Homework 4

Automated Learning and Data Analysis
Dr. Thomas Price

Spring 2020

Instructions

Due Date: April, 22 2020 at 11:45 PM
Total Points: 100 for CSC522; 88 for CSC422.
Submission checklist:

- Clearly list each team member's names and Unity IDs at the top of your submission.
- Your submission should be a single zip file containing a PDF of your answers, your code, and (if needed) a README file with running instructions. **Name your file**: G(homework group number)_HW(homework number), e.g. G1_HW4.
- If a question asks you to explain or justify your answer, **give a brief explanation** using your own ideas, not a reference to the textbook or an online source.
- In addition to your group submission, please also *individually* submit your Peer Evaluation form on Moodle, evaluating yours and your teammates' contributions to this homework.
- See below for instructions on submitting your group's meeting times.

Meeting Schedule (5 points)

You are required to set at least **two virtual meeting times** to work with you group on this homework. You should decide these times at the beginning of the homework period. When you submit your homework, you should include the dates, times, and attendees of these meetings.

Note: If you feel you are unable to find times when everyone can meet, please make a note on Piazza, at least 1 week before the deadline, and we will work out an arrangement.

Problems

1. SVM Theory (20 points CSC 522 / 8 points CSC 422) [**Yang Shi**].

   (a) Support vector machines (SVM) learn a decision boundary leading to the largest margin between classes. In this question, you will train a SVM on a tiny dataset with 4 data points, shown in Figure 1. This dataset consists of two points with Class 1 (y = 1) and two points with Class 2 (y = -1). Each data point has two non-class attributes: $X1$ and $X2$.

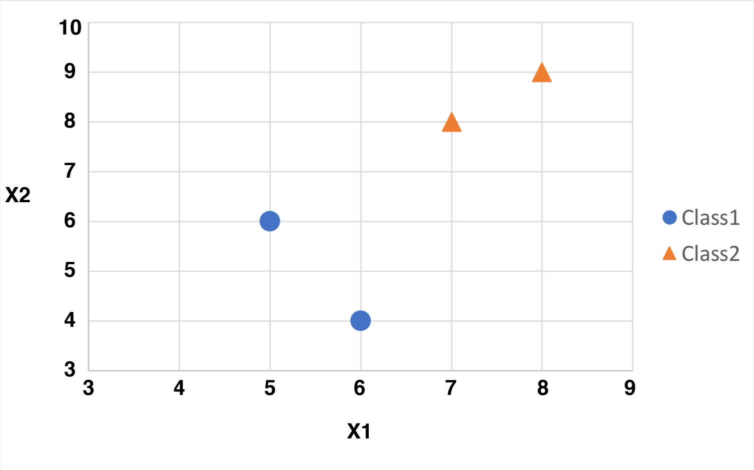


Figure 1: SVM data points for 1(a)

   i. Find the weight vector $\mathbf{w}$ and bias w_0 for the decision boundary of a hard-margin SVM. What is the equation corresponding to this decision boundary?

   ii. Circle the support vectors and draw the decision boundary.

   (b) (12 points) Required for CSC522, Extra Credit for CSC 422): You are given 1-dimensional data points $X^i, i \in [1, 2, 3, 4, 5, 6]$ as shown in Table 1 ,also shown in Figure 2 in this question.

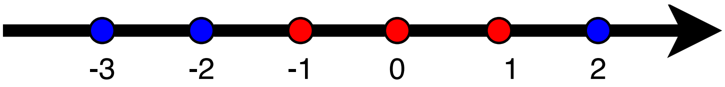


Figure 2: SVM data points for 1(b)

Data ID	x	y
X_1	-3	-1
X_2	-2	-1
X_3	-1	1
X_4	0	1
X_5	1	1
X_6	2	-1

Table 1: Six Data Points

Use this data to answer the following questions:

   i. Calculate the equation for the decision boundary of a *hard-margin* SVM, or if this is not possible, explain why in 1-2 sentences.

   ii. If you were to train a *soft-margin* SVM on this data, would you select a C value where $C \to 0$ or $C \to \infty$. Explain why in 1 sentence.

iii. Imagine you want to transform the 6 given data points to a higher dimensional space. You decide to use the kernel function $K(X_i, X_j) = (1 + 2X_iX_j)^2$, which is equal to $\phi(X_i) \cdot \phi(X_j)$. What is the function $\phi(X)$? How many dimensions is the transformed data?

iv. Use the function $\phi(X)$ to calculate $\phi(X_i)$ for $i \in [1, 2, 3, 4, 5, 6]$. Graph these data points in the higher-dimensional space. (**Hint**: If the data is more than 2-dimensional, can you simplify your visualization to show it in 2D?)

