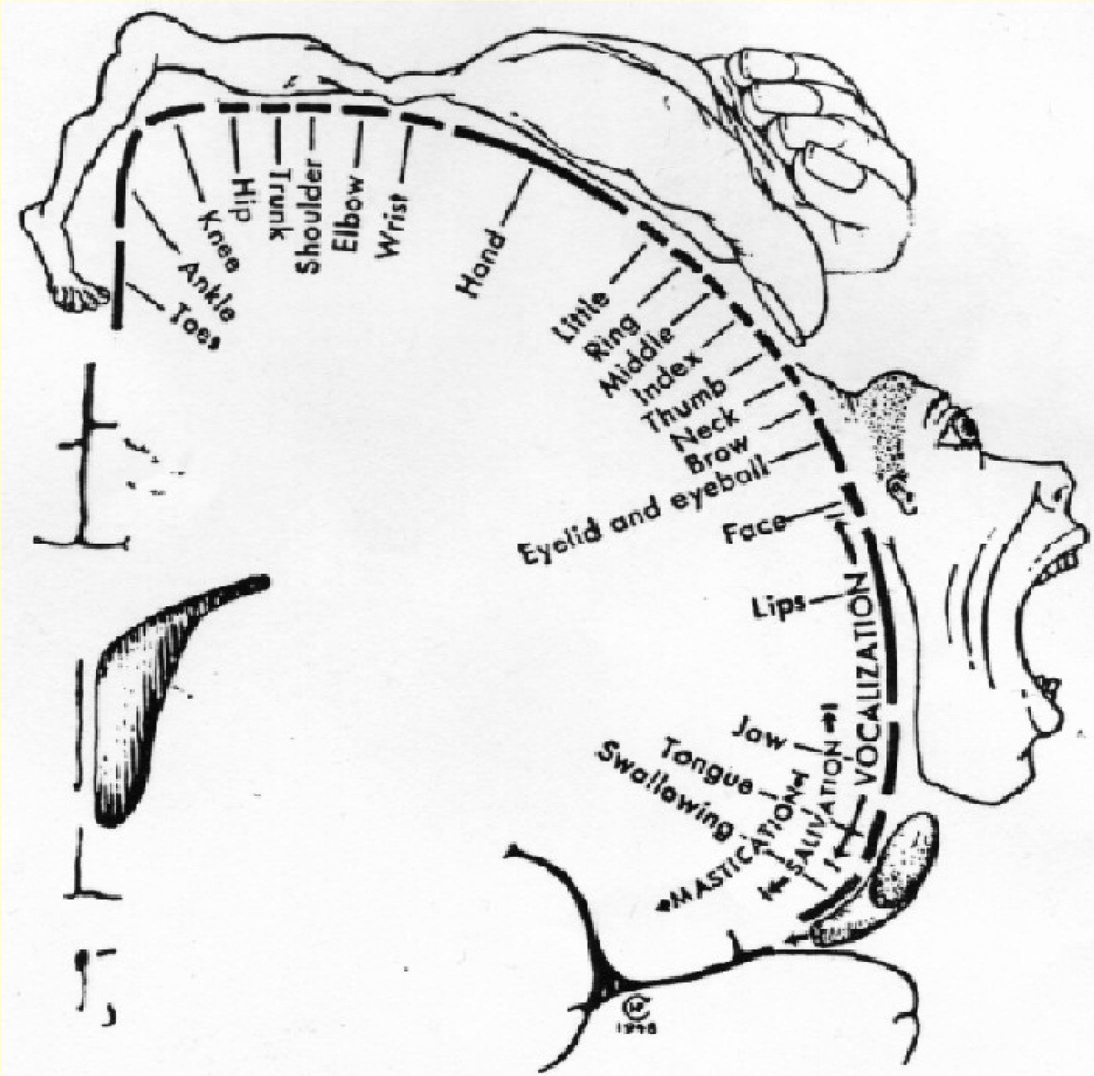
CSCI 252: Neural Networks
Prof. Levy

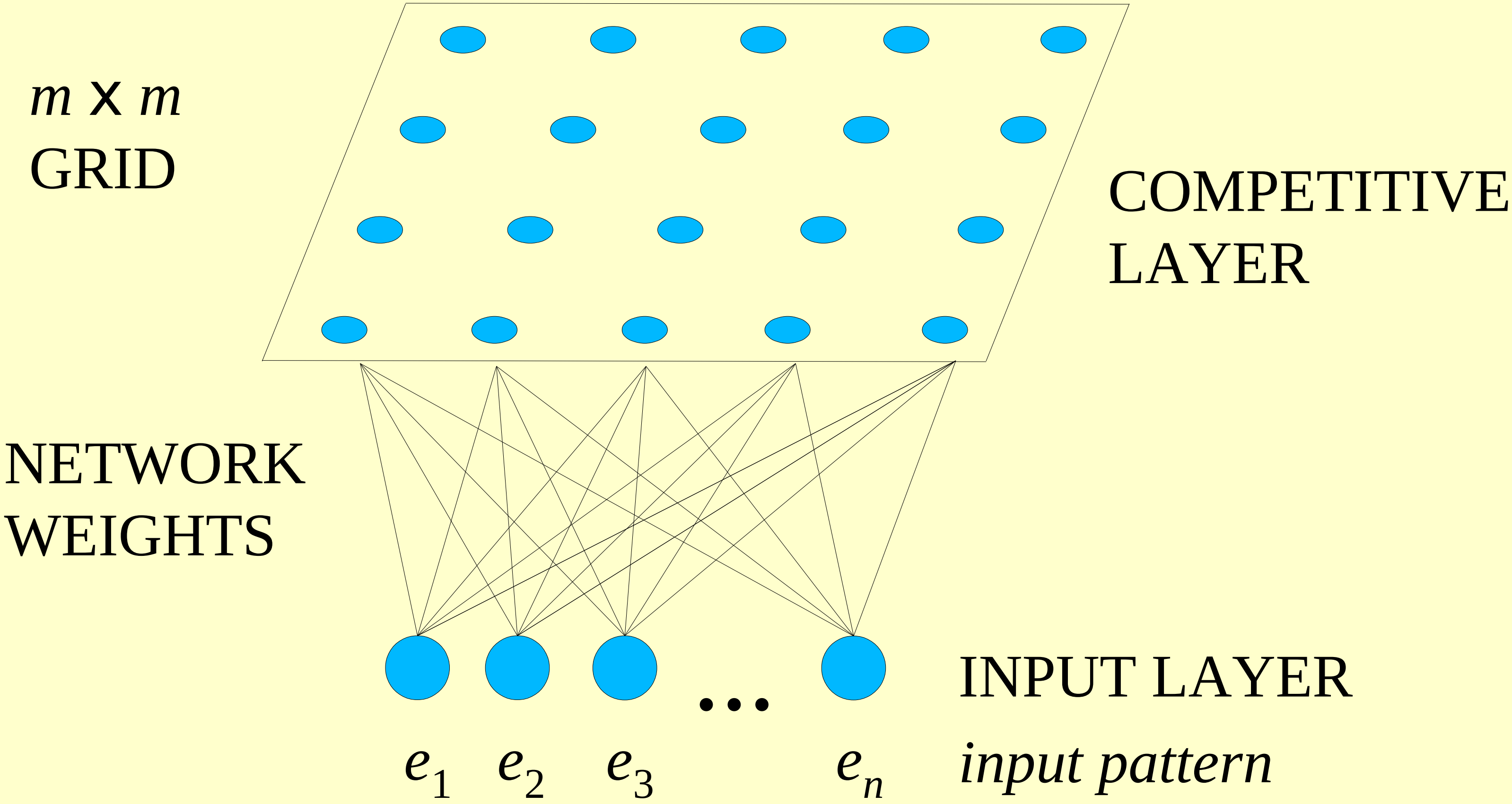
Architecture #1:
Kohonen's Self-Organizing Map
(Kohonen 1988;
notes from Dayhoff 1990)

