
K=3, 5, 7, 10, 15 // iterator = 10 // L1-NORM으로 거리를 구할때

```

In [23]: for K in [3,5,7,10,15]:
    energy = []
    iterator = 10
    # 랜덤으로 배정받을때 중복된 좌표를 얻을 경우가 생기므로 중복이면 다시 랜덤하게 배정함
    while True:
        label_matrix , x1, x2 = init(rows,cols,K)
        dic = get_locate_by_label(x1,x2,label_matrix)
        centroid = centroid_select(dic)
        if len(list(set([tuple(set(item)) for item in [*centroid.values() ]])) == K:
            break

    for x in range(0,iterator):
        label_matrix = locate2cluster(label_matrix,x1,x2,centroid,distance)

        energy.append(get_energyfunction_val(dic,centroid,K))

        dic = get_locate_by_label(x1,x2,label_matrix)
        centroid = centroid_select(dic)

    print("K : " + str(K))
    # 결과 그래프를 그려주는 부분
    cmap = plt.cm.get_cmap("hsv", K+1)
    plt.xlabel("X")
    plt.ylabel("Y")

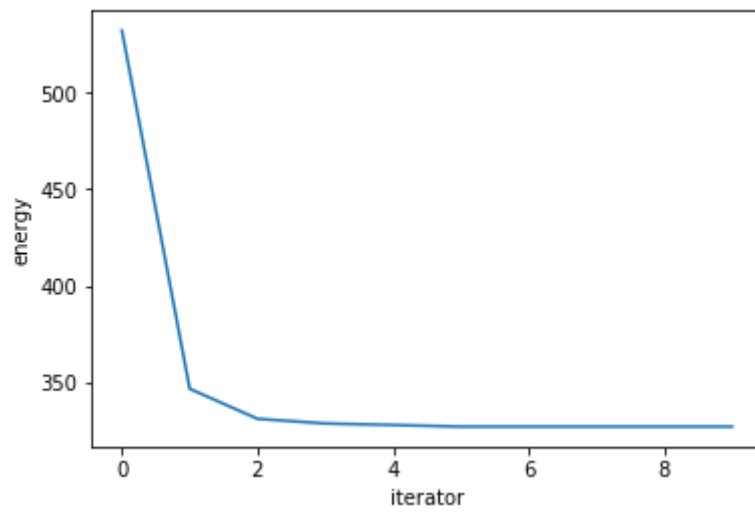
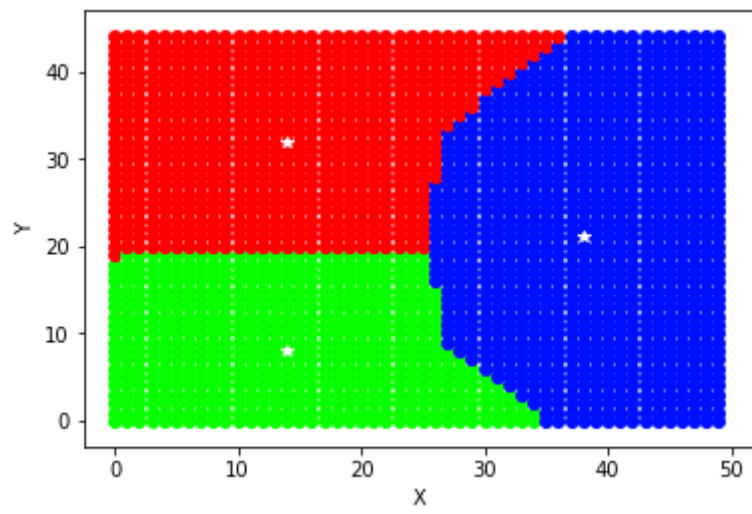
    for i in range(0, len(x1)):
        row = x1[i]
        col = x2[i]
        for i in range(0, K):
            if label_matrix[row][col] == i:
                plt.scatter(row, col,c=cmap(i))

    # 중심 좌표를 *으로 표현해주는 부분
    for x in range(0,len(centroid)):
        plt.plot(centroid[x][0], centroid[x][1], "w*")
    plt.show()

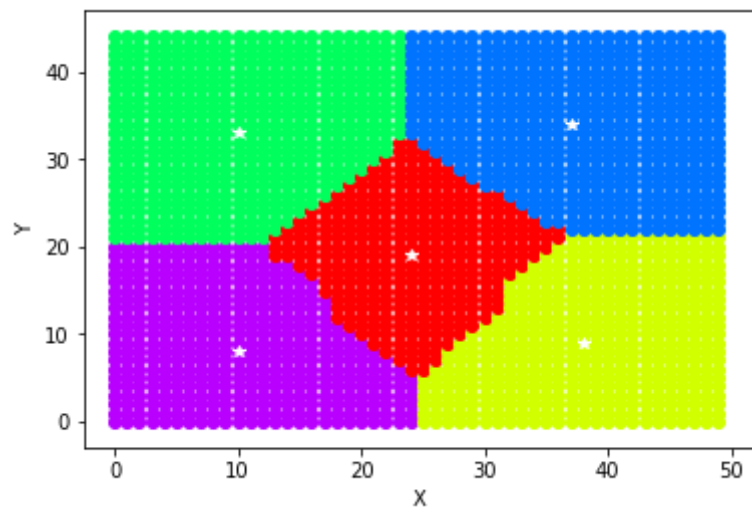
    show_energyfunction(iterator,energy)

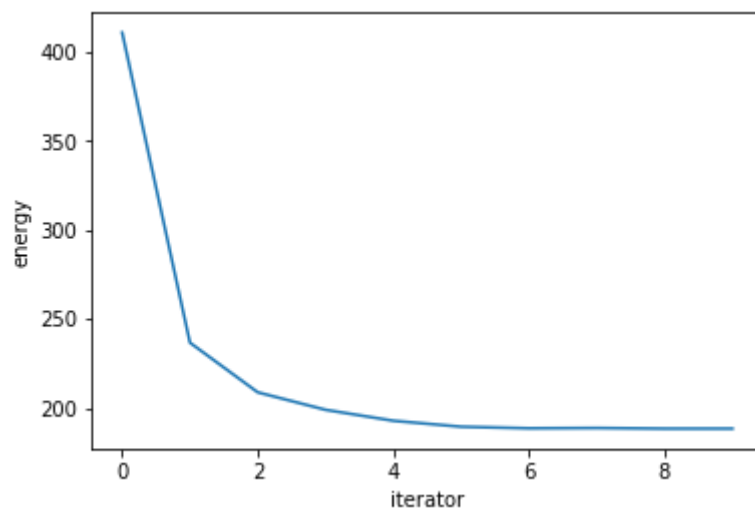
```

