SEEM2460 Introduction to Data Science

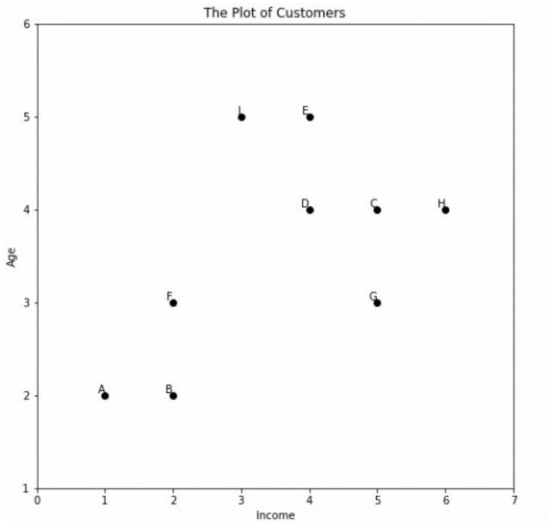
Assignment 3

Assignment 3 is due at 23:59 15 April, 2020. You need to submit a PDF version of your hand-calculation process and your codes (Assignment3-K-means-Clustering.ipynb) to the Blackboard.

Question 1. (K-means clustering)

Suppose we have 9 customers and have two attributes (standardized income and standardized age).

Customer	Income	Age
A	1	2
B	2	2
C	5	4
D	4	4
E	4	5
F	2	3
G	5	3
H	6	4
I	3	5



Our goal is to group these customers into K=3 groups.

(1) [15pts] Please use K-means algorithm and Euclidean distance to cluster the 9 customers into 3 clusters **by hand calculation**. The initial centroids **MUST** be A, D, and H. Please report the detailed distance matrix and centroids at each step.

(2) [15pts] Please first read and run the supplemented Python codes (K-means-Clustering-demonstration.ipynb) which demonstrating the example introduced in week-8 lecture. Then, write your codes in the Jupyter notebook Assignment3-K-means-Clustering.ipynb to solve this problem. Please only write codes in the fenced parts by # like below:

```
# Step-1: Assign 3 initail centroids.
#########################################################################################################
#################################### Please write codes here ! ##########################################
#########################################################################################################
centroids = {
    # Please specify three centroids below
}
#########################################################################################################
#########################################################################################################
```

```
# Step-2: Continue until all assigned categories don't change any more
#########################################################################################################
#################################### Please write codes here ! ##########################################
#########################################################################################################
df = assignment(df, centroids)
while True:
    closest_centroids = df['closest'].copy(deep=True)
    # Please determine the order of update() and assignment() below, Hint: 2 lines of codes


    if closest_centroids.equals(df['closest']):
        break
#########################################################################################################
#########################################################################################################
```

-- End --