CS5228-KDDM, 2024/25-2, Coursework 1

Introduction

- This coursework comprises three parts. Parts 1 and 2 involve Python programming for data mining, and Part 3 contains four MCQs.
- Total CA marks of this coursework is 12. Details of marks/parts are below.
- A Canvas quiz will be open for your coursework submission.
- For Python programming parts, I urge you to complete a Jupyter notebook and submit it. cw1-template.ipynb is the template for your answer. You have to run your codes and get sure that answers are available in the notebook before submission.
- Regarding MCQs, there is one and only one correct answer for each question. So select the best option. There is no penalty for wrong answers.
- The deadline for this coursework is **16/2/2025**. Please be aware that no delayed submission is possible.
- Good luck, my friends.

CW1, Part 1: Data Preprocessing using Python (2+2=4 marks)

For the following tasks, we consider a dataset (a1-condo-resale*.csv) containing information for 20,000 past resale transactions of condo flats. Each record (i.e., data samples) consists of 12 attributes. The following **data description** lists all attributes together with a brief description of each attribute's data type/domain:

- **transaction_id**: Unique ID of the resale transactions; an 8-digit integer number uniquely assigned to each transaction.
- **url**: Unique link to a website documenting this transaction as a string value.
- **name**: The name of the condo as a string value (e.g., "Estella gardens", "Eedon green").
- **type**: The type of condo as a string value (e.g., "condominium", "apartment").
- **postal_district**: The postal district the condo is located in as an integer value; Singapore has 28 postal districts: 1, 2, ..., 28.
- **subzone**: The subzone the condo is located in as a string value.
- **planning_area**: The planning area the condo is located in as a string value.
- **date_of_sale**: The date (month & year) of the transaction as a string value (e.g., "mar-19", "oct-20").
- **area_sqft**: The size of the condo flat in square feet as a positive integer value.
- **floor_level**: The range of floors in which the flat is located in the condo as a string value (e.g., "06 to 10", "11 to 15").
- **eco_category**: The eco category of the condo as a single-character string value (e.g., "A", "B", "C", "D").
- **price**: Resale price of the condo flat in Singapore Dollar as an integer value.

CW1-1-1: Data Cleaning (2 marks)

Datasets: a1-condo-resale-dirty.csv, and a1-condo-resale-nan.csv

1- Use a1-condo-resale-dirty.csv , We argued in the lecture that almost all real-world datasets contain some form of noise that might negatively affect any applied data analysis. The very first -- and in some sense -- easiest way to identify noise is to check if all data confirms with the data description. If you check the dataset against its description as given above -- with the help of **Pandas** or by simply inspecting the raw data file -- you will notice that many records are "dirty", meaning they are not in the expected format. Dirty records can negatively affect any subsequent analysis it needs. So, develop a Python program that reads the contents of the data file and **removes** the dirty data samples. (A data sample = a data record = a row in the csv or excel file.) Your program should print any case of removal and at the end show the number of removed records, and save the clean data file as result1-1.csv .

2- Recall from the lecture that data cleaning often involves making certain decisions. As such, you might come up with different steps than other students. This is OK as long as you can reasonably justify your steps.

3- Use a1-condo-resale-nan.csv, and develop a Python program to remove any data sample with empty cells. Your program should print the number and percentage of removed data samples and save the final **nan-free** dataset as result1-2.csv.

CW1-1-2: Data Transformation (2 marks)

Datasets: a1-condo-resale-others.csv

This dataset is assumed to be clean and without any missing or dirty values.

4- Develop a Python program to read the dataset, and convert the contents of columns below from categorical to numerical (encoding), using unique integer labels. For instance, considering column K, eco-category, you may replace 'A' with 0, 'B' with 1, 'C' with 2, and so on. Converted values replace the old ones in columns D, J, and K. Your program should show the first 15 records of the converted dataset and save the results in result1-3.csv file too.
   a. Column D, type
   b. Column J, floor_level
   c. Column K, eco-category

5- Develop a Python program to normalize the contents of columns **I,** area_sqft, and **L,** price, using z-transform. Your code should show the mean and standard deviation of those 2 columns before and after normalization. Also, it should show the last 15 records of the transformed dataset, then save the normalized dataset as result1-4.csv.