K : 3

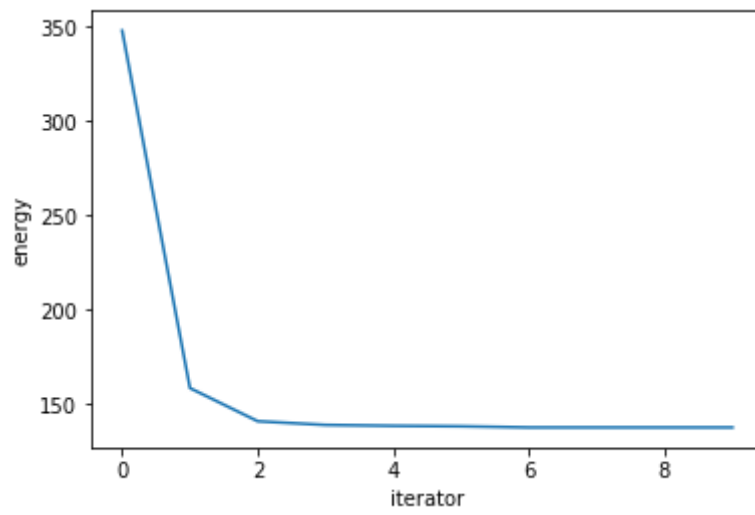
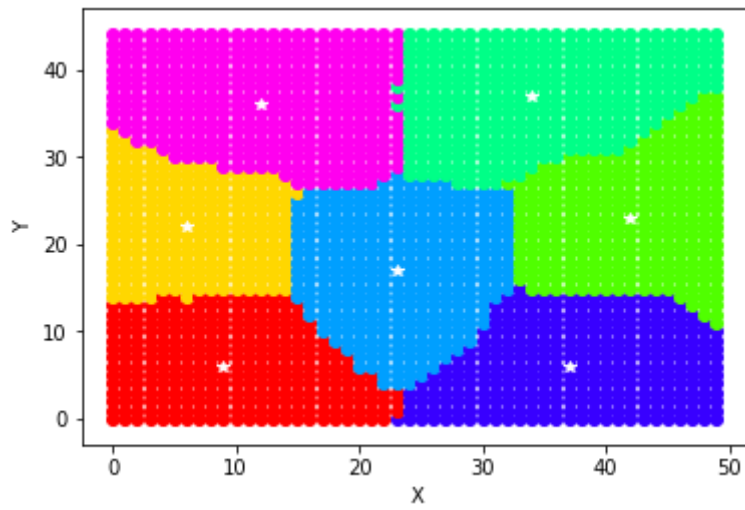


