



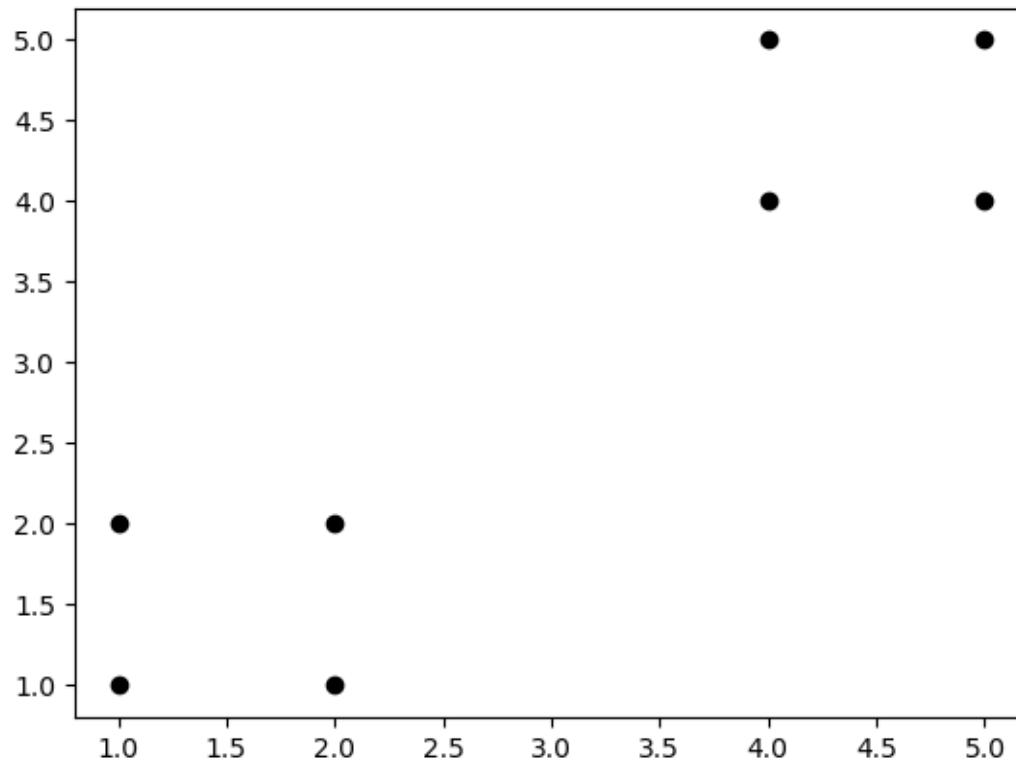
Self-Organizing Maps

Randy Cahya Wihandika, S.ST., M.Kom.

