Grooming an ant colony

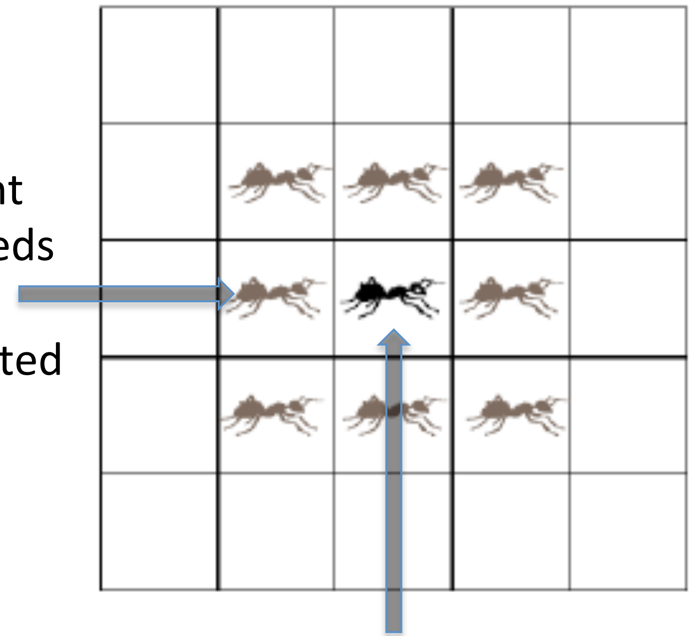
Result = dead ants collected in one or a few places.

Collecting dead ants together -

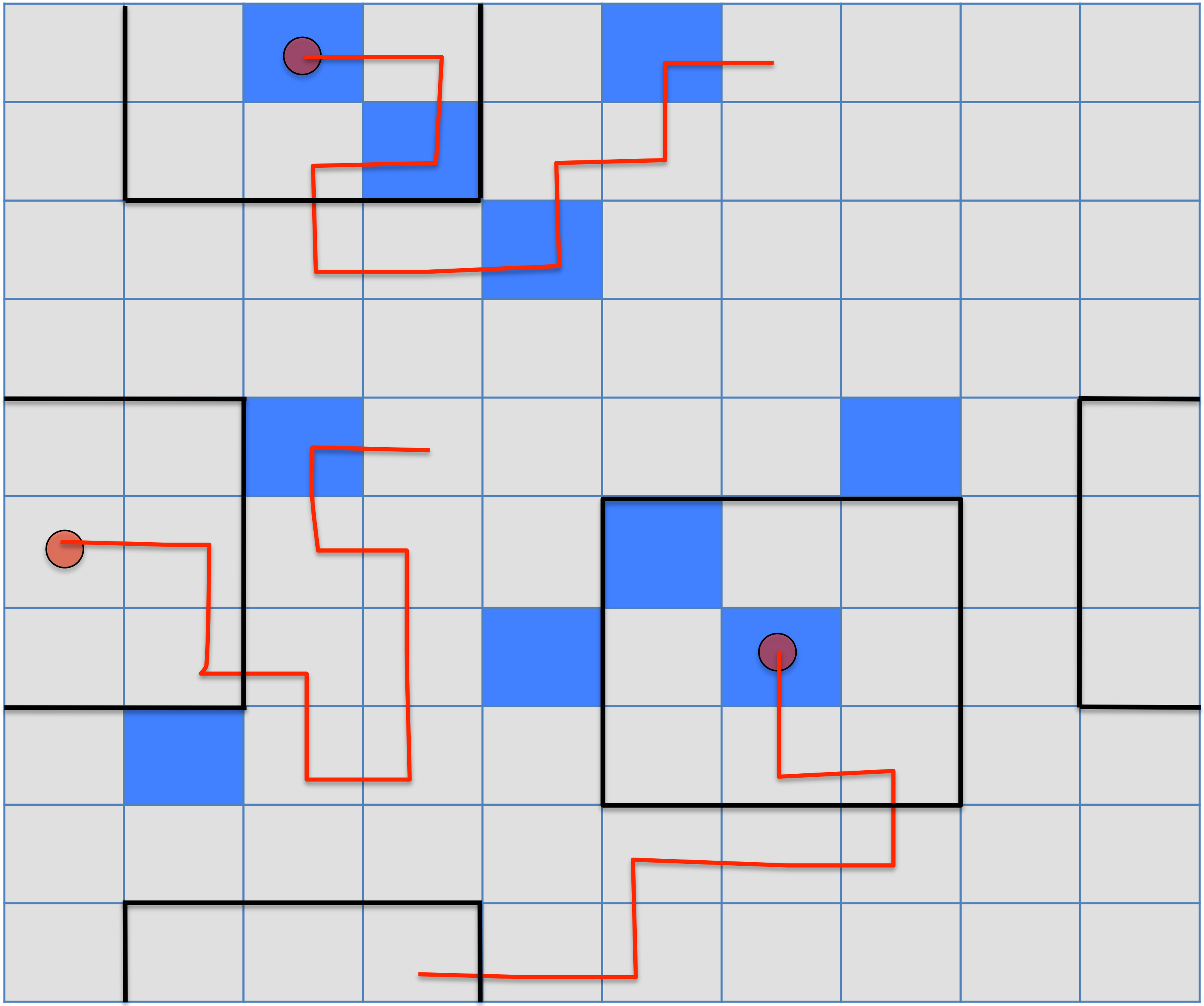
• If you are not already carrying a dead ant, and you bump into one, *pick* it up if you see *few* other dead ants in that area.