K : 5

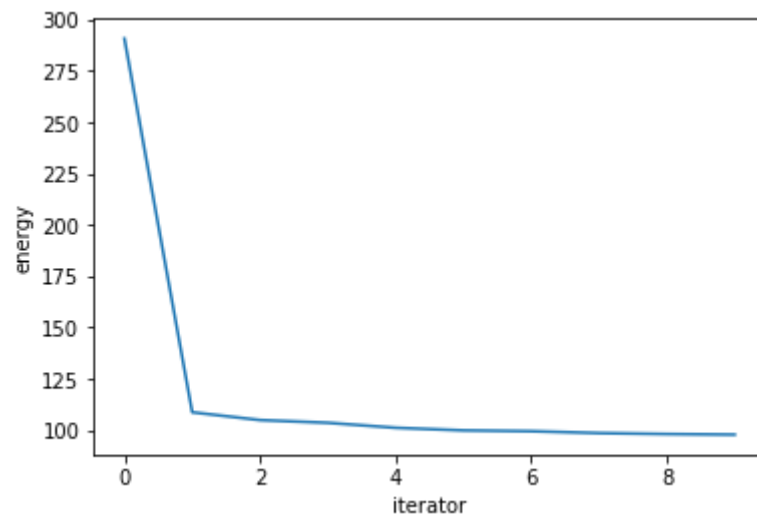
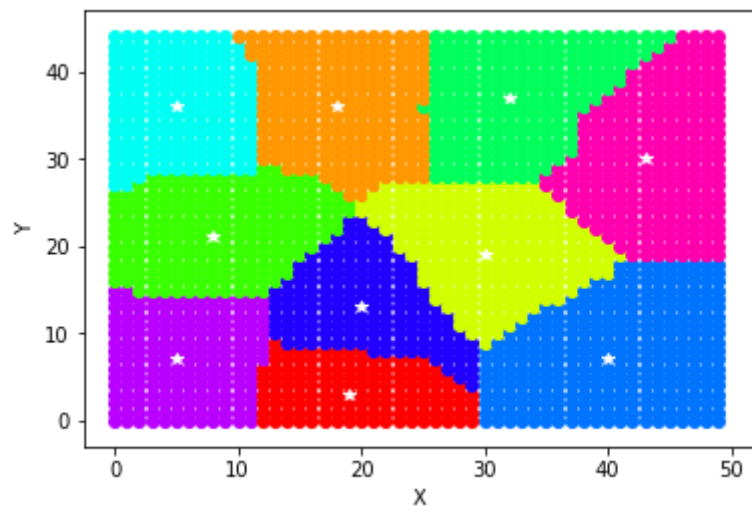