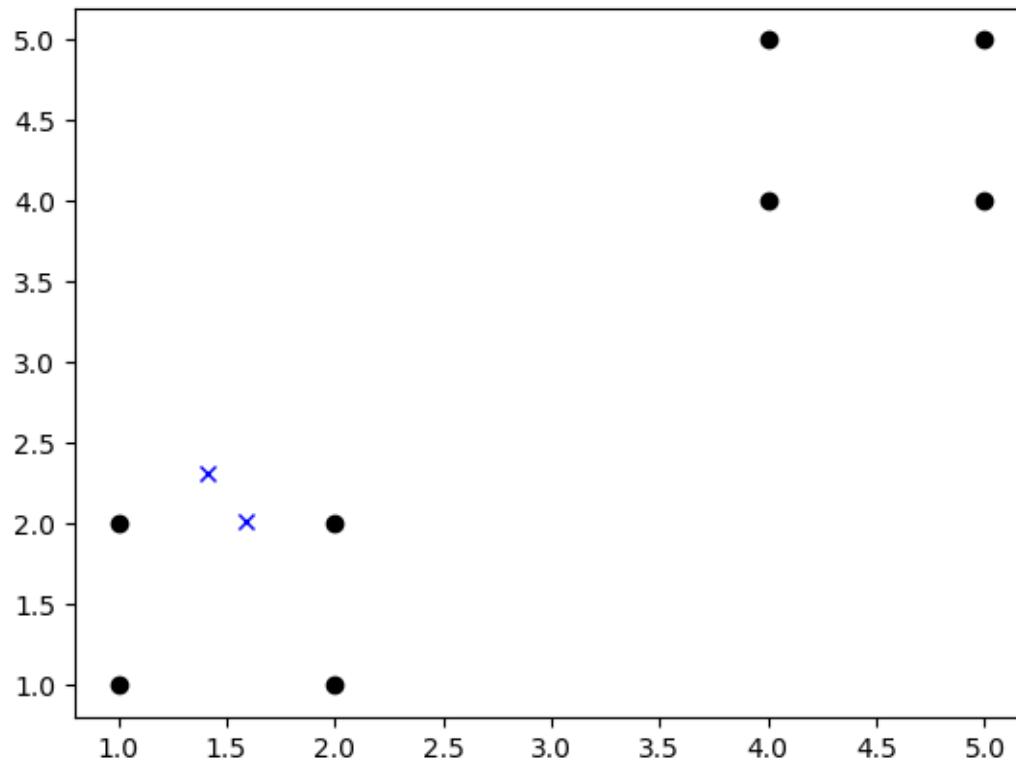
Kohonen SOM

- Digunakan untuk melakukan *clustering* sehingga terjadi secara *unsupervised*

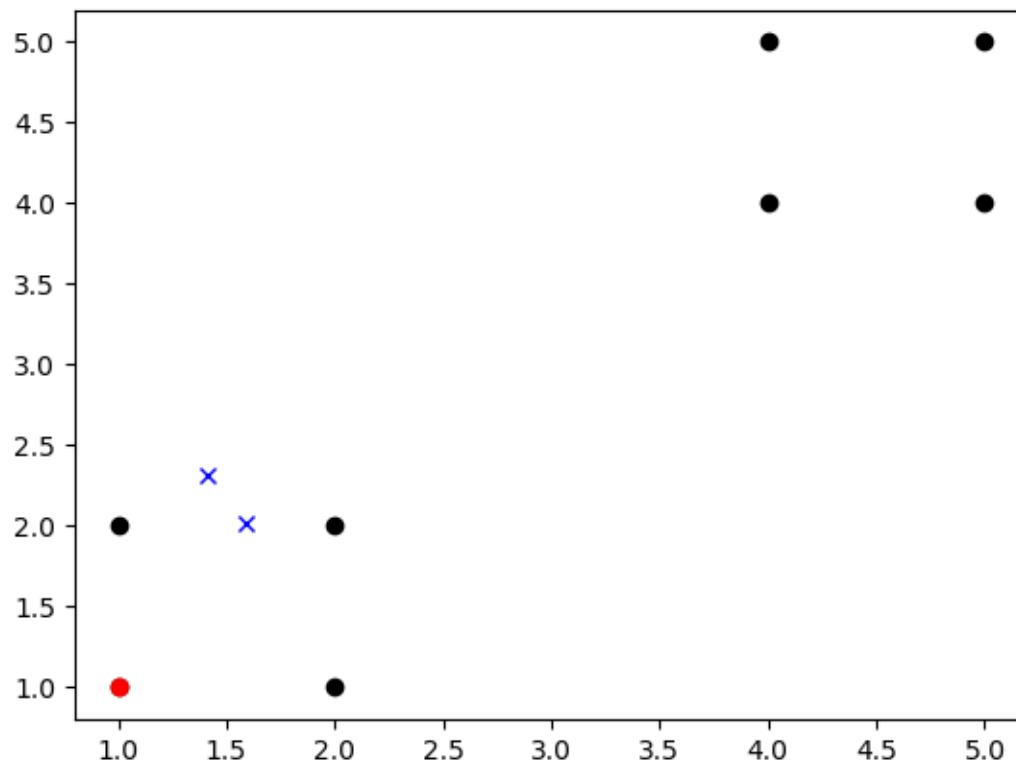
Epoch: 0



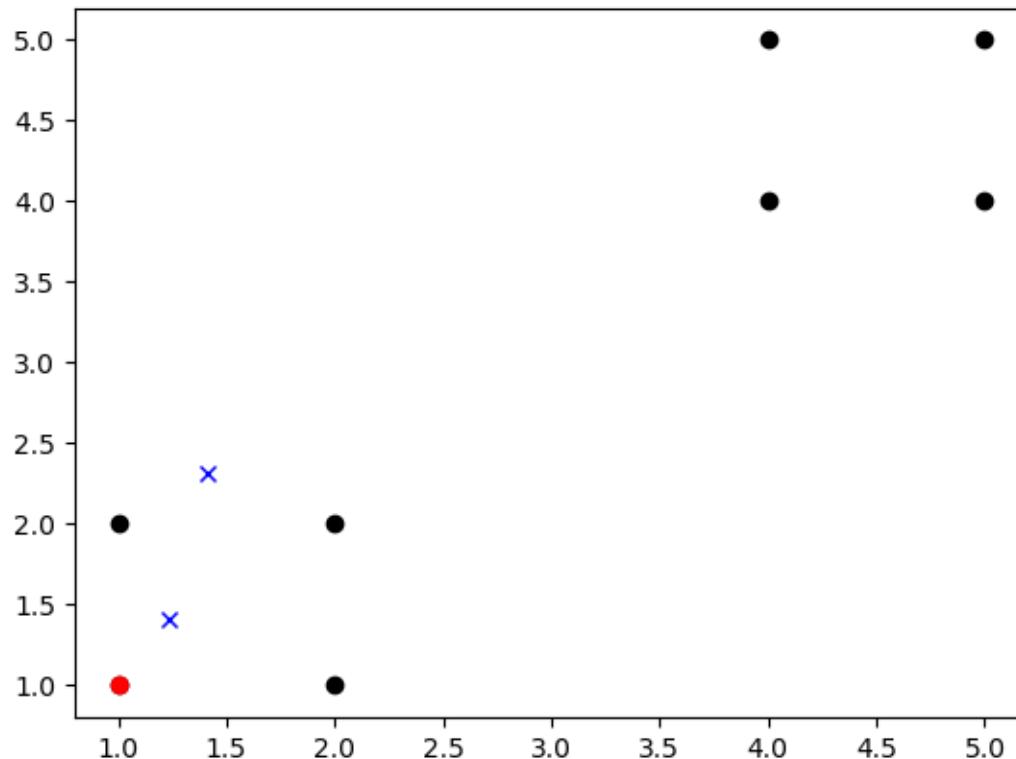
Epoch: 0



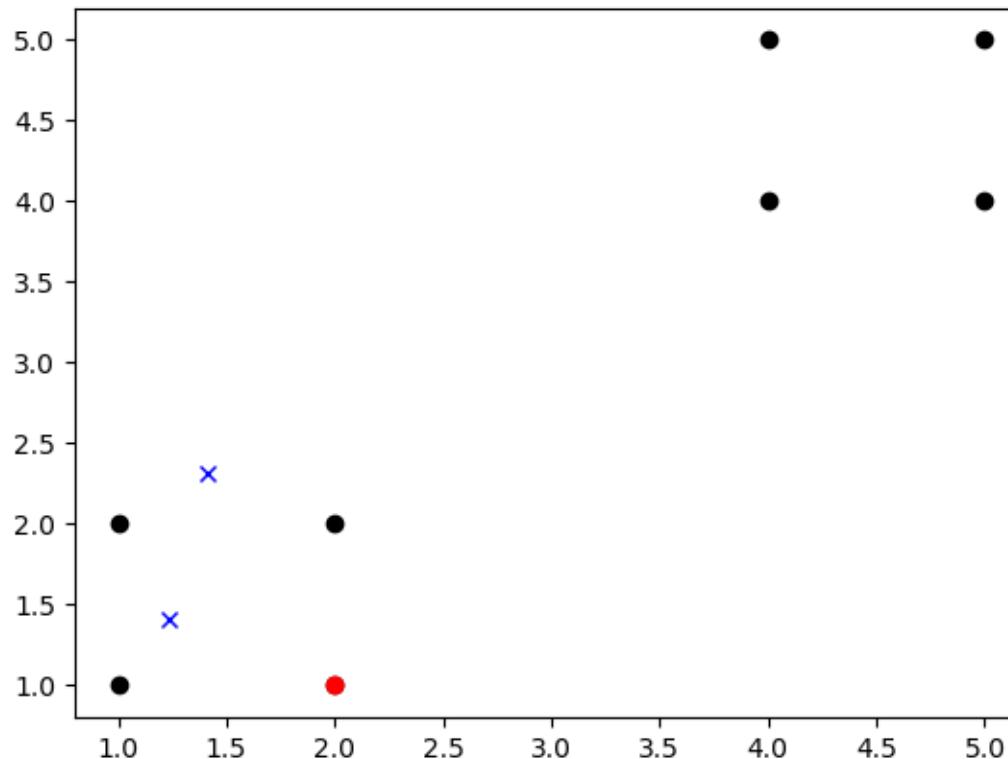
Epoch: 1, data: 1



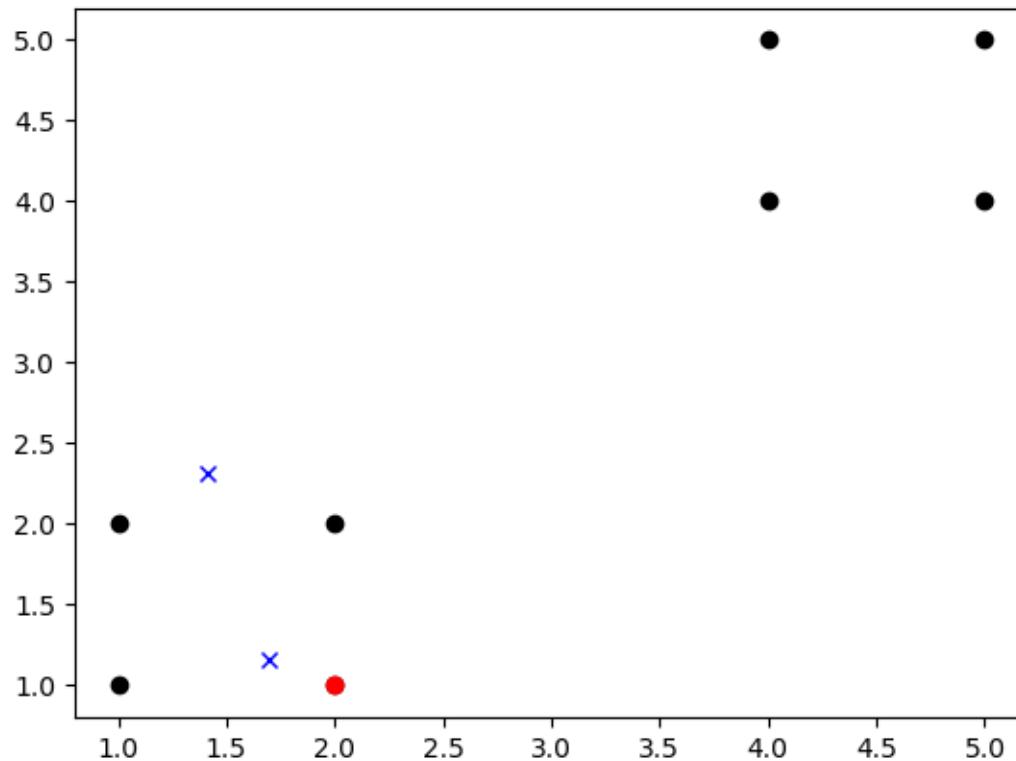
Epoch: 1, data: 1



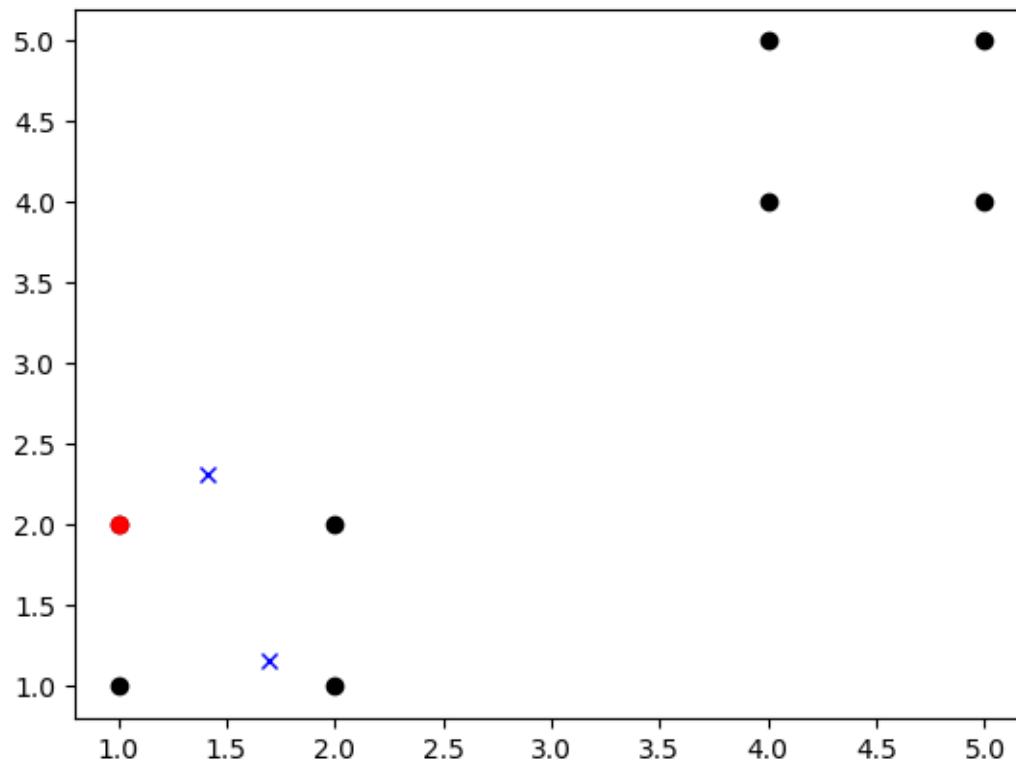
Epoch: 1, data: 2



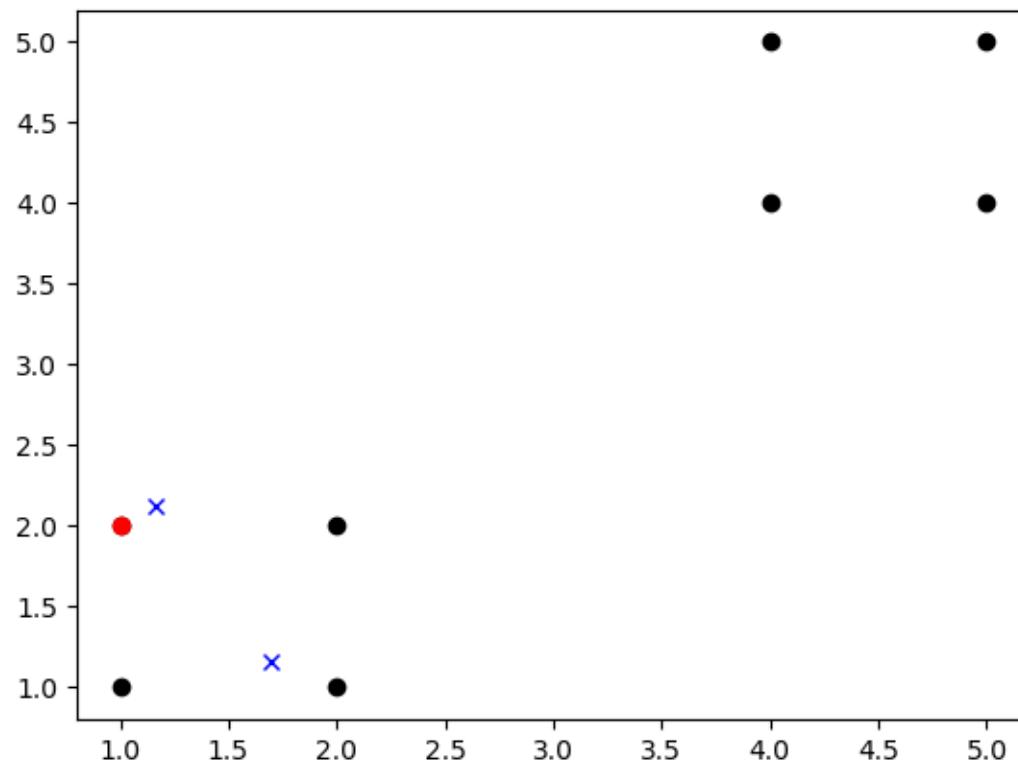
Epoch: 1, data: 2



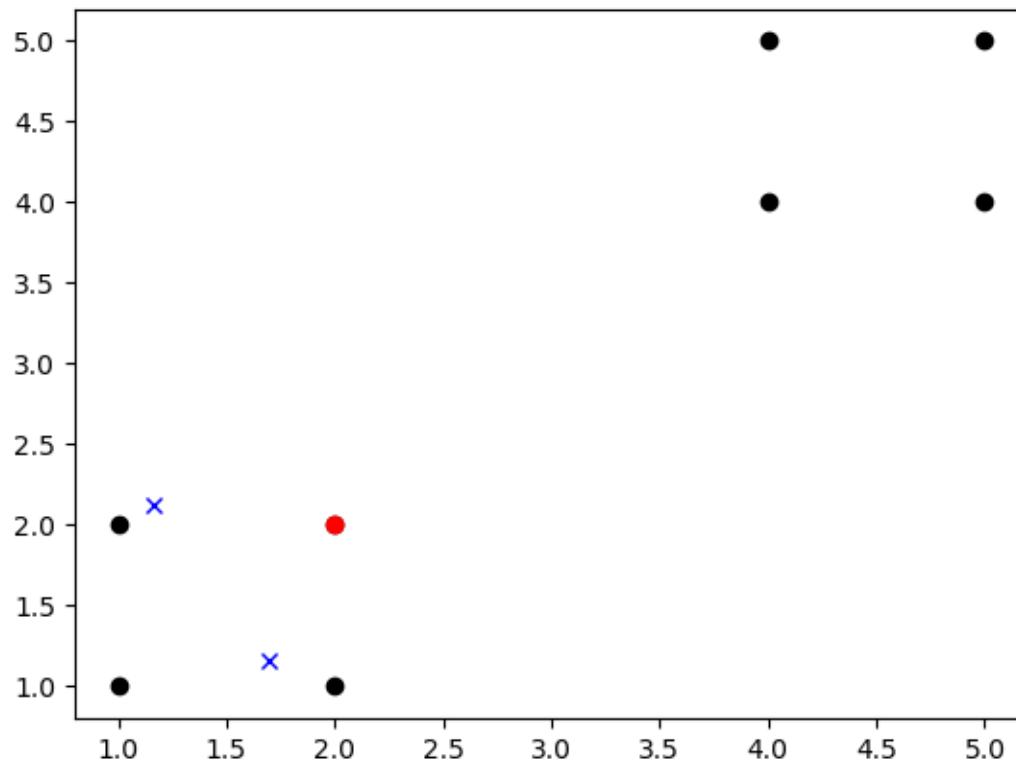
Epoch: 1, data: 3



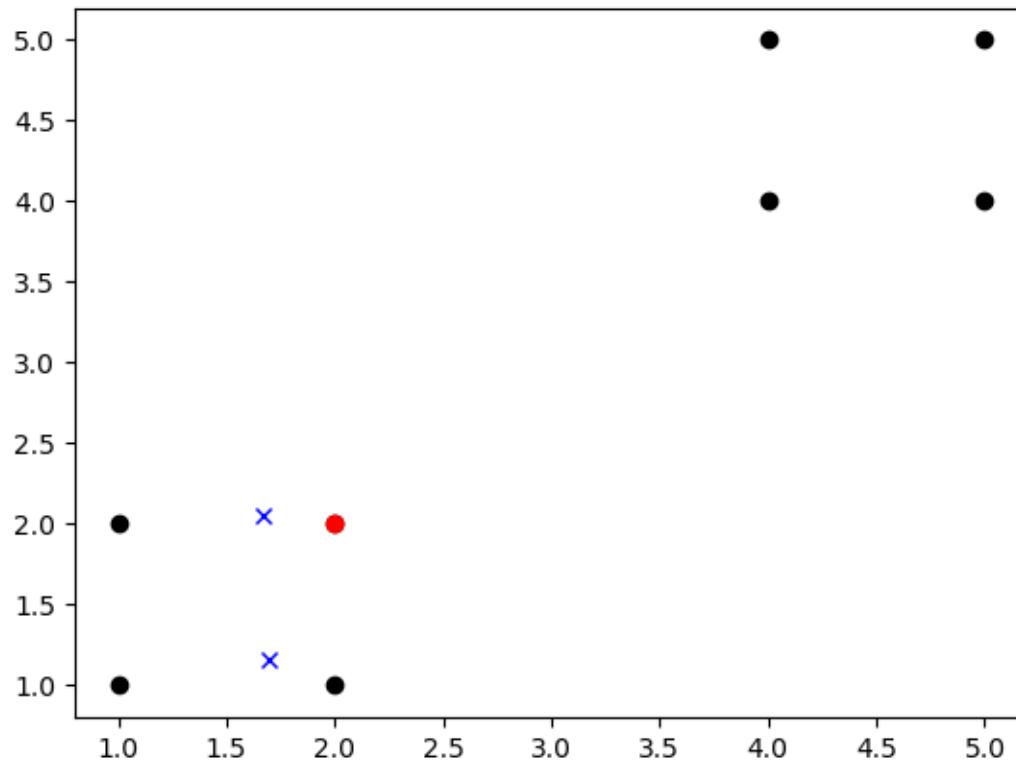
Epoch: 1, data: 3



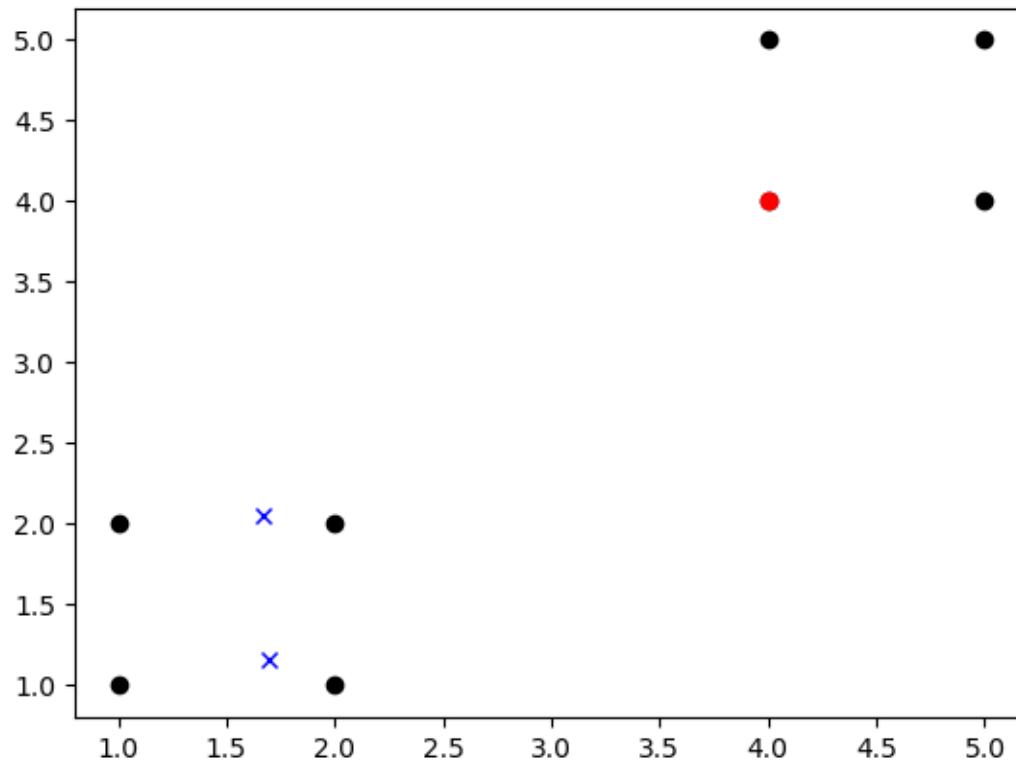
Epoch: 1, data: 4



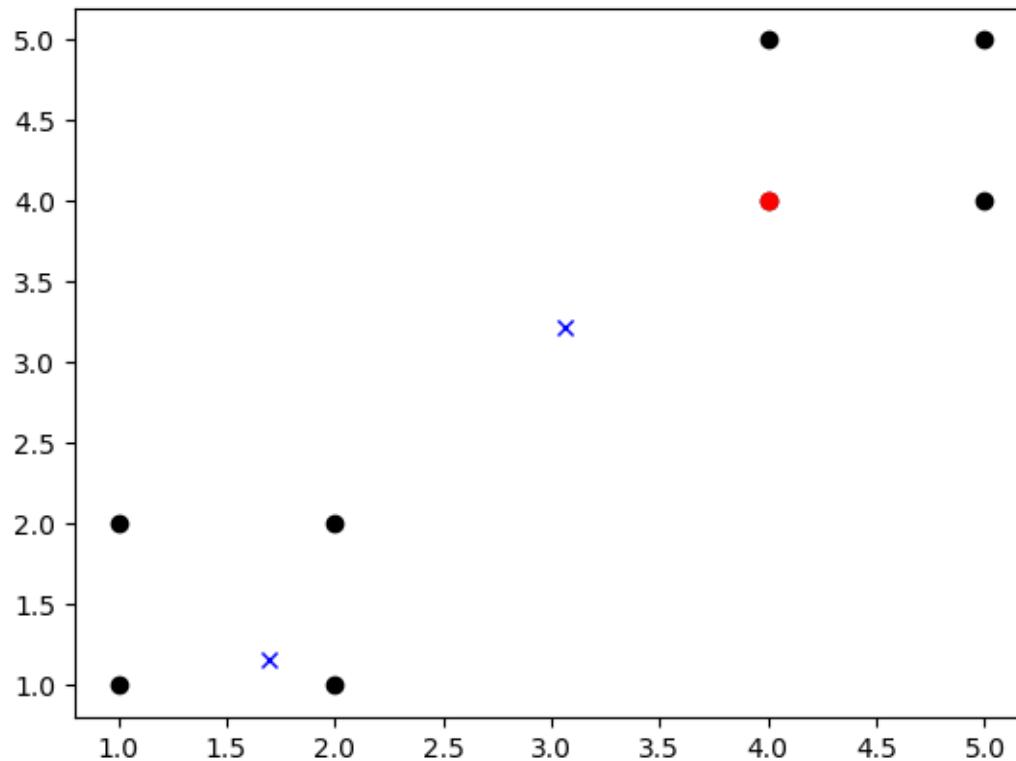
Epoch: 1, data: 4



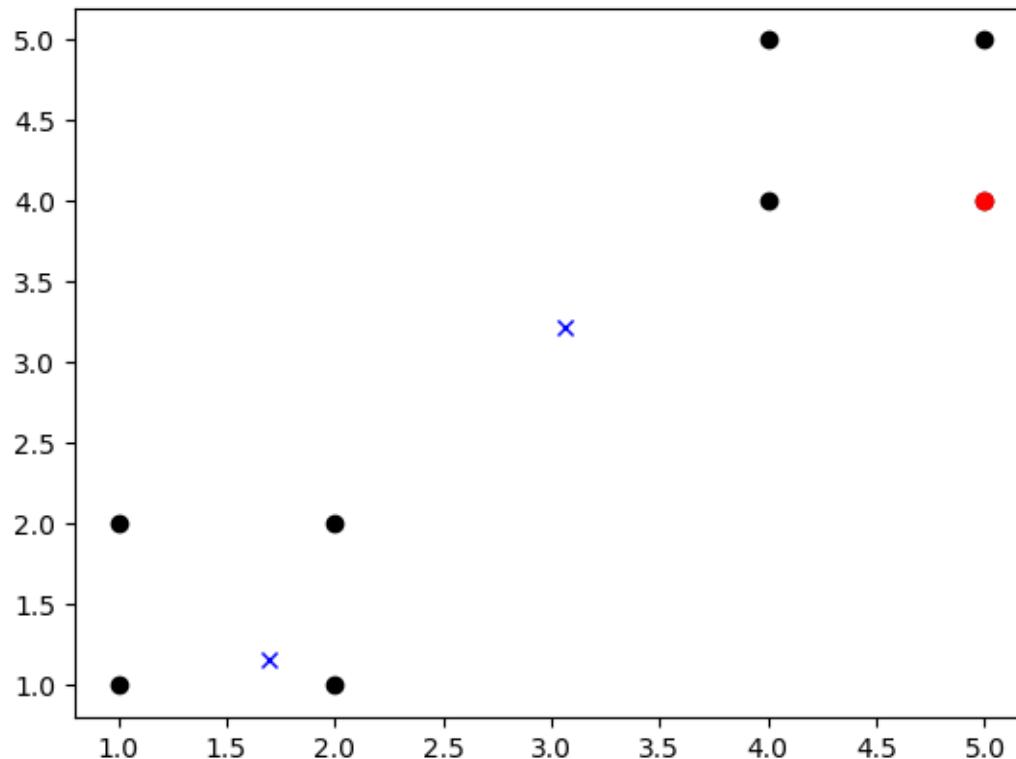
Epoch: 1, data: 5



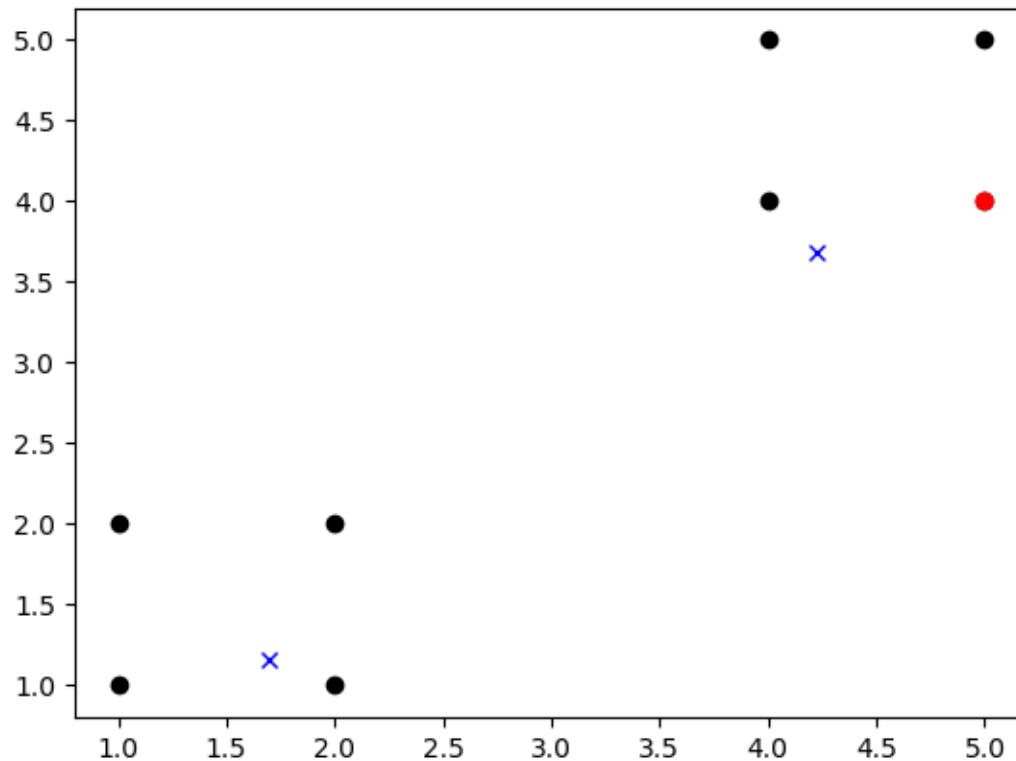
Epoch: 1, data: 5



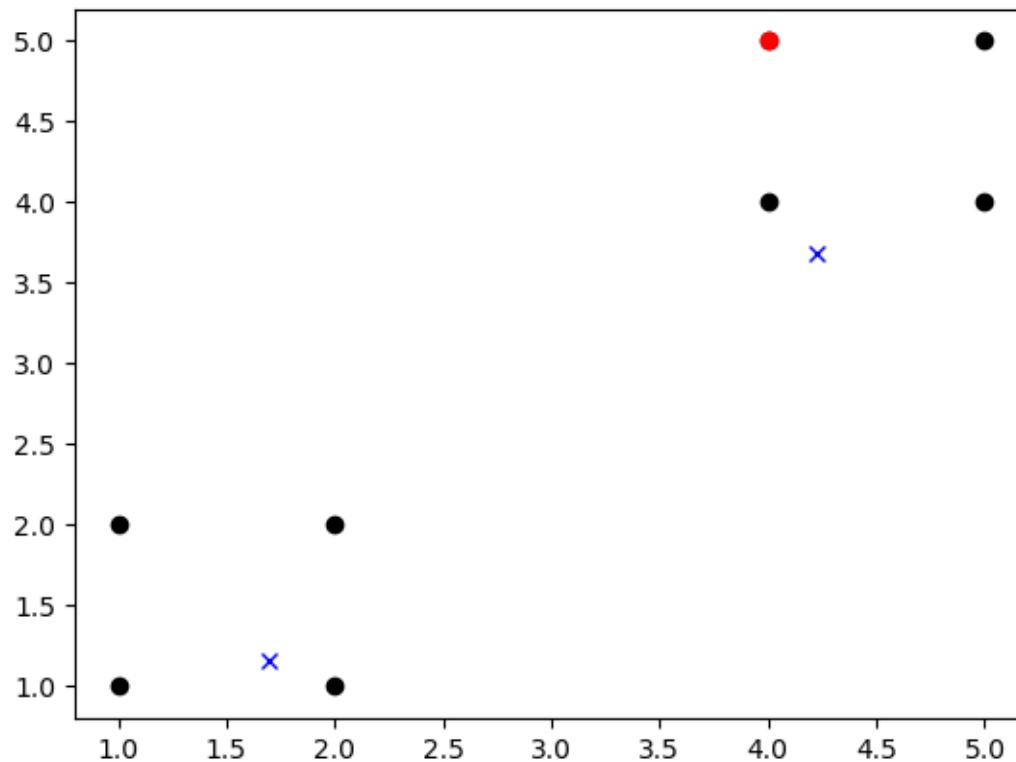
Epoch: 1, data: 6



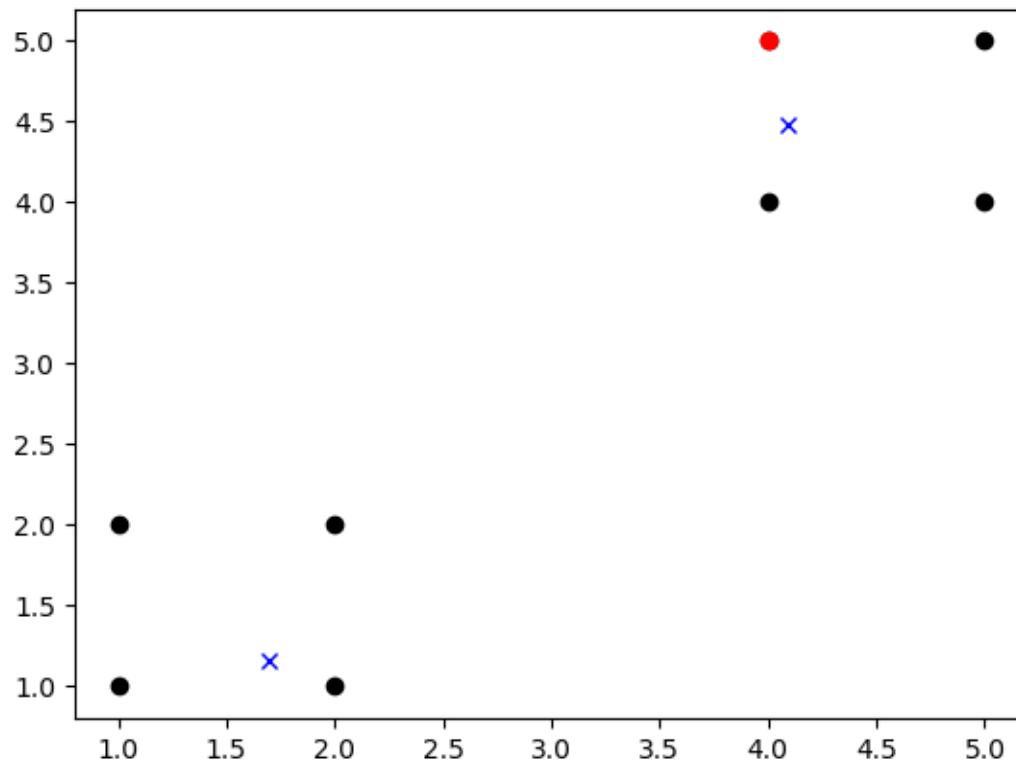
Epoch: 1, data: 6



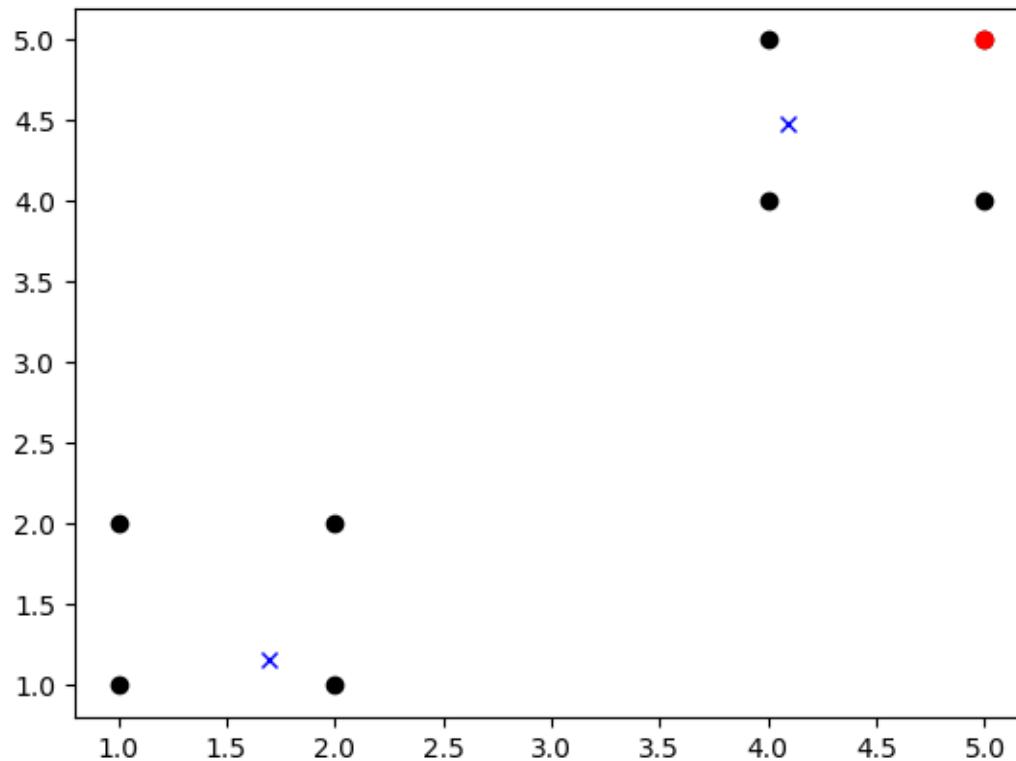
Epoch: 1, data: 7



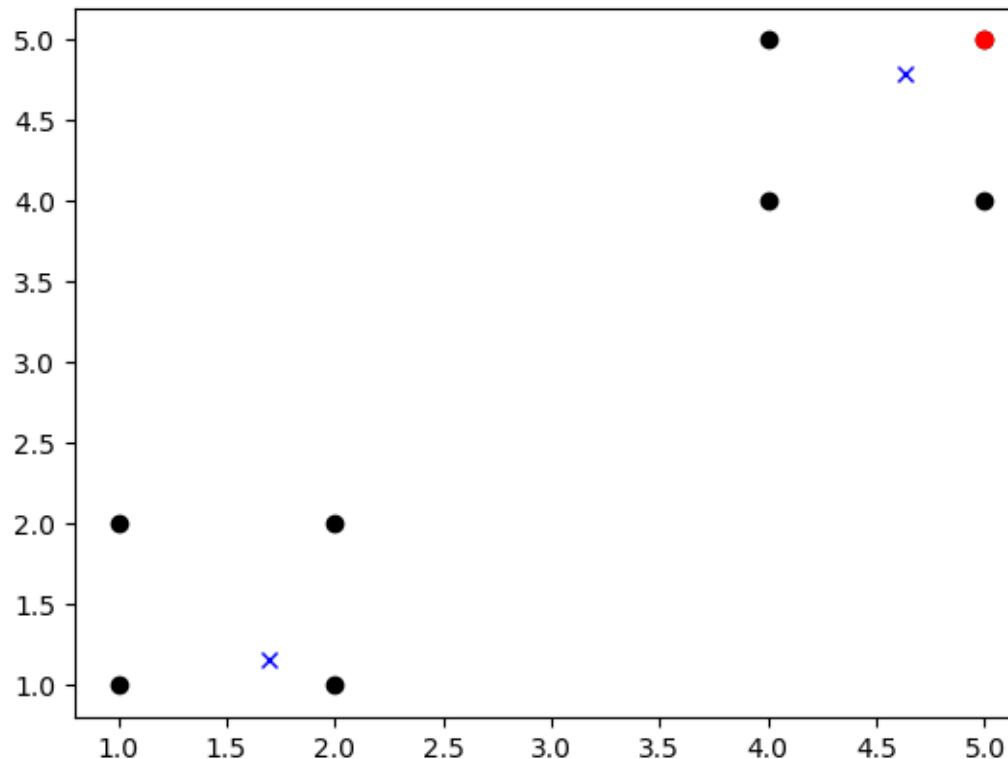
Epoch: 1, data: 7



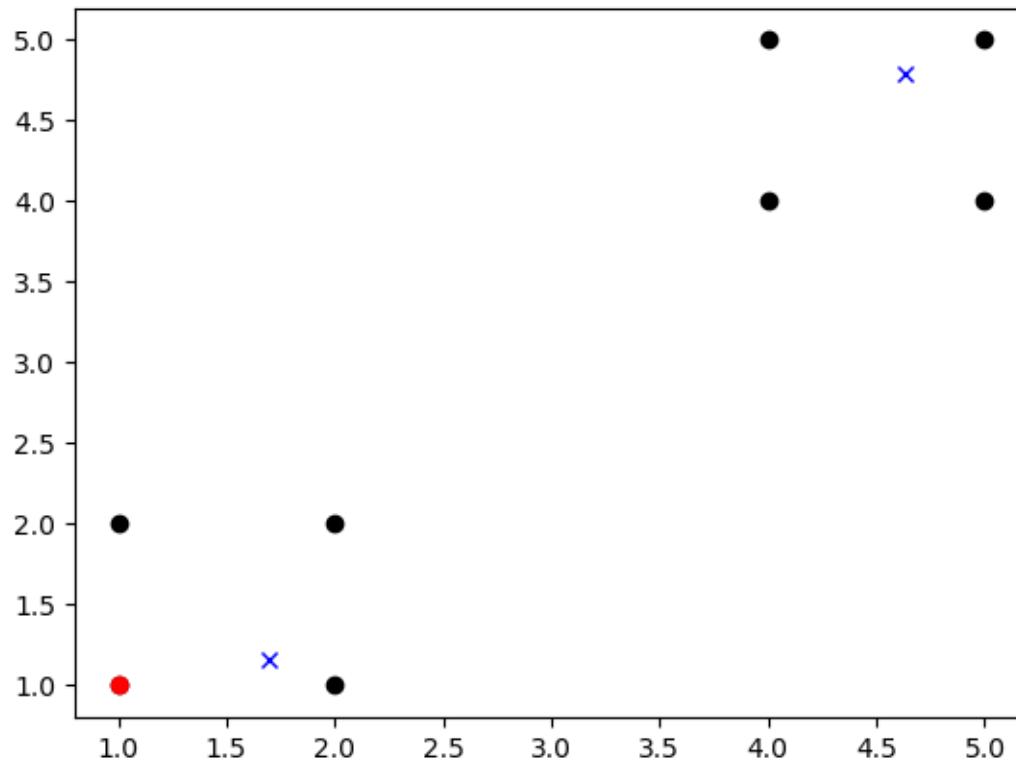
