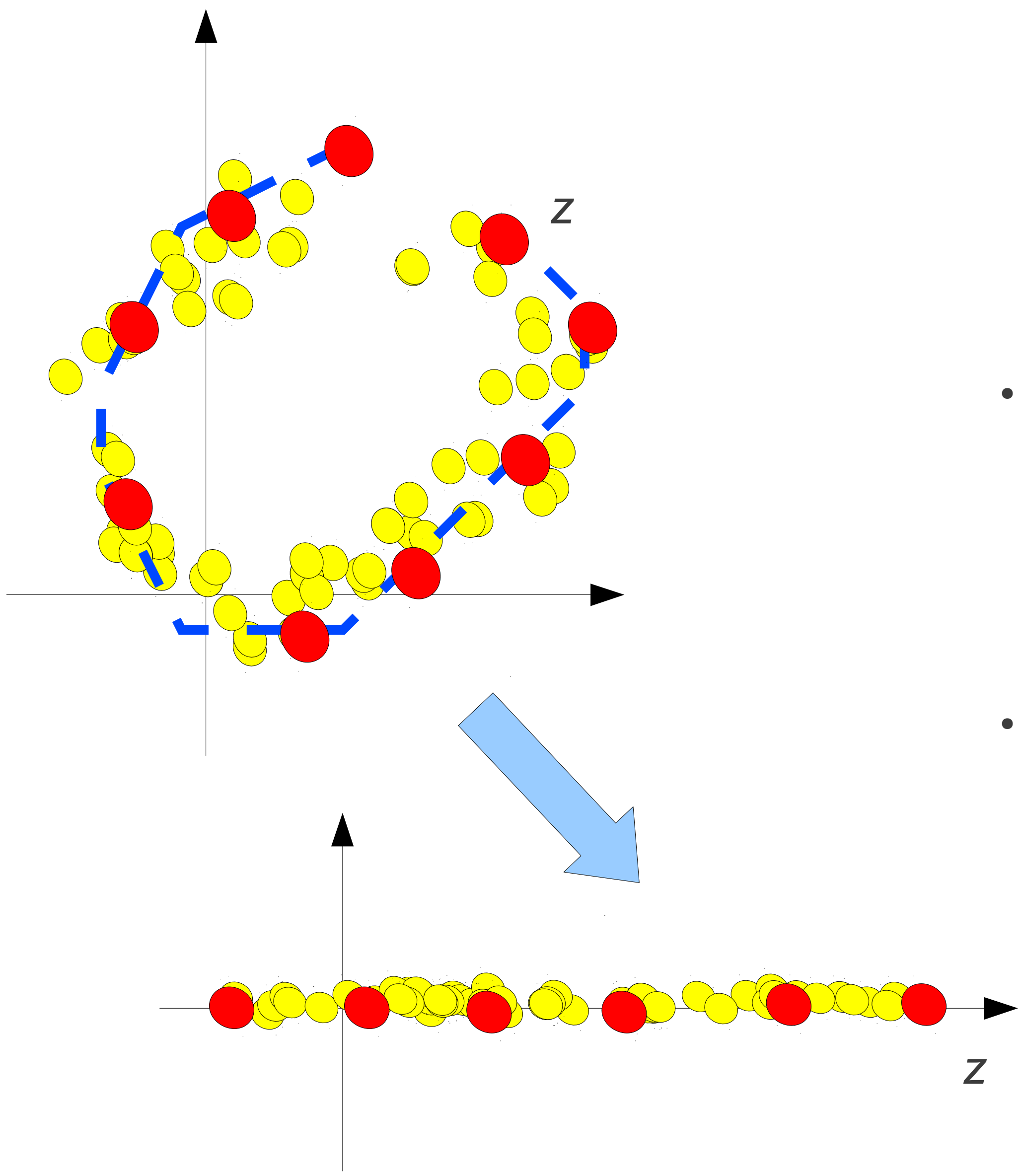
Linear vs Non-linear



- **Linear techniques like PCA project data onto *hyper-plane***
  - Works well when data contains linear relationships
  - (e.g. (A) → two dimensional case, relationship written:

    y=mx+c (+noise)
- **Alternative: *nonlinear* structure/data**
  - (B) is an example
  - Linear projections → weird things can happen
  - For e.g., if projected onto v, p_1 and p_2 will look around the same
- **Solution:**
  - Use nonlinear axes! (C)