• If you are carrying a dead ant, *drop* it if you see *many* other dead ants in that area.

Dead ant that needs to be segregated

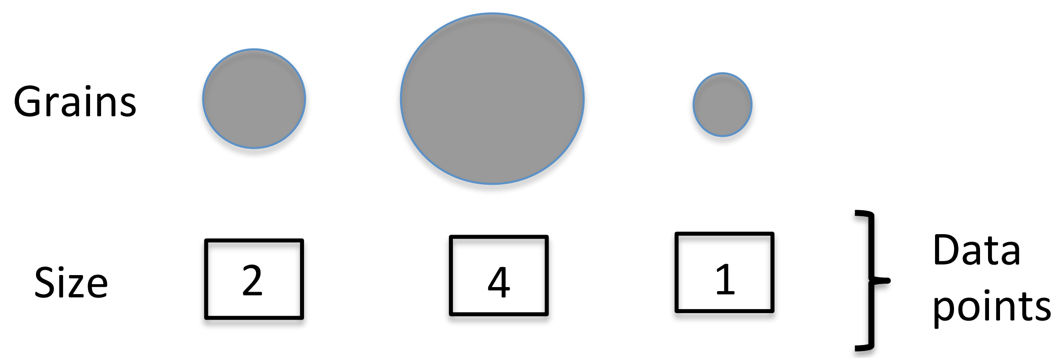


Live ant that wanders and segregates dead ants

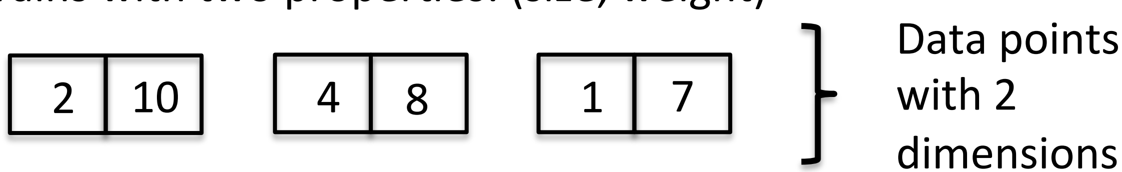


From dead ants to data points

Collecting grains of similar size together



Grains with two properties: (size, weight)



“Grooming” a data set

Collecting similar data points together -

• If you are not already carrying a data point, and if bump into one, *pick* it up if you see *few* other similar data points in that area.