Epoch: 1, data: 8



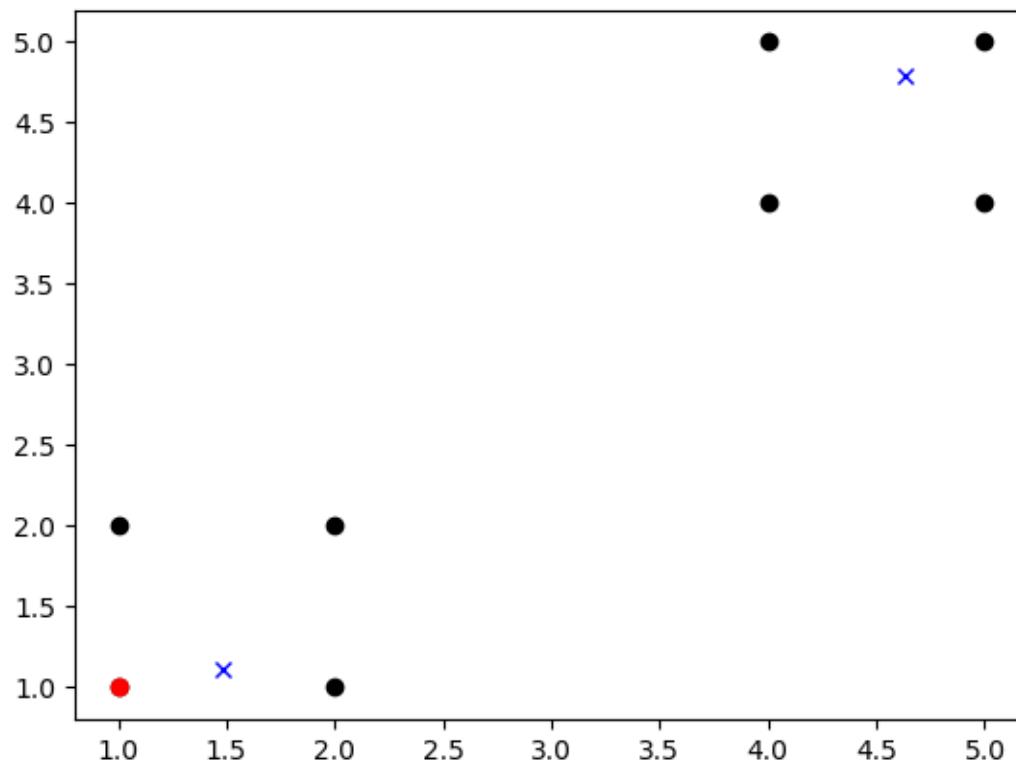
Epoch: 1, data: 8



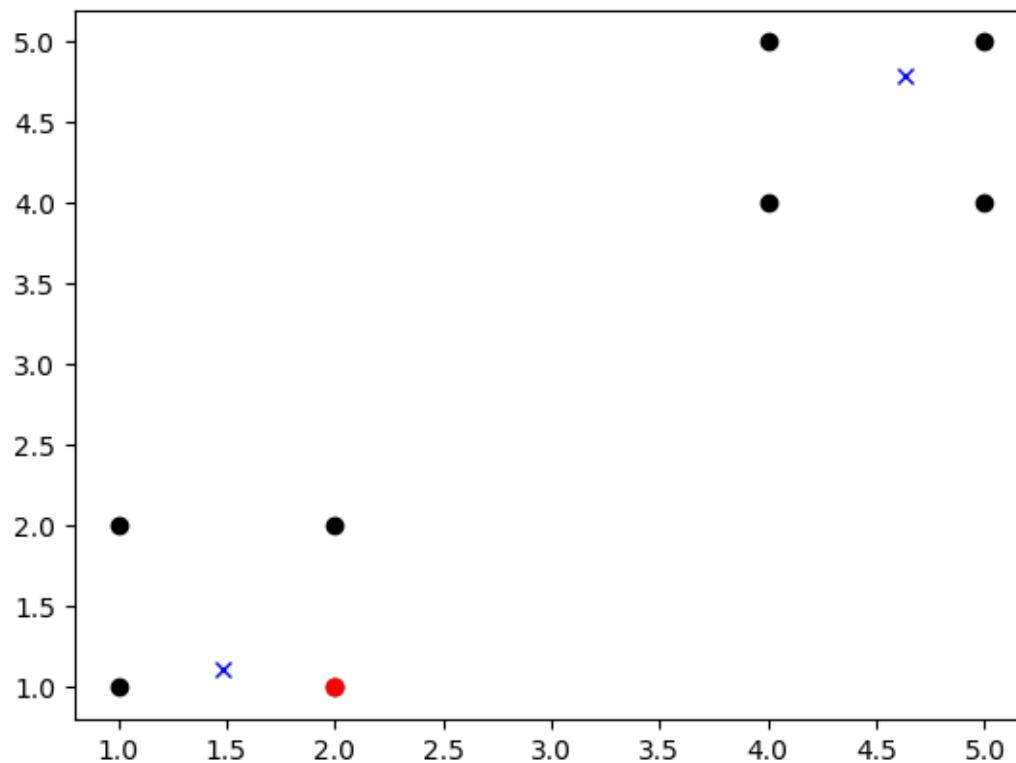
Epoch: 2, data: 1



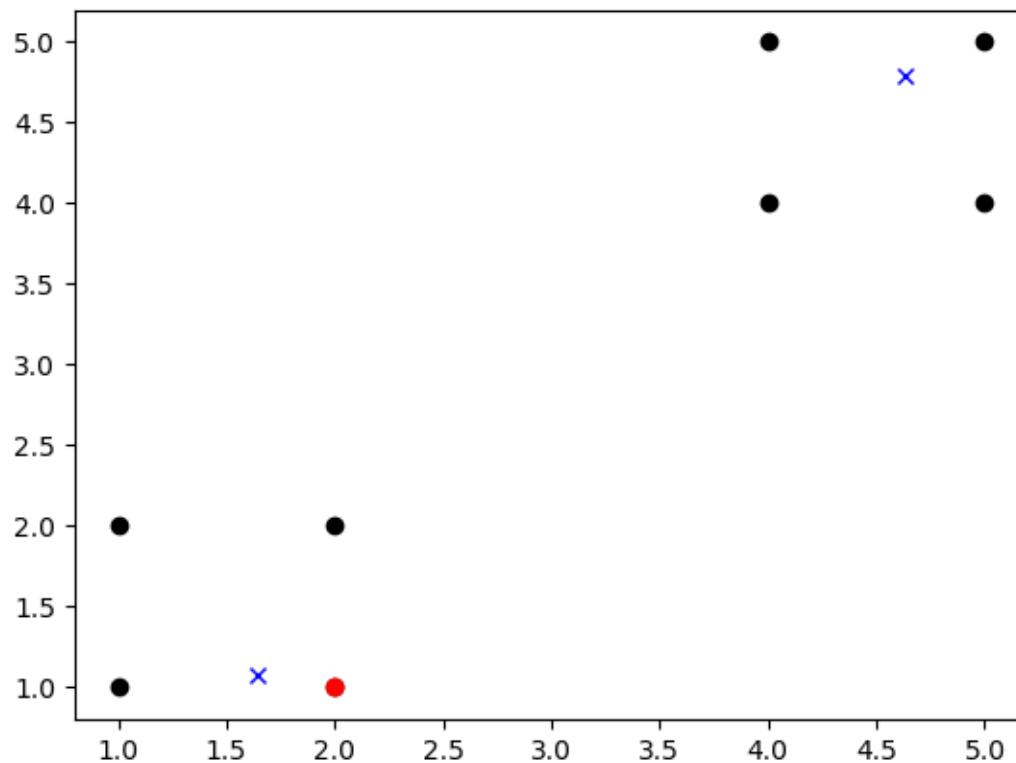
Epoch: 2, data: 1



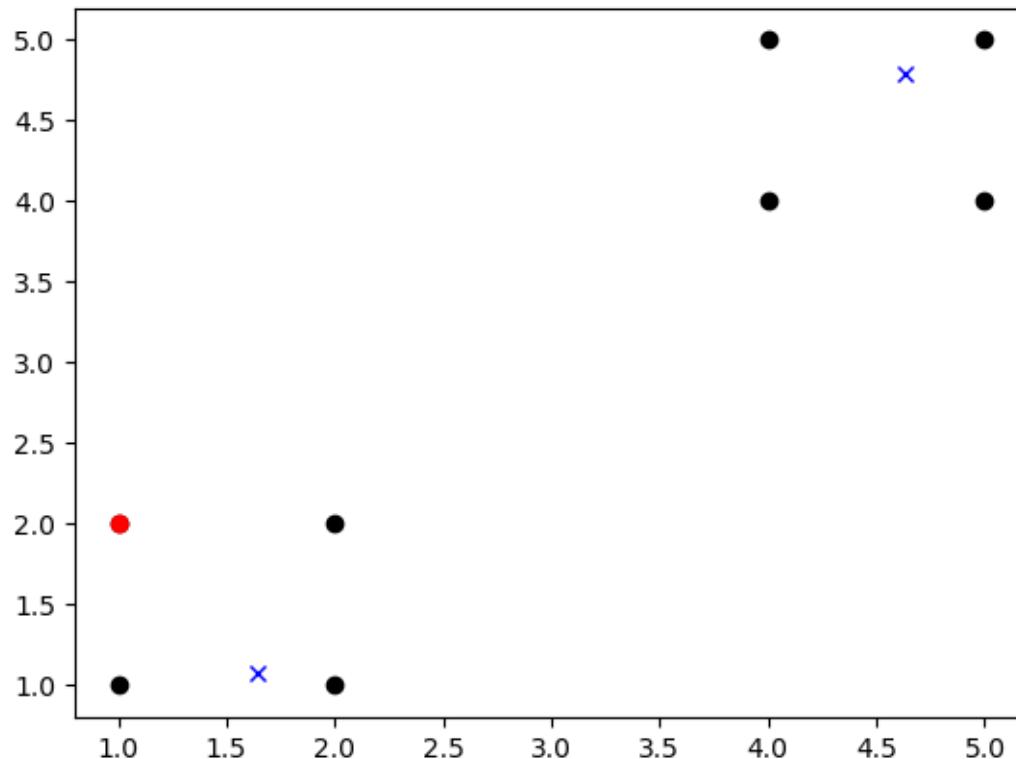
Epoch: 2, data: 2



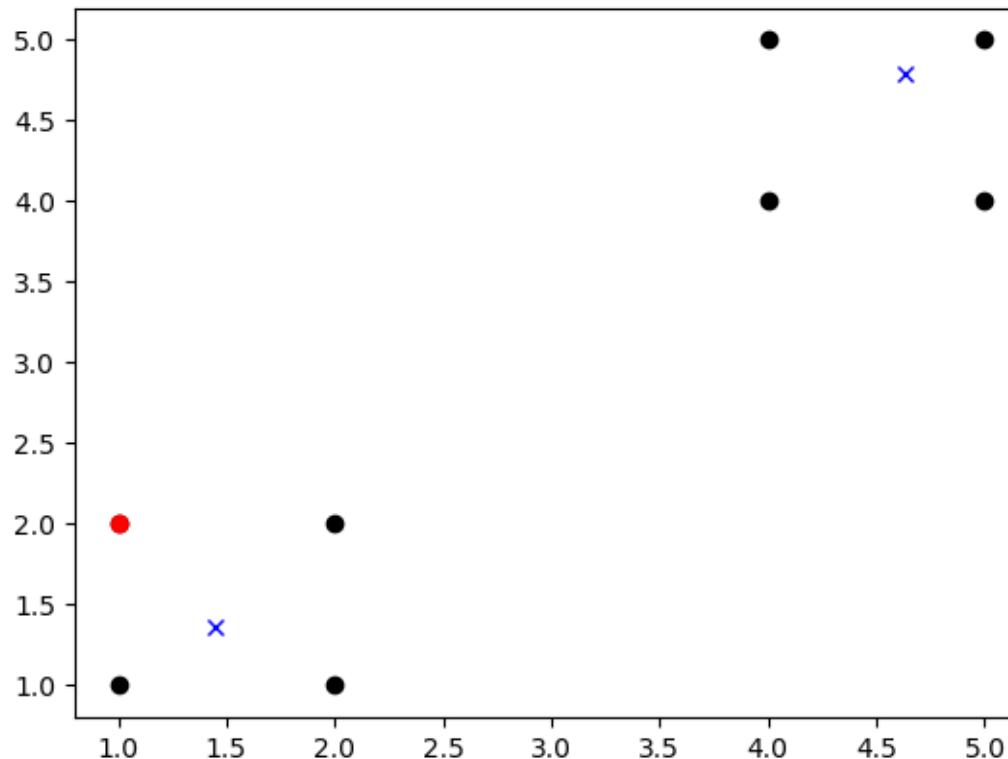
Epoch: 2, data: 2



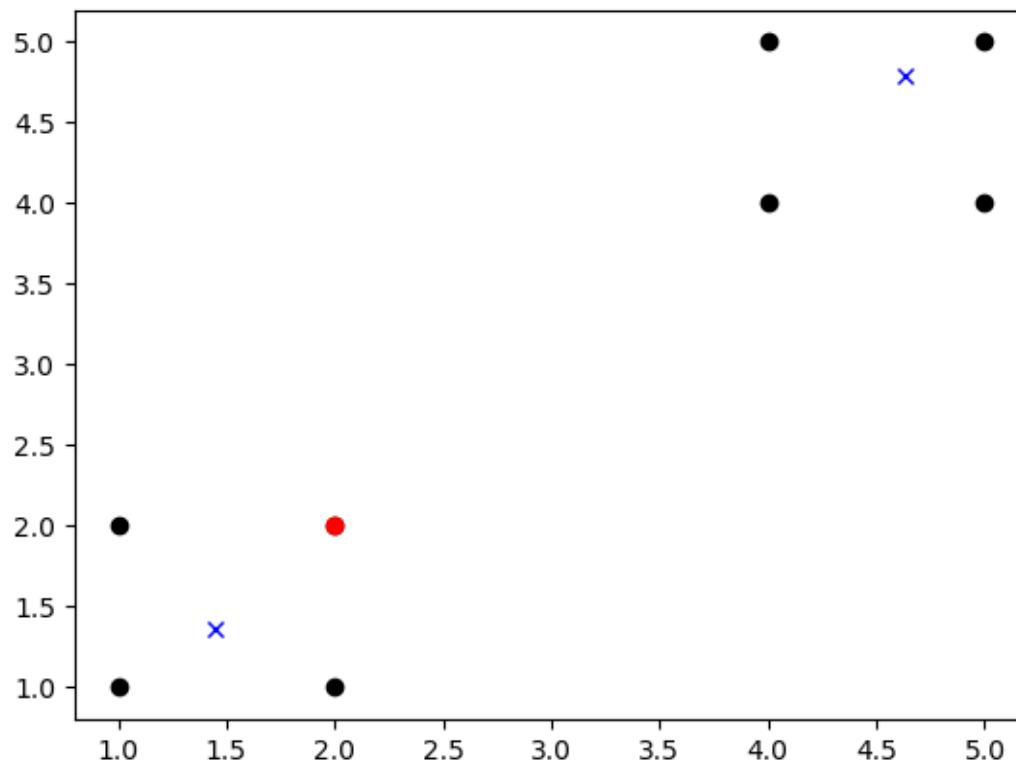
Epoch: 2, data: 3



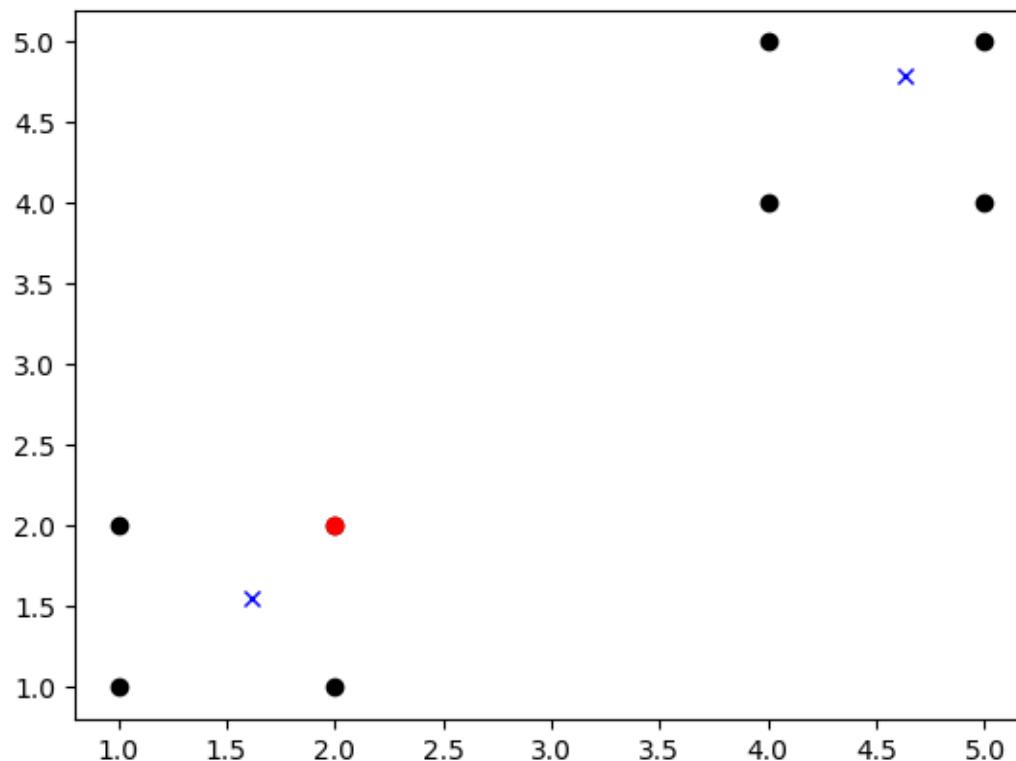
Epoch: 2, data: 3



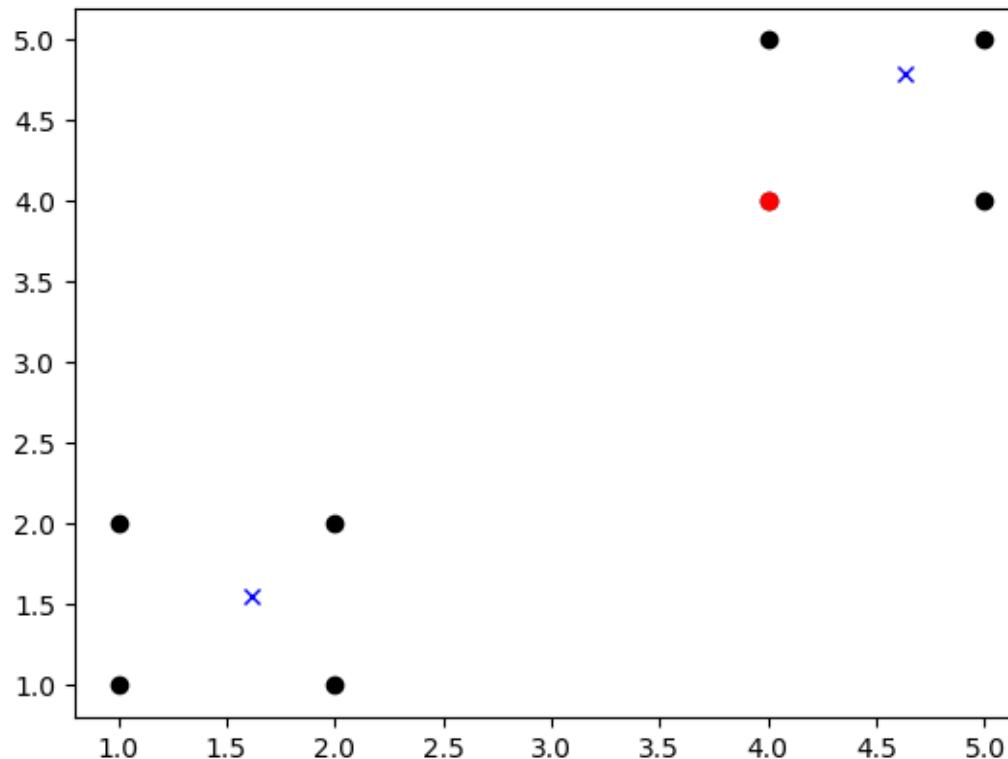
Epoch: 2, data: 4



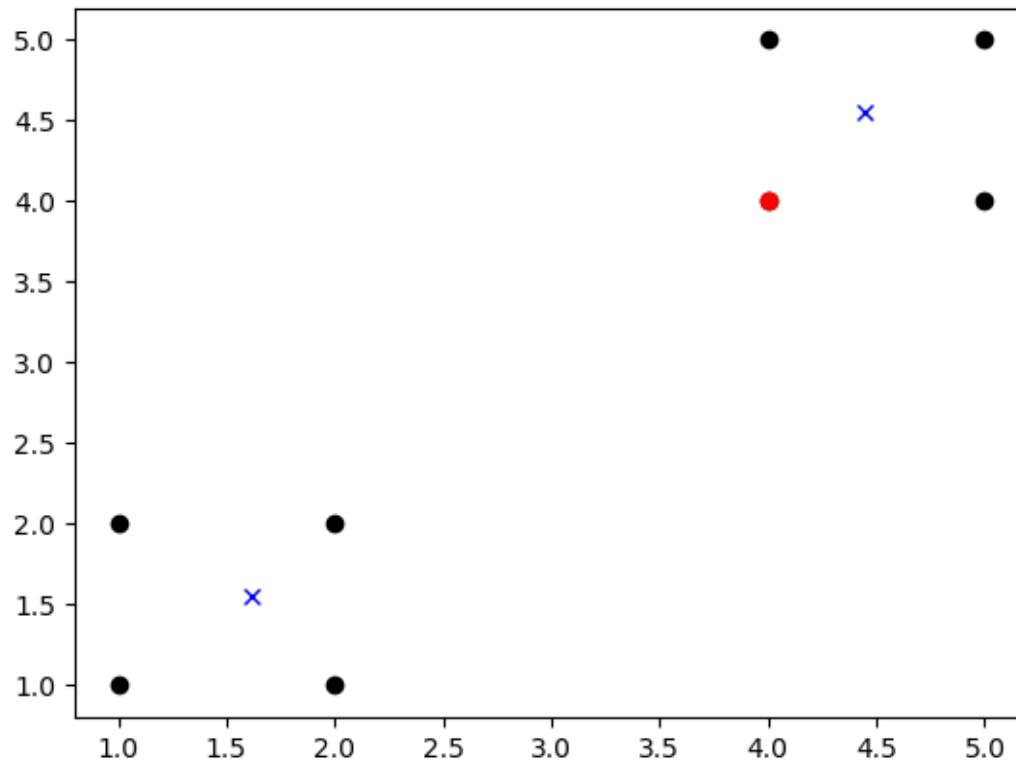
Epoch: 2, data: 4



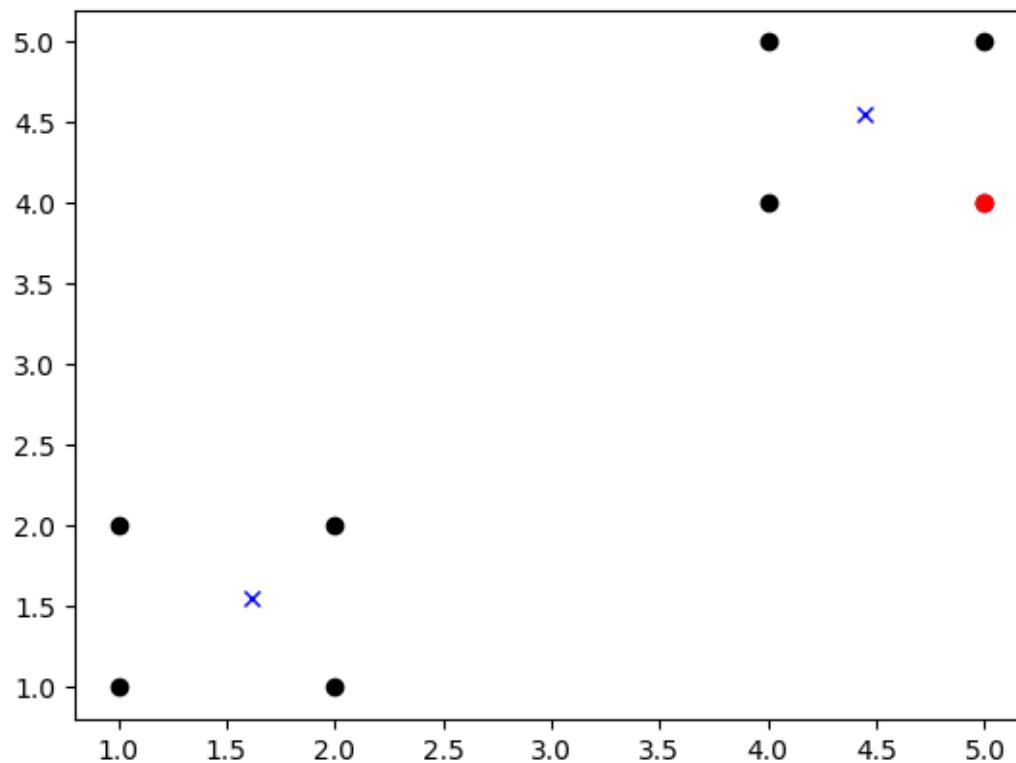
Epoch: 2, data: 5



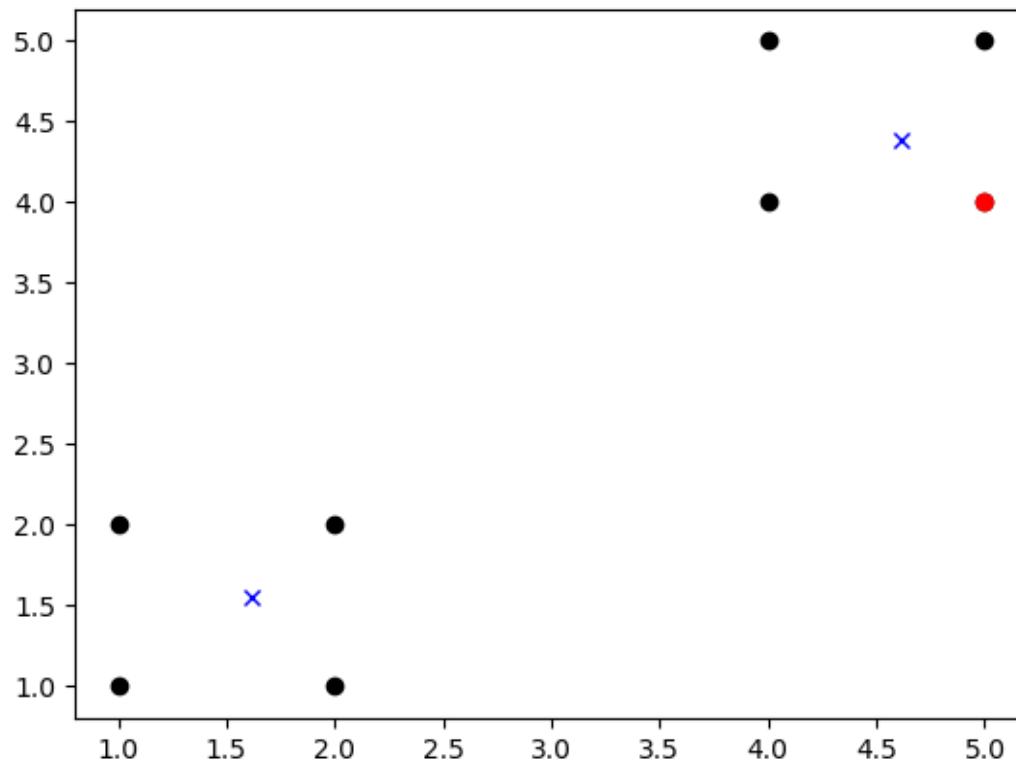
Epoch: 2, data: 5



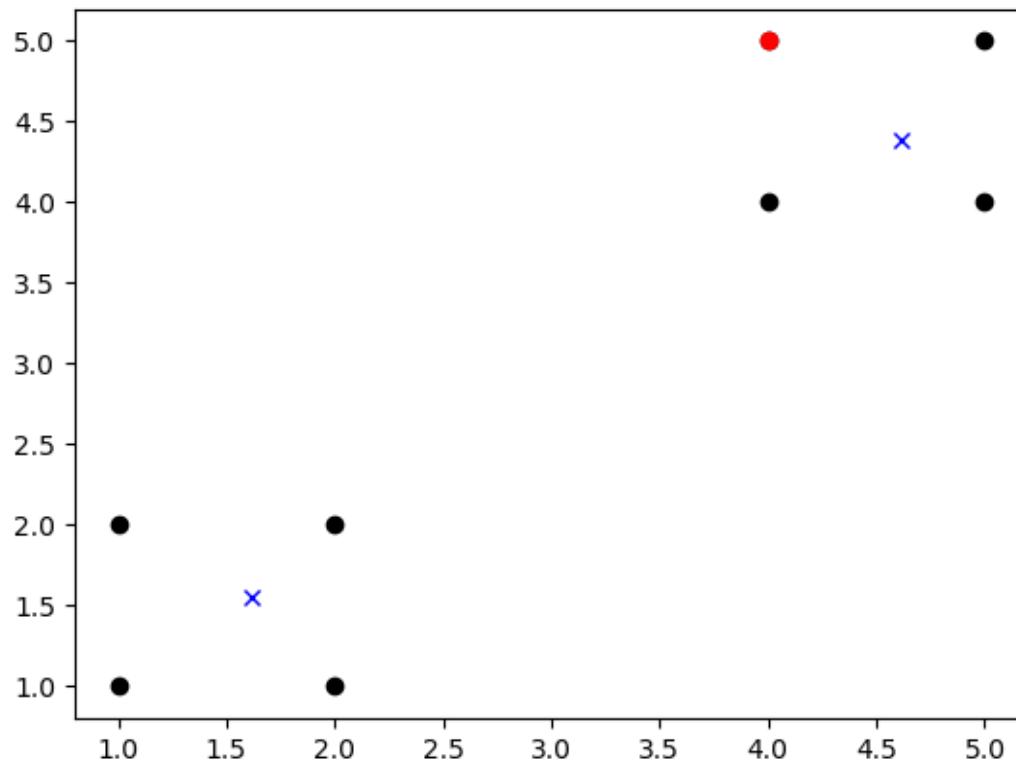
Epoch: 2, data: 6



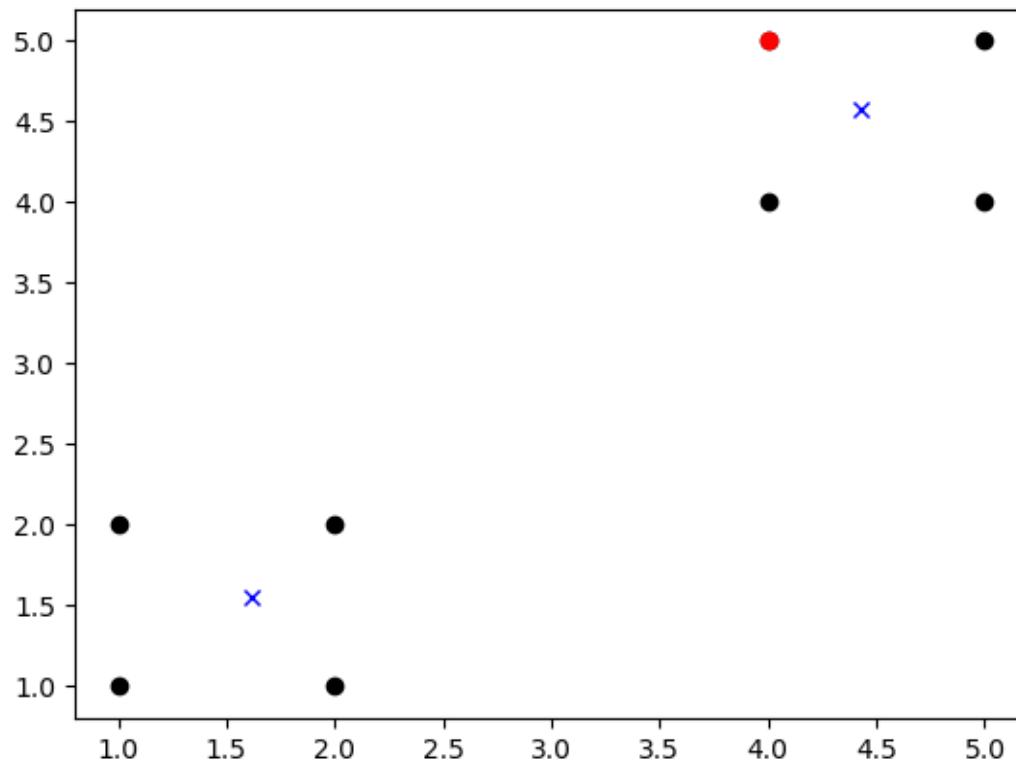
Epoch: 2, data: 6



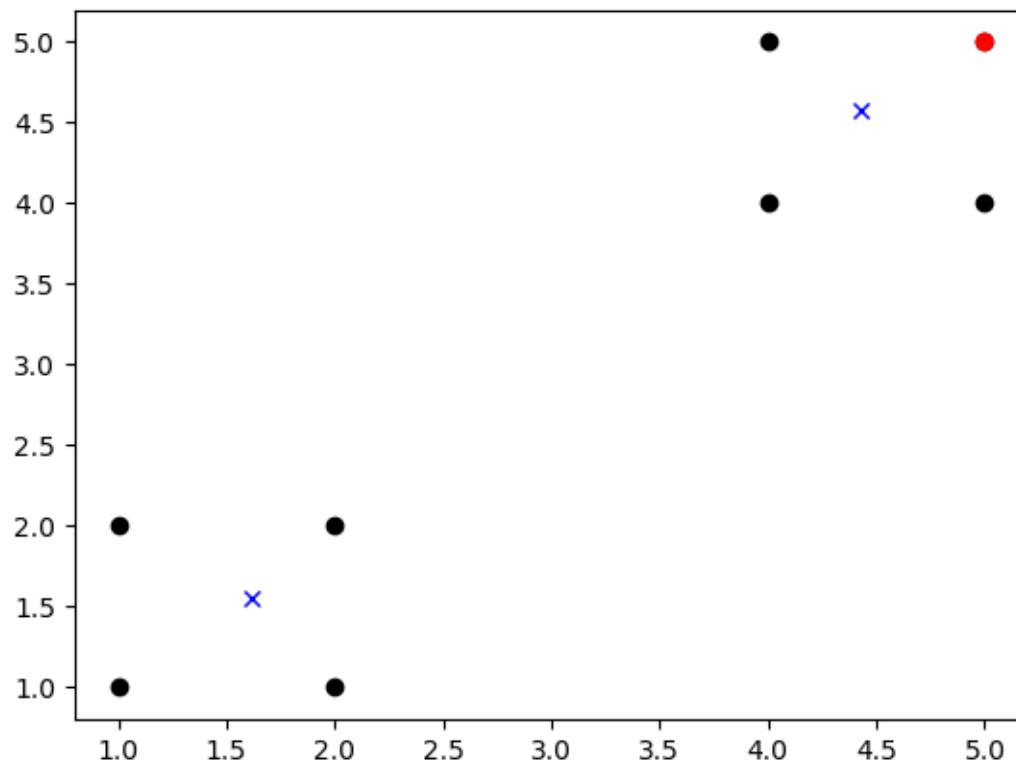
Epoch: 2, data: 7



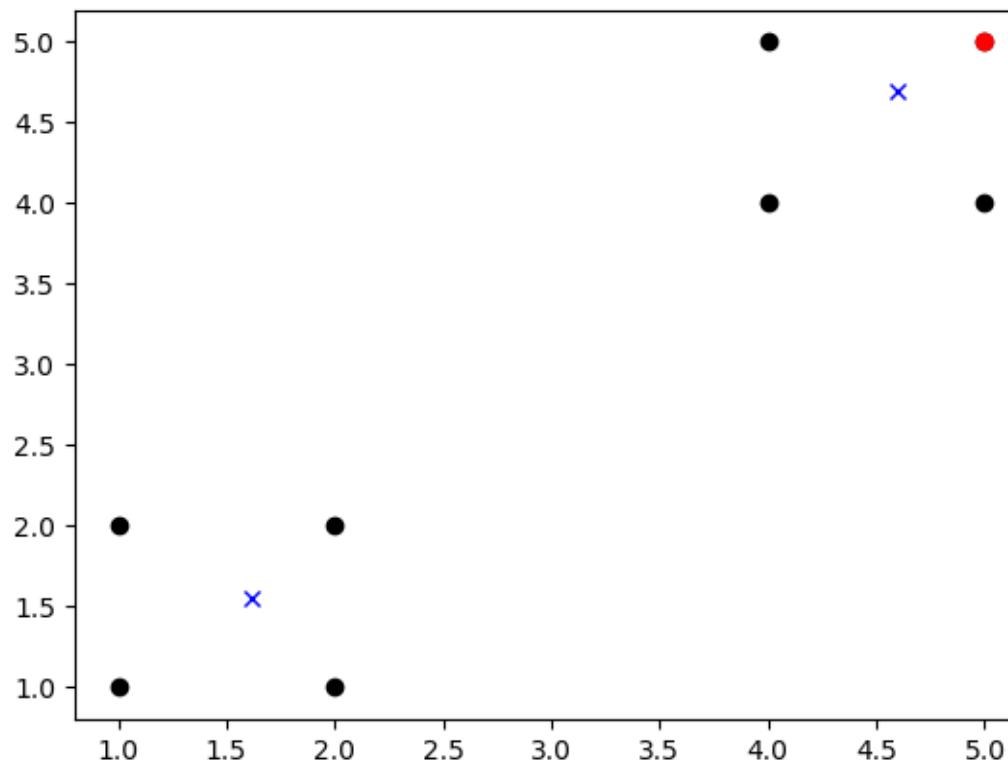
Epoch: 2, data: 7



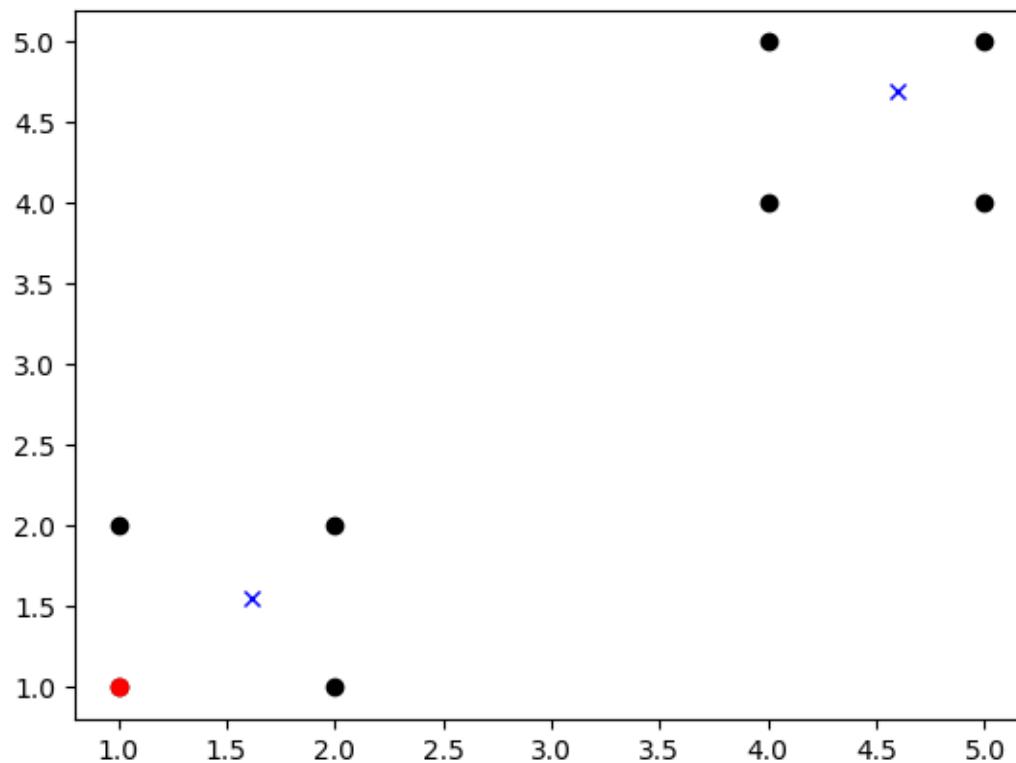
Epoch: 2, data: 8



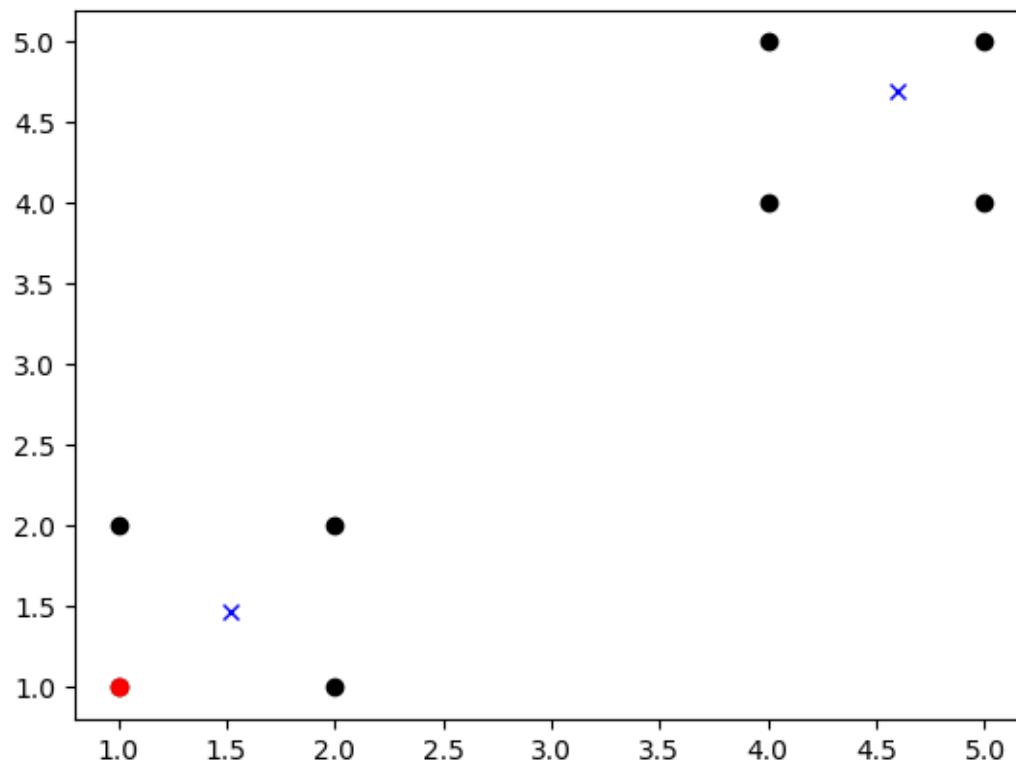
Epoch: 2, data: 8



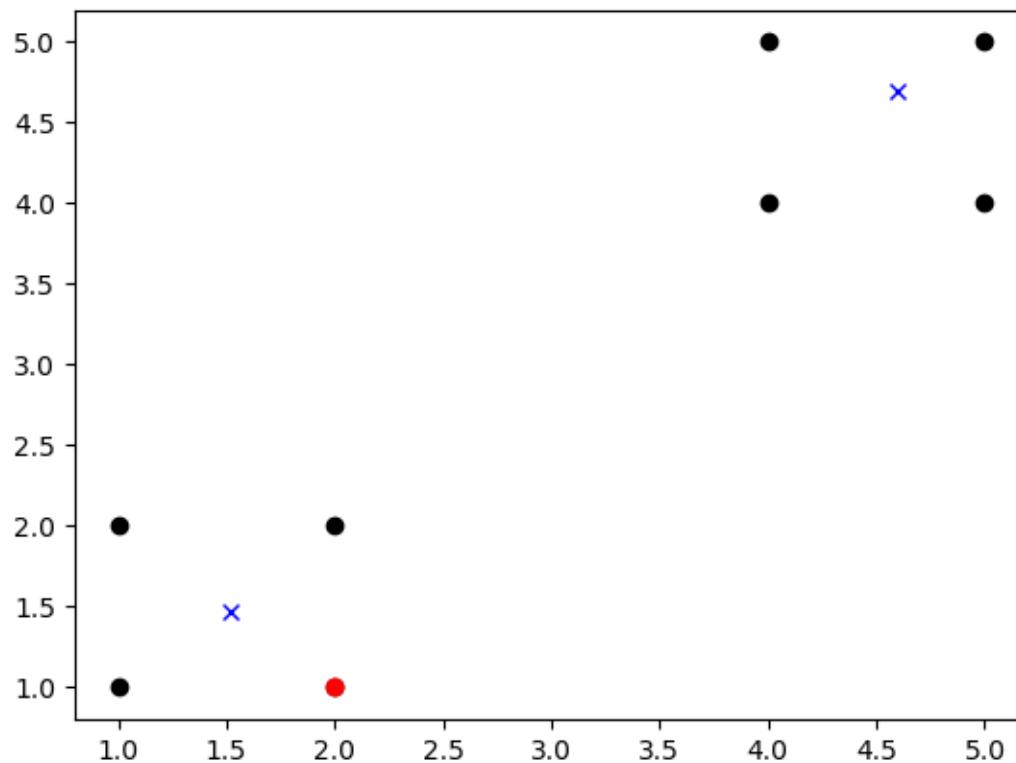
Epoch: 3, data: 1