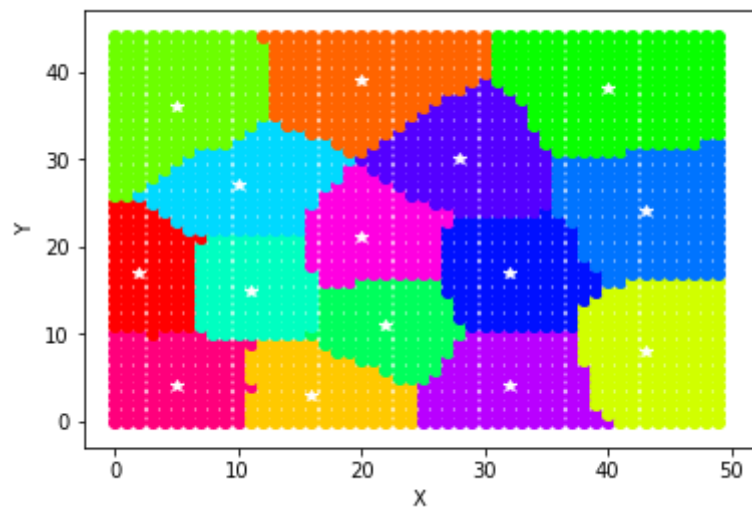
K : 7

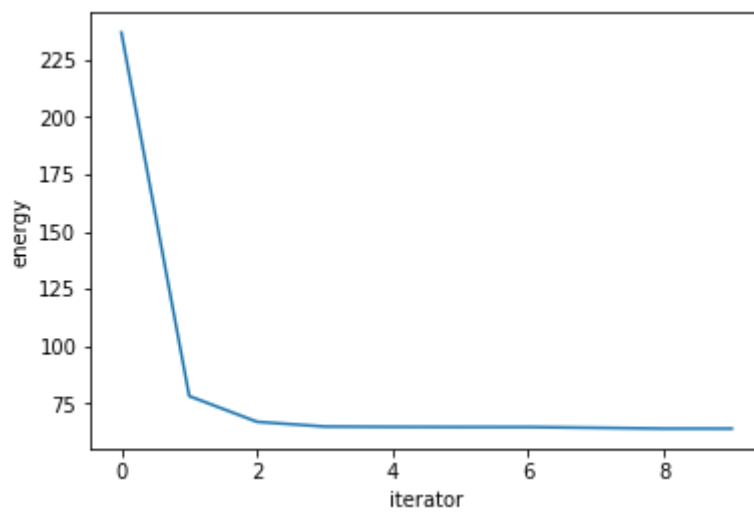


K : 10



K : 15





K=3, 5, 7, 10, 15 // iterator = 10 // L2-NORM으로 거리를 구할때

```

In [24]: for K in [3,5,7,10,15]:
    energy = []
    iterator = 10
    # 랜덤으로 배정받을때 중복된 좌표를 얻을 경우가 생기므로 중복이면 다시 랜덤하게 배정함
    while True:
        label_matrix , x1, x2 = init(rows,cols,K)
        dic = get_locate_by_label(x1,x2,label_matrix)
        centroid = centroid_select(dic)
        if len(list(set([tuple(set(item)) for item in [*centroid.values() ]])) == K:
            break

    for x in range(0,iterator):
        label_matrix = locate2cluster(label_matrix,x1,x2,centroid,2)
        energy.append(get_energyfunction_val(dic,centroid,K))

