CS 4210 – Assignment #5

Maximum Points: 100 pts.

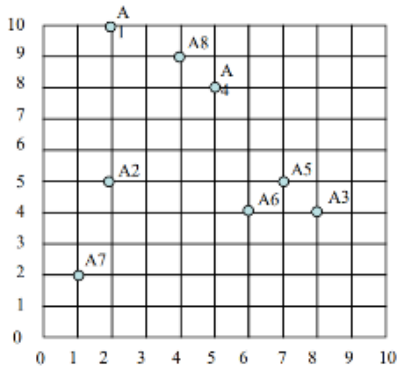
Bronco ID: |__|__|__|__|__|__|__|__|__|

Last Name: _____

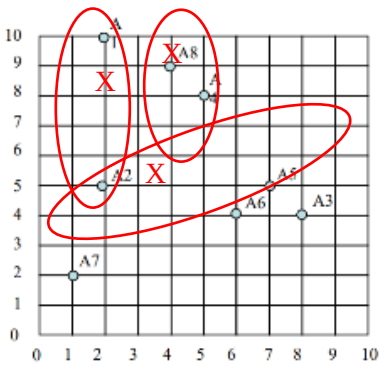
First Name: _____

Note 1: Your submission header must have the format as shown in the above-enclosed rounded rectangle.
Note 2: Homework is to be done individually. You may discuss the homework problems with your fellow students, but you are NOT allowed to copy – either in part or in whole – anyone else's answers.
Note 3: Your deliverable should be a .pdf file submitted through Gradescope until the deadline. In addition, source code (.py files) should be added to an online repository (e.g., github) to be downloaded and executed later.
Note 4: All submitted materials must be legible. Figures/diagrams must have good quality.
Note 5: Please use and check the Blackboard discussion for further instructions, questions, answers, and hints.

1. [20 points] By considering the following 8 2D data points below do:
   a. [15 points] Group the points into 3 clusters by using k-means algorithm with Euclidean distance. Show the intermediate clusters (by drawing ellipses on this 10 by 10 space) and centroids (by drawing marks like X on this 10 by 10 space) in each iteration until convergence. Consider the initial centroids as: C1 = A1, C2 = A4, and C3 = A7.



Example only!



Solution format:

1st iteration								
Centroid: (C1, C2, C3)								
Instance	A1	A2	A3	A4	A5	A6	A7	A8
C1 dist.								
C2 dist.								
C3 dist.								
Cluster Assigned								

2nd iteration centroid: (C1, C2, C3)

   b. [5 points] Calculate the SSE (Sum of Square Errors) of the final clustering.

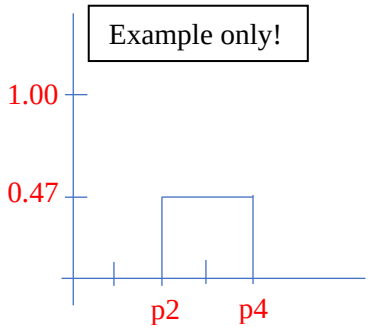
2. [20 points] Use the distance matrix below to perform the following operations:
   a. [14 points] Group the points by using single link (MIN) hierarchical clustering. Show your results by informing the updated distance matrix after each merging step and by drawing the corresponding dendrogram that should clearly present the order in which the points are merged.

	p1	p2	p3	p4	p5
p1	0.00	0.10	0.41	0.55	0.35
p2	0.10	0.00	0.64	0.47	0.98
p3	0.41	0.64	0.00	0.44	0.85
p4	0.55	0.47	0.44	0.00	0.76
p5	0.35	0.98	0.85	0.76	0.00

Solution format:

(1st iteration) Suppose the first two points to be merged are p2 and p4, then:

	p1	p2 ∪ p4	p3	p5
p1	0.00	?	0.41	0.35
p2 ∪ p4	?	0.00	?	?
p3	0.41	?	0.00	0.85
p5	0.35	?	0.85	0.00



   b. [6 points] Show the clusters when k = 2, k = 3, and k = 4.