Self-Organizing (Feature) Maps

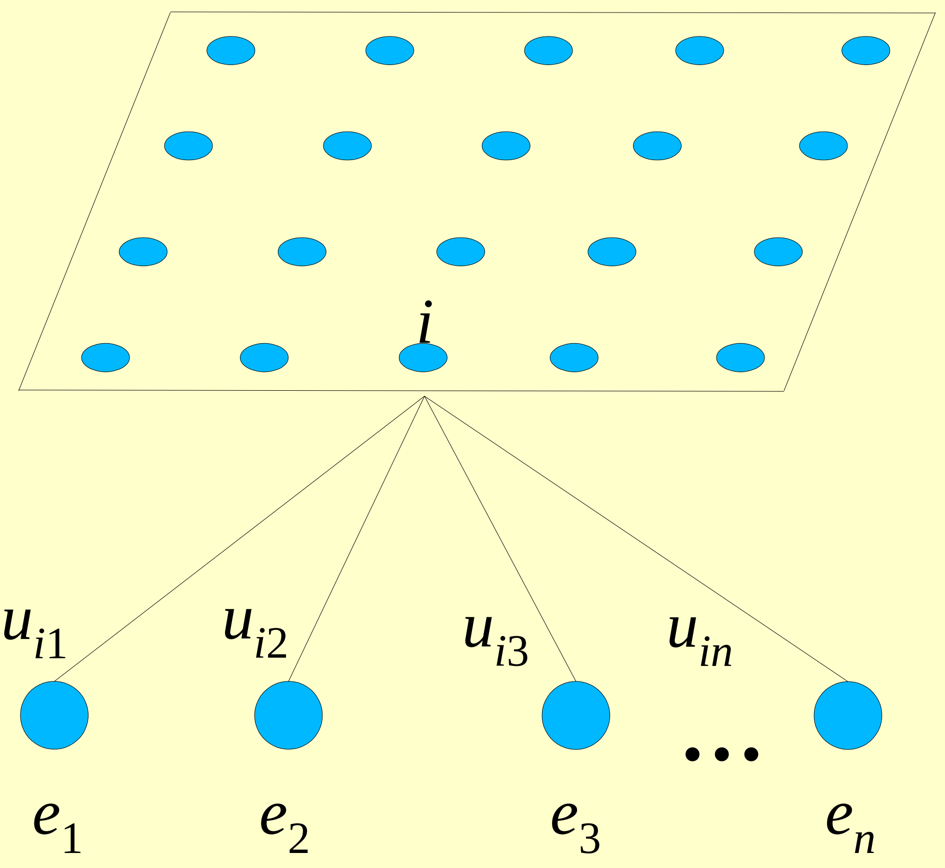
- An *unsupervised* learning method
- Goal is to reveal underlying relationships in high-dimensional data by mapping to lower (2) dimensions
- Has some justification in neuroscience (somatotopic maps; noun/verb localization)


Toes
Ankle
Knee
Hip
Trunk
Shoulder
Elbow
Wrist
Hand
Little
Ring
Middle
Index
Thumb
Neck
Brow
Eyelid and eyeball
Face
Lips
VOCALIZATION
Jaw
Tongue
Swallowing
MASTICATION
SALIVATION
VOCALIZATION

Network Architecture



Network Weights



Network Operation

- An *iterative* algorithm: start with some initial conditions (random weights), then repeat till satisfied.
- On each iteration, units U_i "compete" for each input *e*. In practice, we pick an *e* at random.
- Winner *c* is unit whose weight vector u_i is closest to *e* :

$$\min_{i,j} \| E - U_{i,j} \| \equiv \min_{i,j} \sqrt{\sum_k (e_k - u_{ijk})^2}$$

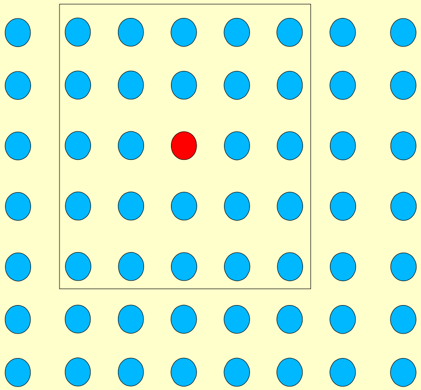
- Adjust weights of winner and units in its "neighborhood".

