CS5228-KDDM, 202526-2, Coursework 1

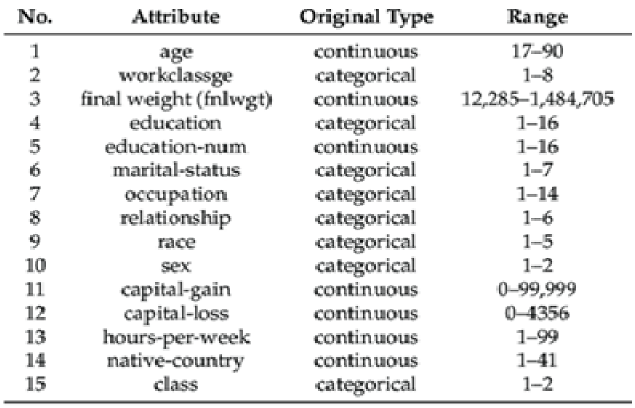
Introduction

- 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 10. 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 make sure that answers are available in the notebook before submission.
- Please submit an individual notebook for every question in Parts 1 and 2. The notebooks should assume that the datasets are in the same directory.
- 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 Sunday, 15/2/2026. 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 (census-94-1.csv) containing the results of the 1994 US census. Each record (i.e., data sample) consists of 15 attributes. The following **table** lists all attributes together with a brief description of each attribute's data type/domain. You may also find more information about that in census-94-description.txt file as well as UCI Machine Learning Repository. (https://archive.ics.uci.edu/dataset/2/adult)

You have to submit your Python notebook at the end of the day. You may call your notebooks **CW1-Q1-1.ipynb** and **CW1-Q1-2.ipynb,** respectively.



No.	Attribute	Original Type	Range
1	age	continuous	17–90
2	workclassge	categorical	1–8
3	final weight (fnlwgt)	continuous	12,285–1,484,705
4	education	categorical	1–16
5	education-num	continuous	1–16
6	marital-status	categorical	1–7
7	occupation	categorical	1–14
8	relationship	categorical	1–6
9	race	categorical	1–5
10	sex	categorical	1–2
11	capital-gain	continuous	0–99,999
12	capital-loss	continuous	0–4356
13	hours-per-week	continuous	1–99
14	native-country	continuous	1–41
15	class	categorical	1–2

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

Datasets: **census-94-1.csv**

1- Use **census-94-1.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 some 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. However, something which is definitely necessary is removing the records with missing data fields.

3- Study the situation of columns **K and L**, Capital Gain and Capital Loss. Mention what makes those 2 columns different from the other variables. Develop a Python code to show the histogram of those columns. Justify your points based on the histograms.

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

Datasets: **result1-1.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 (dummy or pseudo encoding). For instance, considering column B, Work Class, you may replace each work class with an integer number. Converted values replace the old ones in columns **B, D, and J**. Your program should show the first 15 records of the converted dataset and save the results in the **result1-2.csv** file, too.
   a. Column B, work class
   b. Column D, education
   c. Column J, sex

5- Develop a Python program to normalize the contents of columns **K,** capital gain, and **L**, capital loss, using the 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 **result1-3.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 had better 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 (Why?). 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. You may call your Python notebook **CW1-Q2.ipynb.**

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

1- Data file **cw1-mcq1.csv** contains data samples of a given experiment in the 2d feature space, <f1,f2>. There are 39 data samples in that file. Which clustering algorithm may be the most successful one, and what is a good guess for **k** or the number of clusters?

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

2- Regarding the **Part 3-MCQ 1** data, above, apply a Z-transform (normalization) on both f1 and f2 features/attributes to bring them between 0 and 1. What is the summation of the features f1 and f2 after normalization?

a. sum(f1)= 0 , sum(f2)= 0
b. sum(f1)= -0.038, sum(f2)= -0.17
c. sum(f1)= 1.45 , sum(f2)= 1.152
d. sum(f1)= -1.6 , sum(f2)= 0.523

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

	Features		
Data samples	F1	F2	F3
A	3.14	-2.21	4.14
B	8	-6.5	4.1
C	-1.4	-2.2	-3.9

a. cosine(A,B)= 0.9 , cosine(A,C)= -0.59
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)= 0.87 , 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?

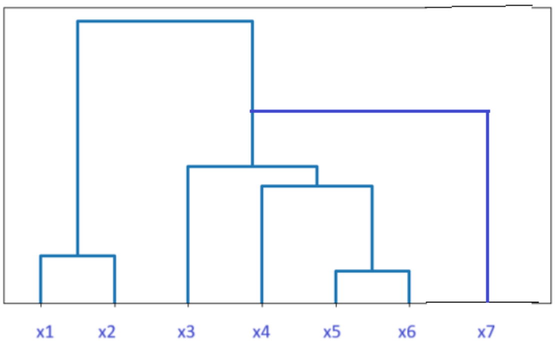
a. [x5=A , x6=G], then [x1=B , x2=C]

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


x1
x2
x3
x4
x5
x6
x7