        dic = get_locate_by_label(x1,x2,label_matrix)
        centroid = centroid_select(dic)

    print("K : " + str(K))
    # 결과 그래프를 그려주는 부분
    cmap = plt.cm.get_cmap("hsv", K+1)
    plt.xlabel("X")
    plt.ylabel("Y")

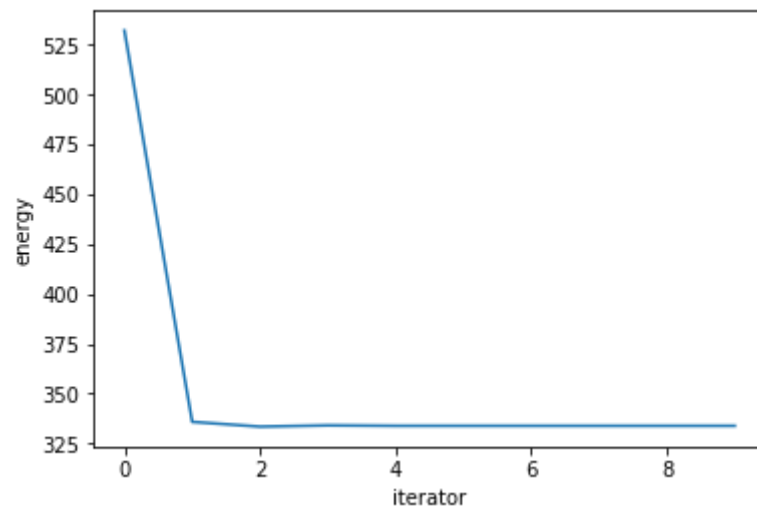
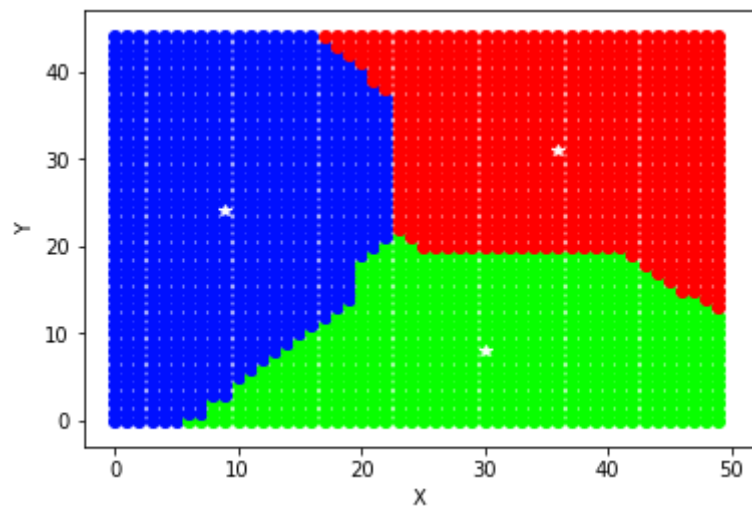
    for i in range(0, len(x1)):
        row = x1[i]
        col = x2[i]
        for i in range(0, K):
            if label_matrix[row][col] == i:
                plt.scatter(row, col,c=cmap(i))

    # 중심 좌표를 *으로 표현해주는 부분
    for x in range(0,len(centroid)):
        plt.plot(centroid[x][0], centroid[x][1],"w*")
    plt.show()

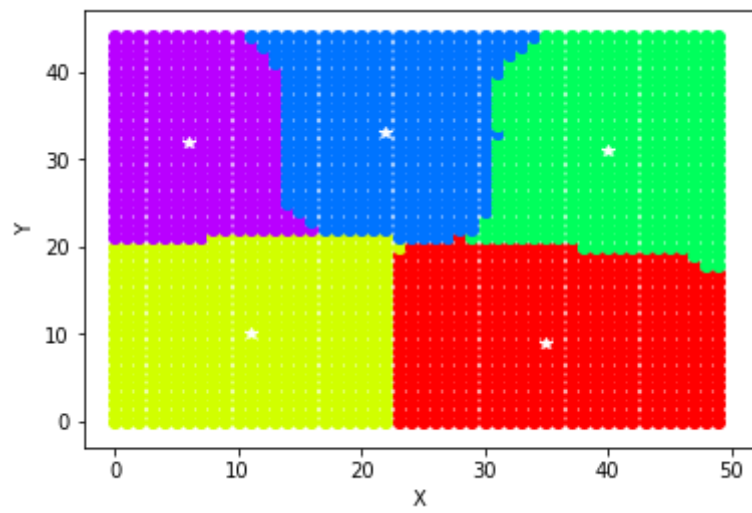
    show_energyfunction(iterator,energy)

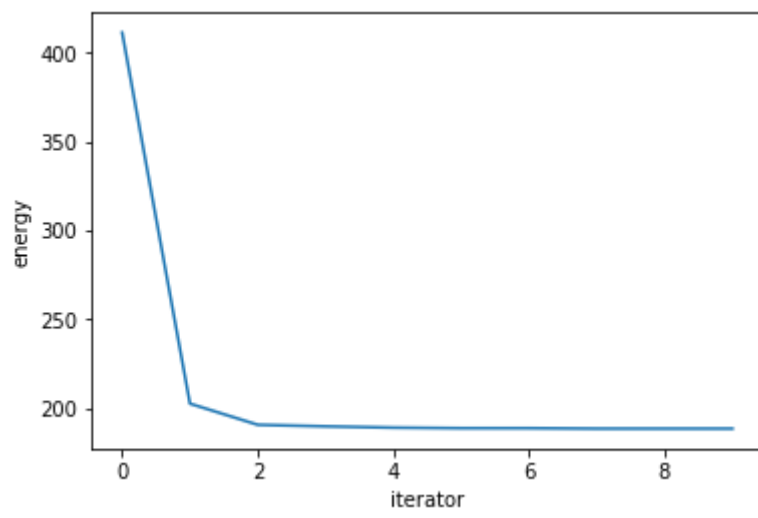
```