Weight Adjustment

$$\Delta u_{ij} = \alpha(e - u_{ij})$$ if u_i is in neigbhorhood of c

$$\Delta u_{ij} = 0$$ otherwise

Neighborhood: $d = 2$



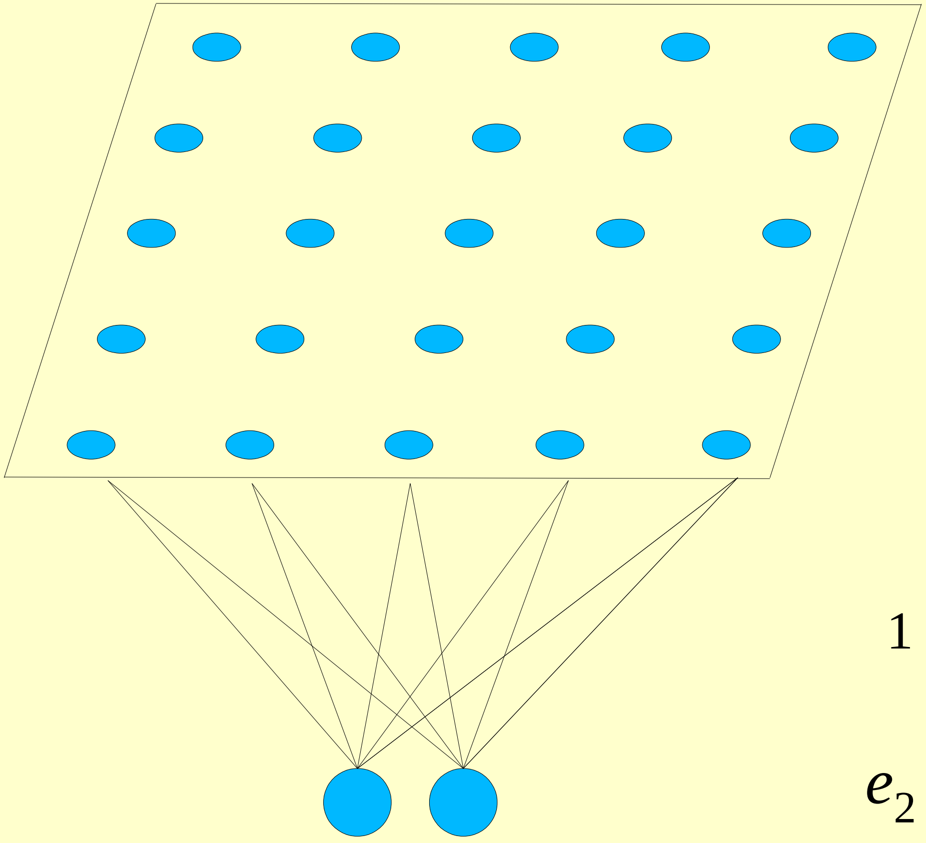
current iteration

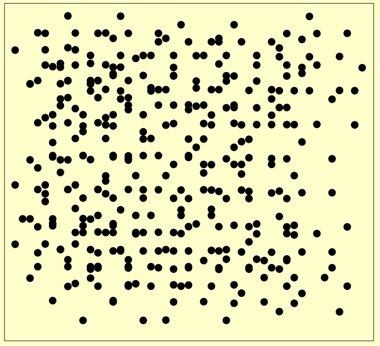
$$\alpha_t = \alpha_0 \left(1 - \frac{t}{T}\right)$$