Kohonen's self-organizing map



- **How can we create this nonlinear mapping?**
  - Use nonlinear mapping function → difficult!
  - Create a virtual axis of points (on left)
- **This is the principle of the "self-organizing-map" or SOM**
  - Objective: learn nonlinear axis "z"
  - Unfold to form a traditional visualization surface
- **Strategy**
  - Embed a string of markers or nodes along the axes
  - Optimize position of nodes so that each approximates position of nearby points
  - Projection: points attached to closest node

Masdar INSTITUTE

SOM learning algorithm

- **Two main concepts:**
  - Perform *clustering* to fix location of each node
  - Concept of a *neighborhood* so that topological relationships preserved
- **Algorithm:**

(1) Initialize position of the node

(2) Cycle through the input vectors

(3) Determine similarity between the input vector and each of the node

(4) Identify the node that produces the smallest distance.

(5) Update the position of *all* nodes as follows:

$$v^{(t+1)} = v^{(t)} + \eta^{(t)}\Theta^{(t)}(x^{(t)} - v^{(t)})$$

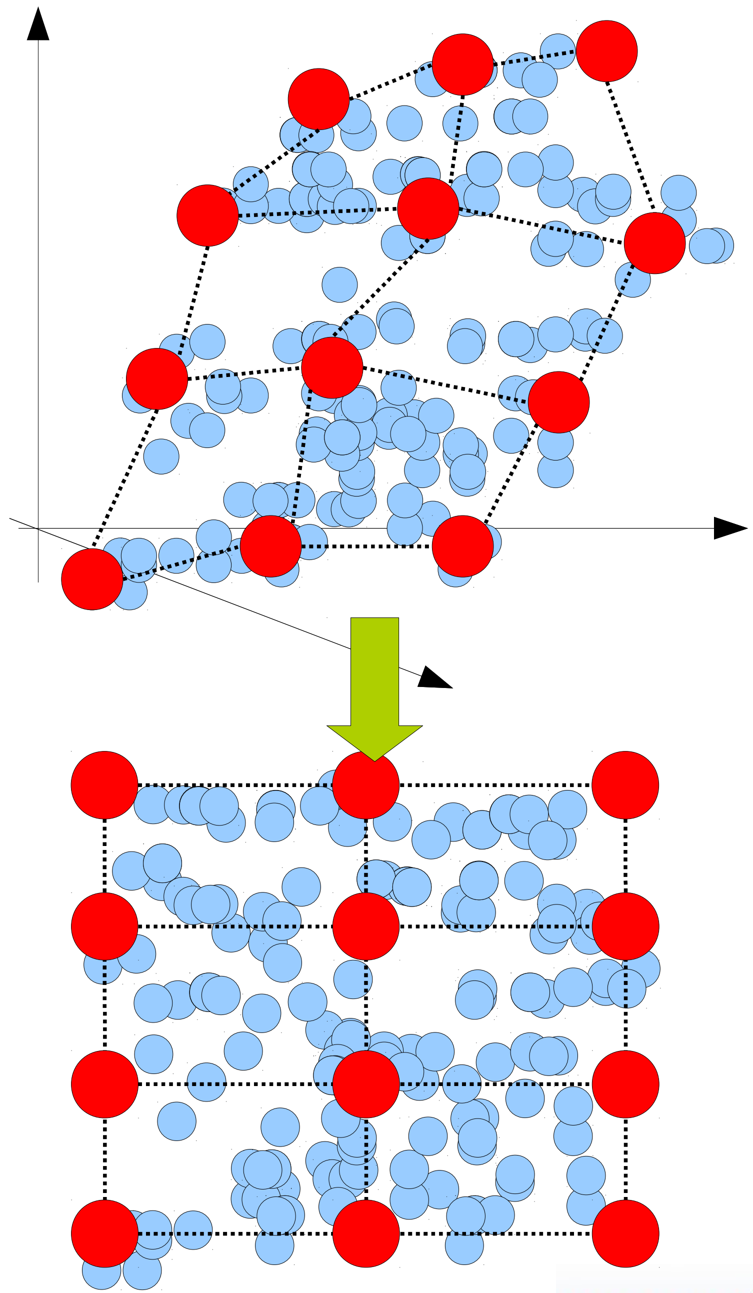
(6) Repeat from 2

- $v^{(t)}$ – position of the node at time t
- $\eta^{(t)}$ – Learning rate (time dependent)
- $\Theta^{(t)}$ – Neighborhood function
- $x^{(t)}$ – Input vector selected at time t