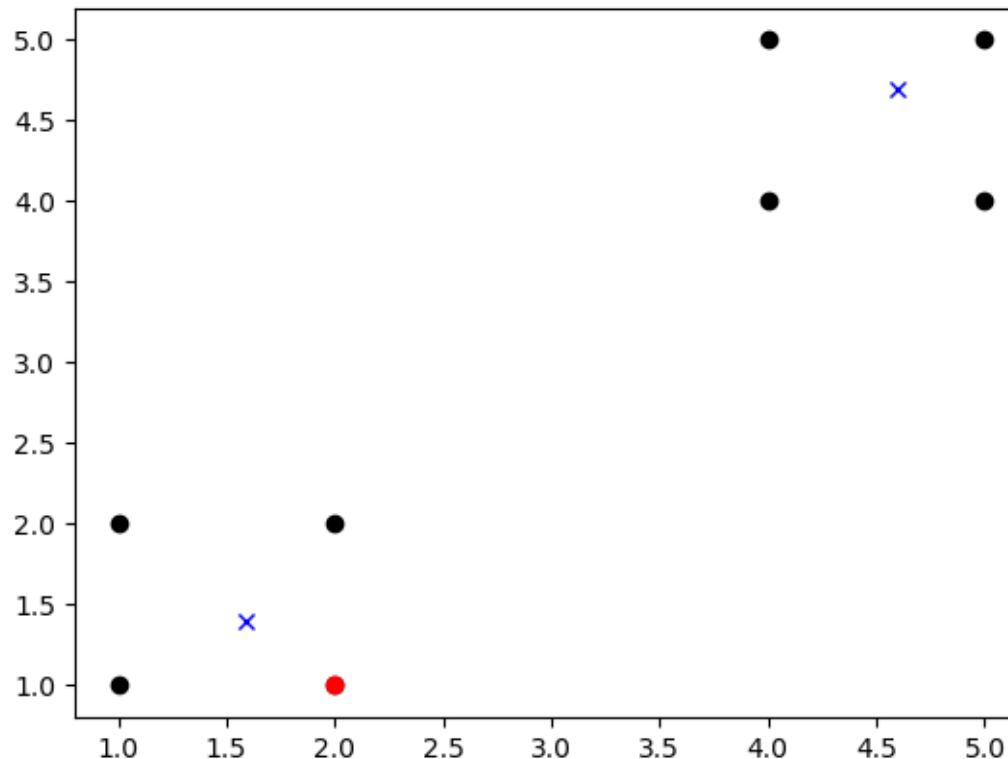
Epoch: 3, data: 1



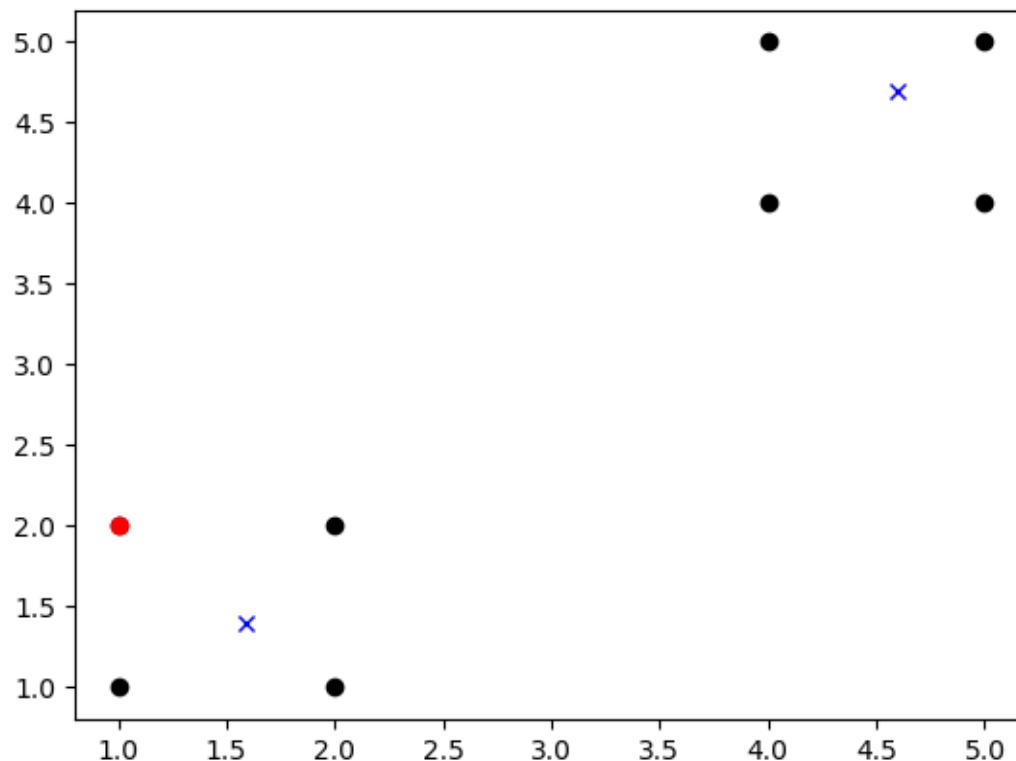
Epoch: 3, data: 2



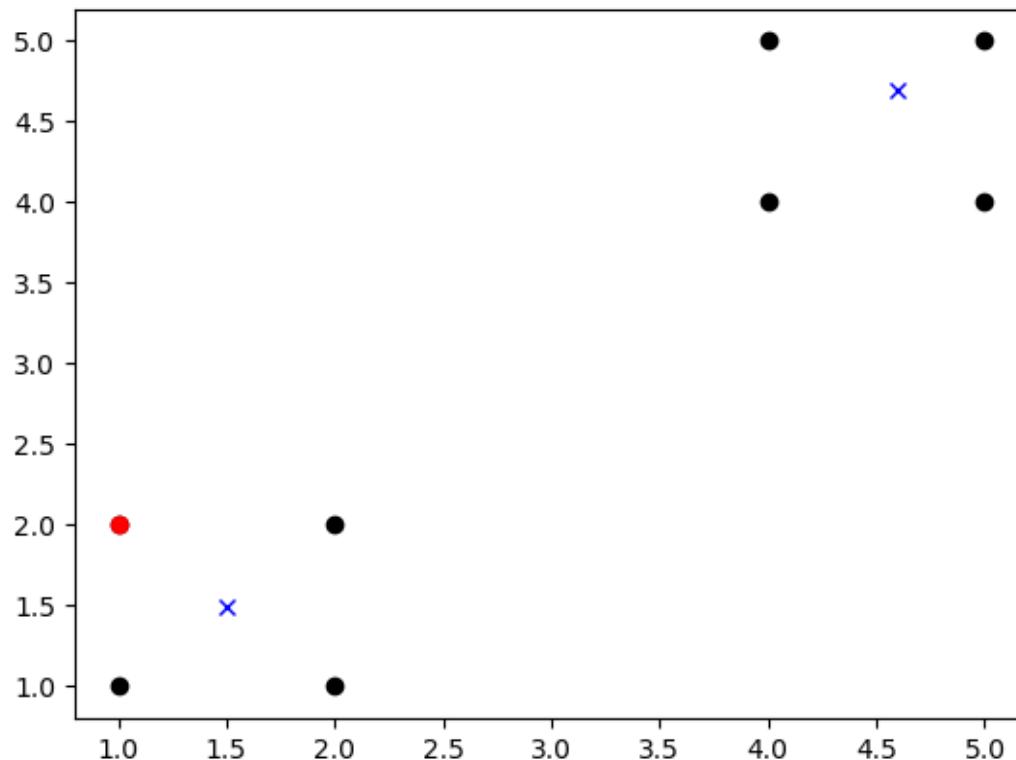
Epoch: 3, data: 2



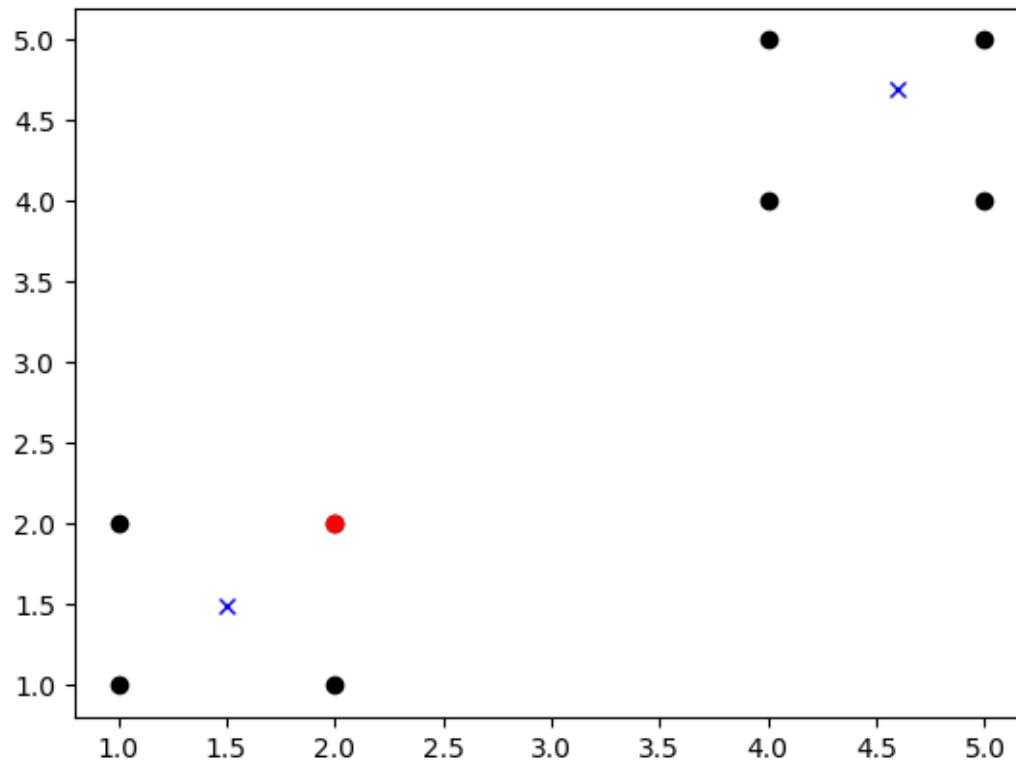
Epoch: 3, data: 3



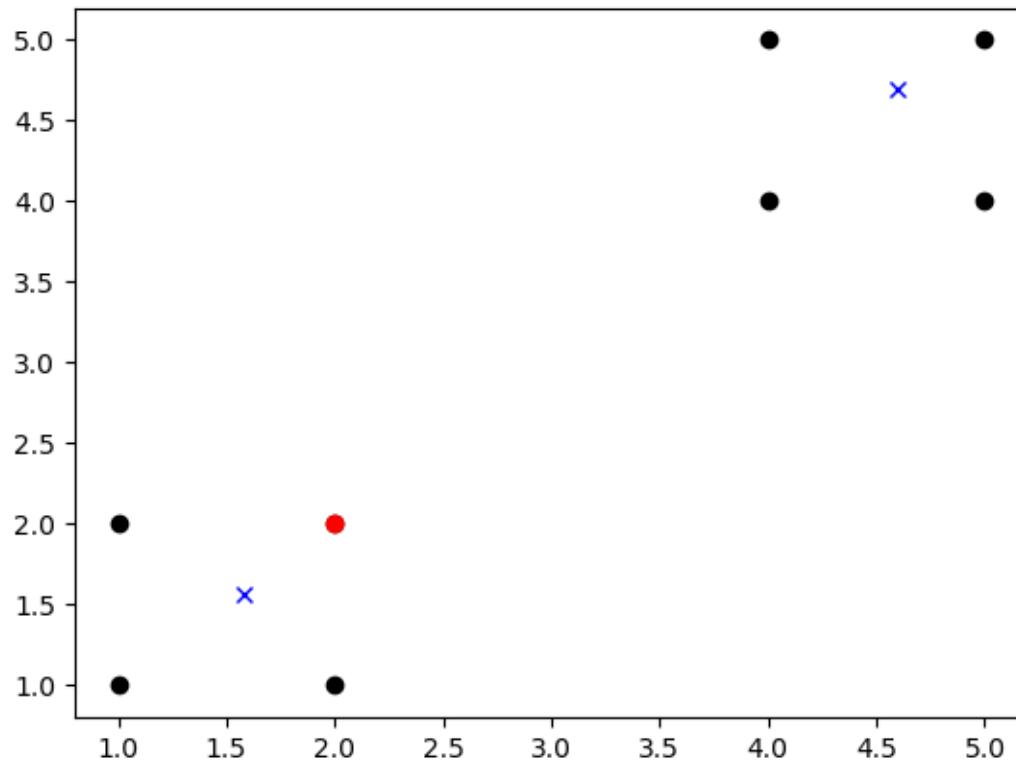
Epoch: 3, data: 3



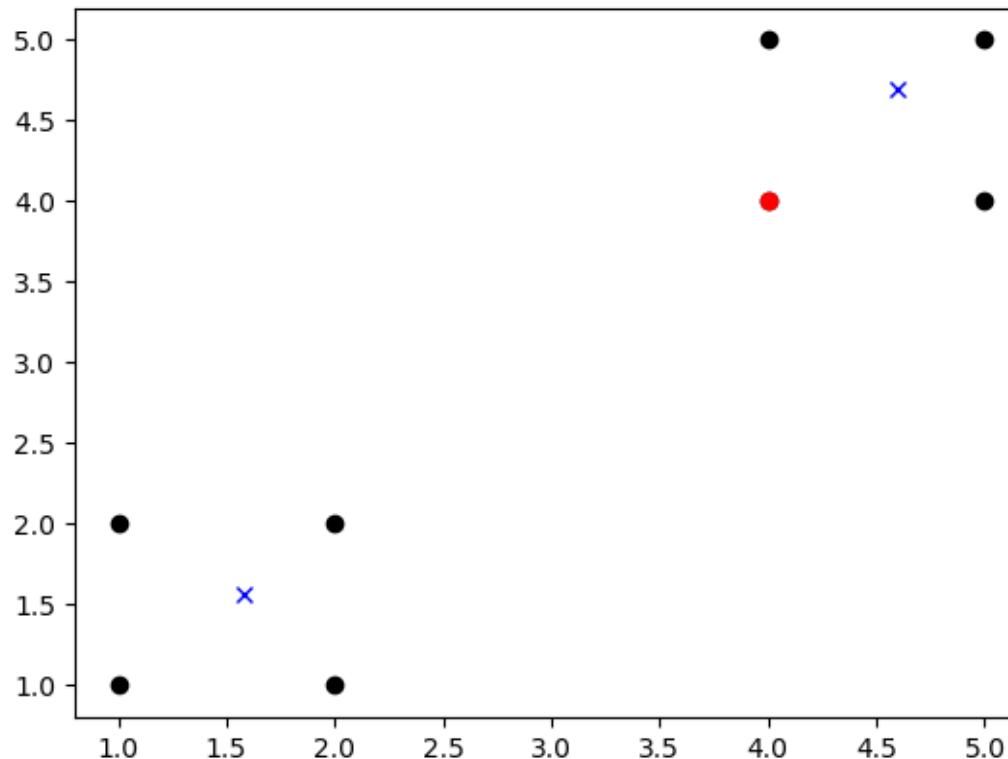
Epoch: 3, data: 4



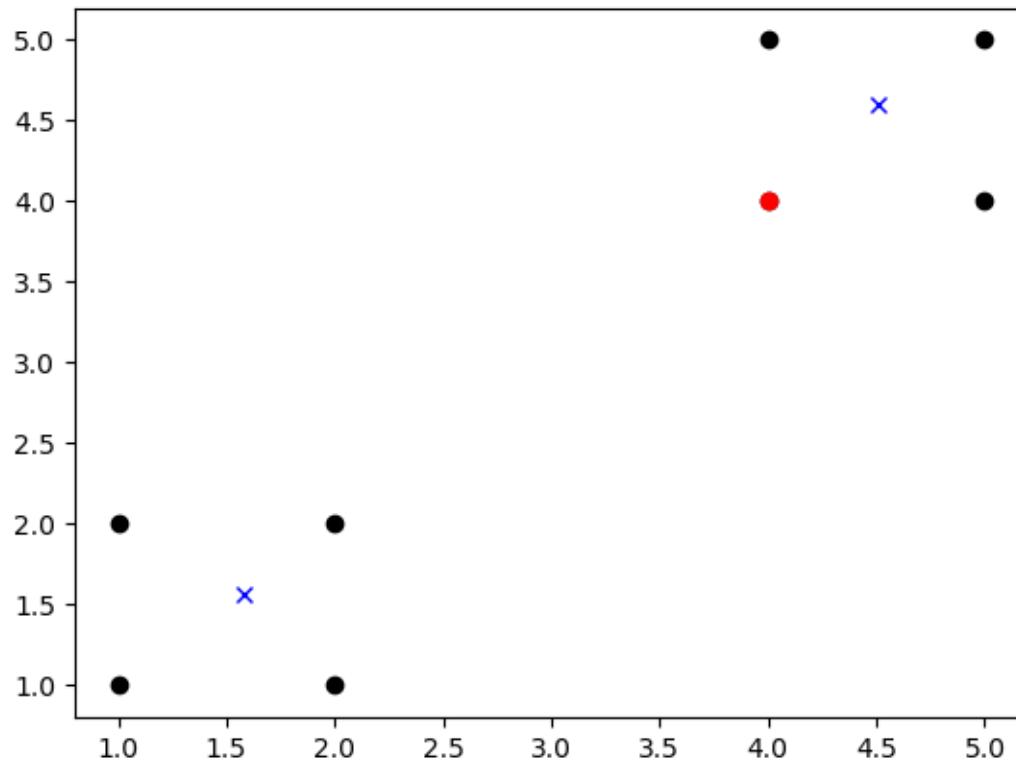
Epoch: 3, data: 4



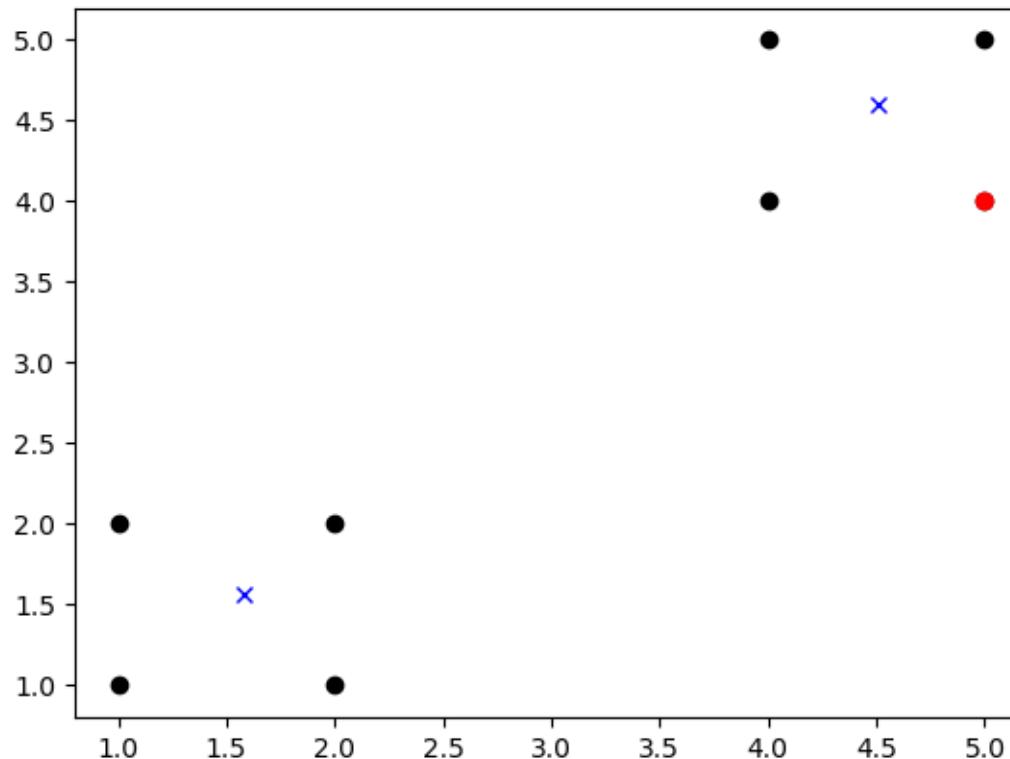
Epoch: 3, data: 5



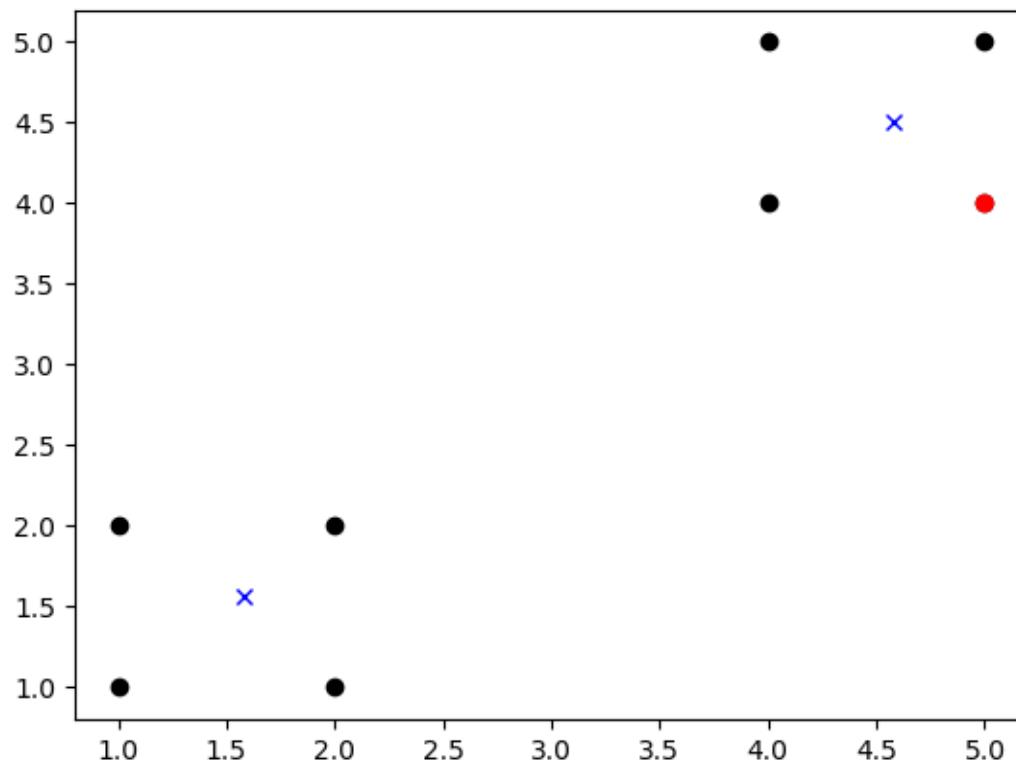
Epoch: 3, data: 5



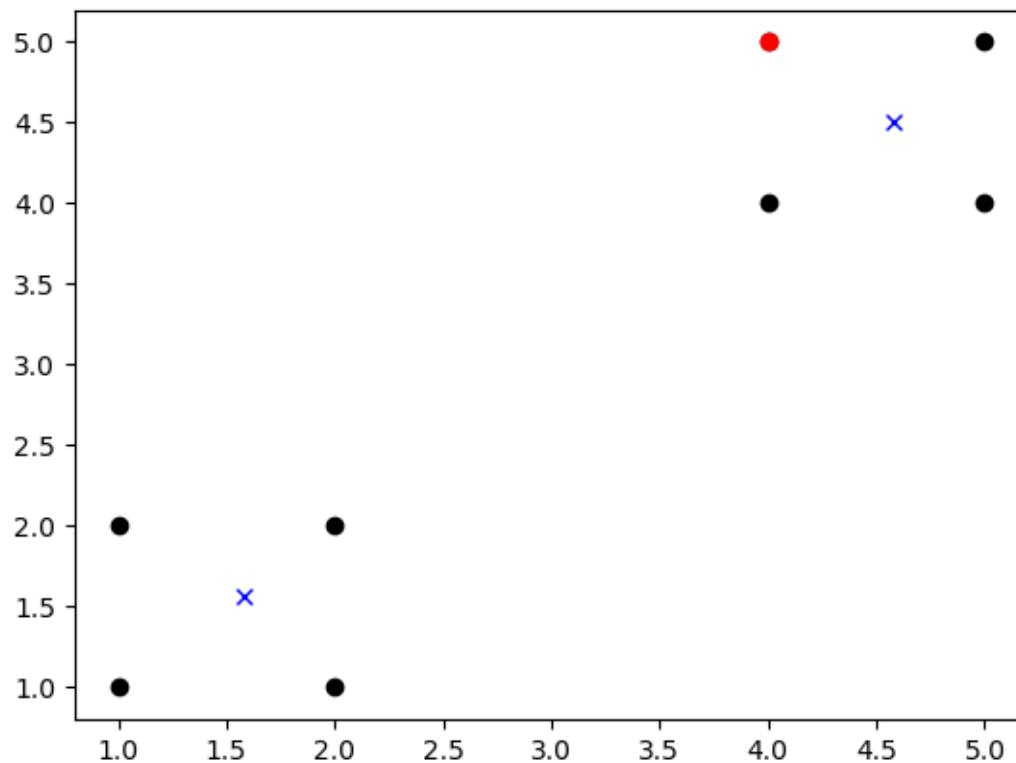
Epoch: 3, data: 6



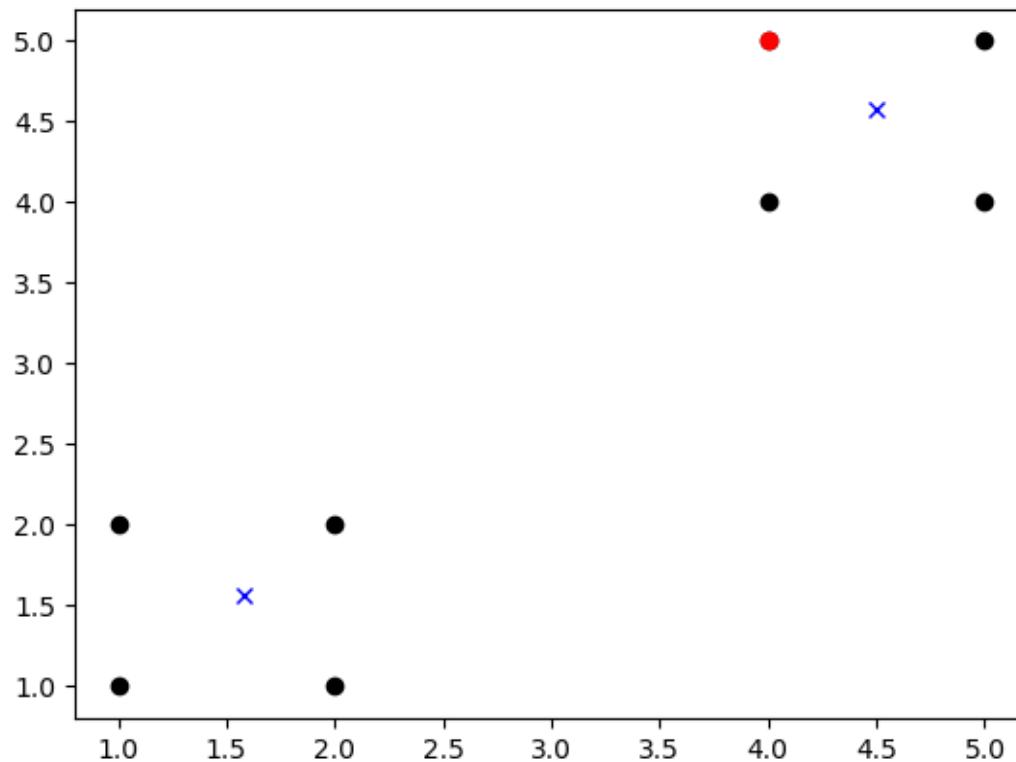
Epoch: 3, data: 6



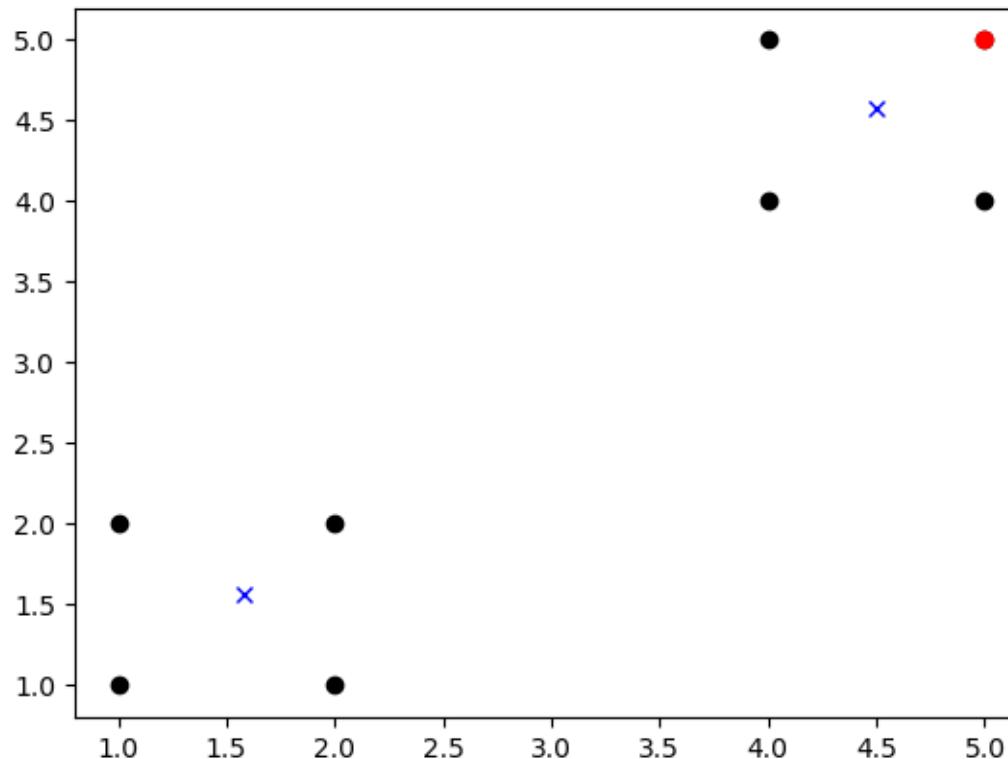
Epoch: 3, data: 7



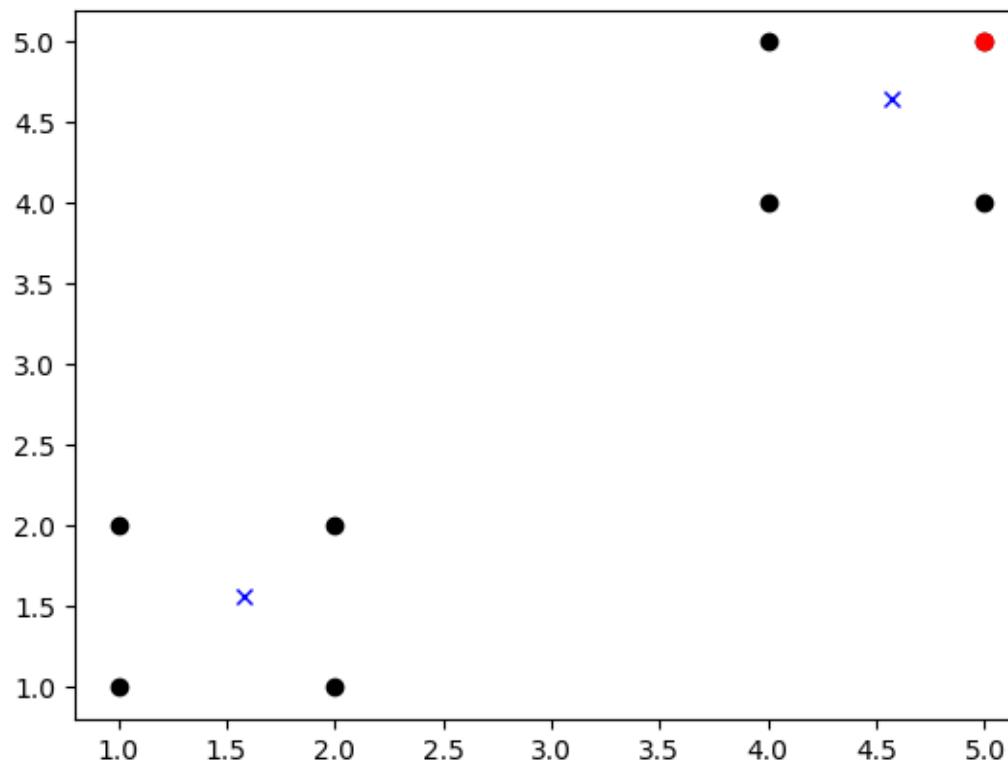
Epoch: 3, data: 7



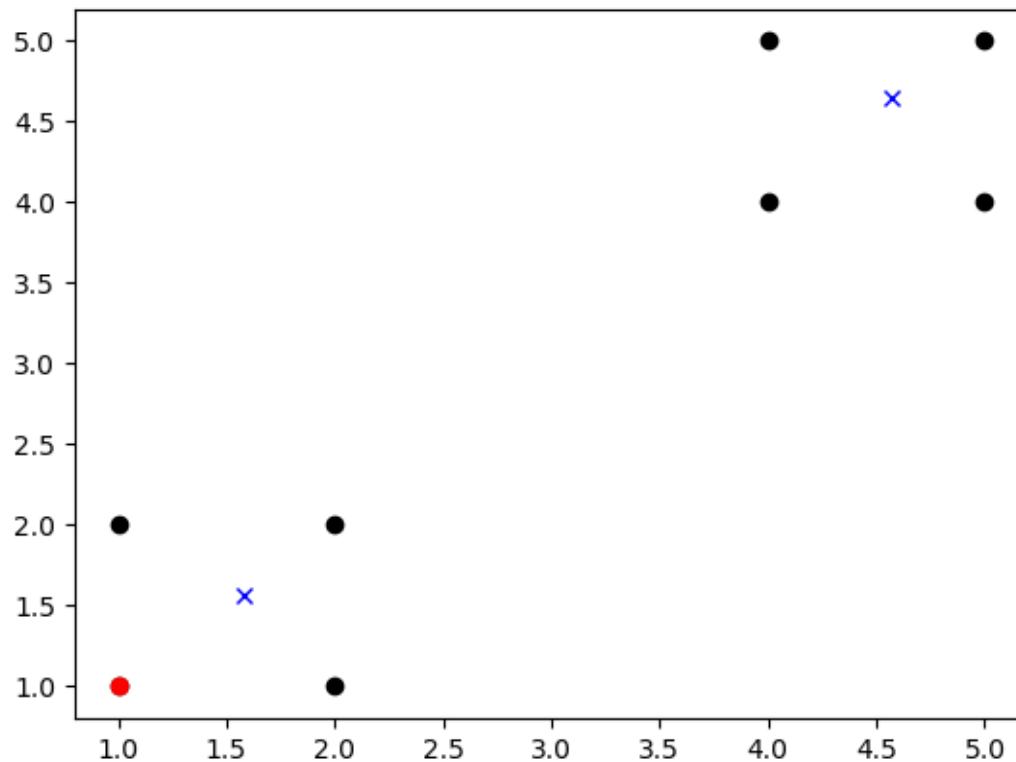
Epoch: 3, data: 8



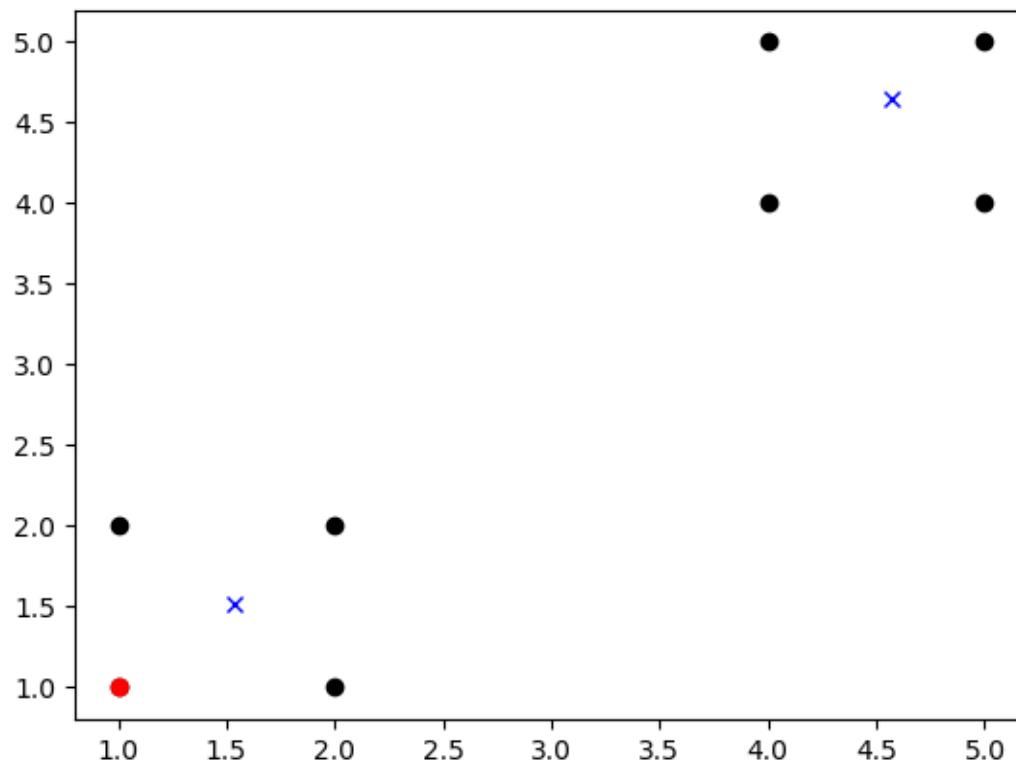
Epoch: 3, data: 8



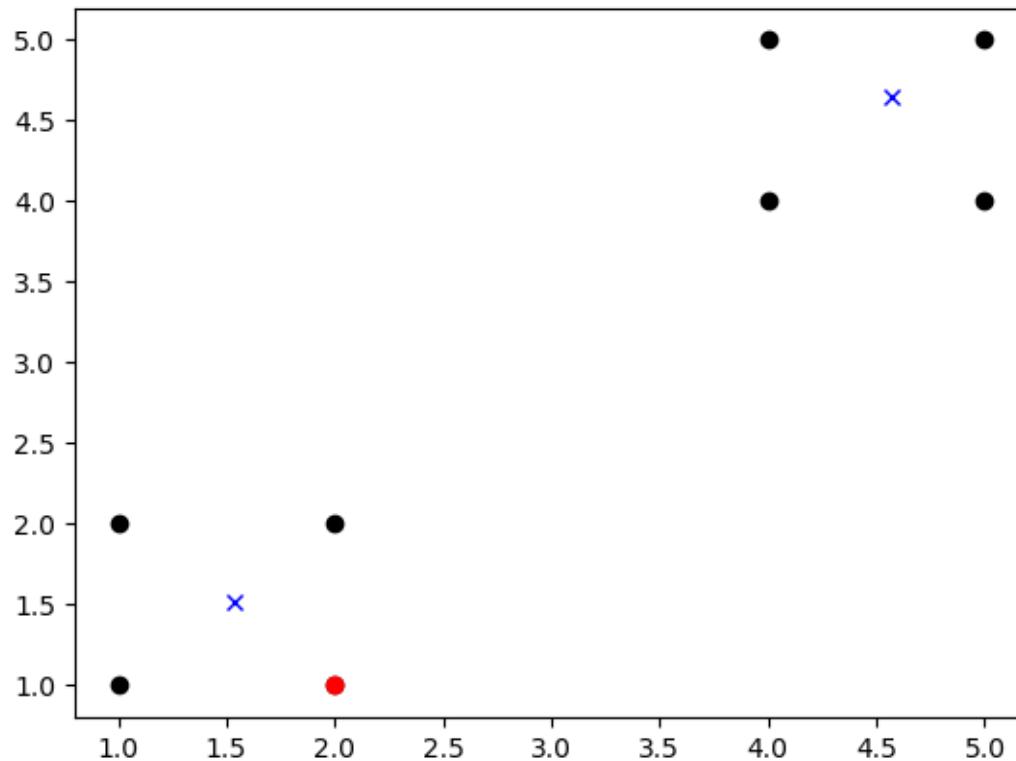
Epoch: 4, data: 1



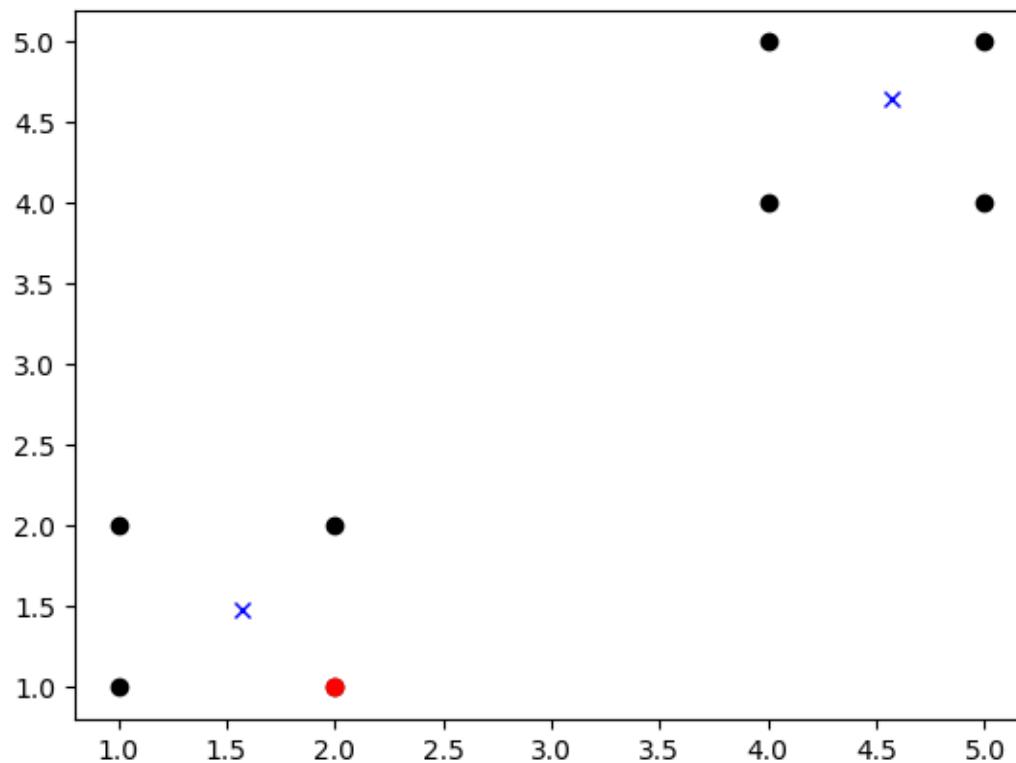
Epoch: 4, data: 1



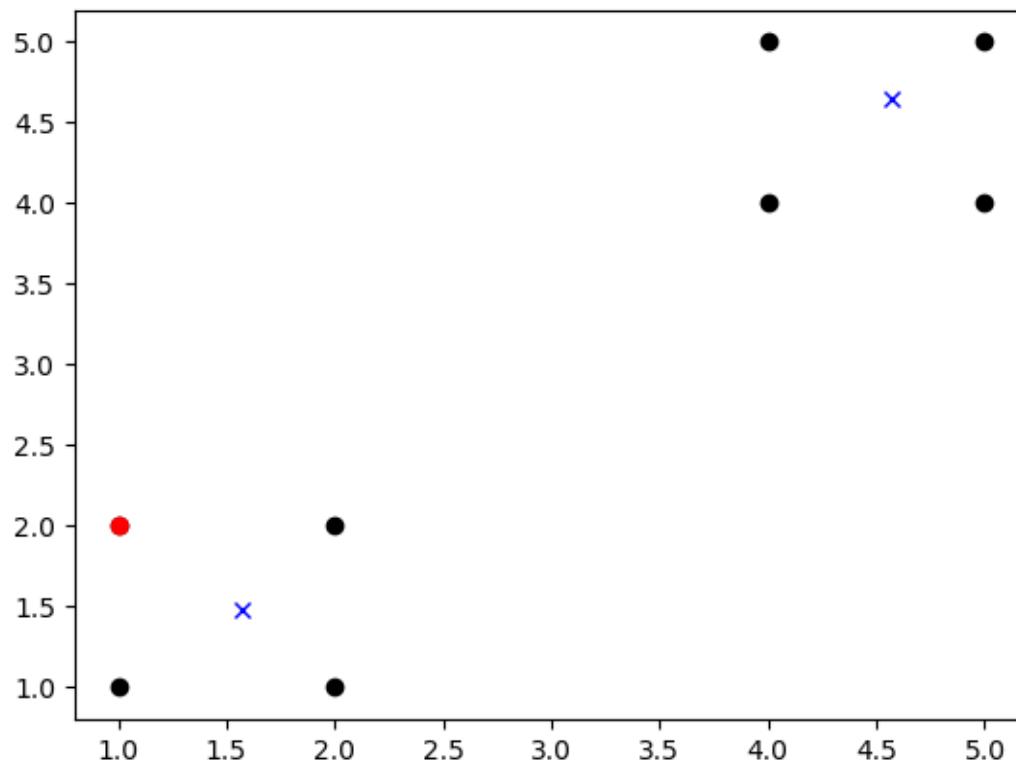
Epoch: 4, data: 2



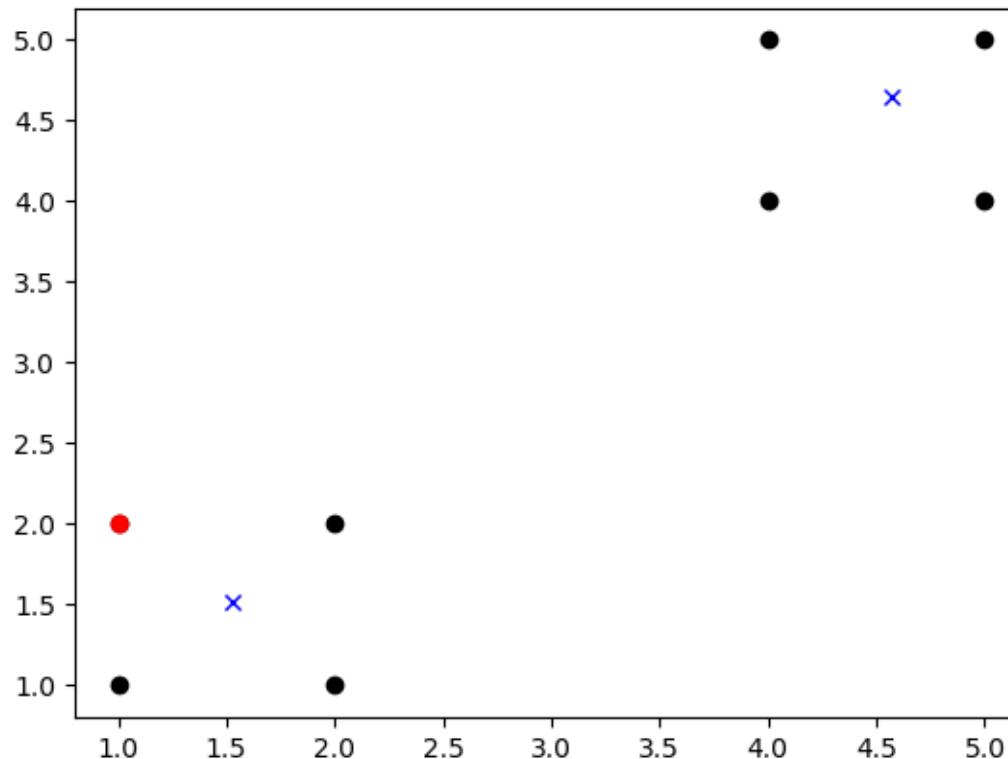