v. Is it possible to linearly separate the data in the higher-dimensional space? If so, draw the decision boundary in your graph. If not, explain why. **Note**: You do not have to calculate the weights, just draw the decision boundary.

vi. You train a hard-margin SVM on the higher-dimensional data using a library and it gives you the following Lagrange multiplier for your data[1]: $\alpha_2 = 0.1$, $\alpha_3 = 0.1$, $\alpha_5 = 0.1$, $\alpha_6 = 0.1$. What are the remaining Lagrange multipliers, α_1 and α_4? Justify your answer in 1-2 sentences. (**Hint**: This should not require any math to calculate.)

vii. Recall that the SVM's prediction (using the Kernel transformation) for a data point $\mathbf{Z}$ can be defined with the following equation:

$$f(\mathbf{Z}) = \text{sign}(\sum_i \alpha_i y_i(\phi(\mathbf{X}_i) \cdot \phi(\mathbf{Z})) + w_0) \tag{1}$$

You are given w_0 = -2/3. You are now asked to classify a new test data point, Z, using the SVM defined earlier by the Lagrange multipliers. You do not know what Z's attributes are, but you do know: $K(X_2, Z) = 25$, $K(X_3, Z) = 4$, $K(X_5, Z) = 16$, $K(X_6, Z) = 49$. Classify Z using the SVM. (**Hint**: If you find yourself trying to solve for Z's x value, you are doing it wrong.)

2. K-Means Clustering (14 points) [**Ge Gao**] Use the K-means clustering algorithm with *Euclidean Distance* to cluster the 10 data points in Figure 3 into 3 clusters. Suppose that the initial seed centroids are at points: C, I and H. The data are also given in tabular format in Table 2.

(a) After each iteration of k-means, report the coordinates of the new centroids and which cluster each data point belongs to. **Stop when the algorithm converges and clearly label on the graph where the algorithm converges.** To report your work, either:

i. Draw the result clusters and the new centroid at the end of each round (including the first round). You can use the image `hw4q2_start.jpg`, included with your homework, to mark clusters and centroids. Additionally, indicate the coordinates (x, y) alongside corresponding centroids.

ii. Give your answer in tabular format with the following attributes: Round (e.g. Round 1, 2, etc), Points (e.g. {A, B, C}), and Cluster_ID. Also report the centroids for each cluster after each round.

[1] Note, these are not the actual Lagrange multipliers for the SVM, but assume they are for the purposes of this question.

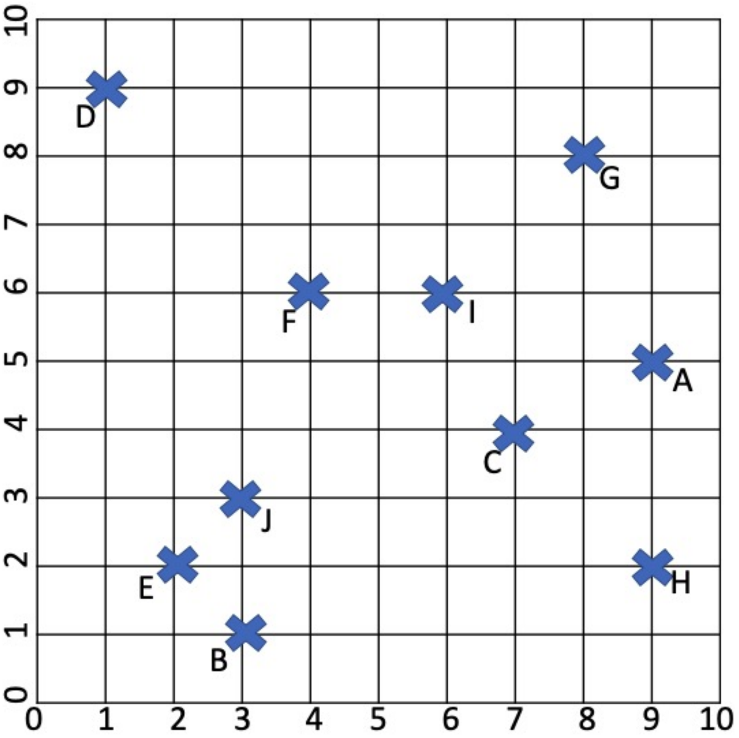


Figure 3: K-means Clustering (a)

Point	x	y
A	9	5
B	3	1
C	7	4
D	1	9
E	2	2
F	4	6
G	8	8
H	9	2
I	6	6
J	3	3

Table 2: K-means Clustering (b)

(b) How many rounds are needed for the K-means clustering algorithm to converge?