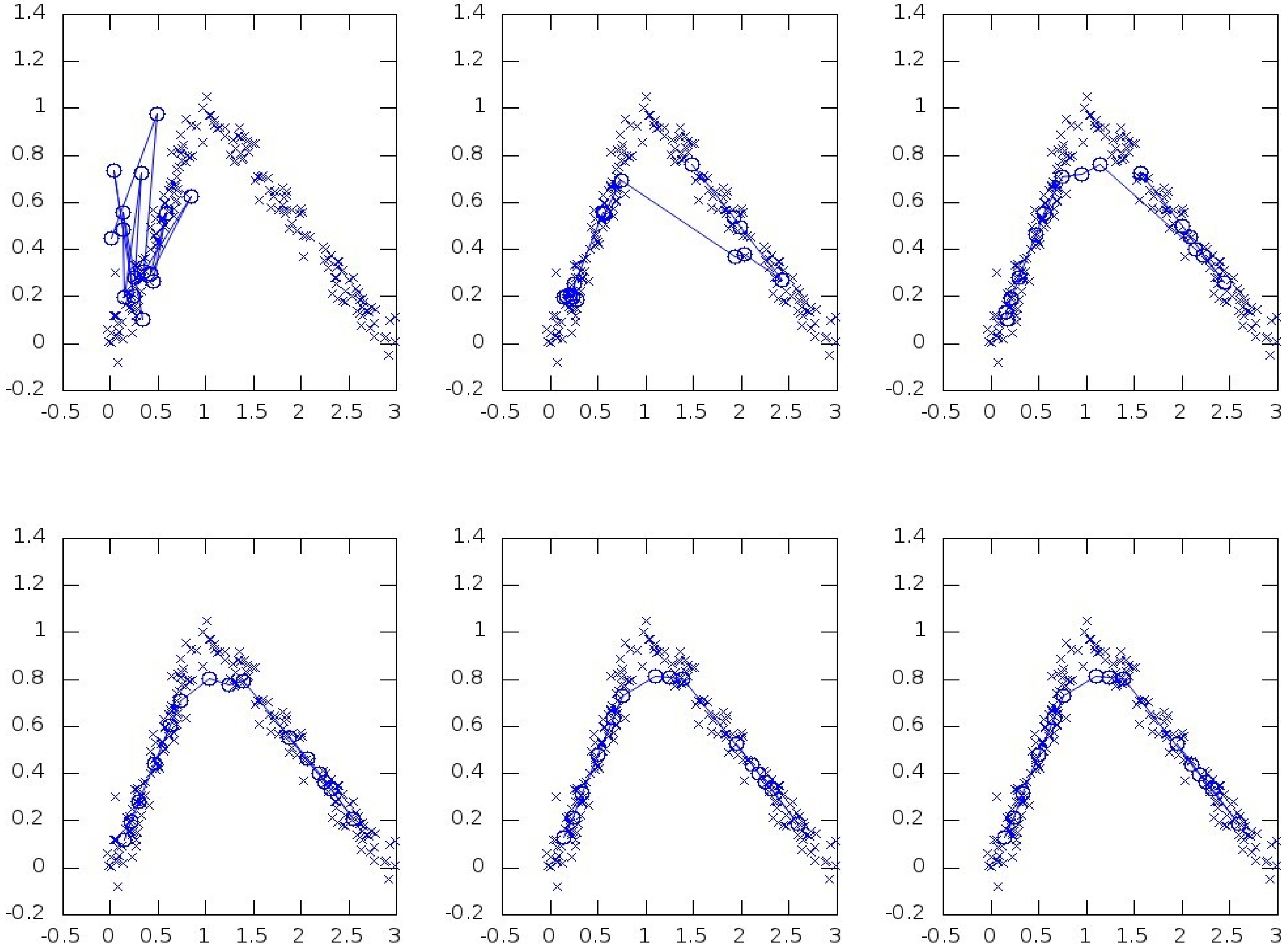
Masdar INSTITUTE

(Cont'd)