Epoch: 4, data: 2



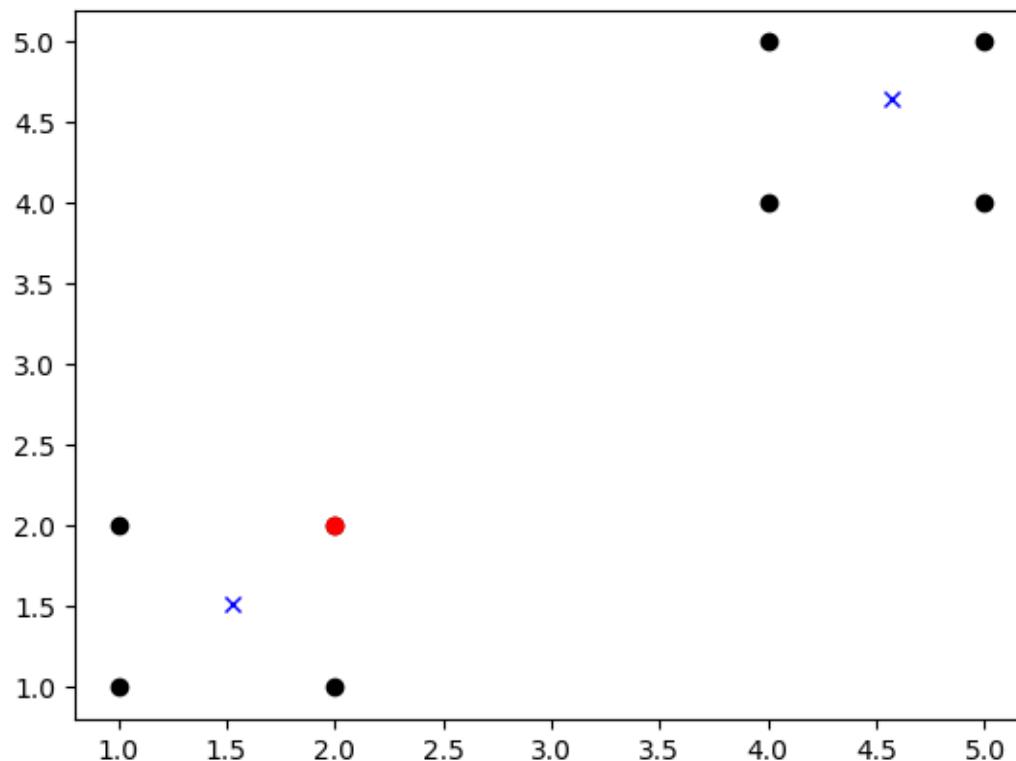
Epoch: 4, data: 3



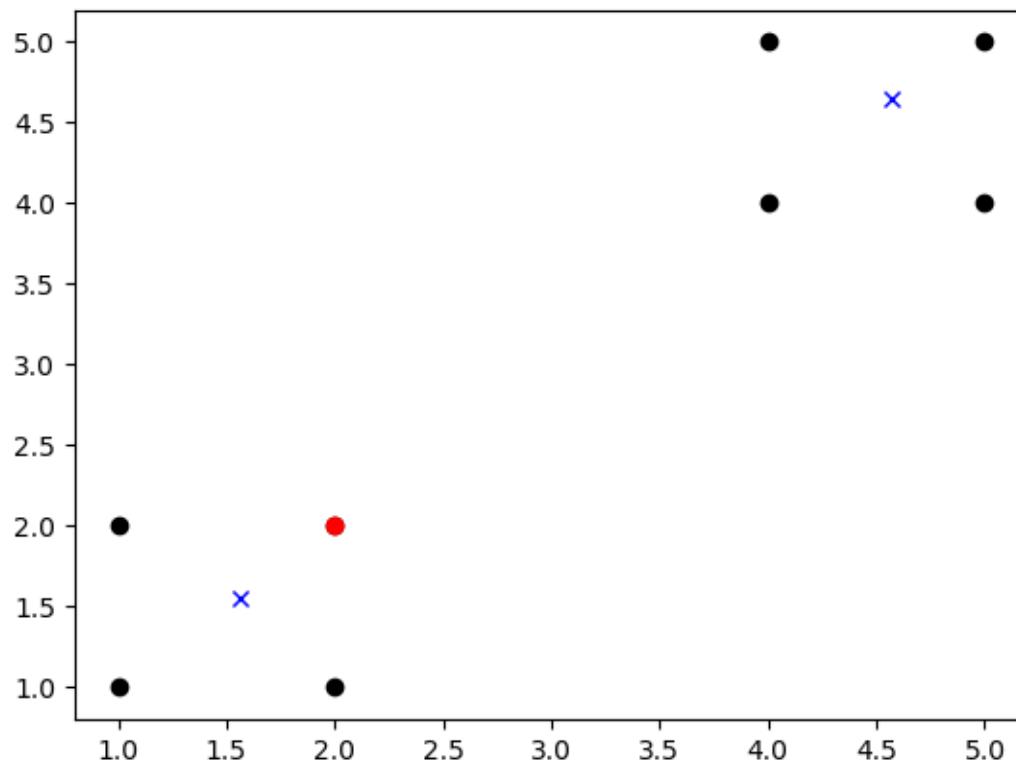
Epoch: 4, data: 3



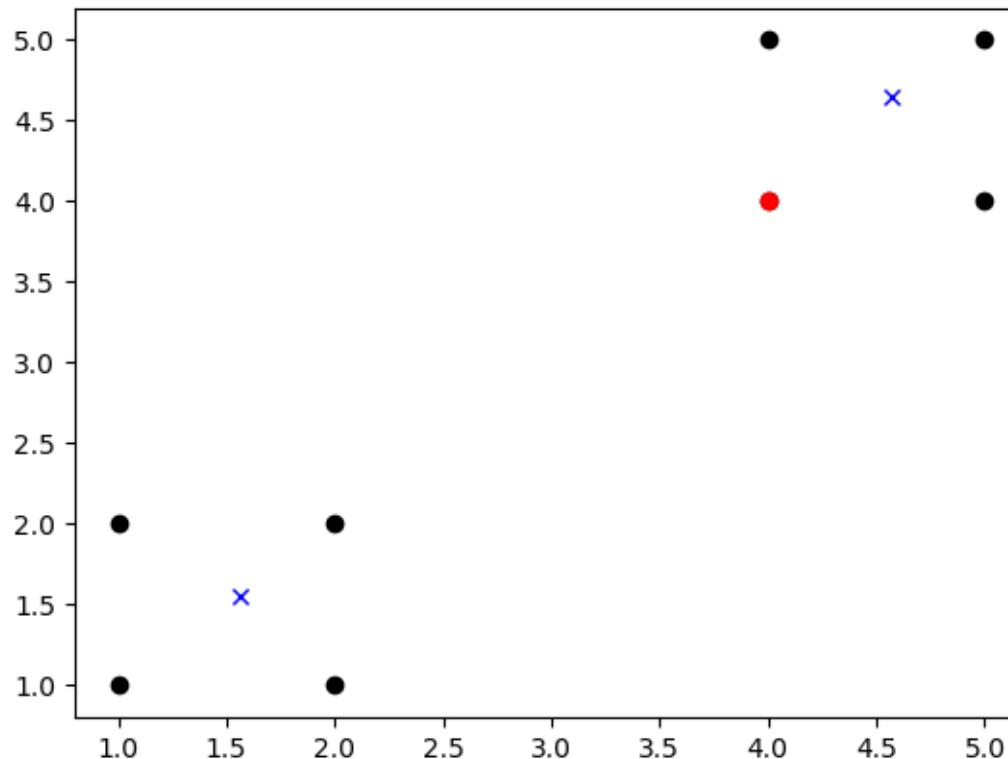
Epoch: 4, data: 4



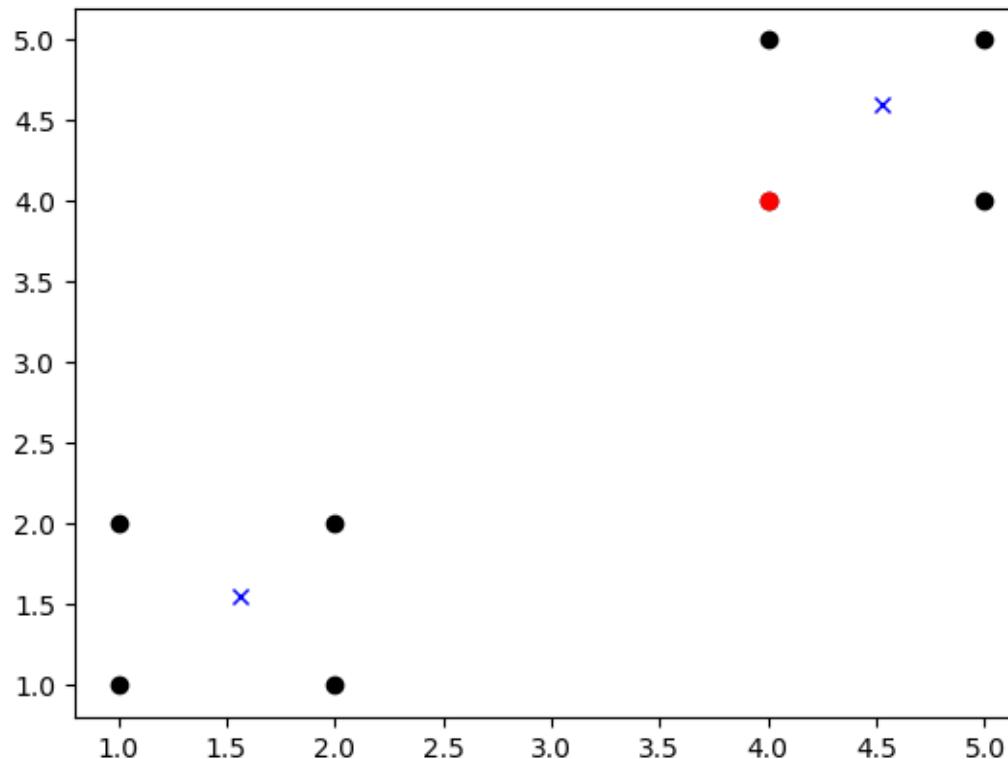
Epoch: 4, data: 4



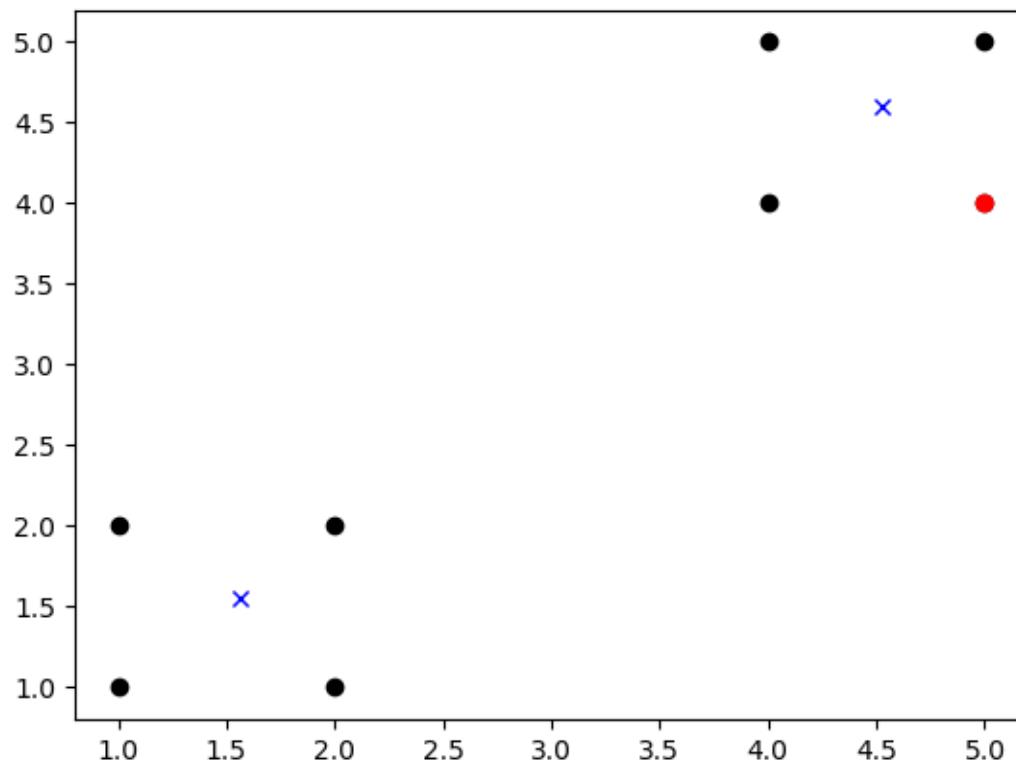
Epoch: 4, data: 5



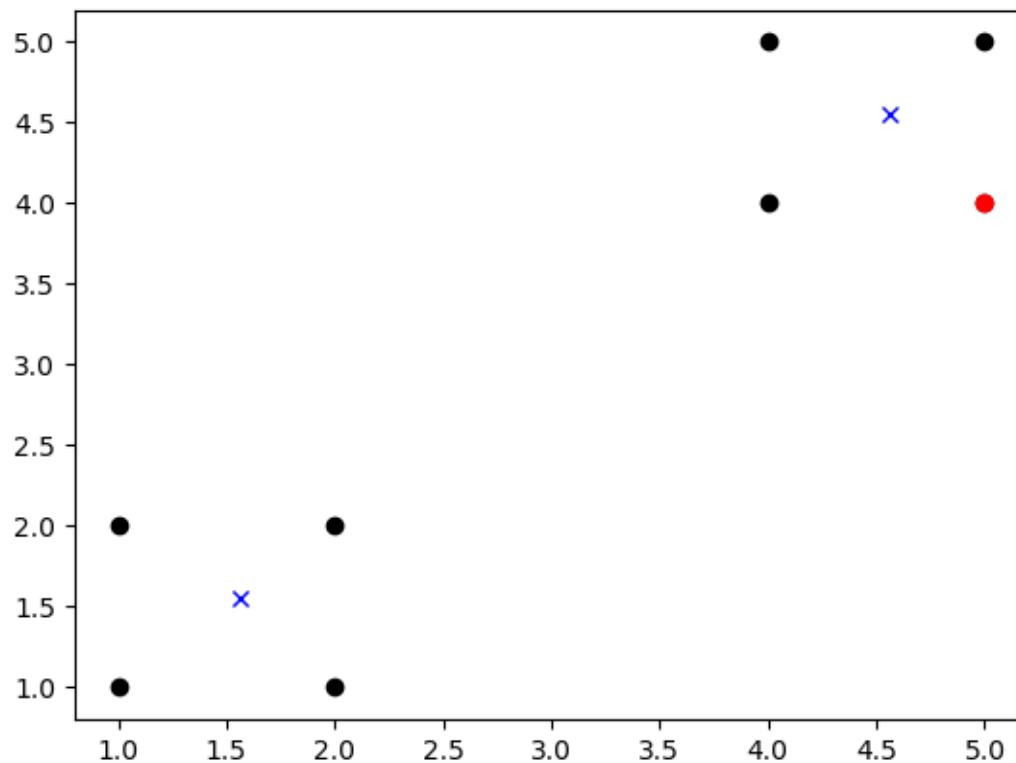
Epoch: 4, data: 5



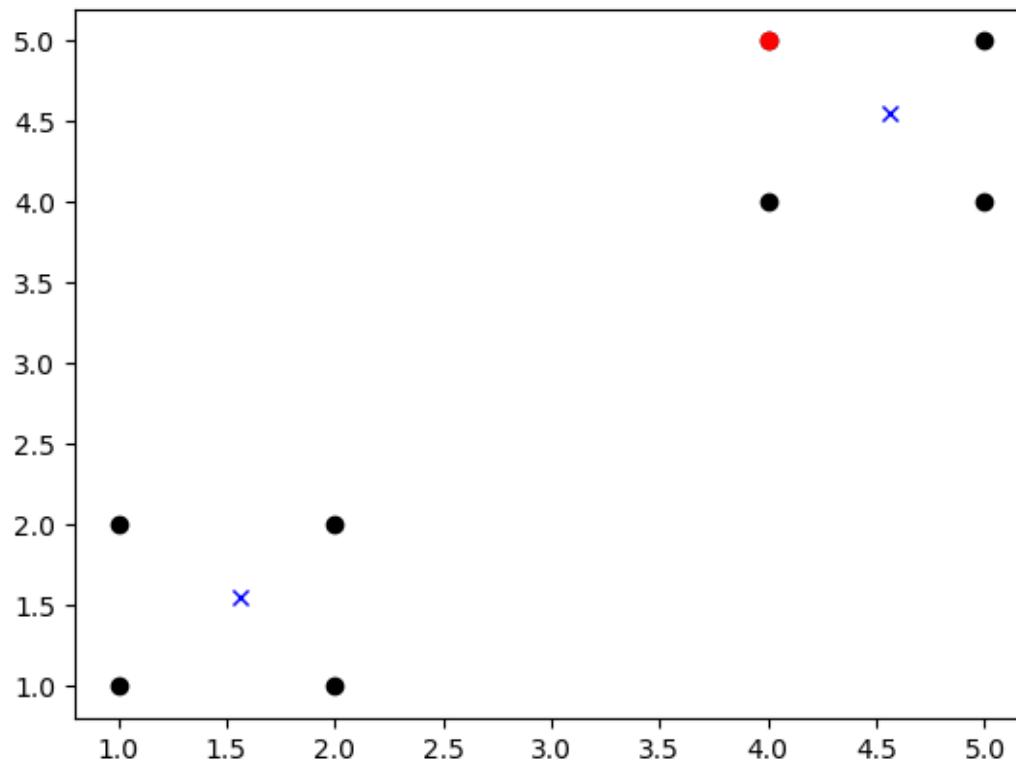
Epoch: 4, data: 6



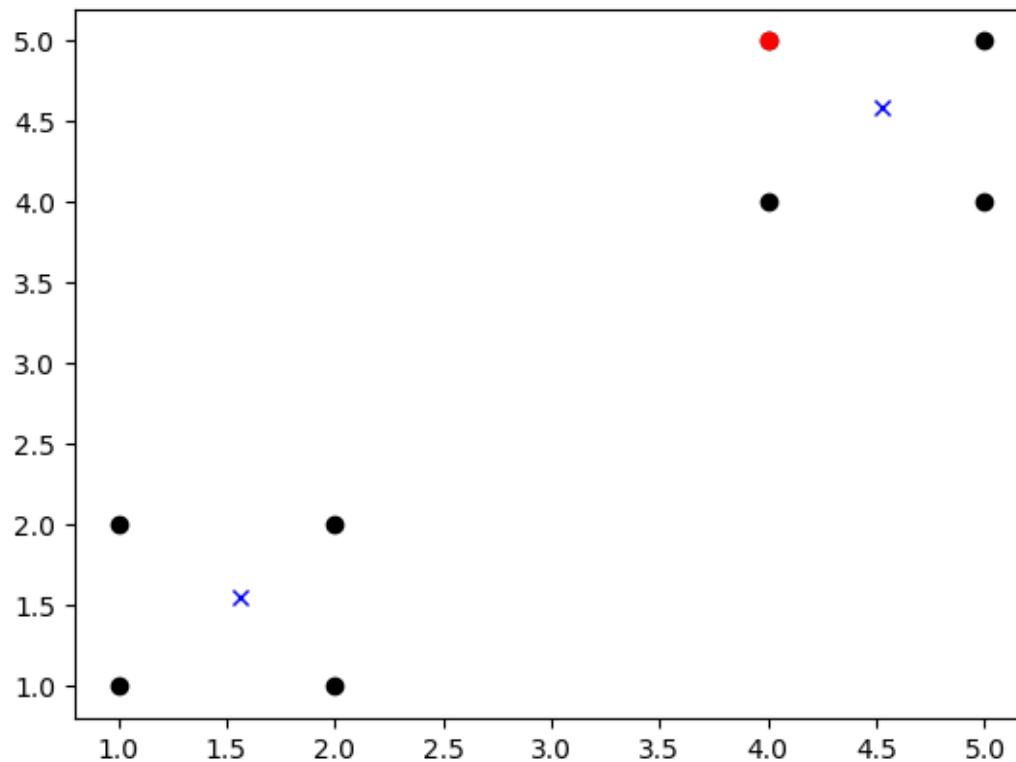
Epoch: 4, data: 6



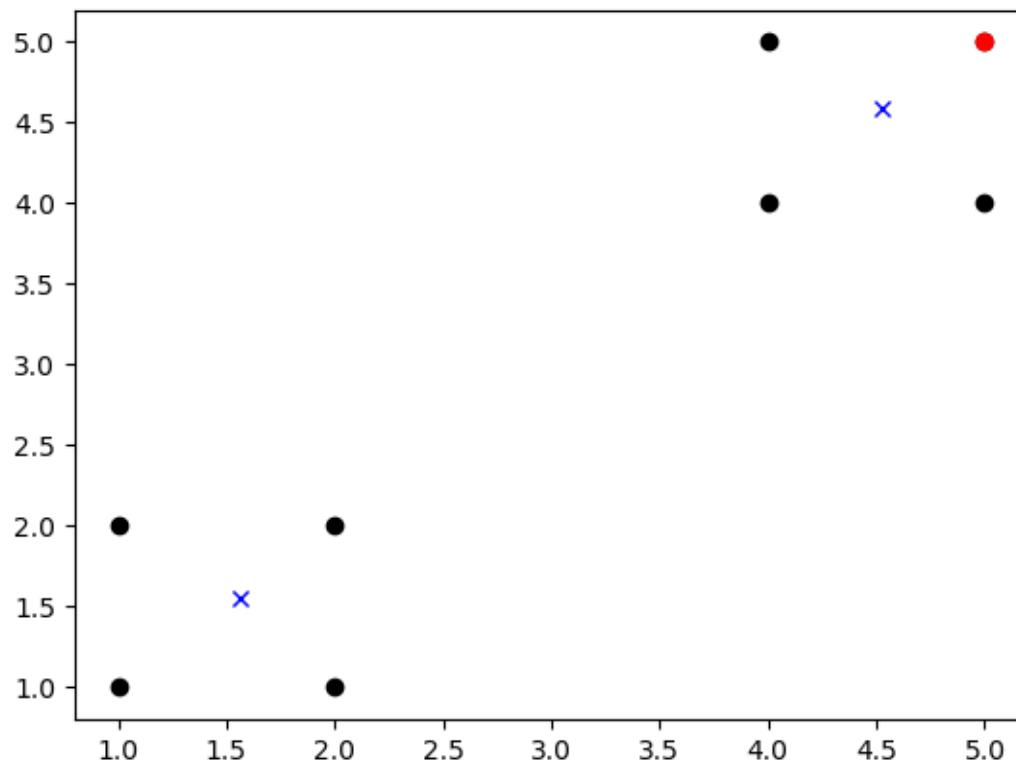
Epoch: 4, data: 7



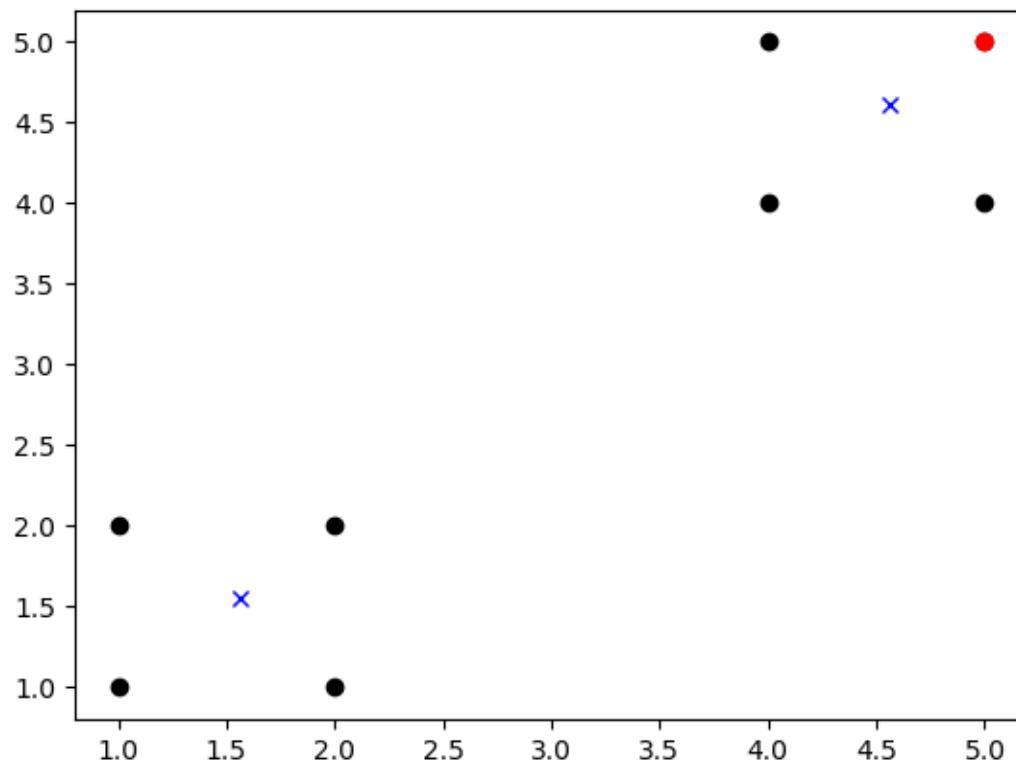
Epoch: 4, data: 7



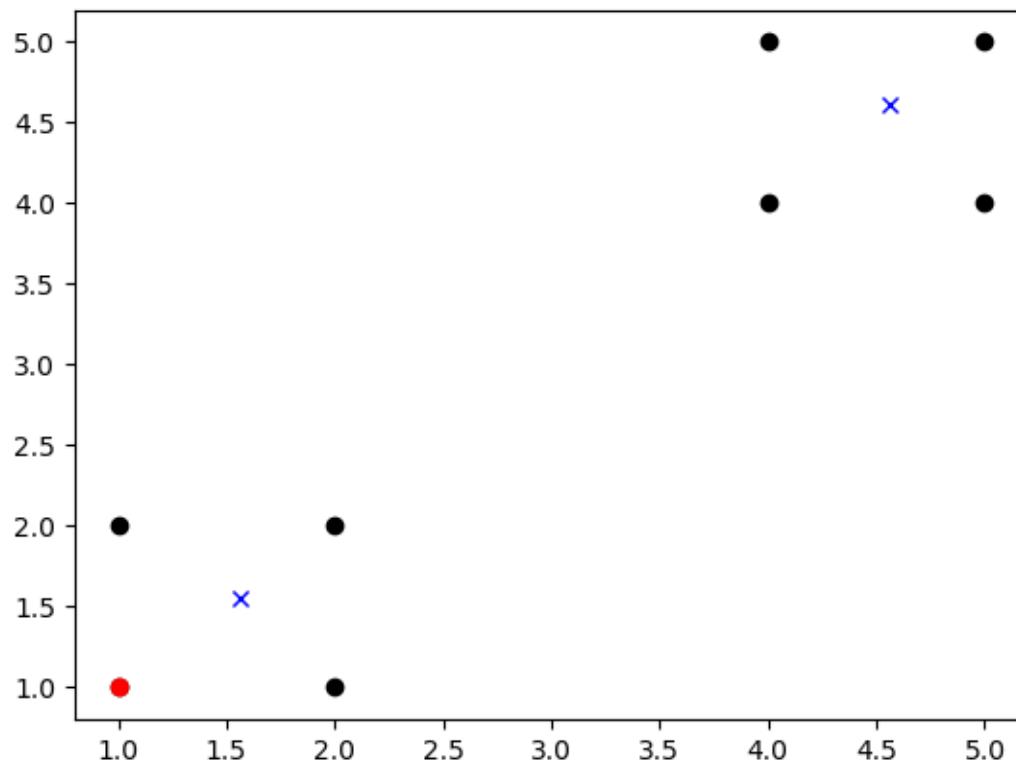
Epoch: 4, data: 8



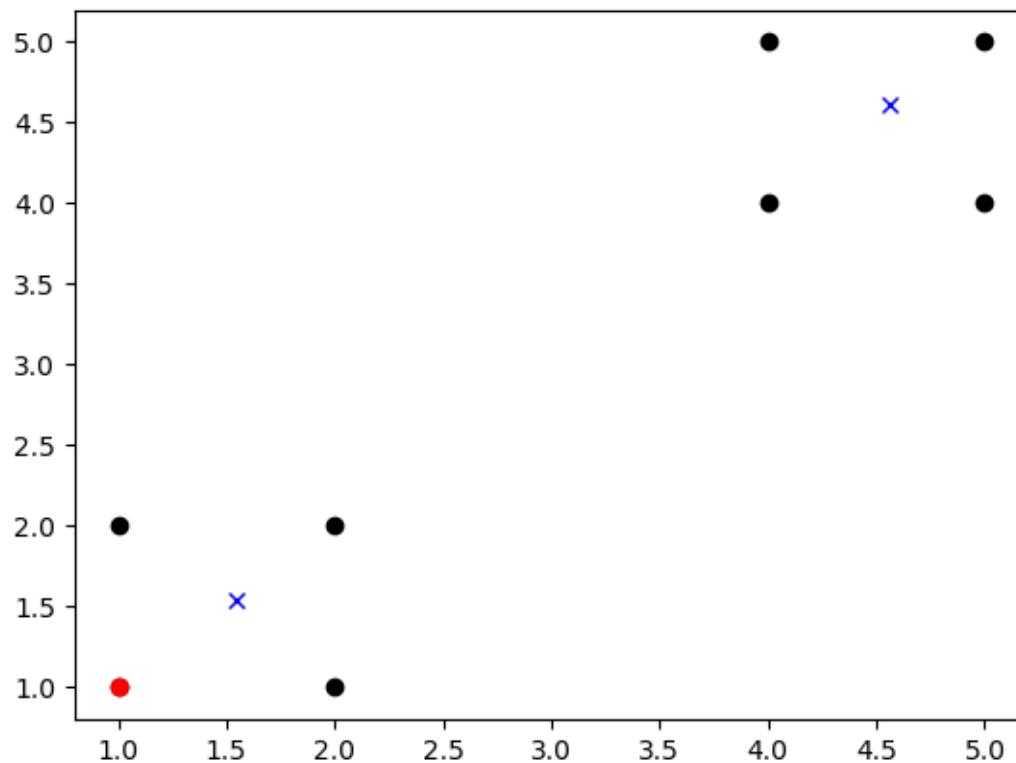
Epoch: 4, data: 8



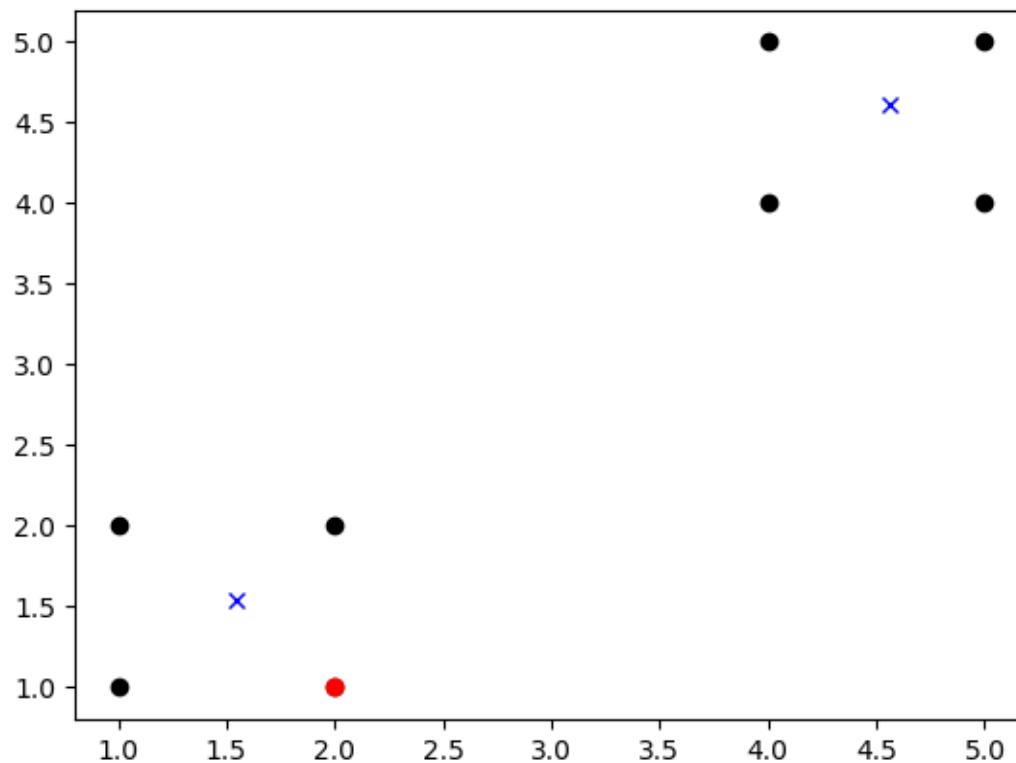
Epoch: 5, data: 1



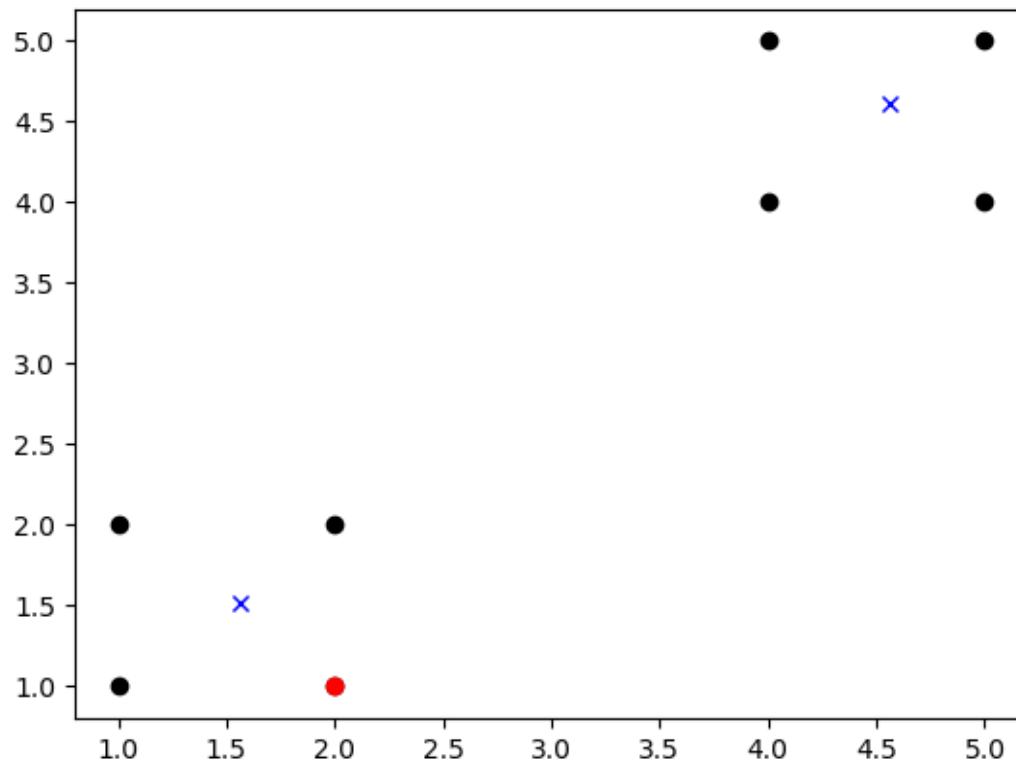
Epoch: 5, data: 1



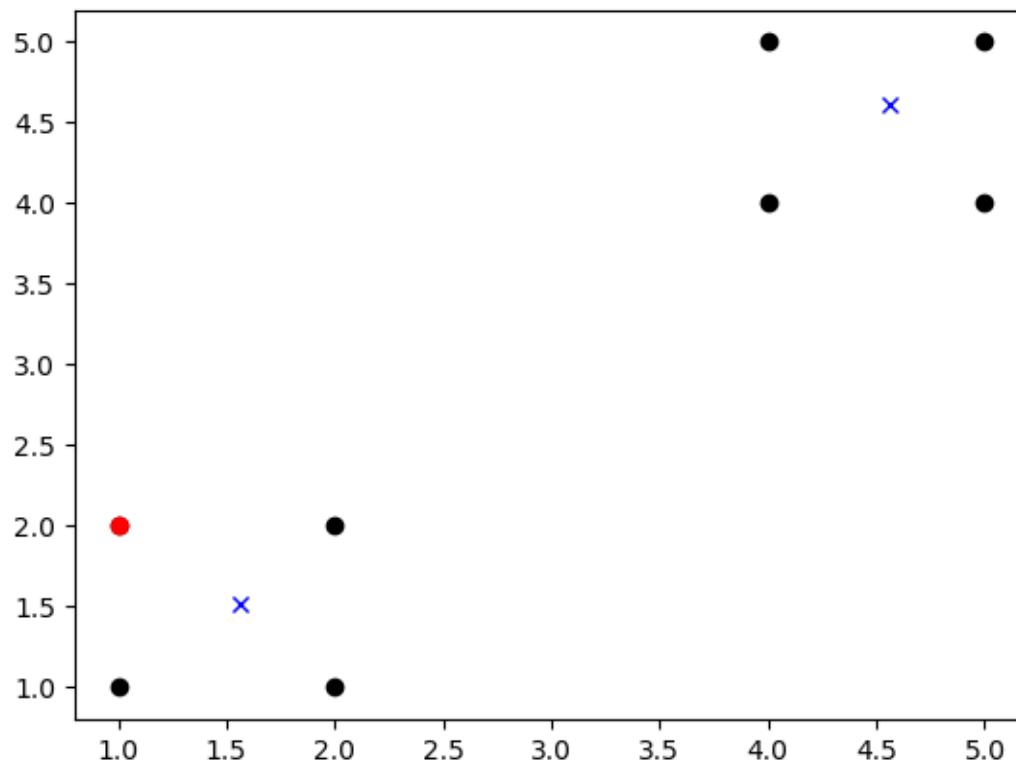
Epoch: 5, data: 2



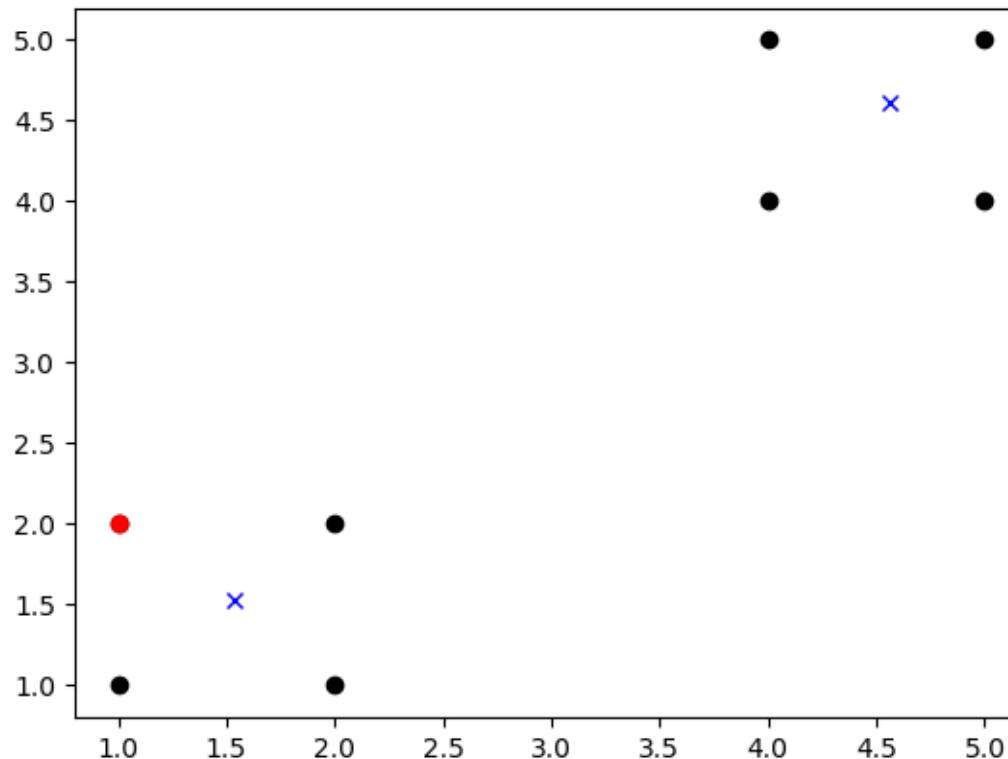
Epoch: 5, data: 2



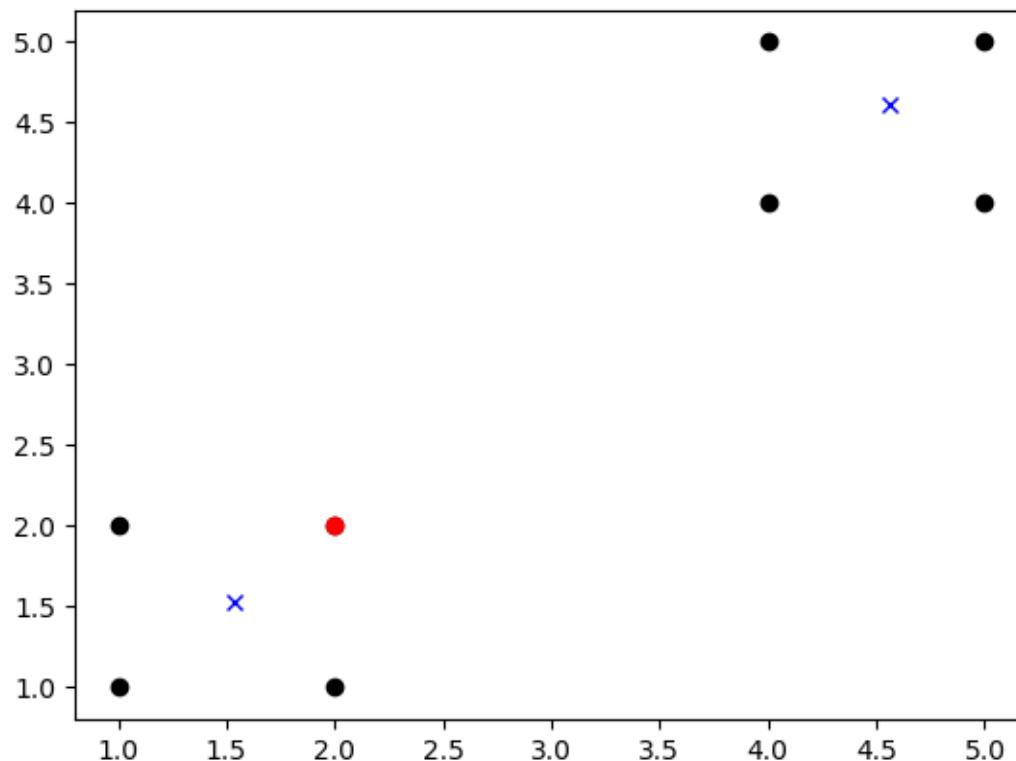
Epoch: 5, data: 3



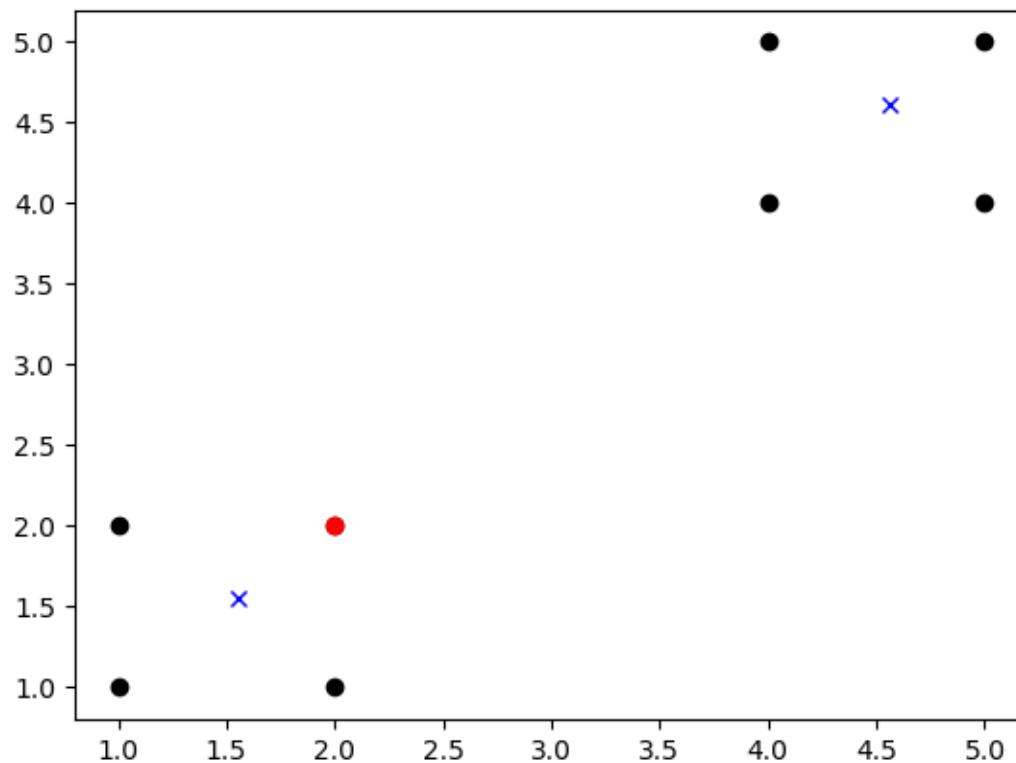