3. Hierarchical Clustering (18 points) [**Ge Gao**] We will use the same dataset as in Question 2 (shown in Figure 3) for Hierarchical Clustering. The *Euclidean Distance* matrix between each pair of the datapoints is given in Figure 4 below:

(a) Perform *single* link hierarchical clustering. Show your work at each iteration by giving the inter-cluster distances. Report your results by drawing a corresponding dendrogram. The dendrogram should clearly show the order and the height in which the clusters are merged. If possible, use a program to construct your dendrogram (e.g. PowerPoint, LucidChart[2], or VisualParadigm[3]). Scanned hand drawings will also be accepted if they are very clear.

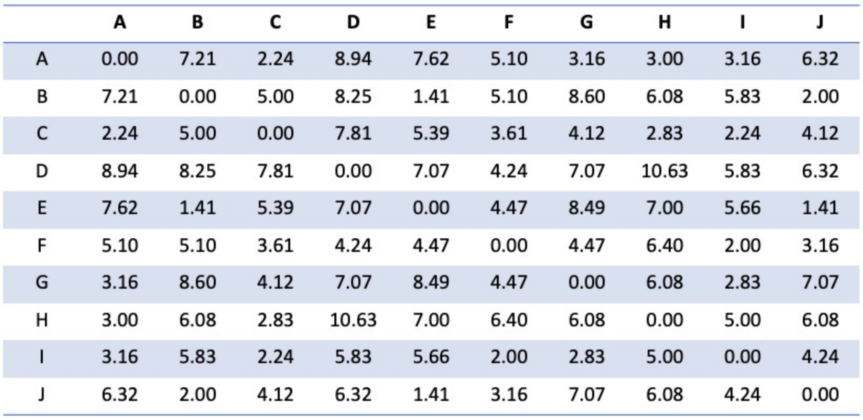
(b) Perform *complete* link hierarchical clustering on the dataset. As above, show your calculations and report the corresponding dendrogram.

[2] https://www.lucidchart.com/

[3] https://online.visual-paradigm.com/features/dendrogram-software/

(c) If we assume there are *two* clusters, will the *single* or *complete* link approach give a better clustering? Justify your answer.

(d) Consider the single-link hierarchical clustering with **3 clusters**. Compare the quality of this single link clustering with your final K-means clustering in Question 2. To evaluate the quality of each clustering, calculate its corresponding Sum of Squared Error (SSE). Based on this measure, which clustering (k-means, single link), is best? Do you agree with this assessment? Explain why in 1-2 sentences. Note: you may want to write some code to help speed up these calculations, which you can include in lieu of showing your work.



	A	B	C	D	E	F	G	H	I	J
A	0.00	7.21	2.24	8.94	7.62	5.10	3.16	3.00	3.16	6.32
B	7.21	0.00	5.00	8.25	1.41	5.10	8.60	6.08	5.83	2.00
C	2.24	5.00	0.00	7.81	5.39	3.61	4.12	2.83	2.24	4.12
D	8.94	8.25	7.81	0.00	7.07	4.24	7.07	10.63	5.83	6.32
E	7.62	1.41	5.39	7.07	0.00	4.47	8.49	7.00	5.66	1.41
F	5.10	5.10	3.61	4.24	4.47	0.00	4.47	6.40	2.00	3.16
G	3.16	8.60	4.12	7.07	8.49	4.47	0.00	6.08	2.83	7.07
H	3.00	6.08	2.83	10.63	7.00	6.40	6.08	0.00	5.00	6.08
I	3.16	5.83	2.24	5.83	5.66	2.00	2.83	5.00	0.00	4.24
J	6.32	2.00	4.12	6.32	1.41	3.16	7.07	6.08	4.24	0.00

Figure 4: Hierarchical Clustering Dataset

4. Association Rule Mining (6 points) [**Ge Gao**]. Consider the following market basket transactions shown in the Table 3 below.

Transaction ID	Bread	Milk	Butter	Eggs	Beer	Cola
1	0	0	0	1	0	1
2	1	1	0	0	0	1
3	1	0	1	0	1	0
4	0	1	0	1	0	0
5	0	0	1	0	1	1
6	1	1	1	0	0	1
7	1	1	0	1	0	0
8	0	1	1	1	0	1
9	1	0	1	1	1	0
10	1	1	0	0	0	1

Table 3: For each transaction (row), a 1 indicates that a given item was present in that transaction, and a 0 indicates that it was not.

(a) What is the maximum number of unique itemsets that can be extracted from this data set (including itemsets that have 0 support)? Briefly explain your answer in 1-2 sentences.

(b) What is the maximum number of association rules that can be extracted from this data set (including rules that have zero support)? Briefly explain your answer in 2-3 sentences.