K : 3

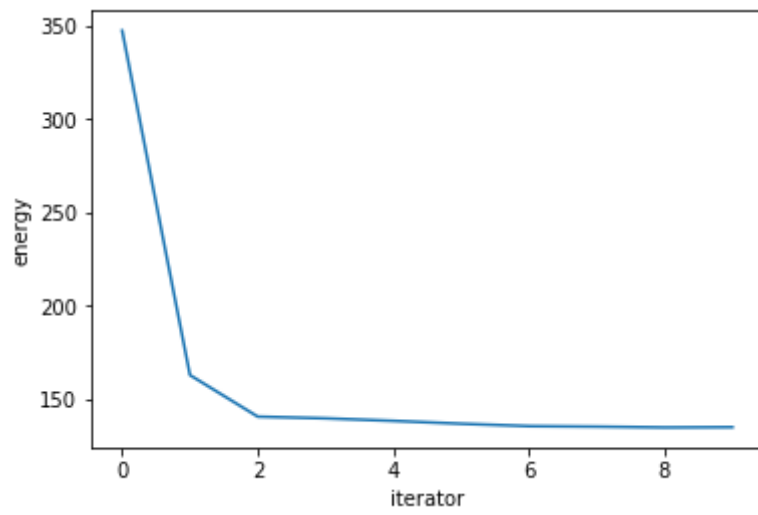
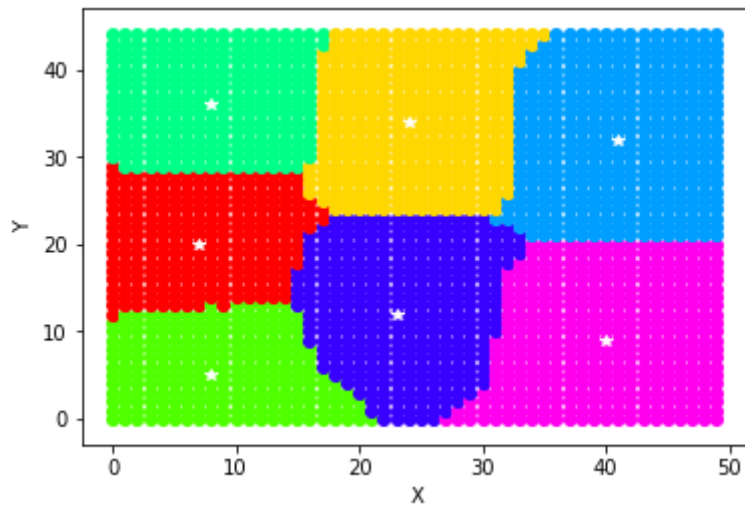