Epoch: 5, data: 3



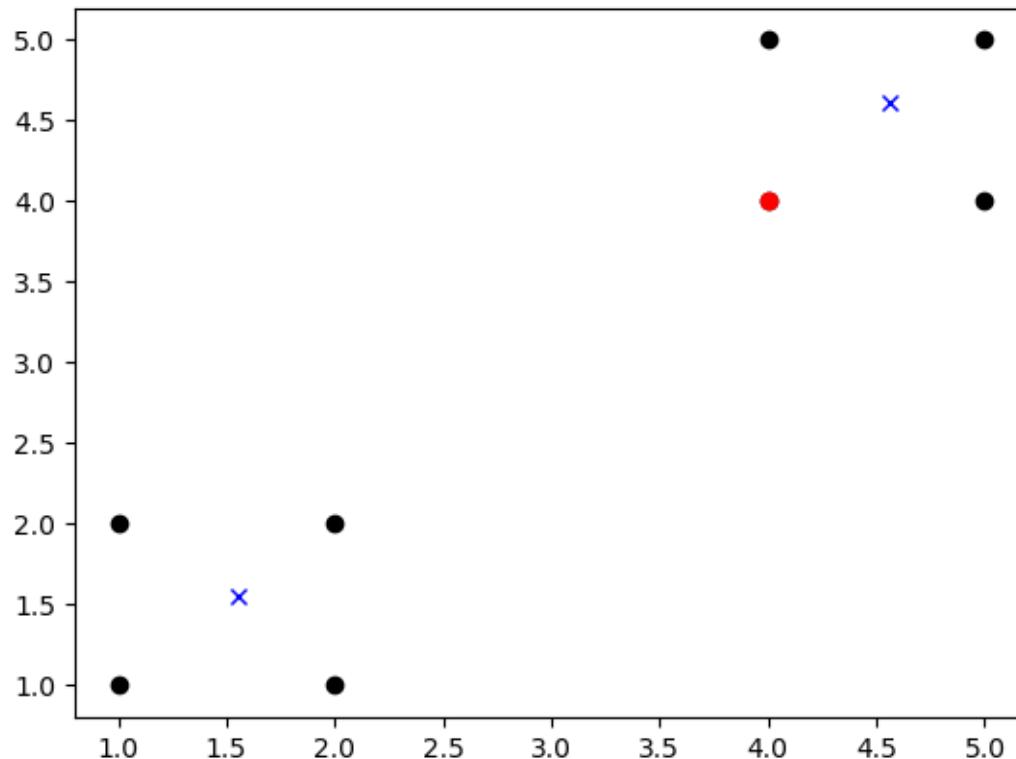
Epoch: 5, data: 4



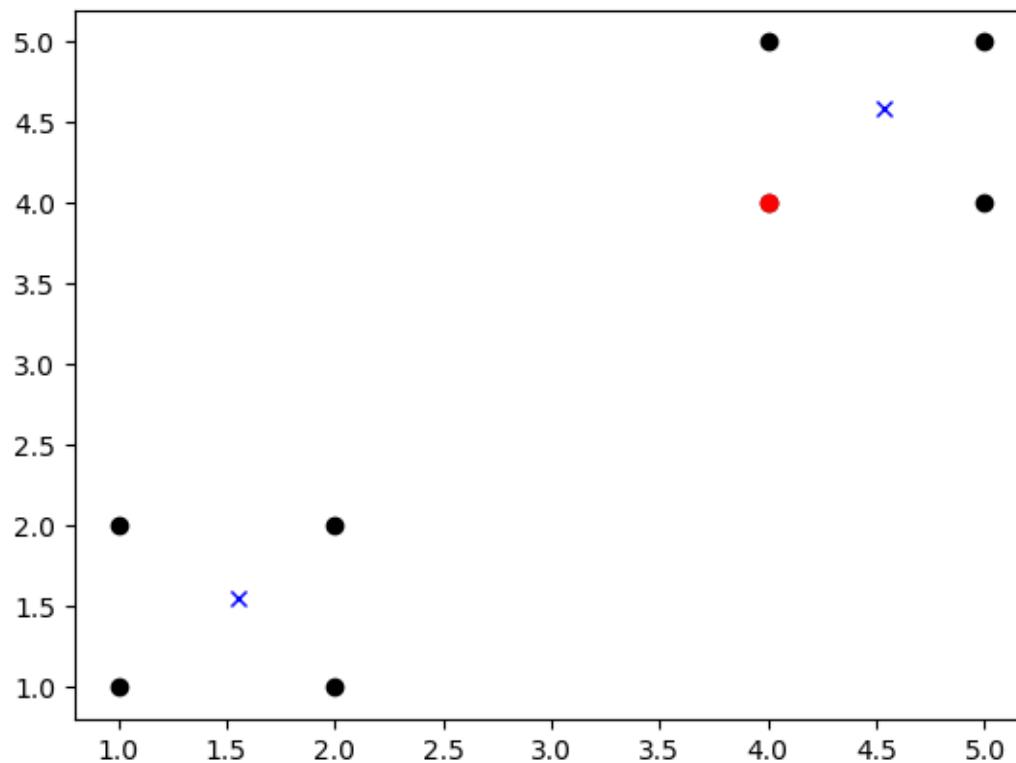
Epoch: 5, data: 4



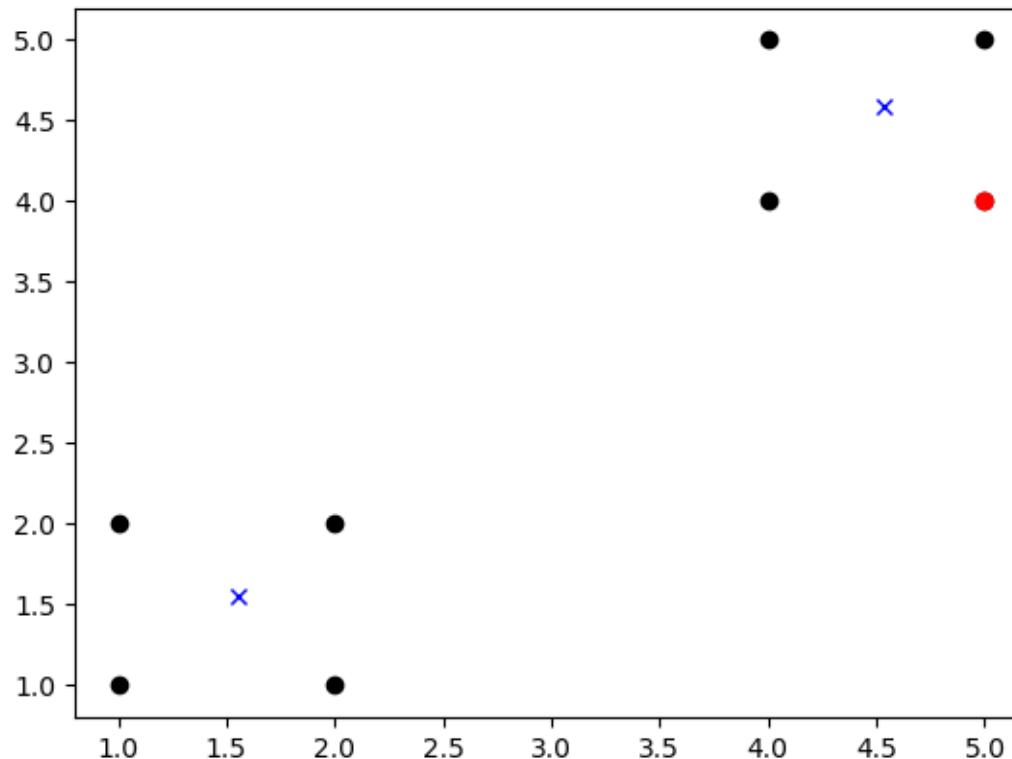
Epoch: 5, data: 5



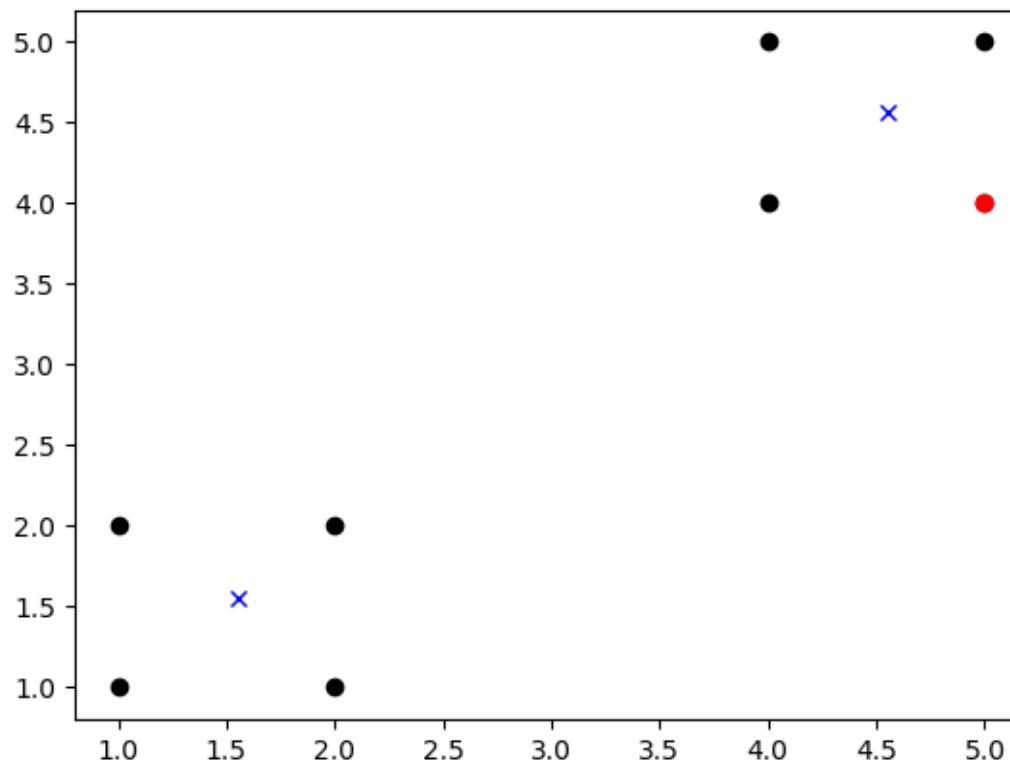
Epoch: 5, data: 5



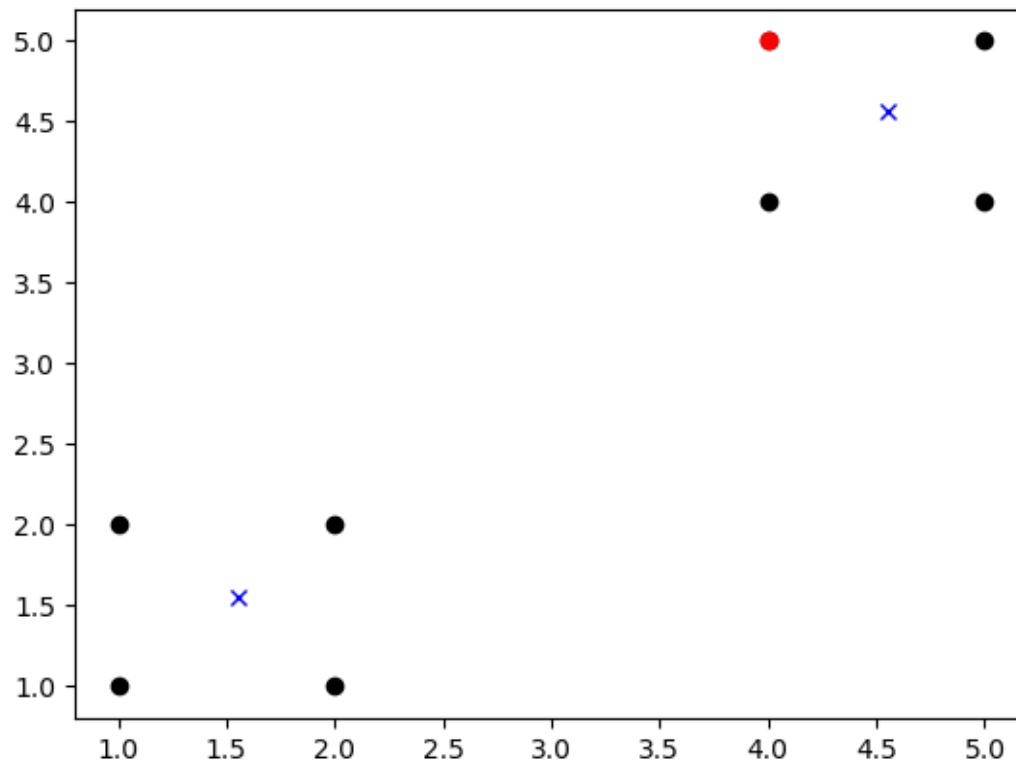
Epoch: 5, data: 6



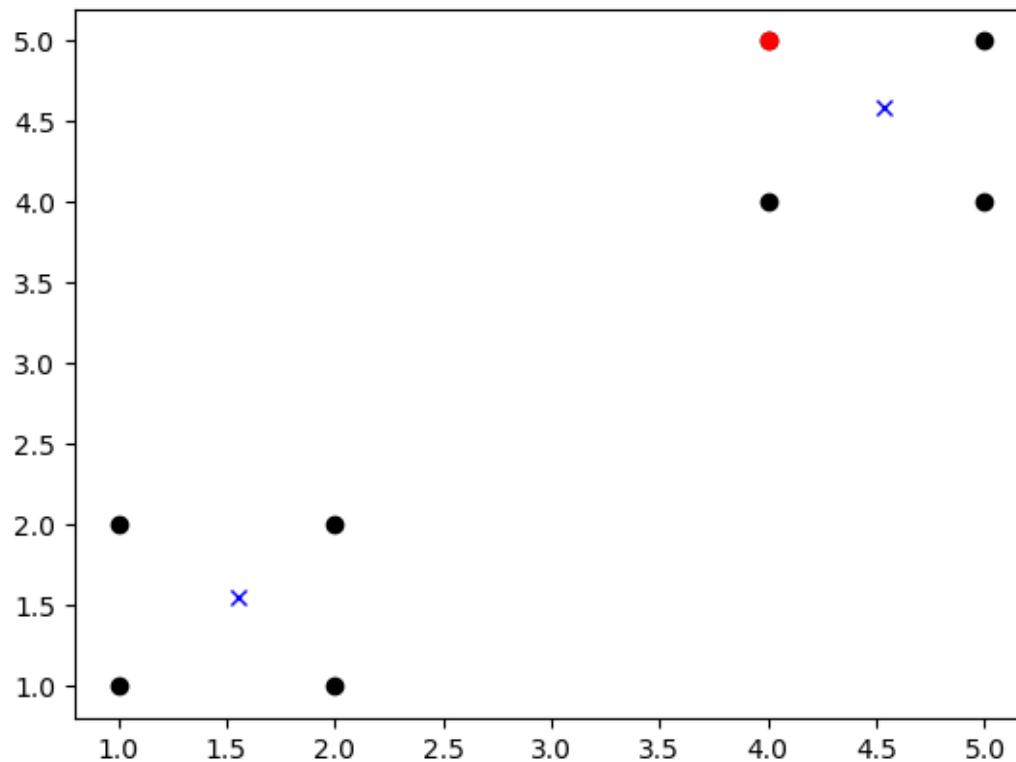
Epoch: 5, data: 6



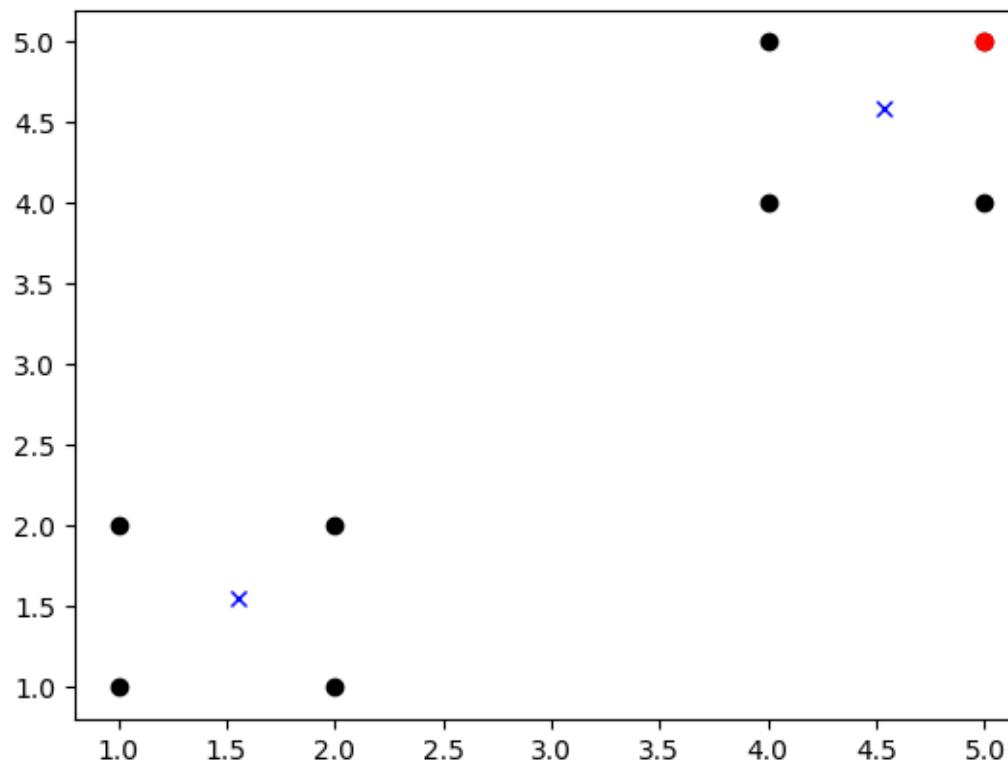
Epoch: 5, data: 7



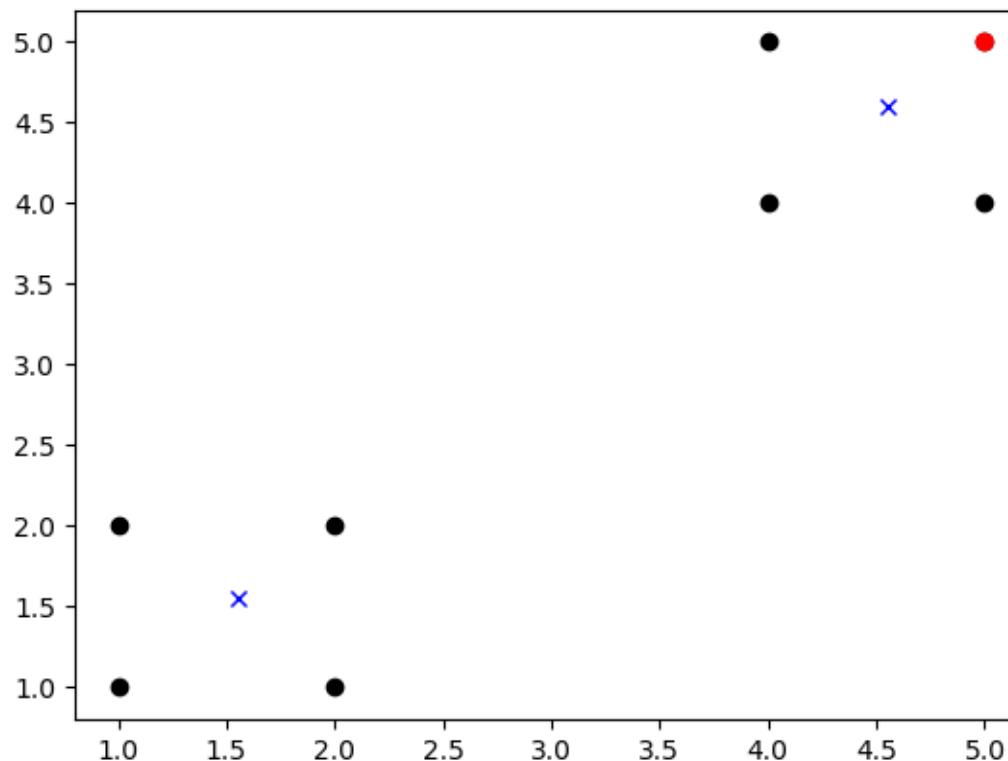
Epoch: 5, data: 7

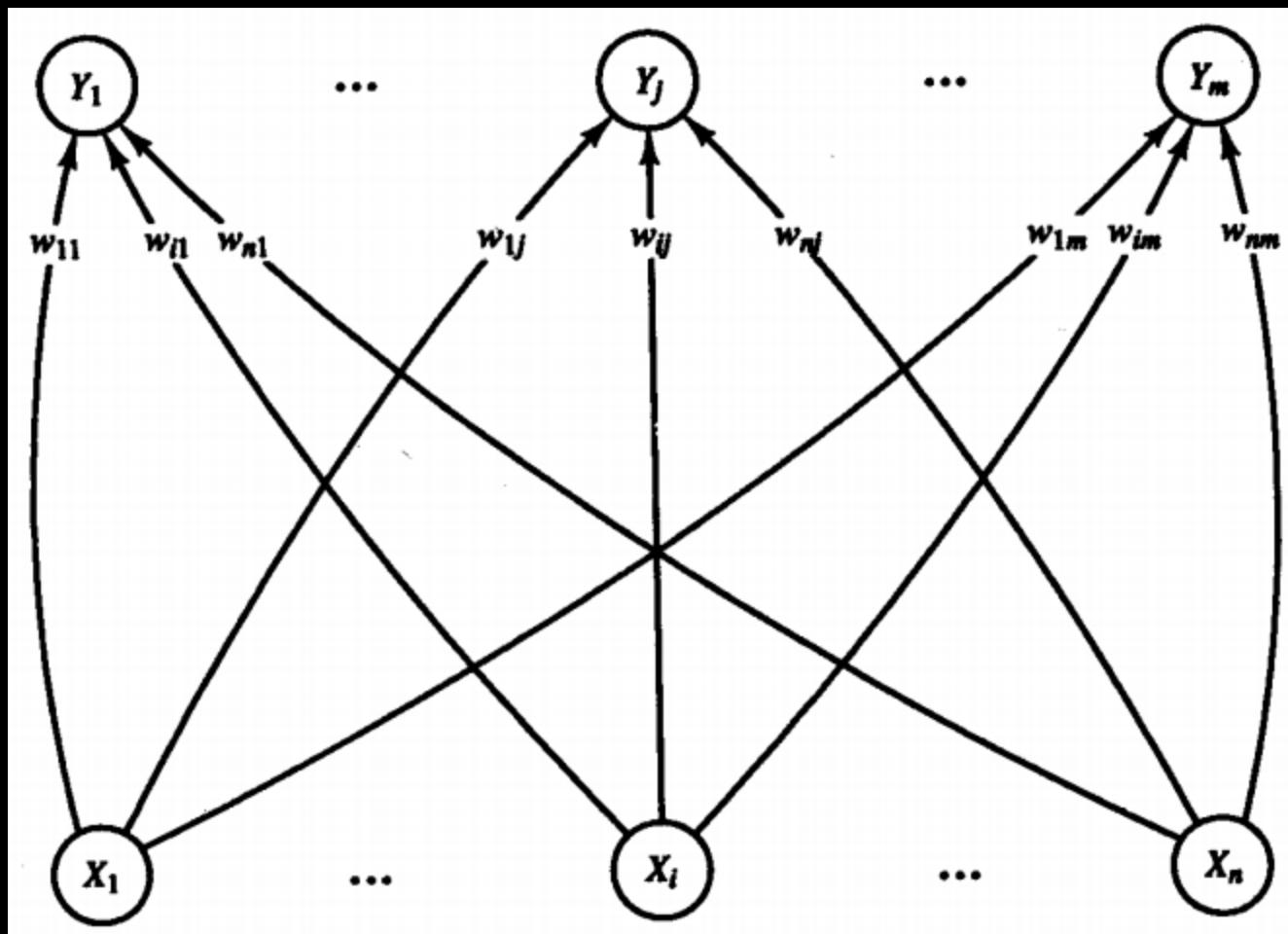


Epoch: 5, data: 8



Epoch: 5, data: 8





Sumber: Fausett (1994)

Algoritme Kohonen SOM

- I. Inisialisasi bobot w_{ij}
Set parameter ketetanggaan
Set *learning rate*
2. Selama kondisi berhenti bernilai false,
lakukan langkah 3–9
3. Untuk setiap vektor input x , lakukan
langkah 4–6

Algoritme Kohonen SOM

4. Untuk setiap nilai j , hitung:

$$D(j) = \sum_i (w_{ij} - x_i)^2$$

5. Tentukan indeks J yang menghasilkan $D(J)$ minimum
6. Untuk semua neuron j yang merupakan tetangga dari J dengan radius R :

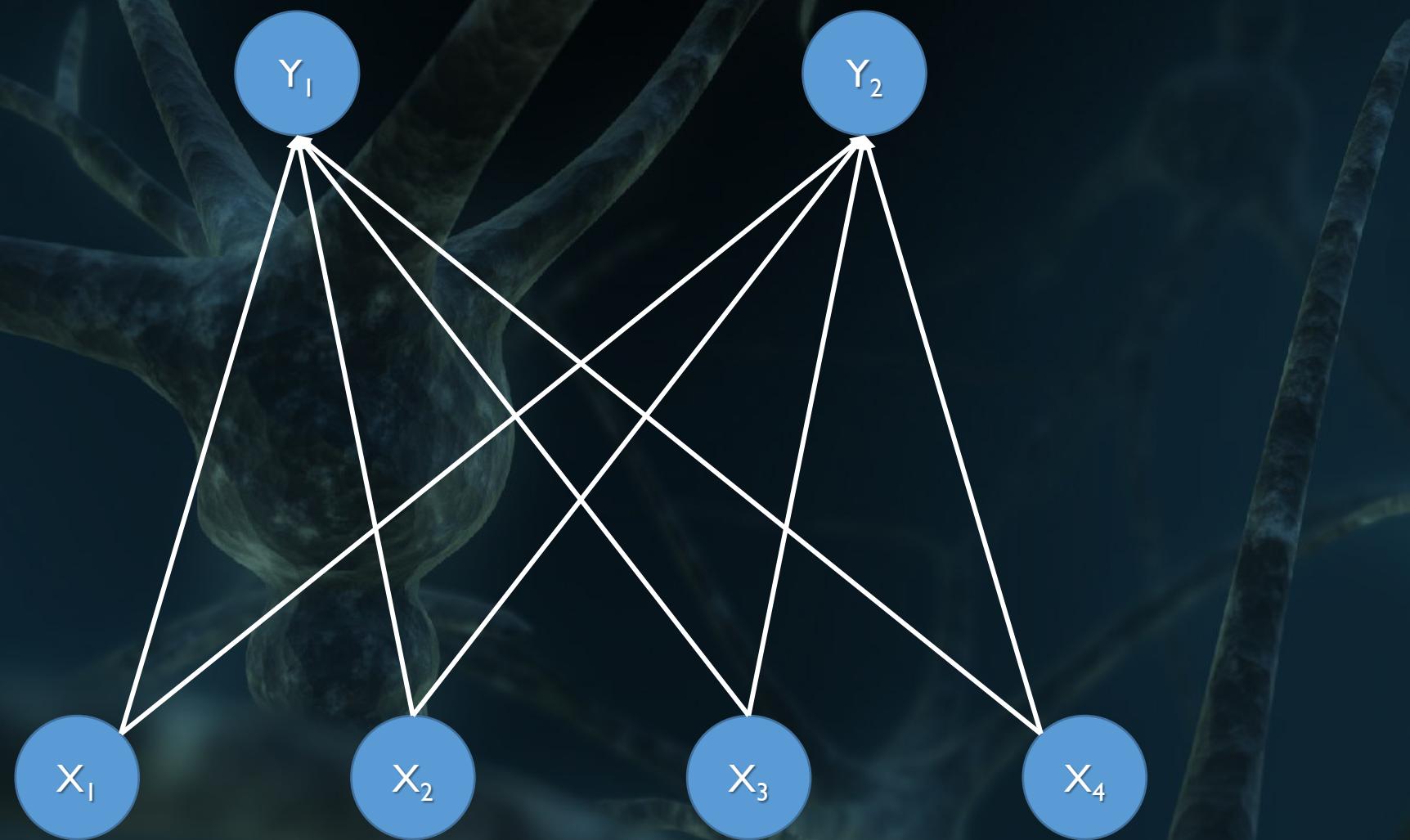
$$w'_{ij} = w_{ij} + \alpha [x_i - w_{ij}]$$

