In [57]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import random
from math import sqrt

df = pd.read_excel('clusters.xlsx', index_col=0)
#number of data points predicted as class 2, while the correct class is 4
error_24= len(df[(df["Class"] == 4) & (df["Predicted_Class"] == 2)])
#number of data points predicted as class 4, while the correct class is 2
error_42 = len(df[(df["Class"] == 2) & (df["Predicted_Class"] == 4)])
#number of data points with predicted class not equal to correct class
error_all = error_24 + error_42
#number of data points
class_all = len(df)
#total error
error_T = (error_all   / class_all) * 100
error_T_rounded = round(error_T,2)

if(error_T_rounded >= 50):
    df["Predicted_Class"].replace({2: 4, 4: 2}, inplace=True)
    error_24= len(df[(df["Class"] == 4) & (df["Predicted_Class"] == 2)])
    error_42 = len(df[(df["Class"] == 2) & (df["Predicted_Class"] == 4)])
    error_all = error_24 + error_42
    error_T = (error_all   / class_all) * 100
    error_T_rounded = round(error_T,2)

#number of data points with predicted class equal to 2
pclass_2 = len(df.loc[df['Predicted_Class'] == 2])
#number of data points with predicted class equal to 4
pclass_4 = len(df.loc[df['Predicted_Class'] == 4])

#error rate for the benign cells
error_B = 0
#error rate for the malign cells
error_M = 0
#total error rate
error_T = 0

error_B = (error_24  / pclass_2) * 100
error_B_rounded = round(error_B,2)
error_M = (error_42   / pclass_4) * 100
error_M_rounded = round(error_M,2)




print("Total errors:\t\t\t\t" + str(error_T_rounded) + "%")
print("Clusters are swapped!")
print("Swapping Predicted_Class\n")


print("Data points in Predicted Class 2:\t" + str(pclass_2))
print("Data points in Predicted Class 4:\t" + str(pclass_4))

print("\nError data points, Predicted Class 2:\n")
print(df.loc[df['Predicted_Class'] == 2][["Scn", "Class", "Predicted_Class"]])
print("\nError data points, Predicted Class 4:\n")
print(df.loc[df['Predicted_Class'] == 4][["Scn", "Class", "Predicted_Class"]])

print("\nNumber of all data points:\t"+ str(class_all))
print("\nNumber of error data points:\t" + str(error_all))
print("\nError rate for class 2:\t\t" + str(error_B_rounded) + "%")
print("Error rate for class 4:\t\t" + str(error_M_rounded) + "%")
print("Total error rate:\t\t" + str(error_T_rounded) + "%")
```

```
Total errors:    4.15%
```

```
Clusters are swapped!
Swapping Predicted_Class

Data points in Predicted Class 2: 465
Data points in Predicted Class 4: 234

Error data points, Predicted Class 2:

         Scn  Class  Predicted_Class
0    1000025      2                2
2    1015425      2                2
4    1017023      2                2
6    1018099      2                2
7    1018561      2                2
..       ...    ...              ...
690   654546      2                2
692   714039      2                2
693   763235      2                2
694   776715      2                2
695   841769      2                2

[465 rows x 3 columns]

Error data points, Predicted Class 4:

         Scn  Class  Predicted_Class
1    1002945      2                4
3    1016277      2                4
5    1017122      4                4
14   1044572      4                4
18   1050670      4                4
..       ...    ...              ...
681  1371026      4                4
691   695091      4                4
696   888820      4                4
697   897471      4                4
698   897471      4                4

[234 rows x 3 columns]

Number of all data points: 699

Number of error data points: 29

Error rate for class 2:  3.87%
Error rate for class 4:  4.7%
Total error rate:  4.15%
```