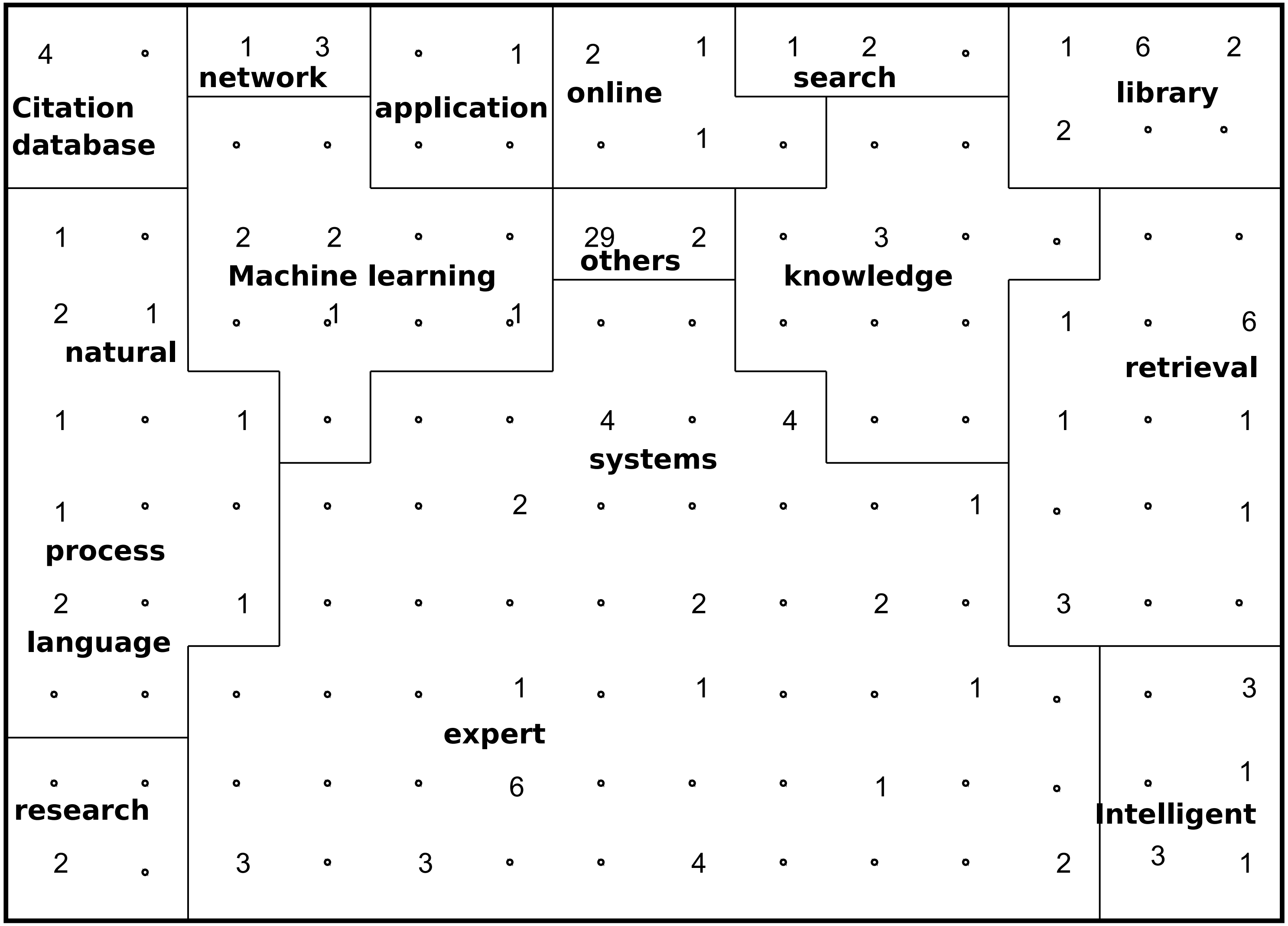
- **Additional notes:**
  - The learning rate $\eta^{(t)}$ and neighborhood function $\Theta^{(t)}$ are shown to be time dependent
  - $\eta^{(t)}$ decreases monotonically as training progresses, while $\Theta^{(t)}$ also reduces in size
  - This allows the nodes to
    - First learn the overall topological structure of the data
    - Smaller values allow "fine-tuning" so that nodes match the data distribution
  - SOMs can be built with any number of dimensions
  - BUT typically 2, which is the most practical and useful
  - Hence, the map in "Self-Organizing-***Map***"