CW1, Part 2: Clustering using Python (2 marks)

Dataset: a1-kmeans-toy-data.csv

In the following, your task is to implement the K-Means clustering algorithm. You can and should explore relevant methods provided by **numpy** or **sklearn**.

Steps:

1- Read the datafile.
2- Visualize the contents using a 2d scatter plot.
3- Apply a k-means clustering on your data file with k=2, 3, and 4, respectively. This would be a simple k-means with random initialization of cluster centroids. Visualize the results again using 2d scatter plots.
4- Try to implement the better k-means++ algorithm and test it with k=2, 3, and 4, respectively. Visualize the results again using 2d scatter plots.

The outputs of your program will be 7 individual scatter plots. Make them readable and understandable. You may refer to the course slides to learn more about the **k-means++** algorithm. Please be aware that there is no single best answer for this part.

CW1, Part 3: MCQs (4x1.5= 6 marks)

1- The table below shows the distribution of data samples of a given experiment in the 2d feature space, <f1,f2>. Which clustering algorithm may be the most successful one and what is a good guess for k or number of clusters? (considering page width, data is shown in 3 rows per feature, e.g., the first data sample's attributes are f1=0.722, f2=32, while the attributes for the last data sample are f1=5.946 and f2=21. We have 39 data samples)

F1	0.722	0.084	0.182	0.397	0.687	0.291	0.587	0.607	0.344	0.040	0.492	0.663	0.383	0.573	0.513	0.469
F2	32	41	42	29	20	29	11	37	23	29	10	33	3	49	23	10
F1	0.135	5.917	5.943	5.387	5.963	5.153	5.673	5.241	5.882	5.752	5.397	5.231	5.938	5.030	5.108	5.564
F2	47	28	16	4	18	1	49	3	48	49	6	14	21	2	17	5
F1	5.312	5.627	5.596	5.352	5.674	5.470	5.946									
F2	20	43	6	15	43	35	21									

a. K-means, 2 clusters
b. K-means, 3 clusters
c. DBSCAN and K-means both perform well
d. DBSCAN, 3 clusters

2- Regarding the Question 1 data table, above, apply a min-max normalization on both **f1** and f2 features/attributes to bring them between 0 and 1. What are the summation of the features f1 and f2 after normalization?
   a. sum(f1)= 21.57 , sum(f2)= 18.6
   b. sum(f1)= 12.93 , sum(f2)= 93.2
   c. sum(f1)= 18.45 , sum(f2)= 31.15
   d. sum(f1)= 17.6 , sum(f2)= 25.52

3- Three data samples, A, B, and C, are represented by 3 numerical attributes/features each. We can clearly assume that each data sample is represented by a 3-element vector, called feature or attribute vector in a 3d feature space. What is the cosine similarity factor between vectors A and B, and A and C, respectively? If you refer to the slides you will realize that the inner product can provide the cosine similarity factor between vectors.

\/Data samples features →	F1	F2	F3
A	2.2	-1.3	2.9
B	8	-6.5	14.1
C	-11.4	4.7	-3.9

a. cosine(A,B)= 0.71 , cosine(A,C)= -0.15
b. cosine(A,B)= 0.99 , cosine(A,C)= -0.85
c. cosine(A,B)= 0.45 , cosine(A,C)= -0.78
d. cosine(A,B)= 0.67 , cosine(A,C)= -0.42

4- The table below shows 7 data samples and 4 attributes/features. If we apply an AGNES hierarchical clustering algorithm and employ Euclidean distance as the linkage metric, we come across the dendrogram below. What are the first [x5,x6] and second [x1,x2] data samples clustered together?

a. [x5=A , x6=G], then [x1=B , x2=C]
b. [x5=B , x6=G], then [x1=D , x2=C]
c. [x5=D , x6=F], then [x1=A , x2=G]
d. [x5=D , x6=G], then [x1=E , x2=F]

Features → Data samples \/	F1	F2	F3	F4
A	156.2	105	172.7	122
B	162.6	122	172.7	120
C	183	130	170.2	125
D	198	243	183	176
E	182	181	193	220
F	192	201	182	174
G	160	105	157	120