K : 5

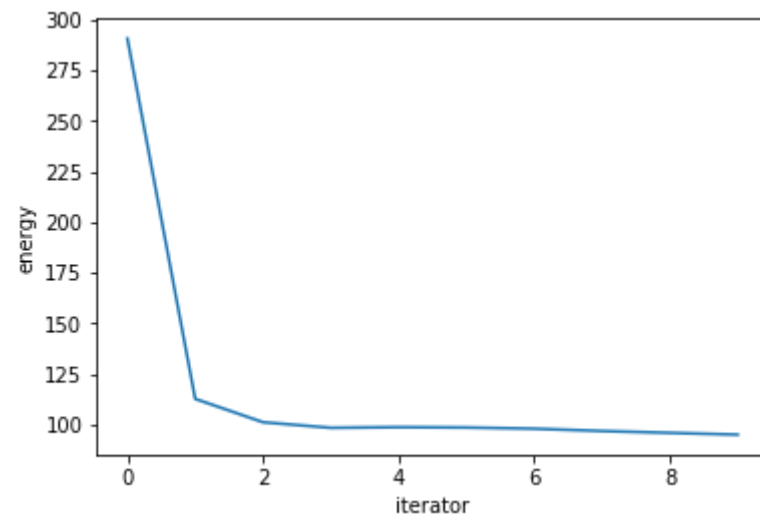
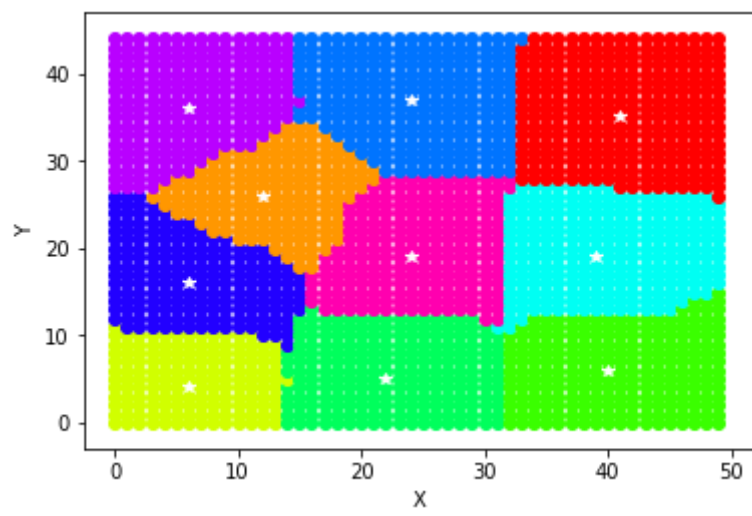




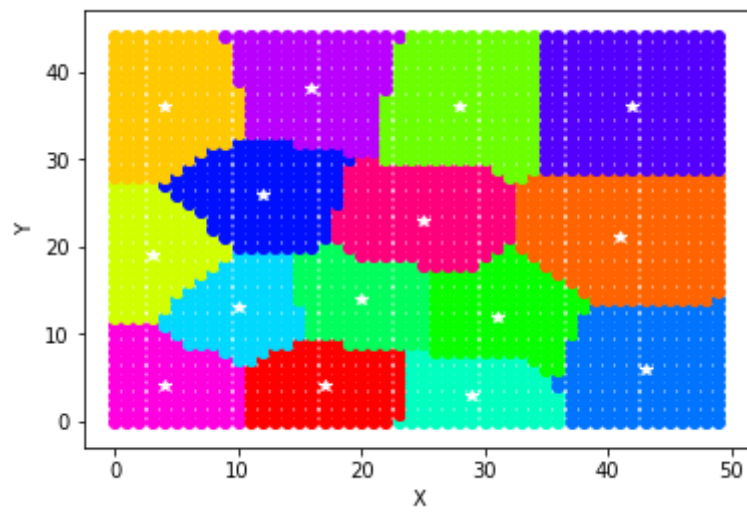
K : 7

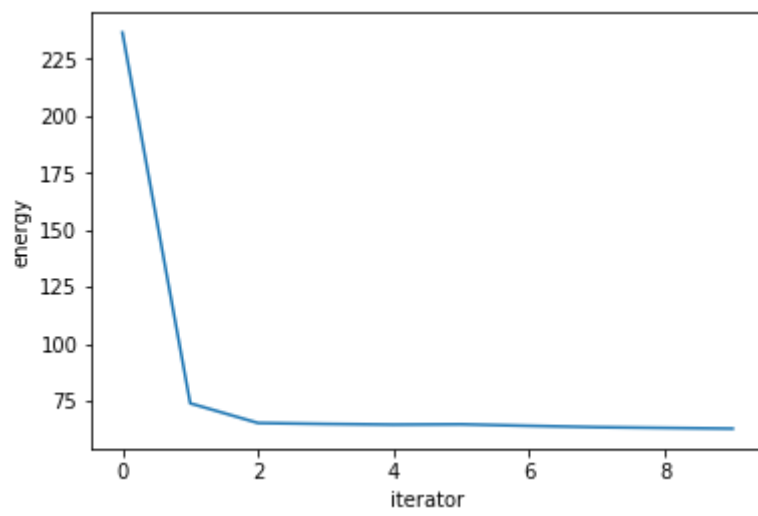


K : 10



K : 15





In []: