Assignment 1

CS 412: Introduction to Data Mining (Spring 2023)
Instructor: Hanghang Tong

Release date: Jan. 17th, 2023
Due date: Feb. 2nd, 2023

- This assignment will cover the content from Chapters #1 (Introduction) and #2 (Data, Measurements, and Data Preprocessing).
- Feel free to discuss with other members of the class when doing the homework. You should, however, write down your own solution **independently**. ***Very Important Notes*: (1) there is a fine line between collaboration and completing the assignment by yourself and (2) aiding others to cheat would have the same consequence as the cheating itself. Please try to keep the solution brief and clear.**
- Please use Piazza first if you have questions about the assignment. Also feel free to send us e-mails and come to office hours.
- The assignment is due at 11:59 PM on the due date. We will be using Canvas for collecting homework assignments. **Please do not hand in a scan of your hand-written solution, only the typed solution (e.g., Microsoft Word, Latex, etc) will be graded**. The datasets for HW1 are in the HW1 folder on Canvas. Contact the TAs if you are having technical difficulties in submitting the assignment. We do **NOT** accept late assignment!
- The assignment should be submitted as a **single** PDF file using the name convention yourNetID_HW1.pdf. If you use additional source code for solving problems, you are required to submit them and use the file names to identify the corresponding questions. For instance, 'yourNetID_HW1.py' refers to the python source code for Problem 1, replace netid with your netid. Compress all the files (PDF and source code files) into one zip file. Submit the compressed file **ONLY**. (If you did not use any source code, submitting the PDF file without compression will be fine)
- For each question, **you will NOT get full credits if you only give out a final result.** Necessary calculation steps are required. If the result is not an integer, round your result to 4 decimal places.

Problem 1. True or False (20 points)
Please justify your answers with **at most** 2 sentences.

(a) (2 points) Binary attribute is equivalent to discrete attribute.

False

A binary attribute is restricted to only two values, while a discrete attribute can have a finite number of values, which can be more than two.

(b) (2 points) For a unimodal curve that is not symmetric, according to the empirical formula $mean - mode = 3(mean - median)$, its median and mode are usually on the same side of its mean.

True

The median and mode are usually on the same side of its mean for a unimodal curve that Is not symmetric.

(c) (2 points) For a group of scalars, the median is always smaller than its max and larger than its min.

False

The median is the middle value of a sorted dataset and can be either greater than or equal to the first quartile and less than or equal to the third quartile. It may or may not be smaller than the maximum or larger than the minimum.

(d) (2 points) Given two real-valued vectors, if their histogram are exactly the same, these two vectors are the same.

False

Histograms only depict the frequency distribution of the data and do not provide all the information about the data. Two vectors can have the same histogram yet still have different values.

(e) Histograms only depict the frequency distribution of the data and do not provide all the information about the data. Two vectors can have the same histogram yet still have different values.(2 points) Given a similarity measure between two objects A and B, e.g., $S(A, B)$, we can always define a valid distance measure as $D(A, B) = \frac{1}{S(A,B)}$.

True

Similarity and distance can be used interchangeably.

(f) (2 points) Given a vector of real numbers, if a raw number in the vector is smaller than the mean, its corresponding normalized value will be negative after z-score standardization.

True

The corresponding normalized value will be negative after z-score standardization if a raw number in the vector is smaller than the mean

(g) (2 points) Given any two vectors of the same length, the L_2 distance between them is always smaller than or equal to the L_1 distance between them.

True

The L_2 distance between them is always smaller than or equal to the L_1 distance if the two vectors have the same length.

(h) The covariance matrix (of one random vector) has square shape.

True

The covariance matrix has square shape.

(i) (2 points) Let p and q be two different distributions. The KL divergence between p and q is the same as the KL divergence between q and p.

False

Since KL divergence is not symmetric, The KL divergence between p and q is not the same as the KL divergence between q and p.

(j) (2 points) Principal Component Analysis (PCA) can generate new features.

True

Since PCA can reduce dimension and extract features, the extracted features will become new features.

Problem 2. Data Measurement (24 points)

Table 1 lists the heights and weights of 12 NBA players [1].