Algoritme Kohonen SOM

7. Ubah *learning rate* α
8. Perkecil radius ketetanggaan R
9. Cek kondisi berhenti

Contoh

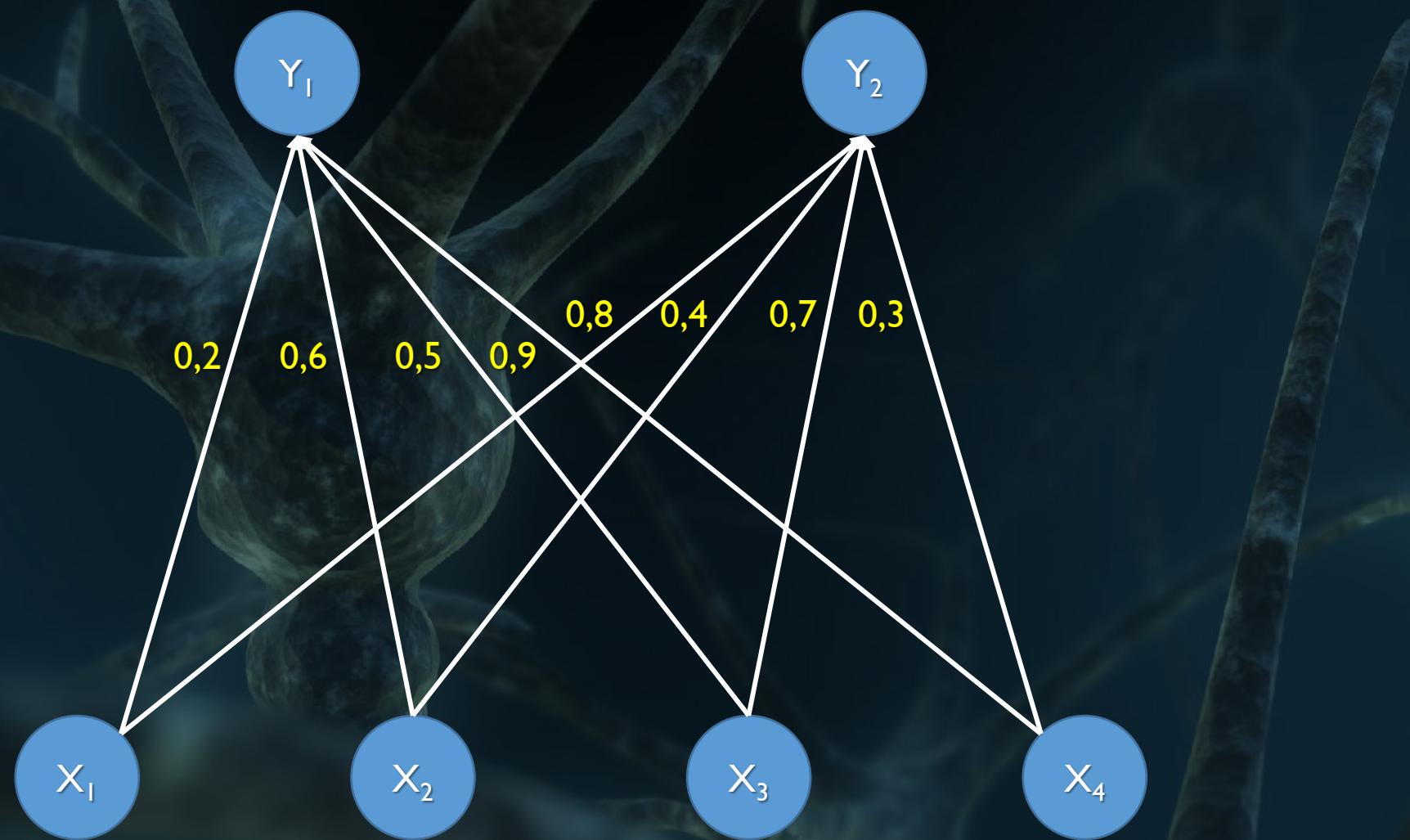
- *Clustering* empat vektor berikut menjadi dua *cluster*
 $(1, 1, 0, 0), (0, 0, 0, 1), (1, 0, 0, 0), (0, 0, 1, 1)$
- *Learning rate* (α) awal: 0,6
- Radius (R) awal: 0
- Perubahan *learning rate*:
$$\alpha' = 0,5 \times \alpha$$

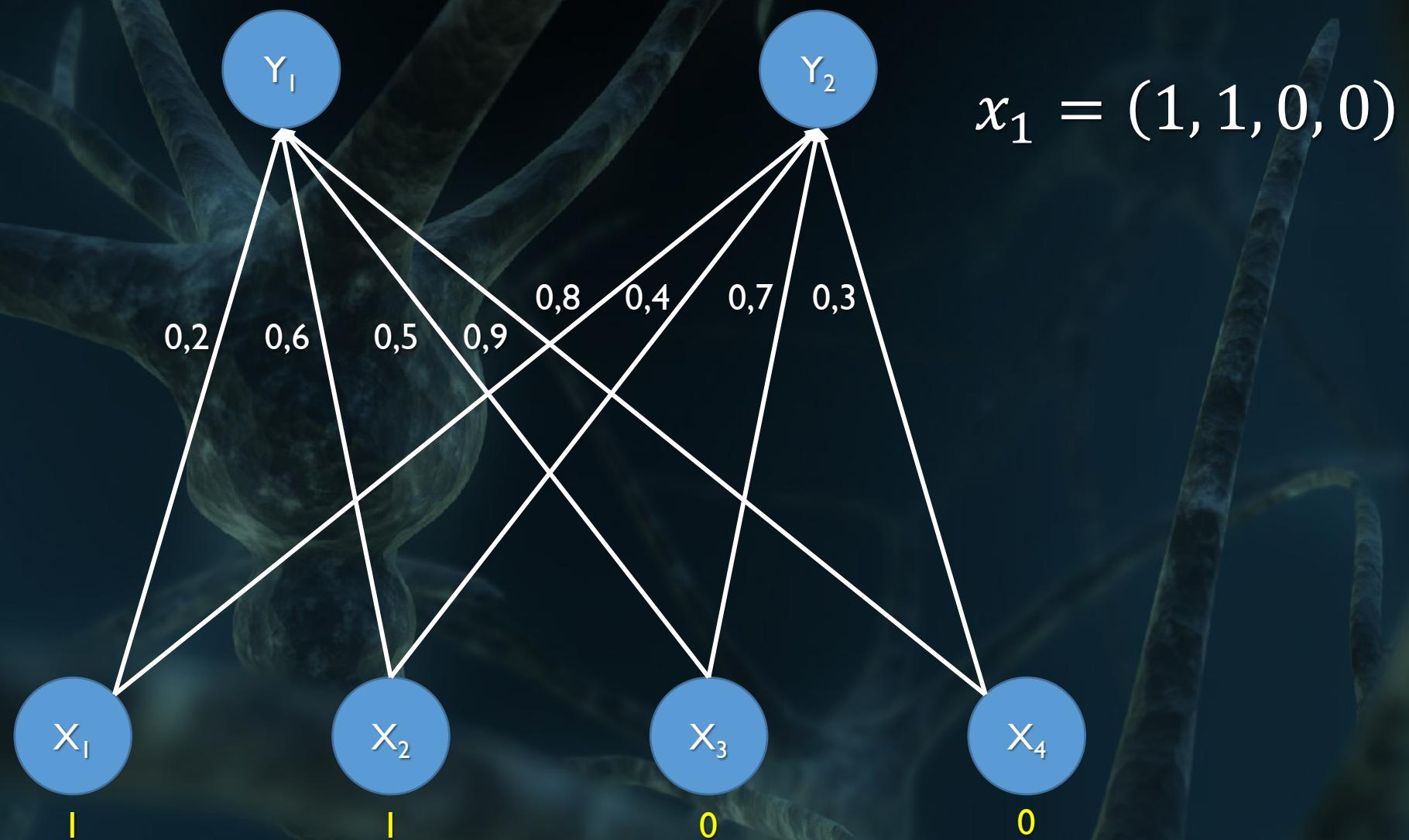


Contoh

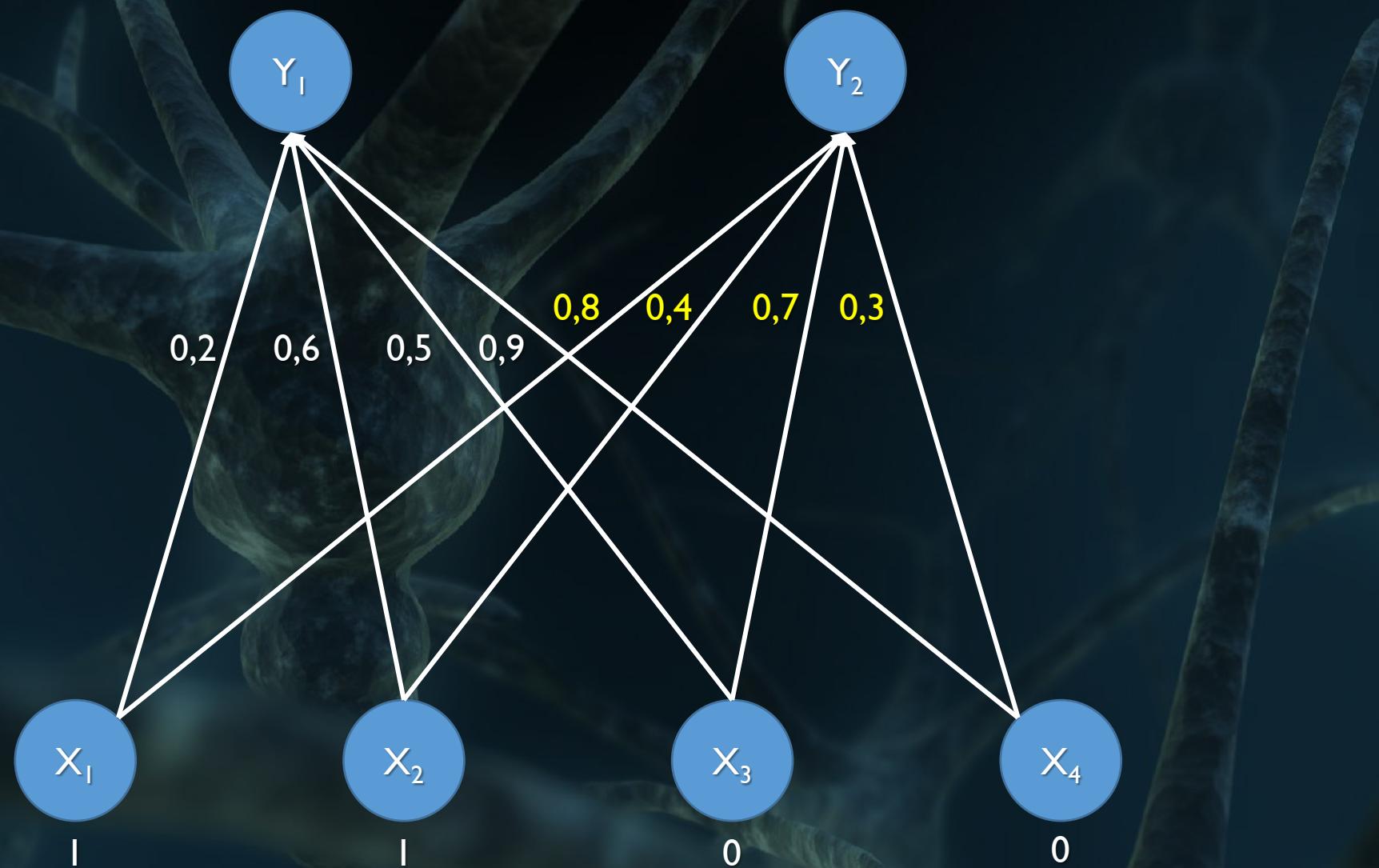
- Bobot awal (acak):

$$\begin{bmatrix} 0,2 & 0,8 \\ 0,6 & 0,4 \\ 0,5 & 0,7 \\ 0,9 & 0,3 \end{bmatrix}$$

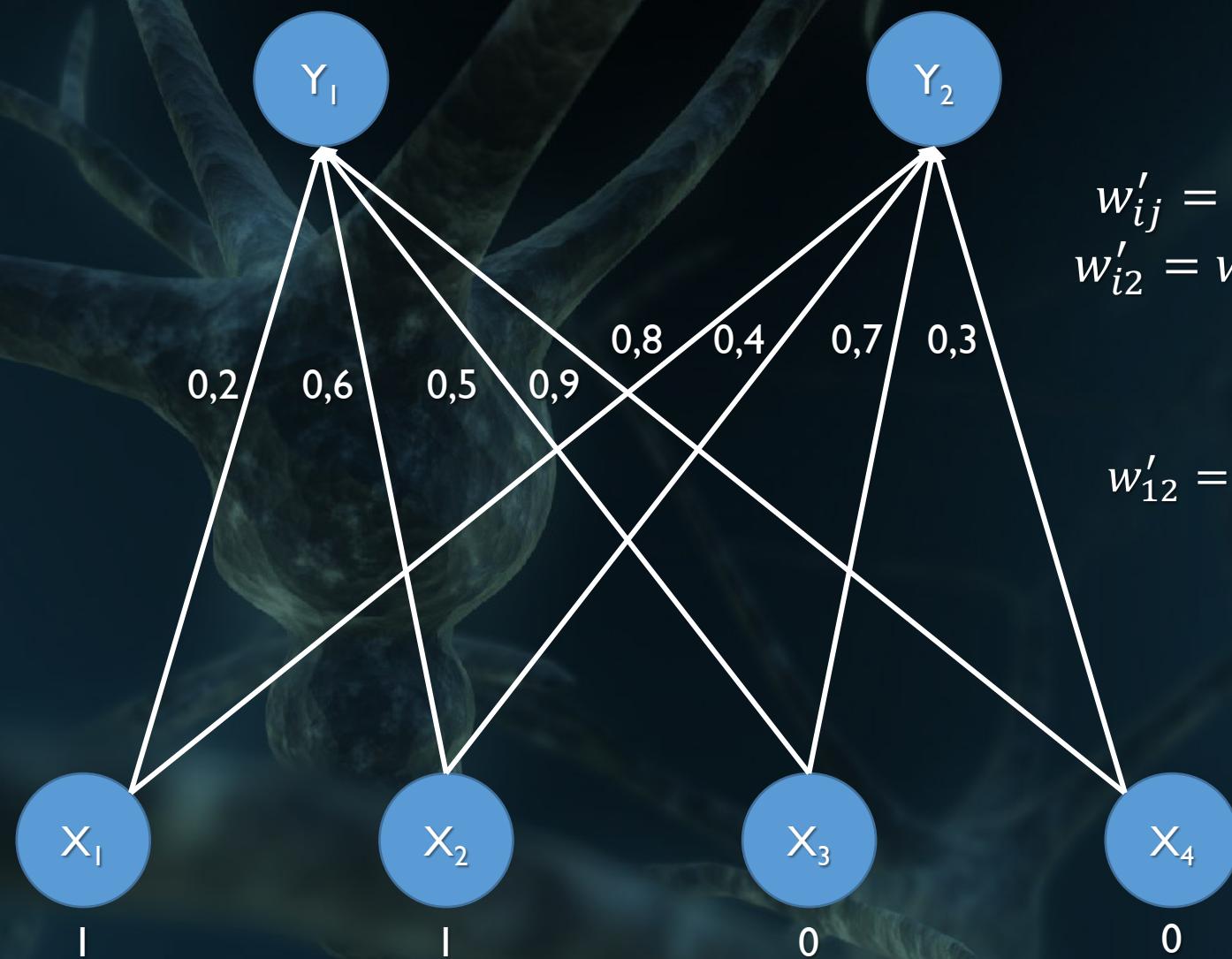




$$x_1 = (1, 1, 0, 0)$$



$$D(1) = (0,2 - 1)^2 + (0,6 - 1)^2 + (0,5 - 0)^2 + (0,9 - 0)^2 = 1,86$$
$$D(2) = (0,8 - 1)^2 + (0,4 - 1)^2 + (0,7 - 0)^2 + (0,3 - 0)^2 = 0,98$$