Simple demonstration



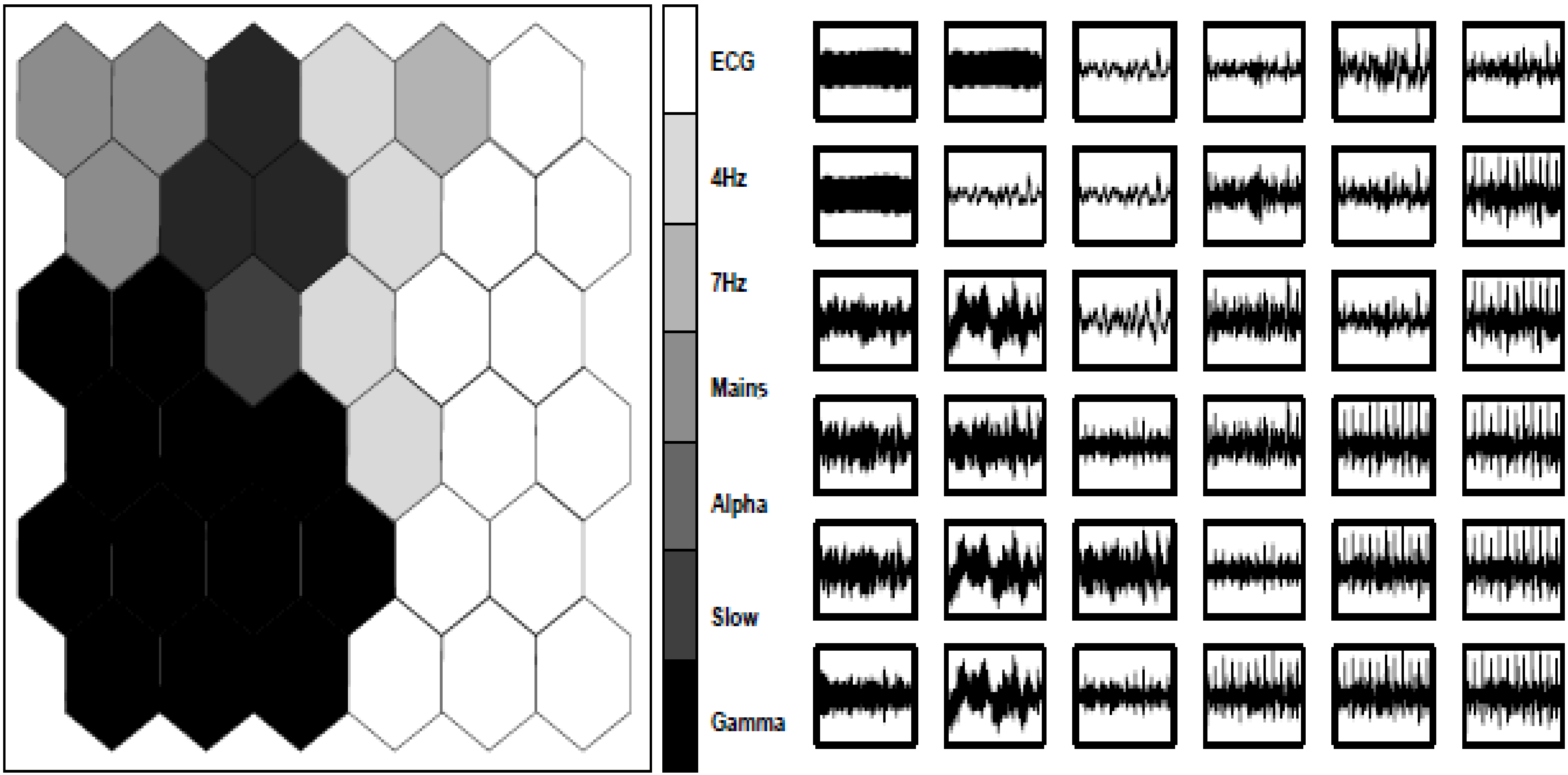
Visualization example: map of 140 A.I. Documents



Source: http://faculty.cis.drexel.edu/~xlin/presentations/KohonenMapping(asis99).ppt

Masdar INSTITUTE

Visualization example: Brain Signals



(a) Class assignments

(b) ICs corresponding to the SOM lattice neurons

Unsupervised learning: Round-up

- **Time was tight, but we covered two main concepts:**
  1. Clustering:
     - Partitional clustering → *k*-means and *k*-centroids
     - Hierarchical clustering → methods UPGMA and the "Saitou-Nei" method
  2. Visualization
     - Linear (PCA)
     - Non-linear (SOM)
- **Compared to supervised algorithms, unsupervised algorithms tend to be exploratory**
  - Descriptive in nature rather than prescriptive
  - Often richer probabilistic interpretations than supervised learning

    (not possible to cover in this introductory course)