Table 1: Heights and Weights of 12 NBA players.

Player No.	1	2	3	4	5	6	7	8	9	10	11	12
Heights (in cm)	193	198	190	228	210	198	208	203	216	175	183	195
Weights (in kg)	96	98	86	140	116	92	117	113	136	81	79	92

Compute the following statistical properties for **both heights and weights**.

(6 points) Mean, mode and median.

$Mean_{heights} = \frac{193+198+190+228+210+198+208+203+216+175+183+195}{12} = \frac{799}{4}$ = 199.7500

$Mode_{heights}$ =198

$Median_{heights} = \frac{198+198}{2} = 198$

$Mean_{weights} = \frac{96+98+86+140+116+92+117+113+136+81+79+92}{12} = \frac{623}{6}$ = 103.8333

$Mode_{weights}$ =92

$Median_{weights} = \frac{96+98}{2} = 97$

(6 points) First quartile, third quartile and inter-quartile range.
For heights:

First quartile = (190 + 193) / 2 = 191.5000

Third quartile = (208 + 120) / 2 = 209

Inter-quartile range = 209 - 191.5 = 17.5000

For weights:

First quartile = (86 + 92) / 2 = 89

Third quartile = (116 + 117) / 2 = 116.5000

Inter-quartile range = 116.5 - 89 = 27.5000

(6 points) Suppose we use z-score normalization to normalize the data for all these NBA players. After the normalization, what are the normalized heights and weights? What are the sample variance and sample standard deviation of the normalized data.

```
heights_data = [193, 198, 190, 228, 210, 198, 208, 203, 216, 175, 183, 195]
```

```
mean_height = sum(heights_data) / len(heights_data)
differences_height = [(value - mean_height)**2 for value in heights_data]
sum_of_differences_height = sum(differences_height)
standard_deviation_height = (sum_of_differences_height / (len(heights_data)))
** 0.5
zero_scores_height = [(value  - mean_height) / standard_deviation_height for
value in heights_data]
print(zero_scores_height)

weights_data = [96, 98, 86, 140, 116, 92, 117, 113, 136, 81, 79, 92]

mean_weight = sum(weights_data) / len(weights_data)
differences_weight = [(value - mean_weight)**2 for value in weights_data]
sum_of_differences_weight = sum(differences_weight)
standard_deviation_weight = (sum_of_differences_weight / (len(weights_data)))
** 0.5
zero_scores_weight = [(value  - mean_weight) / standard_deviation_weight for
value in weights_data]
print(zero_scores_weight)
```

Normalized heights:
[-0.4867 -0.1262 -0.7030 2.0369 0.7390 -0.1262 0.5948
0.2343 1.1717 -1.7845 -1.2077 -0.3425]
Normalized weights:
[-0.4010 -0.2986 -0.9129 1.8513 0.6228 -0.6057 0.6740
0.4692 1.6466 -1.1688 -1.2712 -0.6057]

```
mean_height_norm = sum(zero_scores_height) / len(zero_scores_height)
var_height = sum((i - mean_height_norm) ** 2 for i in zero_scores_height) /
(len(zero_scores_height))
print(var_height)
mean_weight_norm = sum(zero_scores_weight) / len(zero_scores_weight)
var_weight = sum((i - mean_weight_norm) ** 2 for i in zero_scores_weight) /
(len(zero_scores_weight))
print(var_weight)
```

Sample variance of heights = 1

Sample variance of weights = 1

```
stdev_height = (var_height)**0.5
print(stdev_height)
stdev_weight = (var_weight)**0.5
print(stdev_weight)
```

Sample standard deviation of heights = 1

Sample standard deviation of weights = 1

(6 points) Suppose we use min-max normalization to normalize the data for all these NBA players. After the normalization, what are the normalized heights and weights? What are the population variance and population standard deviation of the normalized data.

```
import numpy as np

def normalize(x):
    min = np.min(x)
    max = np.max(x)
    range = max - min

    return [round((a - min) / range, 4) for a in x]


height = [193, 198, 190, 228, 210, 198, 208, 203, 216, 175, 183, 195]
weight = [96, 98, 86, 140, 116, 92, 117, 113, 136, 81, 79, 92]
normalized_height = normalize(height)
normalized_weight = normalize(weight)
print(normalized_height, normalized_weight)
```