max iterations

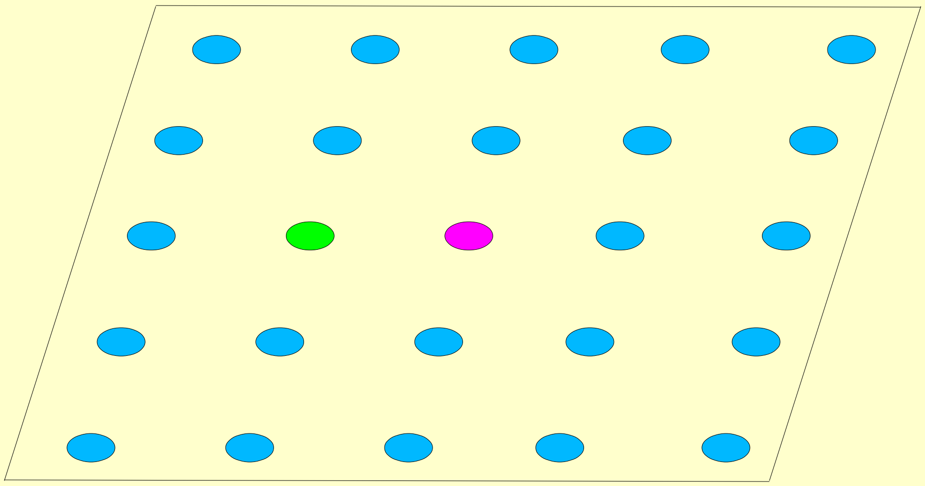
$$d_t = \left\lceil d_0 \left(1 - \frac{t}{T}\right) \right\rceil$$

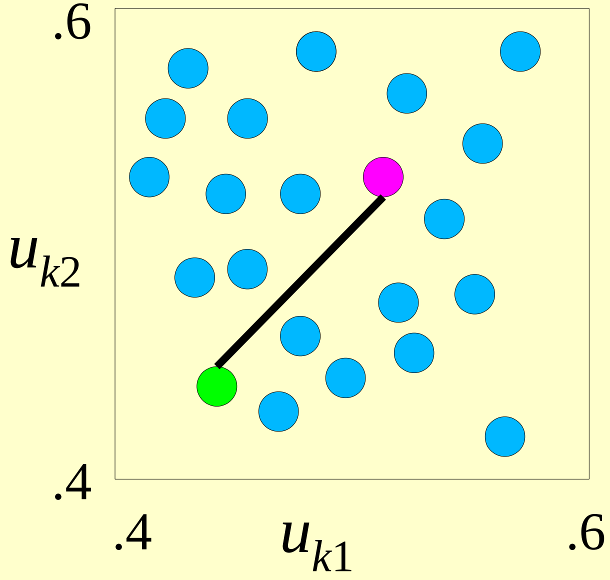
Example: $n = 2$, $m = 5$



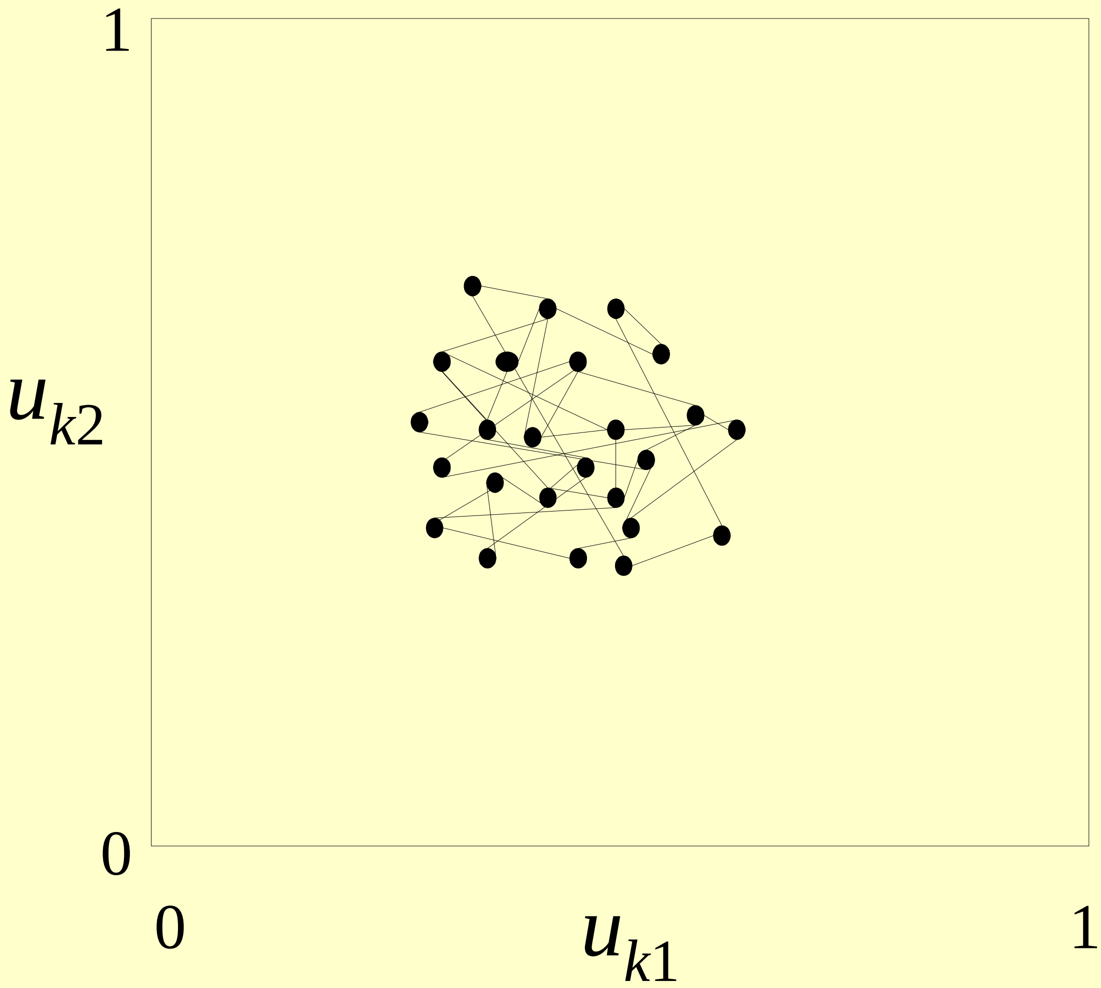


- With n = 2, we have two weights for each unit
- So we can track the learning process as follows:
  - For each unit k, plot its weights u_{k1}, u_{k2} (book confusingly uses w here instead of u.)
  - If two units are adjacent, draw a line between their weight plots

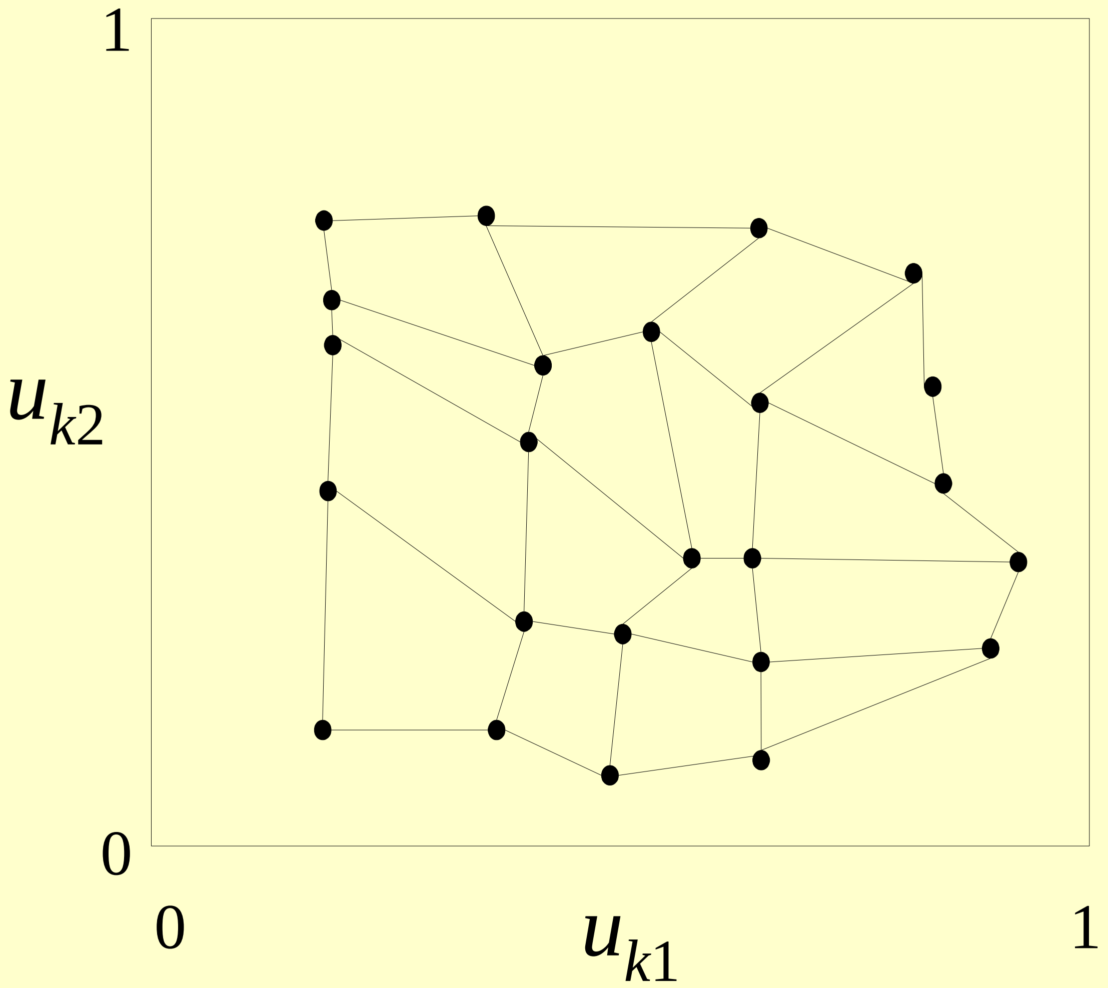




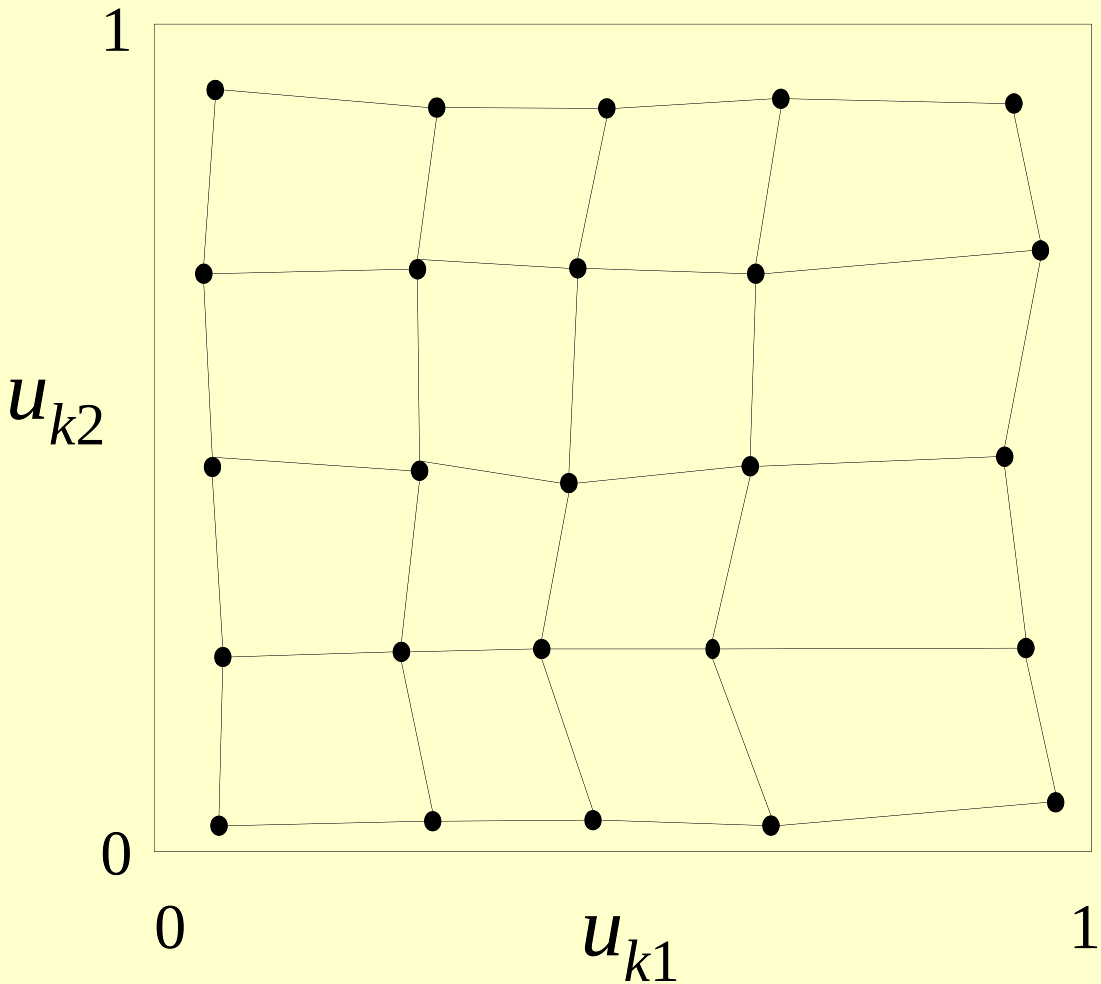
$t = 0$



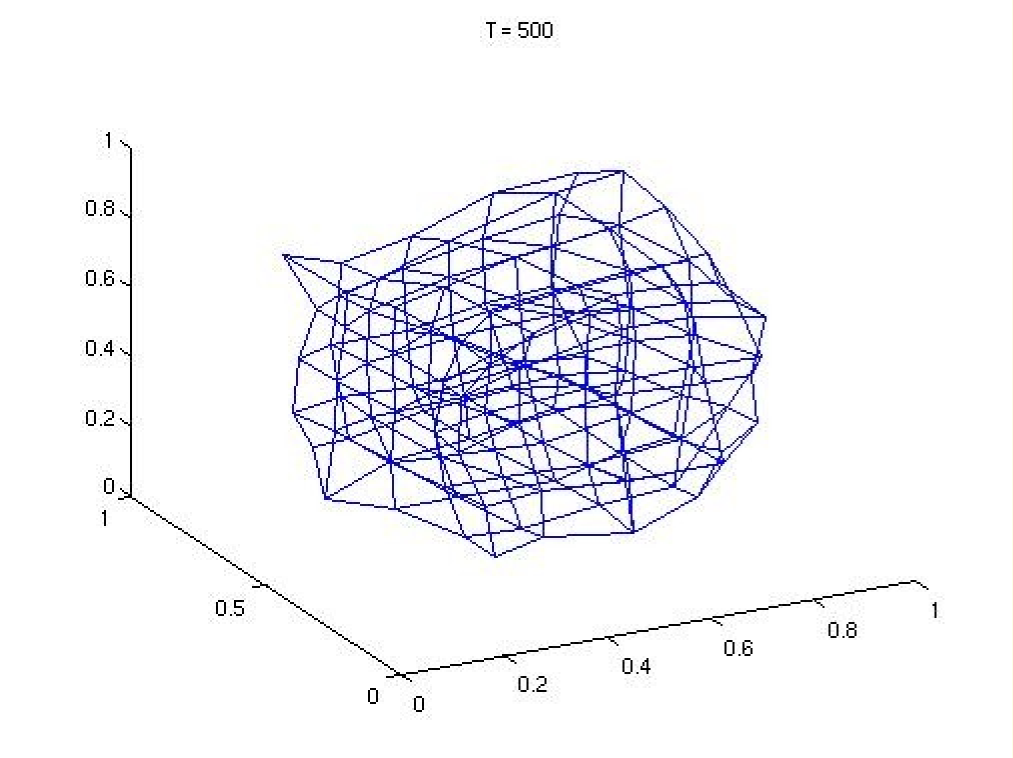
$t = 1000$



$t = 5000$



Three-Dimensional Version



Notes

- These examples are somewhat misleading: for n = 2 (or 3), weights have same dimensions as $m \times m$ (or $m \times m \times m$) grid.
- But in general, this is not the case.
- It is usually more interesting to see how higher-dimensional vector spaces are mapped into $m \times m$ grid.
- For this, we plot each e at the u_i that "wins" it:

e_{15}	e_2	e_6	e_3
e_1	e_{11}	e_{16}	e_8
e_9	e_4	e_5	e_{14}
e_{12}	e_{13}	e_7	e_{10}