$$w'_{ij} = w_{ij} + \alpha[x_i - w_{ij}]$$

$$w'_{i2} = w_{i2} + 0,6[x_i - w_{i2}]$$

$$w'_{12} = 0,8 + 0,6[1 - 0,8]$$

$$= 0,8 + 0,1$$

$$= 0,92$$

$$w'_{22} = 0,4 + 0,6[1 - 0,4]$$

$$= 0,4 + 0,36$$

$$= 0,76$$

$$w'_{32} = 0,7 + 0,6[0 - 0,7]$$

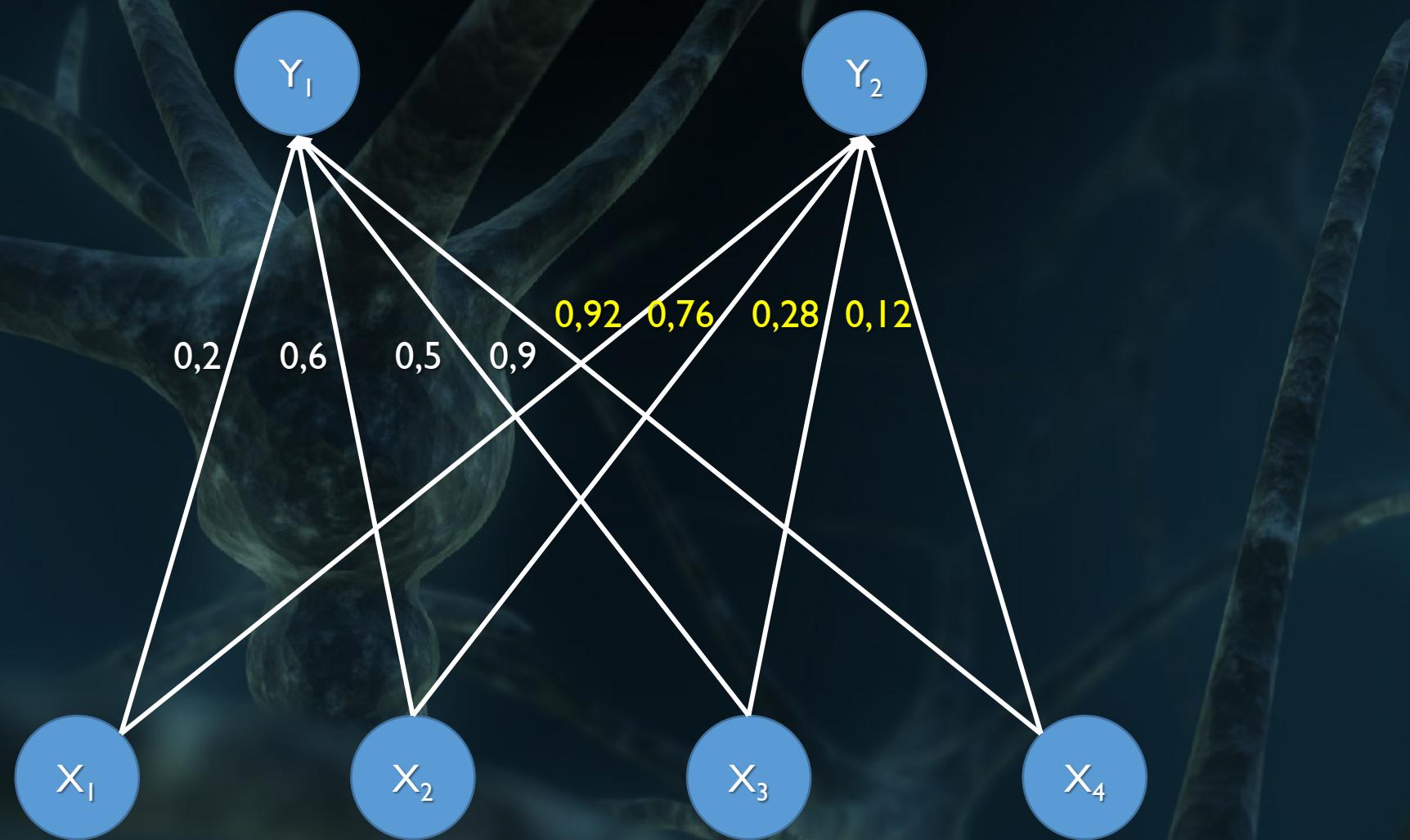
$$= 0,7 - 0,42$$

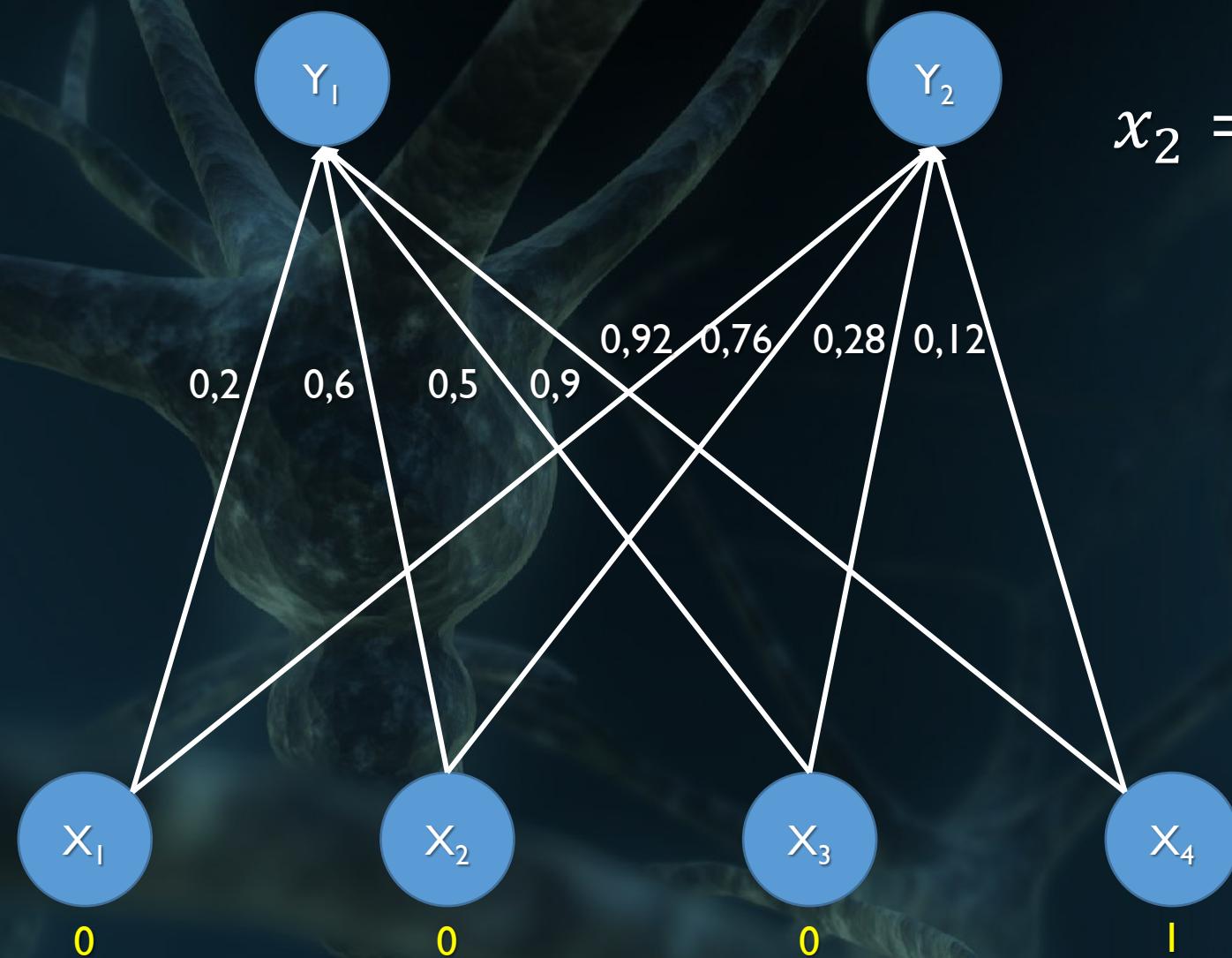
$$= 0,28$$

$$w'_{42} = 0,3 + 0,6[0 - 0,3]$$

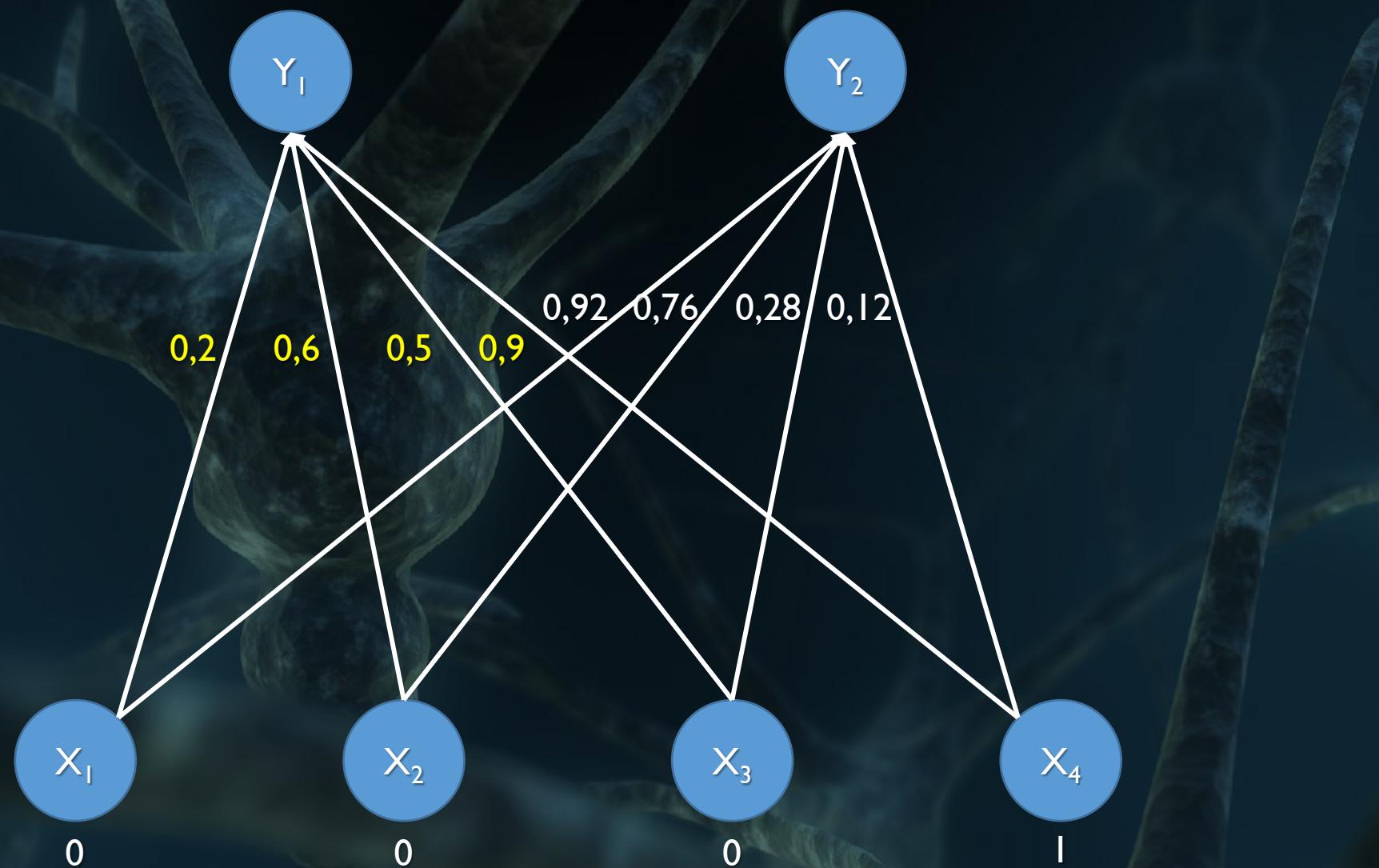
$$= 0,3 - 0,18$$

$$= 0,12$$



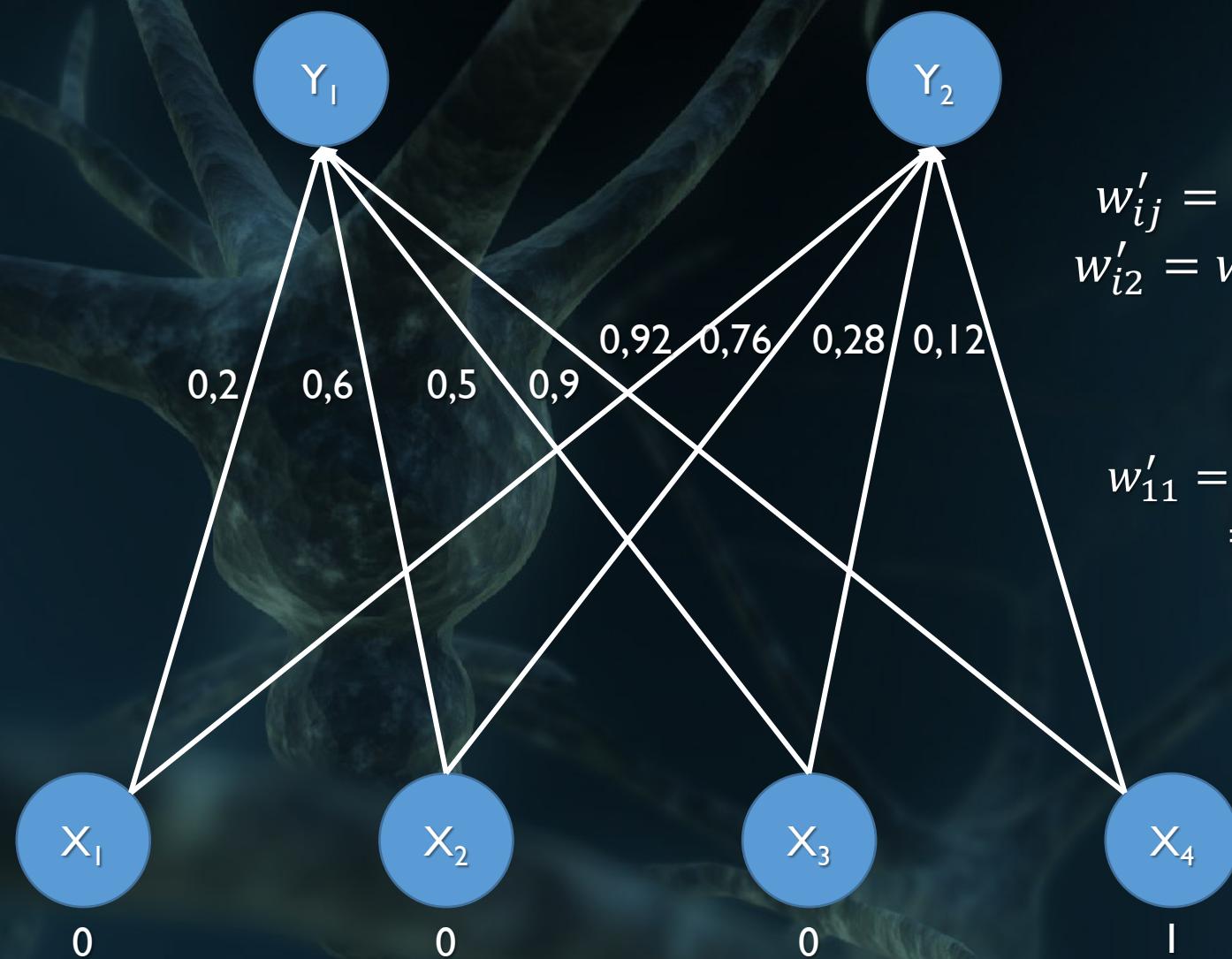


$$x_2 = (0, 0, 0, 1)$$



$$D(1) = (0,2 - 0)^2 + (0,6 - 0)^2 + (0,5 - 0)^2 + (0,9 - 1)^2 = 0,66$$

$$D(2) = (0,92 - 0)^2 + (0,76 - 0)^2 + (0,28 - 0)^2 + (0,12 - 1)^2 = 2,2768$$



$$w'_{ij} = w_{ij} + \alpha[x_i - w_{ij}]$$

$$w'_{i2} = w_{i2} + 0,6[x_i - w_{i2}]$$

$$w'_{11} = 0,2 + 0,6[0 - 0,2]$$

$$= 0,2 - 0,12$$

$$= 0,08$$

$$w'_{21} = 0,6 + 0,6[0 - 0,6]$$

$$= 0,6 + 0,36$$

$$= 0,24$$

$$w'_{31} = 0,5 + 0,6[0 - 0,5]$$

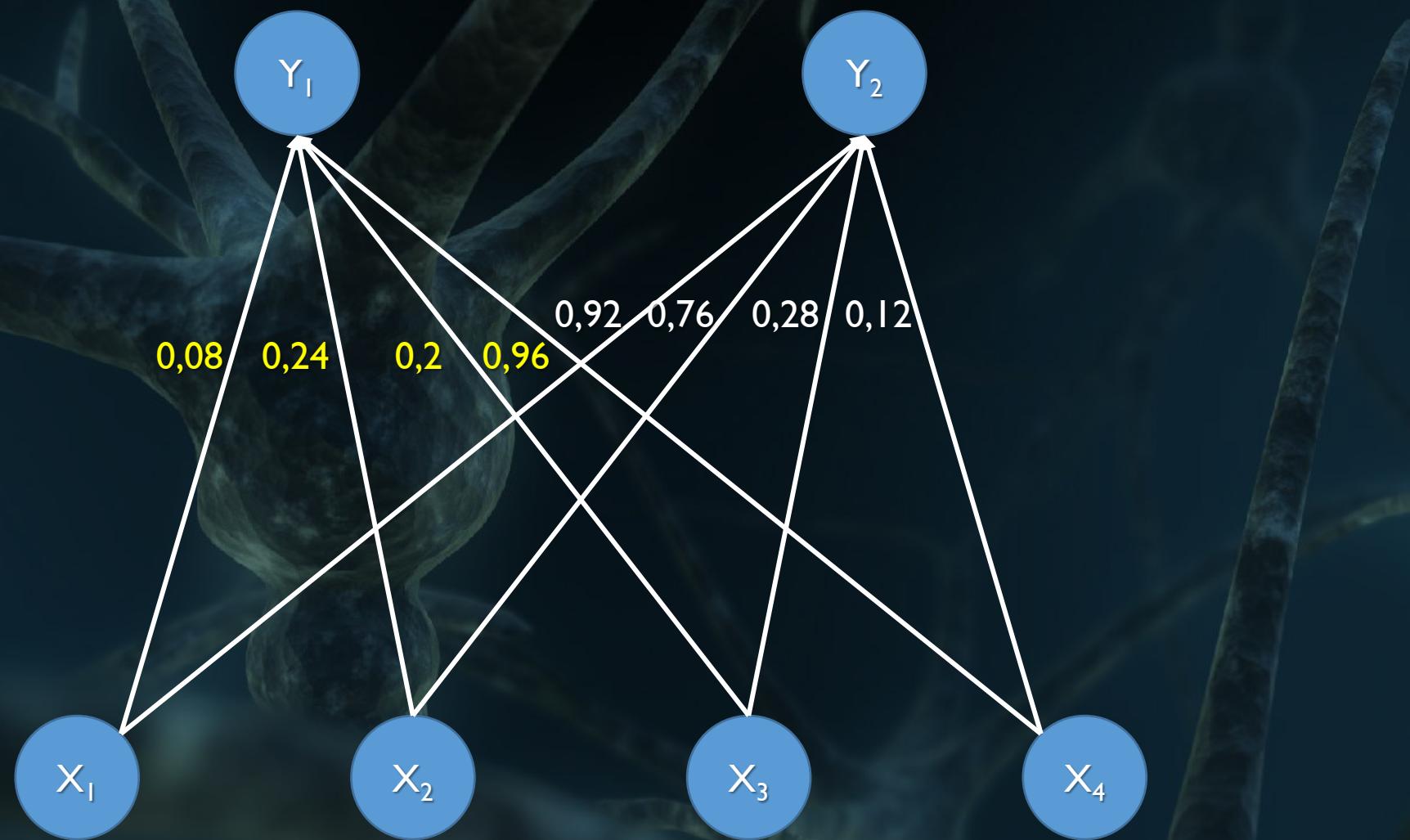
$$= 0,5 - 0,3$$

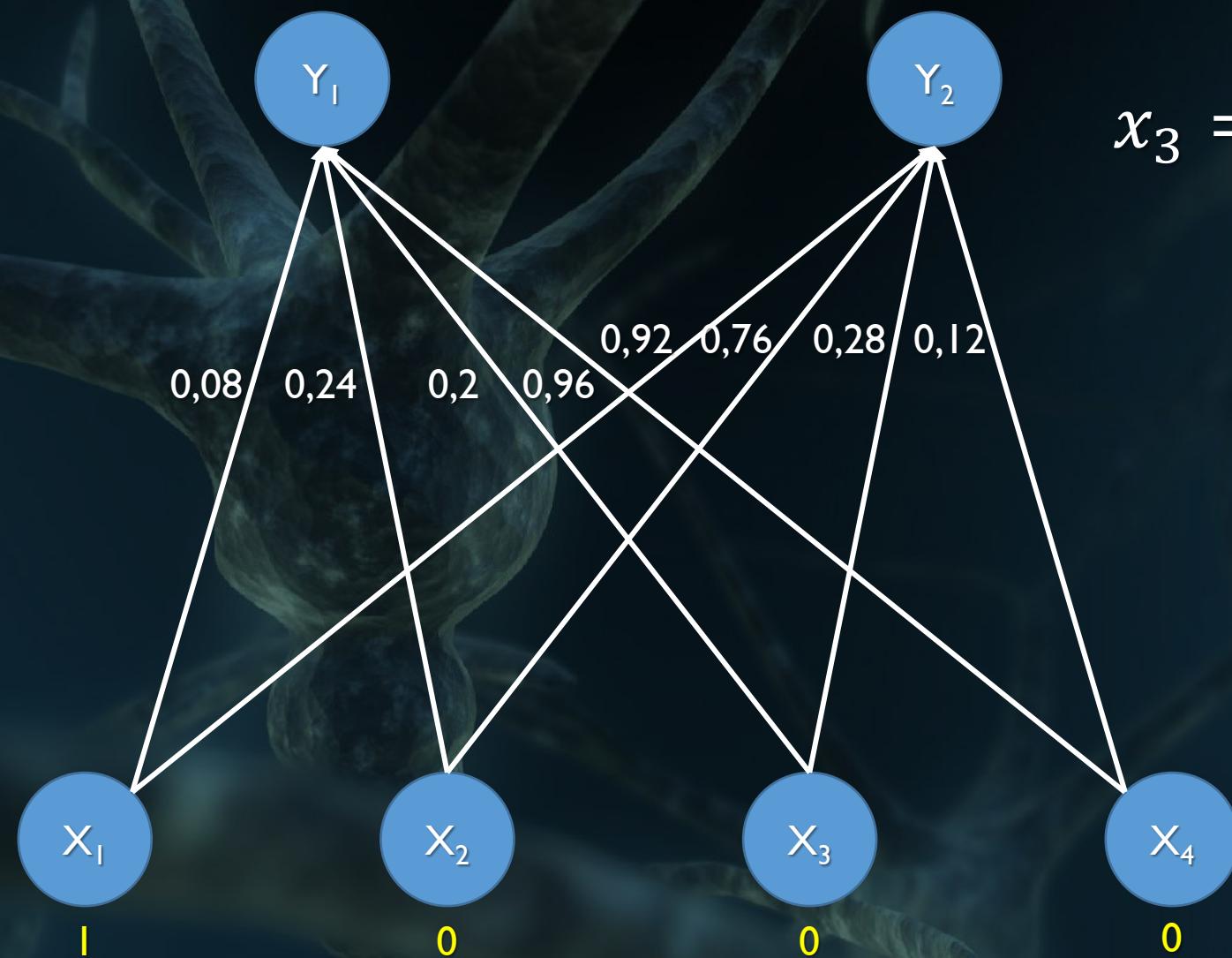
$$= 0,2$$

$$w'_{41} = 0,9 + 0,6[1 - 0,9]$$

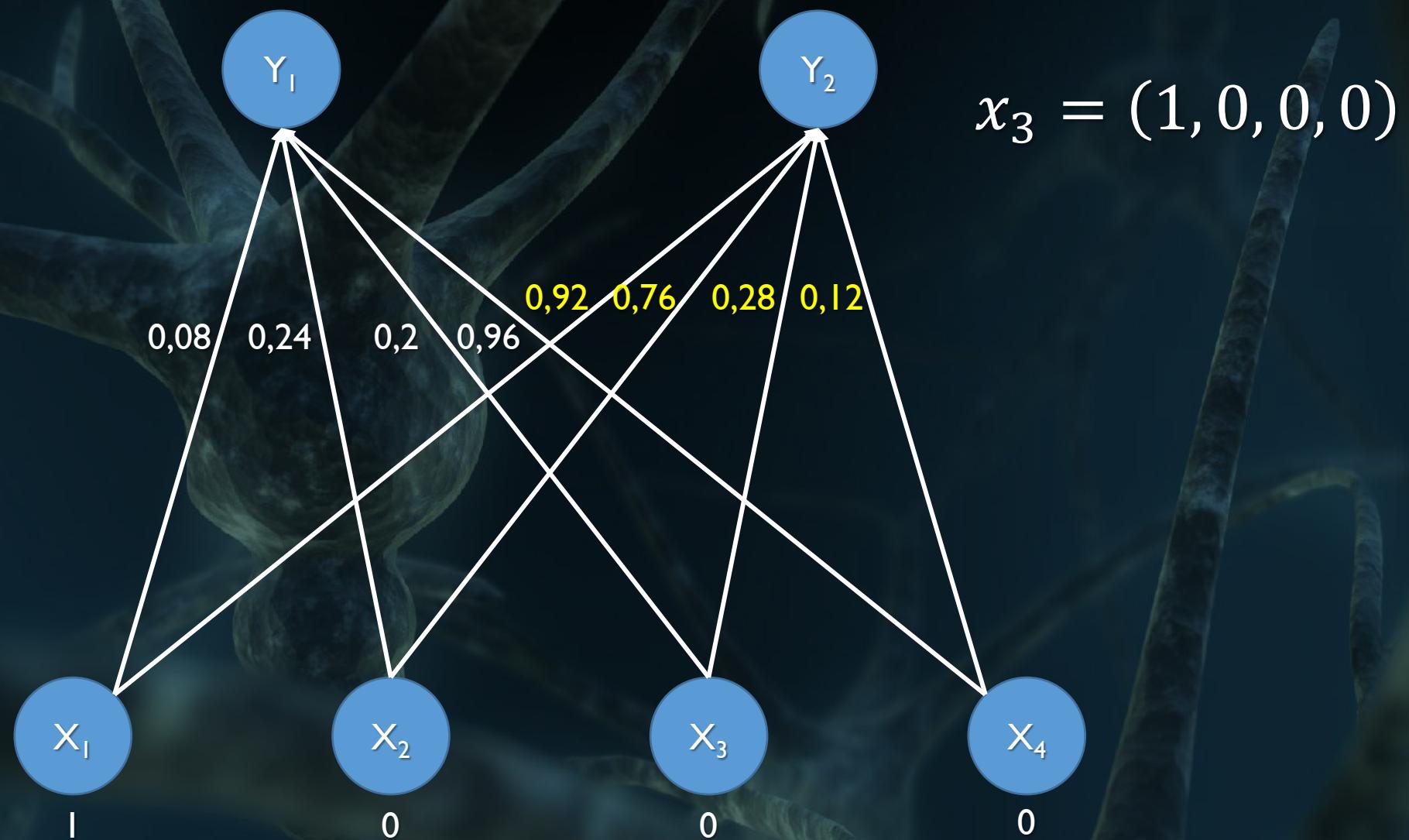
$$= 0,9 + 0,06$$

$$= 0,96$$



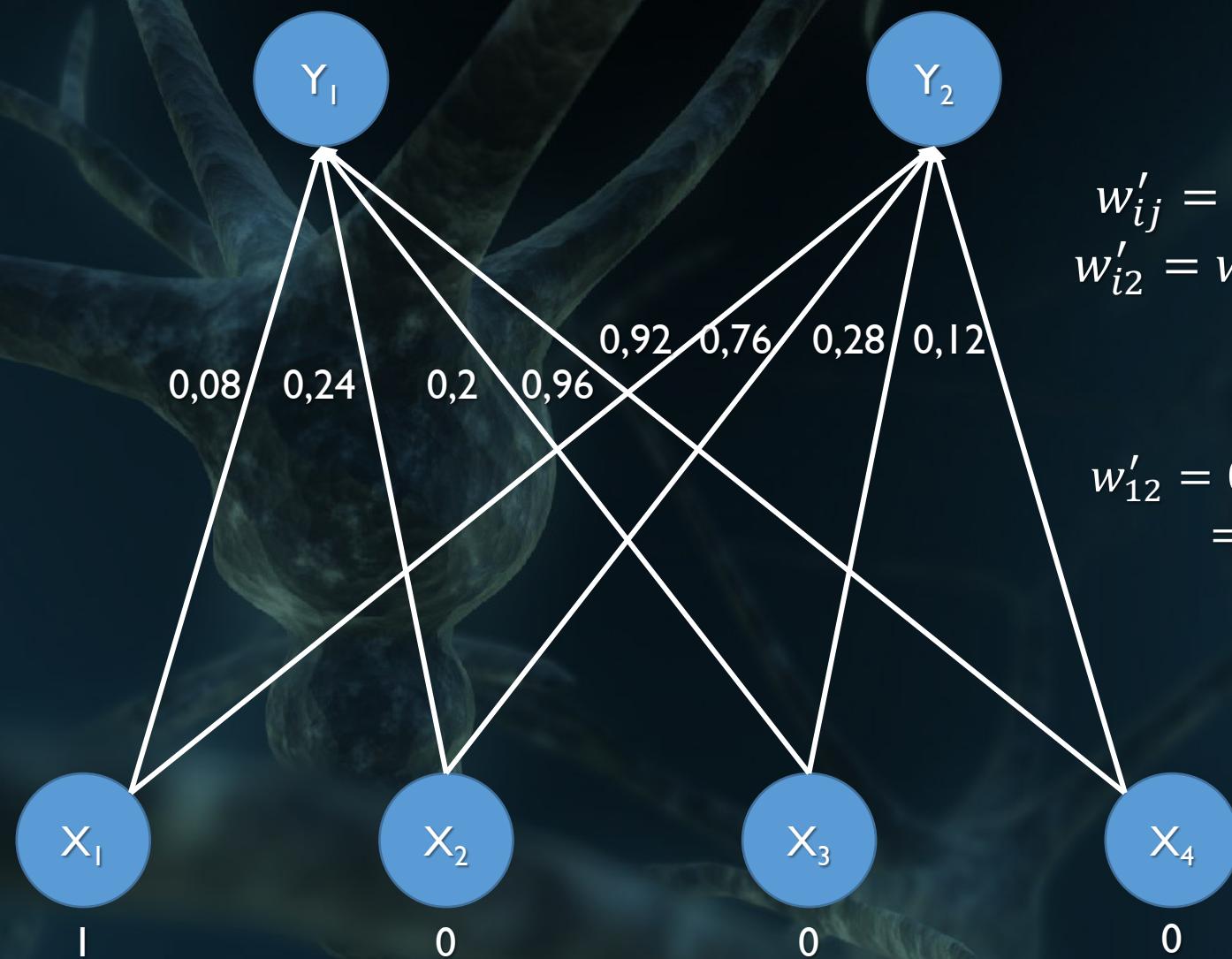


$$x_3 = (1, 0, 0, 0)$$



$$D(1) = (0,08 - 1)^2 + (0,24 - 0)^2 + (0,2 - 0)^2 + (0,96 - 1)^2 = 1,8656$$

$$D(2) = (0,92 - 1)^2 + (0,76 - 0)^2 + (0,28 - 0)^2 + (0,12 - 0)^2 = 0,6768$$



$$w'_{ij} = w_{ij} + \alpha[x_i - w_{ij}]$$

$$w'_{i2} = w_{i2} + 0,6[x_i - w_{i2}]$$

$$w'_{12} = 0,92 + 0,6[1 - 0,92]$$

$$= 0,92 + 0,048$$

$$= 0,968$$

$$w'_{22} = 0,76 + 0,6[0 - 0,76]$$

$$= 0,76 - 0,456$$

$$= 0,304$$

$$w'_{32} = 0,28 + 0,6[0 - 0,28]$$

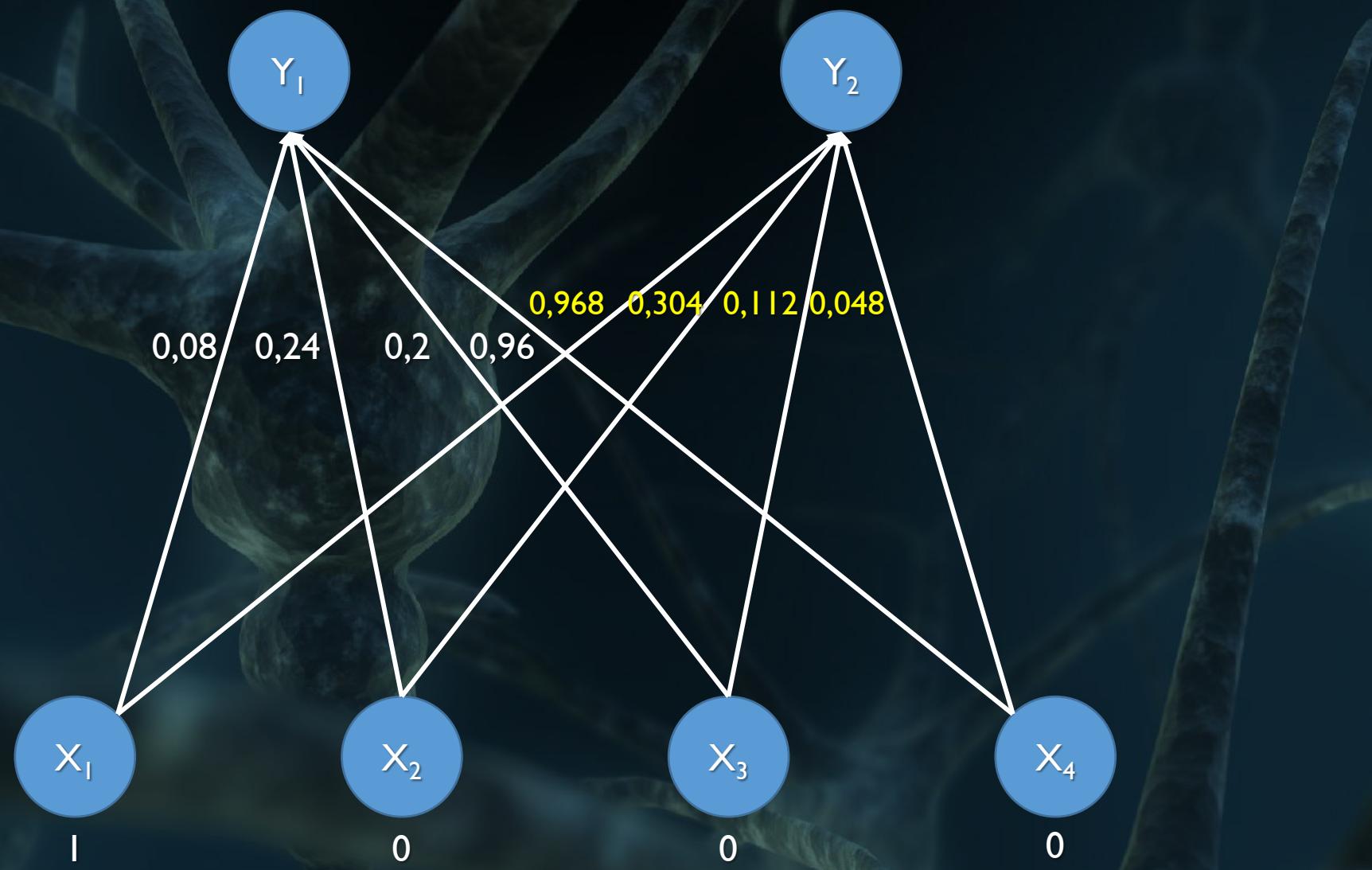
$$= 0,28 - 0,168$$

$$= 0,112$$

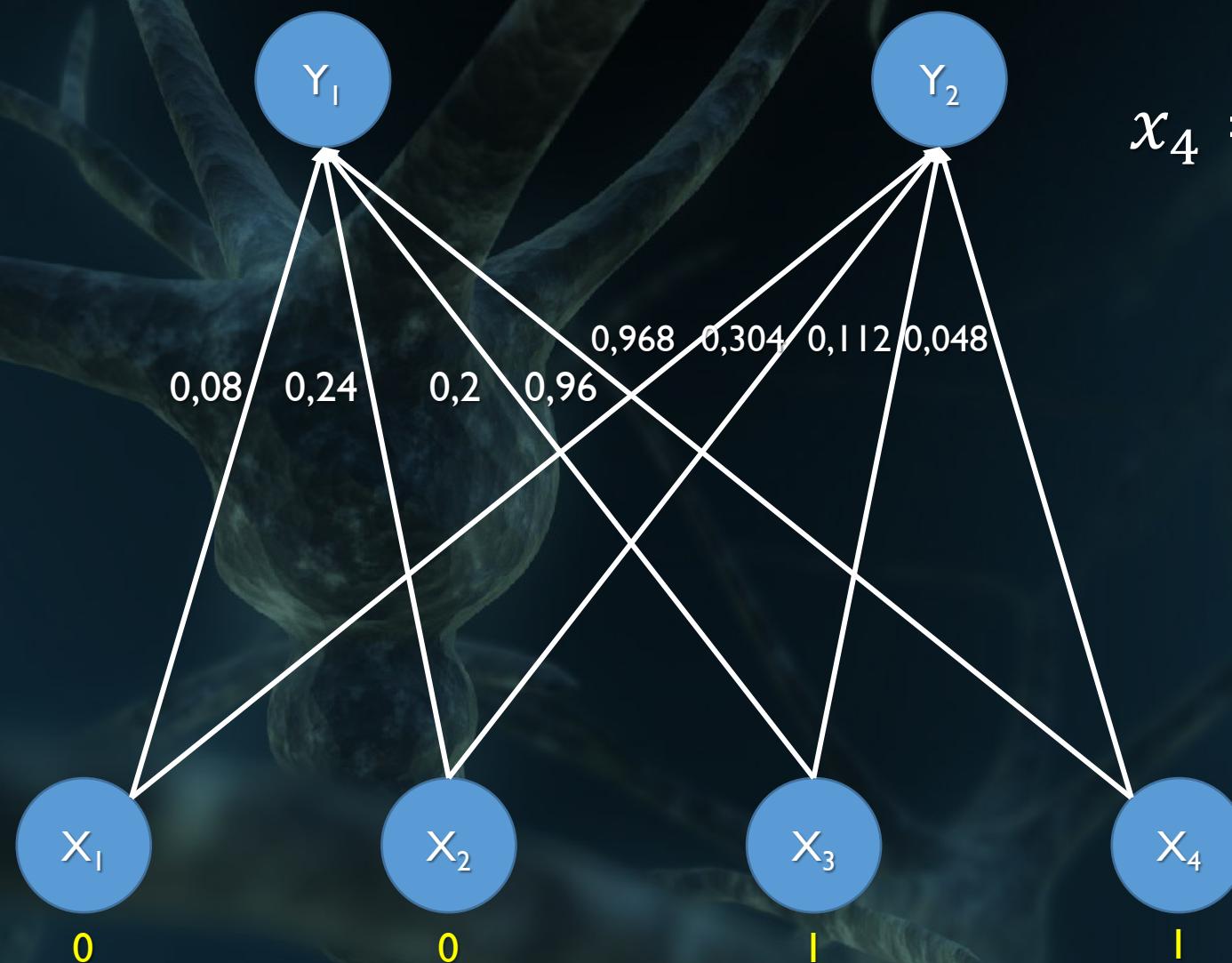
$$w'_{42} = 0,12 + 0,6[0 - 0,12]$$

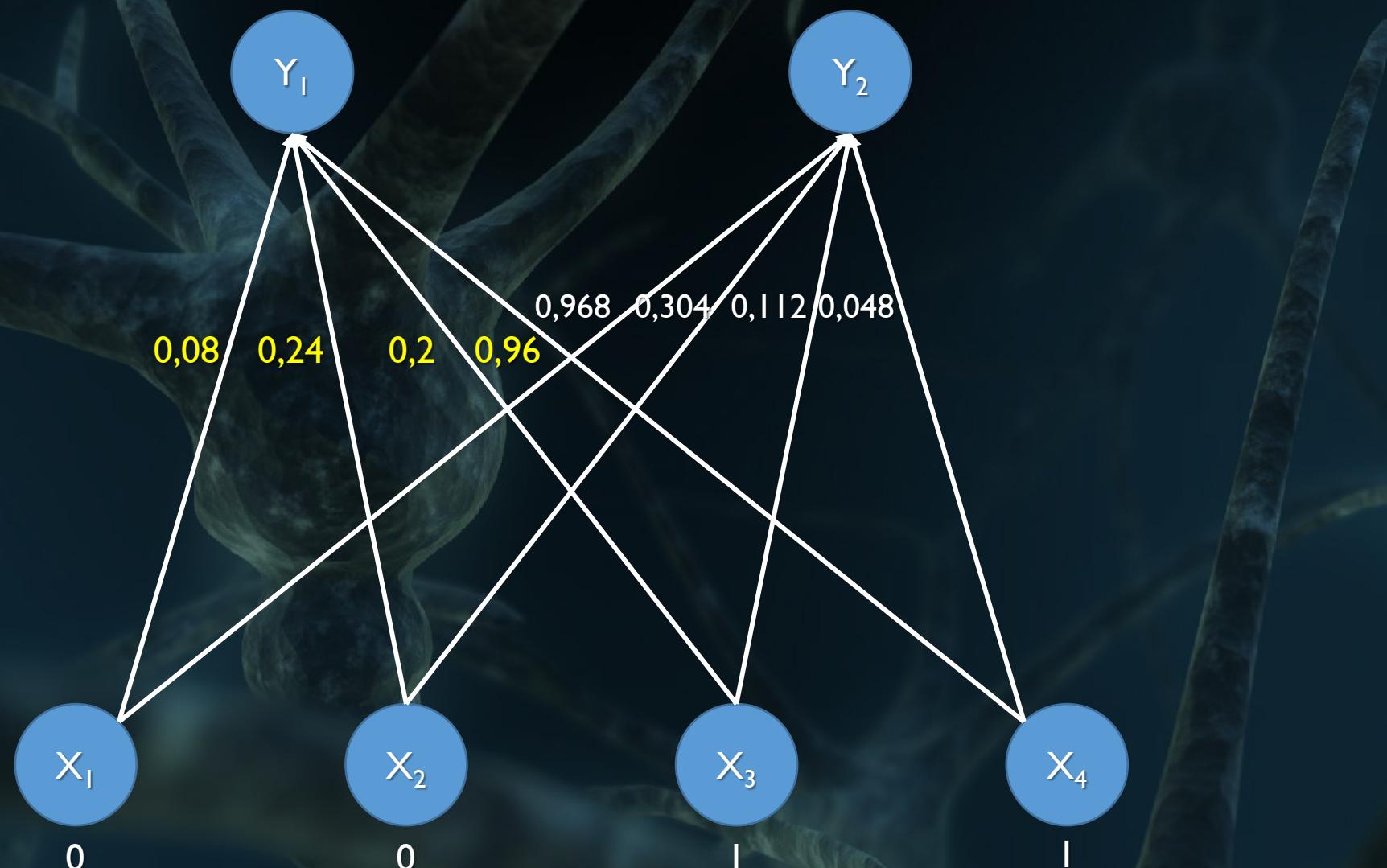
$$= 0,12 - 0,072$$

$$= 0,048$$



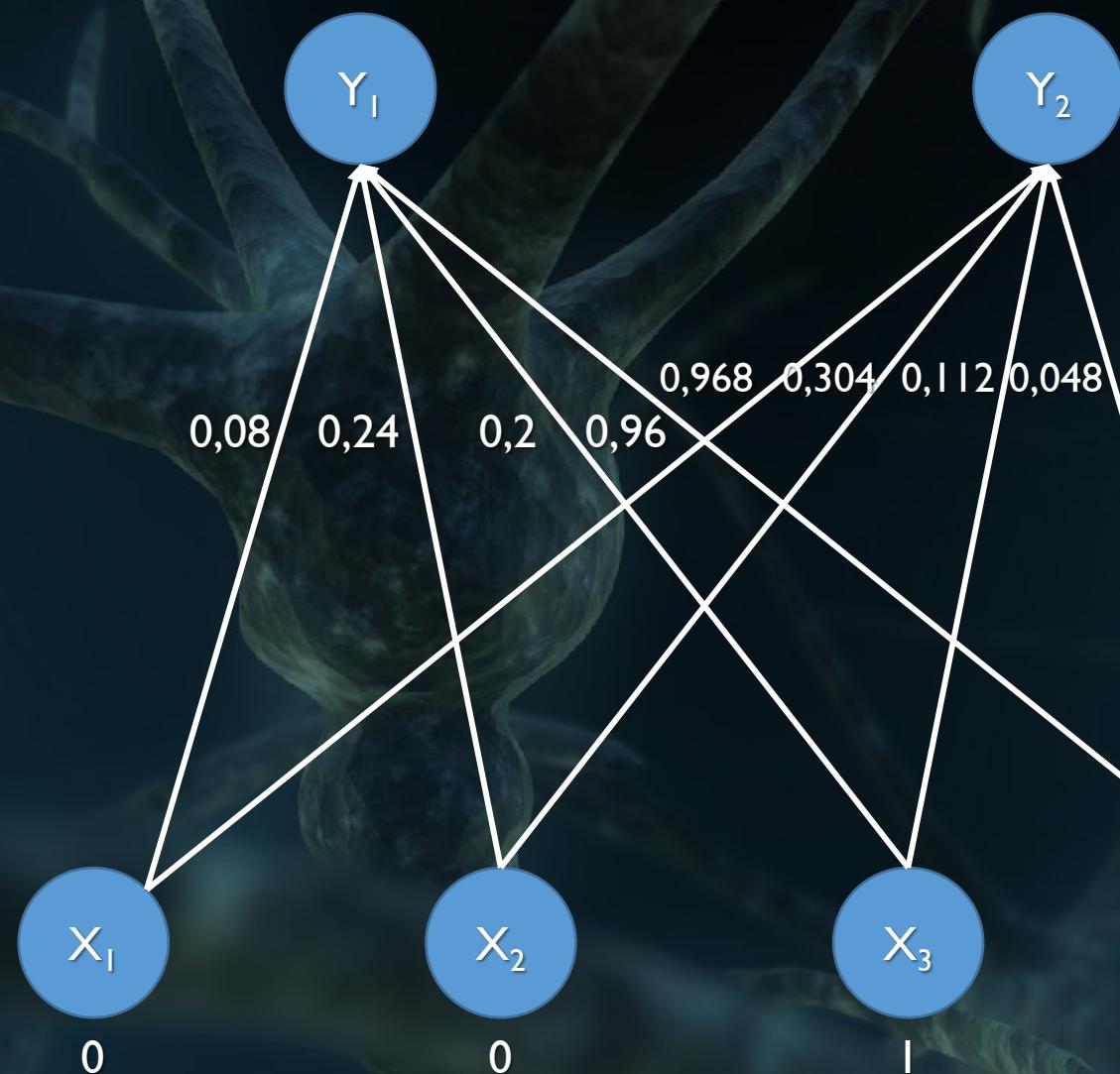
$$x_4 = (0, 0, 1, 1)$$





$$D(1) = (0,08 - 0)^2 + (0,24 - 0)^2 + (0,2 - 1)^2 + (0,96 - 1)^2 = 0,7056$$

$$D(2) = (0,968 - 0)^2 + (0,304 - 0)^2 + (0,112 - 1)^2 + (0,048 - 1)^2 = 2,724$$



$$w'_{ij} = w_{ij} + \alpha[x_i - w_{ij}]$$

$$w'_{i2} = w_{i2} + 0,6[x_i - w_{i2}]$$

$$w'_{12} = 0,08 + 0,6[0 - 0,08]$$

$$= 0,08 - 0,048$$

$$= 0,032$$

$$w'_{22} = 0,24 + 0,6[0 - 0,24]$$

$$= 0,24 - 0,144$$

$$= 0,096$$

$$w'_{32} = 0,2 + 0,6[1 - 0,2]$$

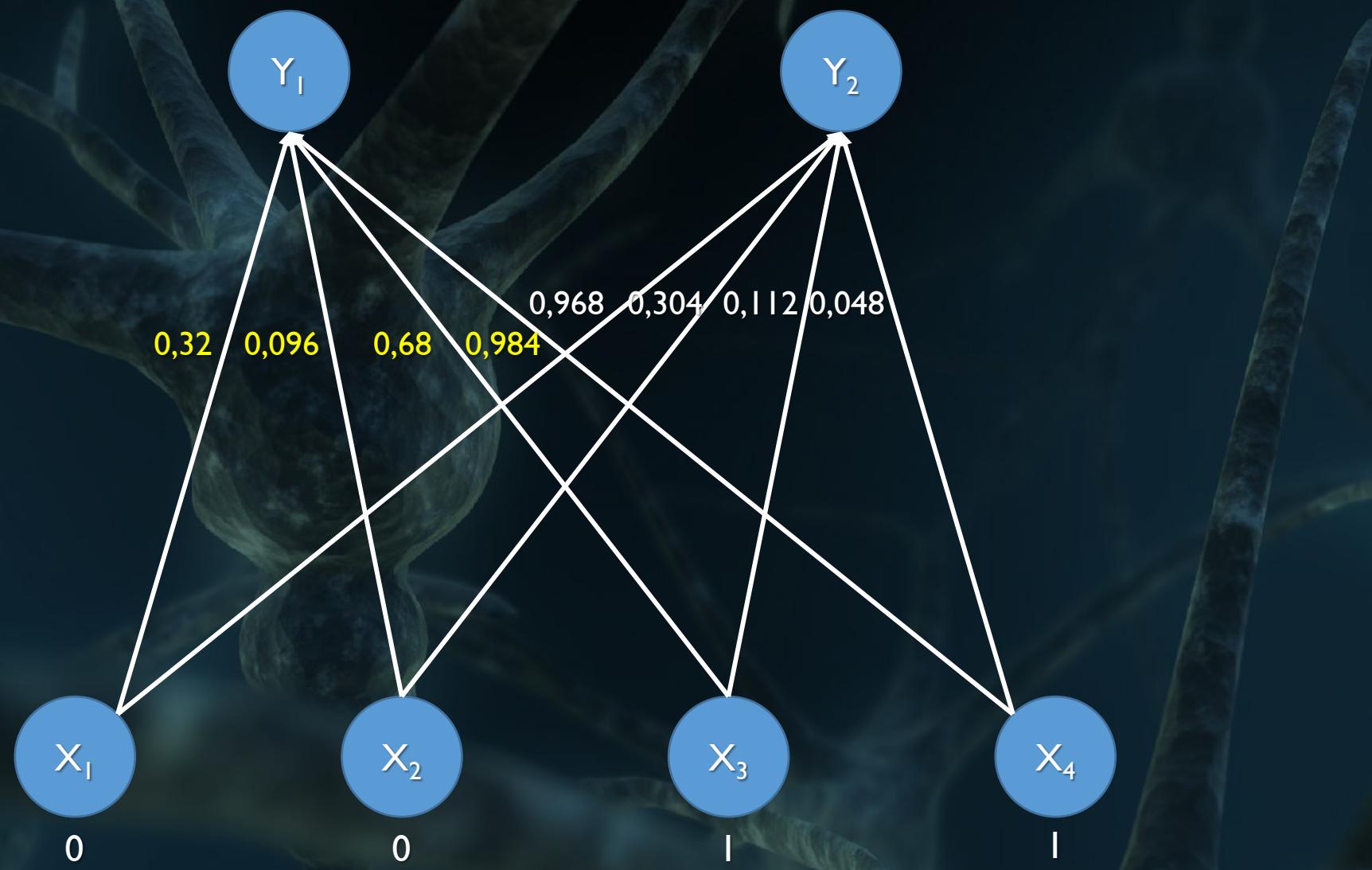
$$= 0,2 + 0,48$$

$$= 0,68$$

$$w'_{42} = 0,96 + 0,6[1 - 0,96]$$

$$= 0,96 + 0,024$$

$$= 0,984$$



Contoh

- *Epoch 1* selesai
- Ubah $\alpha = 0,5 \times 0,6 = 0,3$
- Lanjutkan ke *epoch* berikutnya

- Bobot iterasi 0=
$$\begin{bmatrix} 0,2 & 0,8 \\ 0,6 & 0,4 \\ 0,5 & 0,7 \\ 0,9 & 0,3 \end{bmatrix}$$

Contoh

- Bobot iterasi 1 =
$$\begin{bmatrix} 0,032 & 0,968 \\ 0,096 & 0,303 \\ 0,68 & 0,112 \\ 0,984 & 0,048 \end{bmatrix}$$
- Bobot iterasi 100 \approx
$$\begin{bmatrix} 0,0 & 1,0 \\ 0,0 & 0,5 \\ 0,5 & 0,0 \\ 1,0 & 0,0 \end{bmatrix}$$

Contoh

- Vektor-vektor:

1. $(1, 1, 0, 0)$
2. $(1, 0, 0, 0)$
3. $(0, 0, 0, 1)$
4. $(0, 0, 1, 1)$

- Bobot:
$$\begin{bmatrix} 0,0 & 1,0 \\ 0,0 & 0,5 \\ 0,5 & 0,0 \\ 1,0 & 0,0 \end{bmatrix}$$

- Nilai bobot berfungsi sebagai *centroid*

Implementasi

```
def som(X, a, b, max_ep, W):
    W = np.random.uniform(size=(W, len(X[0])))
    ep = 0

    while ep < max_ep:
        for x in X:
            d = [sum((w - x) ** 2) for w in W]
            min = np.argmin(d)
            W[min] += a * (x - W[min])

        a *= b
        ep += 1

    return W
```

Implementasi

```
x = np.array([[1, 1, 0, 0],  
              [0, 0, 0, 1],  
              [1, 0, 0, 0],  
              [0, 0, 1, 1]])  
  
a = .6  
b = .5  
w = som(x, a, b, 100, 2)  
  
print(w)
```



Alhamdulillah.