3. [15 points] Complete the Python program (clustering.py) that will read the file training_data.csv to cluster the data. Your goal is to run k-means multiple times and check which k value maximizes the Silhouette coefficient. You also need to plot the values of k and their corresponding Silhouette coefficients so that we can visualize and confirm the best k value found. Next, you will calculate and print the Homogeneity score (the formula of this evaluation metric is provided in the template) of this best k clustering task by using the testing_data.csv, which is a file that includes ground truth data (classes). Finally, you will use the same k value found before with k-means to run Agglomerative clustering a single time, checking and printing its Homogeneity score as well.

4. [10 points] The dataset below presents the user ratings on a 1-3 scale for 6 different rock bands.

	Bon Jovi	Metallica	Scorpions	AC/DC	Kiss	Guns n' Roses
Fred	1	3	-	3	1	3
Lillian	3	-	2	2	3	1
Cathy	2	2	2	3	-	2
John	3	2	2	2	?	?

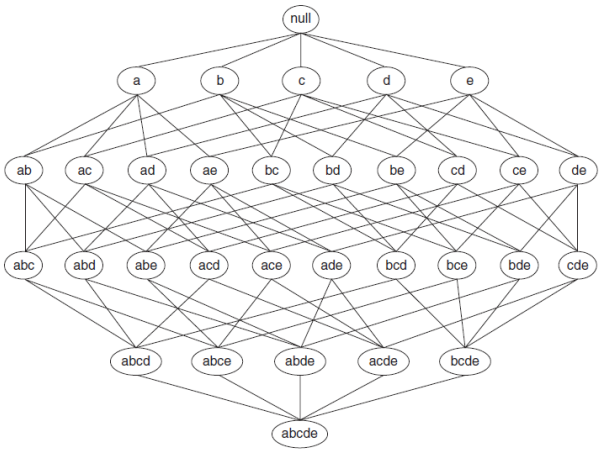
   a. [5 points] Apply **user-based** collaborative filtering on the dataset to decide about recommending the bands Kiss and Guns n' Roses to John. You should make a recommendation when the predicted rating is greater than or equal to 2.0. Use cosine similarity, a neutral value (1.5) for missing values, and the top 2 similar neighbors to build your model.
   b. [5 points] Now, apply **item-based** collaborative filtering to make the same decision. Use the same parameters defined before to build your model.

5. [20 points] Consider the following transaction dataset.

Transaction ID	Items Bought
1	$\{a, b, d, e\}$
2	$\{b, c, d\}$
3	$\{a, b, d, e\}$
4	$\{a, c, d, e\}$
5	$\{b, c, d, e\}$
6	$\{b, d, e\}$
7	$\{c, d\}$
8	$\{a, b, c\}$
9	$\{a, d, e\}$
10	$\{b, d\}$

Suppose that minimum support is set to 30% (*minsup*) and minimum confidence is set to 60%.

a. [5 points] Rank all frequent itemsets according to their support (list their support values).
b. [5 points] For all frequent 3-itemsets, rank all association rules - according to their confidence values - which satisfy the requirements on minimum support and minimum confidence (list their confidence values).
c. [5 points] Show how the 3-itemsets candidates can be generated by the F_{k-1} X F_{k-1} method and if these candidates will be pruned or not.
d. [5 points] Consider the lattice structure given below. Label each node with the following letter(s): *M* if the node is a maximal frequent itemset, *C* if it is closed frequent itemset, *F* if it is frequent but neither maximal nor closed, and *I* if it is infrequent.



6. [15 points] Complete the Python program (association_rule_mining.py) that will read the file retail_dataset.csv to find strong rules related to supermarket products. You will need to install a python library this time. Just use your terminal to type: pip install mlxtend. Your goal is to output the rules that satisfy *minsup* = 0.2 and *minconf* = 0.6, as well as the priors and probability gains of the rule consequents when conditioned to the antecedents. The formulas for this math are given in the template.

Important Note: Answers to all questions should be written clearly, concisely, and unmistakably delineated. You may resubmit multiple times until the deadline (the last submission will be considered).

NO LATE ASSIGNMENTS WILL BE ACCEPTED. ALWAYS SUBMIT WHATEVER YOU HAVE COMPLETED FOR PARTIAL CREDIT BEFORE THE DEADLINE!