Normalized heights:

[0.3396, 0.434, 0.283, 1.0, 0.6604, 0.434, 0.6226, 0.5283, 0.7736, 0.0, 0.1509, 0.3774]

Normalized weights:

[0.2787, 0.3115, 0.1148, 1.0, 0.6066, 0.2131, 0.623, 0.5574, 0.9344, 0.0328, 0.0, 0.2131]

```
mean_height_norm = sum(normalized_height) / len(normalized_height)
var_height = sum((i - mean_height_norm) ** 2 for i in normalized_height) /
(len(normalized_height)-1)
print(round(var_height, 4))

mean_weight_norm = sum(normalized_weight) / len(normalized_weight)
var_weight = sum((i - mean_weight_norm) ** 2 for i in normalized_weight) /
(len(normalized_weight)-1)
print(round(var_weight, 4))
```

Sample variance of heights = 0.0747

Sample variance of weights = 0.1119

```
stdev_height = (var_height)**0.5
print(round(stdev_height, 4))
stdev_weight = (var_weight)**0.5
print(round(stdev_weight, 4))
```

Sample standard deviation of heights = 0.2733

Sample standard deviation of weights = 0.3345

Problem 3. Data Cleaning (16 points)
Suppose a big tech company holds a database about employees' performance for year 2022 in Table 2. Please answer the following questions.

Table 2: Employee Performance

ID	Level	Year of Experience	Department	Rating Scores	Performance
43218	Senior	3.5	Privacy	83	A
26520	Senior		Ads	88	88
70645	Junior	1.5	Privacy	96	A
-08002	Staff	8.0	Retail	92	A
31466		4.0	Ads	78	A

(a) (4 points) (True or False) Does the ID column contain noisy data? Please justify your answer.
Yes, since negative values cannot be considered as ID in database system, -08002 is noisy data.

(b) (4 points) For the 'Level' column, if we fill in the missing value with the value with highest probability, what would it be? Please justify your answer.

It should be "Senior"; ID 43218's year of experience is 3.5, and its level is Senior, and there are 2 Senior Level in there. Therefore, "Senior" is the value with highest probability.

(c) (4 points) For the 'Year of Experience' column, if we fill in the missing value with the attribute mean, what would it be? Please justify your answer.

(3.5 + 1.5 + 8.0 + 4.0) / 4 = 17 / 4 = 4.25

Therefore, missing value in this column will be 4.25.

(d) (4 points) Does the *Performance* column contain any inconsistent data? Why?

Yes, most of the value of Performance are Character type, but ID 26520's Performance is 88, which make the performance column inconsistent.

Problem 4. Data Distribution (20 points)

Table 3 shows 2, 100 student grades and their preferences on courses collected from 10 years of a university's data.

(6 points) Let $\boldsymbol{p} = [p_A \ \ p_B \ \ p_C \ \ p_D \ \ p_F]^T$ be the probability distribution of the course grade. Find $\boldsymbol{p}$.

PA = (290 + 212) / 2100 = 251 / 1050 = 0.23904
PB = (428 + 347) / 2100 = 31 / 84 = 0.36904
PC = (361 + 236) / 2100 = 199 / 700 =0.28428
PD = (103 + 78) / 2100 = 181 / 2100 = 0.08619
PF = (17 + 28) / 2100 = 3 / 140 = 0.02142

(4 points) Assuming the course grade and student's preference on courses are independent, what is the expected number of students who both select CS412 and gets A?

E[A|CS412] = $\frac{sum(CS412)*sum(A)}{sum(students)} = \frac{1199*(290+212)}{2100} = 286.6181$

Table 3: Student grades and their preference on courses.

	A	B	C	D	F
CS412	290	428	361	103	17
CS512	212	347	236	78	28

(10 points) Calculate the χ^2 correlation value for "CS412" and "CS512"

```
A = [290, 428, 361, 103, 17]
B = [212, 347, 236, 78, 28]
sum = []
for i in range(len(A)):
  sum.append(A[i] + B[i])
#print(sum)
Exp = []
for i in range(len(sum)):
  Exp.append((1199 * sum[i])/2100)
for i in range(len(sum)):
  Exp.append((901 * sum[i])/2100)
#print(Exp)
chi_square = 0
for i in range(len(A)):
  chi_square += (A[i]-Exp[i])**2/Exp[i]
for i in range(len(B)):
```

```
  chi_square += (B[i]-Exp[len(A) + i])**2/Exp[len(A) + i]
chi_square
```

χ^2 correlation value for “CS412” and “CS512” is 10.8303.

Problem 5. Principal Component Analysis (PCA) (20 points)

(a) (8 points) Suppose we have 5 data points in a 2-dimensional Euclidean space: $\mathbf{x}_1 = [1\ -2]^T$, $\mathbf{x}_2 = [-\frac{3}{7}\ \frac{6}{7}]^T$, $\mathbf{x}_3 = [-3\ 6]^T$, $\mathbf{x}_4 = [\frac{6}{5}\ -\frac{12}{5}]^T$, $\mathbf{x}_5 = [\frac{20}{3}\ -\frac{40}{3}]^T$. What are the first and second principal components? (Please do not use code to solve this sub-problem.)

mean(x) = [1.08761905, -2.1752381]
x – mean(x) = [[1-1.08761905 -2+2.1752381]
[-3/7-1.08761905 6/7+2.1752381]
[-3-1.08761905 6+2.1752381]
[6/5-1.08761905 -12/5+2.1752381]
[20/3-1.08761905 -40/3+2.1752381]]
= [[-0.0876 0.1752]
[-1.5162 3.0324]
[-4.0876 8.1752]
[0.1124 -0.2248]
[5.5790 -11.1581]]

$$\text{var(x[0])} = \frac{(-0.0876)^2+(-1.5162)^2+(-4.0876)^2+(0.1124)^2+(5.579)^2}{4} = 12.5381$$

$$\text{var(x[1])} = \frac{(0.1752)^2+(3.0324)^2+(8.1752)^2+(-0.2248)^2+(-11.1581)^2}{4} = 50.1534$$

cov(x[0], x[1]) =

$$(-0.08761950 * 0.1752381) + (-1.51619048 * 3.032398095) + (-4.08761905 * 8.1752381)$$

$$+ (0.11238095 * (-0.2247619) + \big(5.57904762 * (-11.15809524)\big)/4$$

= -25.0768

Therefore, the covariance matrix is:
[[12.53838549 -25.07677098]
[-25.07677098 50.15354195]]

Since det(cov) = 0, the eigenvalues are 0 and 12.5384 + 50.1535 = 62.6919.
Since 12.5384/25.0768 = 1/2, the first princ`iple component would be [-0.8944, 0.4472] and the second principle component would be [-0.4472, -0.8944].

(b) (12 points) Consider a dataset of automobiles' information (file: automobile.csv [2]), we will analyze it through PCA. Use these five feature columns: curb-weight, horsepower, city-mpg, highway-mpg, price to obtain a subdataset, and answer the following question: what are the first principal component and second principal component of this subdataset?
If you want to use Python for this problem, please note that only standard Python library such as Numpy, Scikit-learn, Pandas are allowed. Your source code is required to be submitted and please make sure it is bug-free.

```
import pandas as pd

file = pd.read_csv("automobile.csv")

data = file[["curb-weight", "horsepower", "city-mpg", "highway-mpg", "price"]]

cent_data = data - np.mean(data, axis = 0)

cov_mat = np.cov(cent_data.T)
eigvalues, eigvectors = np.linalg.eig(cov_mat)
#print(eigvalues)
#print(eigvectors)
temp_sort = np.argsort(eigvalues)[::-1]
sorted_eigvalues = eigvalues[temp_sort]
sorted_eigvectors = eigvectors[:, temp_sort][:, 0:2]

print(sorted_eigvalues, sorted_eigvectors.T)
```

The first principal component: [-5.3389e-02 -3.7597e-03 5.5377e-04 5.9990e-04 -9.9857e-01]
The second principal component: [9.9830e-01 2.0256e-02 -7.0750e-03 -8.8853e-03 -5.3460e-02]