Name – Sourabh M Zambre
Class – TY B.Tech
Date – 20/02/2019
Roll no - BML12_31
Assignment 2 :

# Implement program of K-Means Clustering(Java/Python) -Use any data set for clustering

Code -

```
# -*- coding: utf-8 -*-
"""
Created on Sun Wed 20 16:06:34 2019

@author: Hp
"""

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
#from sklearn import datasets

df = pd.read_csv('Age_wgt.csv')
print(df)

K = 3


rand_cent = df.sample(n=K)
print(rand_cent)
print(type(rand_cent))
print()
centroids = rand_cent.values.tolist()
c1 = centroids[0]
c2 = centroids[1]
c3 = centroids[2]
allpoints = df.values.tolist()
print(c1,c2,c3)

cl1 = []
cl2 = []
cl3 = []
```

```
C1 = []
C2 = []
C3 = []

while(True):

    for i in range(len(allpoints)):

        a = c1
        b = c2
        c = c3

        d1 = np.sqrt((allpoints[i][0] - c1[0]) ** 2 + (allpoints[i][1] - c1[1]) ** 2 )
        d2 = np.sqrt((allpoints[i][0] - c2[0]) ** 2 + (allpoints[i][1] - c2[1]) ** 2 )
        d3 = np.sqrt((allpoints[i][0] - c3[0]) ** 2 + (allpoints[i][1] - c3[1]) ** 2)
        smd = min(min(d1,d2),d3)


        if (smd==d1):
            C1.append(allpoints[i])
        if (smd==d2):
            C2.append(allpoints[i])
        if (smd==d3):
            C3.append(allpoints[i])

        if(len(C1)!=0) :
            cc1 = [sum(x) for x in zip(*C1)]
            c1[0] = (c1[0]+cc1[0])/(len(C1)+1)
            c1[1] = (c1[1]+cc1[1])/(len(C1)+1)

        if(len(C2)!=0) :
            cc2 = [sum(x) for x in zip(*C2)]
            c2[0] = (c2[0]+cc2[0])/(len(C2)+1)
            c2[1] = (c2[1]+cc2[1])/(len(C2)+1)

        if(len(C3)!=0) :
            cc3 = [sum(x) for x in zip(*C3)]
            c3[0] = (c3[0]+cc3[0])/(len(C3)+1)
            c3[1] = (c3[1]+cc3[1])/(len(C3)+1)


    if (a==c1 and b==c2 and c==c3):
        break
print("\n\nCluster 1","\n")
```

```
print(C1,"\n\n")
print("Cluster 2","\n\n")
print(C2,"\n\n")
print("Cluster 3","\n\n")
print(C3,"\n\n")


col = ['red','blue','yellow']
markers = ['o','v','s']
for i in range(len(C1)):
    plt.plot(C1[i][0],C1[i][1],color=col[0],marker=markers[0])

for i in range(len(C2)):
    plt.plot(C2[i][0],C2[i][1],color=col[1],marker=markers[1])

for i in range(len(C3)):
    plt.plot(C3[i][0],C3[i][1],color=col[2],marker=markers[2])

plt.plot(c1[0],c1[1],color='green',marker='D',markersize=7)
plt.plot(c2[0],c2[1],color='green',marker='D',markersize=7)
plt.plot(c3[0],c3[1],color='green',marker='D',markersize=7)


plt.show()
```

# Output -

```
sourabh@hp-pavilion-notebook: ~/Downloads
File  Edit  View  Search  Terminal  Help
sourabh@hp-pavilion-notebook:~/Downloads$ python3 K_Means_imp.py
    Age in Yrs.  Weight in Kgs.
48        23.73             60
32        23.92             64
74        58.43             60

[23.73, 60.0] [23.92, 64.0] [58.43, 60.0]


Cluster 1

[[36.36, 60.0], [39.67, 51.0], [40.31, 58.0], [34.86, 58.0], [29.73, 60.0], [42.5, 80.0], [21.1, 87.0], [47.23, 68.0], [49.85, 73.0], [37.27, 59.0], [48.09, 77.0], [24.02, 81.0], [43.63, 90.0], [38.65, 80.0], [28.76, 75.0], [34.75, 70.0], [21.93, 82.0], [32.75, 70.0], [25.93, 74.0], [41.19, 60.0], [25.98, 68.0], [32.77, 87.0], [50.7, 87.0], [39.43, 82.0], [22.82, 89.0], [32.61, 77.0]]


Cluster 2


[[22.14, 50.0], [22.21, 61.0], [24.66, 59.0], [22.64, 56.0], [23.77, 56.0], [28.02, 60.0], [27.59, 60.0], [36.24, 49.0], [36.14, 52.0], [23.92, 64.0], [26.24, 62.0], [33.89, 57.0], [35.02, 40.0], [32.58, 50.0], [33.73, 52.0], [27.73, 53.0], [23.73, 60.0], [21.3, 47.0], [28.23, 41.0], [26.25, 43.0], [24.61, 44.0], [22.72, 55.0], [32.08, 52.0], [30.9, 58.0], [34.87, 47.0], [25.1, 56.0], [31.29, 61.0], [25.96, 40.0], [32.29, 48.0], [26.35, 54.0], [33.55, 57.0], [24.01, 65.0], [35.91, 43.0], [37.62, 60.0], [37.84, 59.0], [39.77, 54.0], [37.57, 56.0], [29.6, 57.0], [38.38, 55.0]]


Cluster 3


[[47.02, 60.0], [54.15, 68.0], [59.12, 40.0], [58.18, 42.0], [45.45, 55.0], [59.47, 47.0], [50.97, 42.0], [58.08, 56.0], [44.72, 56.0], [59.42, 53.0], [42.39, 41.0], [53.68, 41.0], [49.69, 54.0], [48.49, 56.0], [55.97, 44.0], [55.14, 45.0], [52.99, 56.0], [56.38, 40.0], [58.15, 51.0], [56.66, 64.0], [59.27, 48.0], [56.27, 45.0], [56.53, 42.0], [47.78, 54.0], [57.37, 62.0], [52.51, 52.0], [43.59, 57.0], [54.16, 60.0], [58.43, 60.0], [50.05, 50.0], [48.79, 48.0], [42.79, 55.0], [42.21, 53.0], [56.0, 55.0], [52.66, 60.0]]
```