• If you are carrying a data point, *drop* it if you see *many* other similar data points in that area.

• In case of ants, you simply *count* the number of dead ants, as they all have a single common property – they are all dead.

• In case of grains, you *calculate the similarity* between the grain you want to pick up or drop and of those in that area.

• You differentiate between the two viewpoints via the function *f*. In the former, *f* is just the perceived fraction of dead ants, and in the latter it is a similarity between the grains.

Distance between a pair of data points

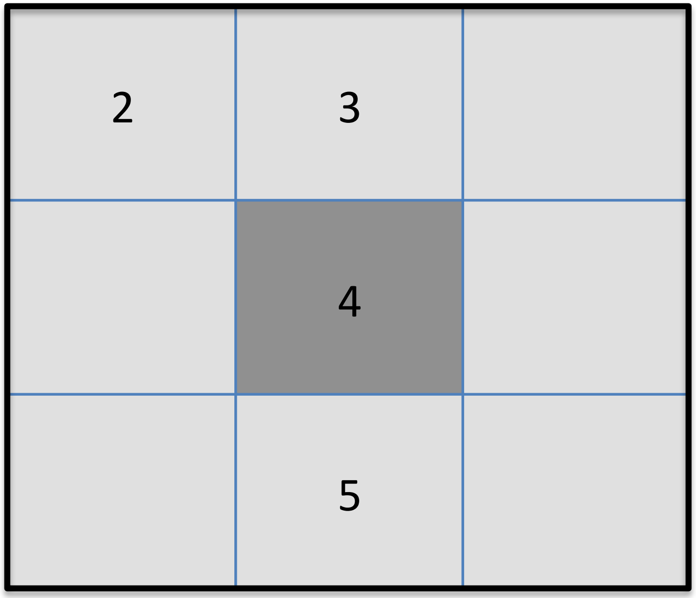
Data Vector X_i

$\boldsymbol{x_{i,1}}$	$\boldsymbol{x_{i,2}}$	$\boldsymbol{x_{i,3}}$	**…**	$\boldsymbol{x_{i,n}}$

Data Vector X_j

$\boldsymbol{x_{j,1}}$	$\boldsymbol{x_{j,2}}$	$\boldsymbol{x_{j,3}}$	**…**	$\boldsymbol{x_{j,n}}$

$$D(x_i, x_j) = \sqrt{\sum_{k=1}^{n} \left(x_{i,k} - x_{j,k}\right)^2}$$



$$D(4,2) = \sqrt{(4-2)^2} = 2$$

$$D(4,3) = \sqrt{(4-3)^2} = 1$$

$$D(4,5) = \sqrt{(4-5)^2} = 1$$

$$D([4,1],[2,1]) = \sqrt{(4-2)^2 + (1-1)^2} = 2$$

$$D([4,1],[3,6]) = \sqrt{(4-3)^2 + (1-6)^2} = 5.1$$

$$D([4,1],[5,4]) = \sqrt{(4-5)^2 + (1-4)^2} = 3.16$$

Similarity between a data point and its neighborhood

$$f\left(x_i\right)=\begin{cases}\dfrac{1}{s^2}\displaystyle\sum_{x_j\in Neigbh_{s\times s}(x_i)}\left(1-\dfrac{D\left(x_i,x_j\right)}{\alpha}\right) & \text{if } f>0\\ 0 & \text{otherwise}\end{cases}$$

where,

$f\left(x_i\right)$: Similarity function (ranges between 0 and 1)

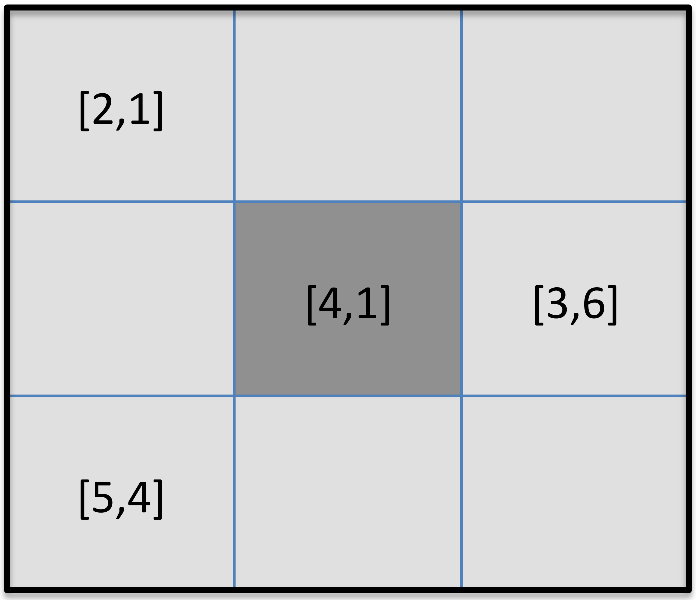
$Neigbh_{s\times s}(x_i)$: Neighborhood square of x_i

$D\left(x_i,x_j\right)$: Euclidean distance between x_i and x_j

α: Scale of dissimilarity or discrimination factor

$\sum$: Sum over terms within $(\ )$, where each term corresponds to the similarity between x_i and a neighbor x_j

Example: $f(x_i)$ for a single data item



$$D([4,1],[2,1]) = \sqrt{(4-2)^2 + (1-1)^2} = 2$$

$$D([4,1],[3,6]) = \sqrt{(4-3)^2 + (1-6)^2} = 5.1$$

$$D([4,1],[5,4]) = \sqrt{(4-5)^2 + (1-4)^2} = 3.16$$

$x_i = [4,6]$, the middle cell

$$\alpha = 4$$

$$f(x_i) = \frac{1}{3^2}\left[\left(1-\frac{2}{4}\right)+\left(1-\frac{5.1}{4}\right)+\left(1-\frac{3.16}{4}\right)\right]$$

$$= \frac{1}{9}\left[1-0.5+1-1.275+0.79\right]$$

$$= \frac{1}{9}\left[0.431\right]$$

$$= 0.04789$$

Picking-up and Dropping

$$p_p(x_i) = \left(\frac{k_1}{k_1 + f(x_i)} \right)^2$$

$$p_d(x_i) = \left(\frac{f(x_i)}{k_2 + f(x_i)} \right)^2$$

$p_p(x_i)$: probability that x_i will be picked

$p_d(x_i)$: probability that x_i will be dropped

k_1, k_2: threshold constants (> 0 for simplicity)

p_p ranges between $\left(\frac{k_1}{k_1 + 1} \right)^2$ and 1

p_d ranges between 0 and $\left(\frac{1}{k_2 + 1} \right)^2$

$f << k_1 \Rightarrow p_p \approx 1$

$f >> k_1 \Rightarrow p_p \approx 0$

$f << k_2 \Rightarrow p_d \approx 0$

$f >> k_2 \Rightarrow p_d \approx 1$

Choosing parameter values

- α determines the threshold value of $D(x_i, x_j)$ beyond which similarity between x_i and x_j is 0.
- k_1 determines the minimum value of $p_p \geq 0$.
- k_2 determines the maximum value of $p_d \leq 1$.

(See p.233 and p.259 of your textbook for some example values)

Ant Clustering Algorithm

- Create a R x C grid
- Strew your data points across the grid
- Randomly place N ants in the grid, with no more than one ant occupying a cell
- Do the following for a given maximum number of iterations:
  - For every ant in the grid, in their respective cells:
    - If ant is not carrying a data point and its cell contains one, x_i, then
      - Compute $f(x_i)$
      - Pick up item with probability $P_p(x_i)$
    - If ant is carrying a data point x_i and its cell is free, then
      - Compute $f(x_i)$
      - Drop item with probability $P_d(x_i)$
    - Walk ant to a random neighboring cell that is not occupied by another ant