(c) Compute the support of the itemset: $\{Eggs, Cola\}$?

(d) Compute the support and confidence of association rule: $\{Bread\} \rightarrow \{Butter\}$?

(e) Given min support = 0.3 and min confidence = 0.6, identify all valid association rules of the form $\{A, B\} \rightarrow \{C\}$.

(f) In a different dataset, the support of the rule $\{a\} \rightarrow \{b\}$ is 0.46, and the support of the rule $\{a, c\} \rightarrow \{b, d\}$ is 0.23. What can we say for sure about the support of the rule $\{a\} \rightarrow \{b, d\}$. Explain in 1-2 sentences.

5. Apriori algorithm (12 points) [**Yang Shi**].

   Consider the data set shown in Table 4 and answer the following questions using apriori algorithm.

TID	**Items**
t_1	A,B,C,D
t_2	A,B,C,E
t_3	A,B
t_4	A,C,E
t_5	A,C,D,E
t_6	B,C,D
t_7	B,C,E
t_8	C,D,E

Table 4: Apriori algorithm

   (a) Show (compute) each step of frequent itemset generation process using the apriori algorithm, with a minimum support count of 3.

   (b) Show the lattice structure for the data given in table above, and mark each node in the lattice as either **F**: Frequent, **IC**: Infrequent due insufficient support count, or **IP**: Infrequent due to pruning (we do not need to calculate the support count). (Scanned hand-drawing is acceptable as long as it is clear.)

6. **Clustering and SVM** (25 points) [**Yang Shi**].

   In this question, you will be performing a variety of machine learning operations - clustering and classification.

   (a) **PART-1: Clustering** (15 points)

   i. **Dataset description:** You are provided a dataset with 2 variables (x, y). Your data is stored in the file `data/clustering-sample.csv`.

   ii. **Note:** The TA will use a different version of `data/clustering-sample.csv`. The format (variables x , y, will be similar, but TA's file may contain different number of data points, and may look visually different than the file supplied to you. Please ensure you take this into account, and do not hard code any dimensions/outputs.

   iii. In this exercise, you will apply three different types of clustering methods to the dataset supplied to you, and then compare their results:

   A. **Clustering:** You will write code in the function `alda_cluster()` to manually implement KMeans, Single Link and Complete Link clustering. Detailed instructions for implementation and allowed packages have been provided in `hw4.R` and in `hw4_checker.R`.

   B. **SSE Calculation:** In the function `alda_calculate_sse()`, we have given you code to calculate the total SSE for a given clustering on a given dataset. The total SSE is the sum of squared error, summed over each cluster. To calculate the SSE for a cluster, first calculate the centroid for the cluster, then calculate to total Euclidean distance (error) from each point to that centroid. Read over the code to check your understanding.

   C. **Analysis-1: KMeans Elbow Plot:** You will write code in the function `alda_kmeans_elbow_plot()` to generate an elbow plot to help you choose a value of k for k-means clustering. Note that you can use `alda_calculate_sse()` to calculate the SSE for each value of k. Generate this elbow plot and save it as instructed in `hw4.R`, and place this plot in your PDF. Using this plot, report what you think is the best value of k and your justification for this choice in the PDF. Detailed instructions for implementation and allowed packages have been provided in `hw4.R` and in `hw4_checker.R`.

   D. **Analysis-2: Comparison of three clustering methods in terms of SSE:** Use the method `alda_calculate_sse()` for calculating SSE. You have been given code in `hw4_checker.R` which prints the SSE values for all the three clustering methods (with $k = 2$ for k-means). Purely based on SSE, which clustering method do you think is the best? Report this answer in your PDF.

E. **Analysis-3: Visual comparison of three clustering methods:** I've given you code for visualizing the clusters in `hw4_checker.R`, which plots the data with their cluster assignments for $k = 2$ for all the three clustering methods. Based on these plots, which clustering method do you think is the best? Report this answer in your PDF.

F. **Analysis-4: Visual comparison vs SSE**: Is your answer for visual comparison (C) the same as for SSE (D) the same? What conclusions can you draw from this? How do visualizations compare with numeric measures of cluster quality? Are numeric measures always reliable? Explain your answers in 3-4 sentences.

(b) **PART-2: Classification** (10 Points)

i. **Dataset description:** You are provided a dataset with 5 variables. Variables $x1 - x4$ refer to the independent variables, while variable *class* is your class variable. Training data is stored in the file `data/classification-train.csv`, and test data is stored in the file `data/classification-test.csv`.

ii. **Note:** The TA will use a different version of `data/classification-test.csv`. The format (independent variables $x1 - x4$, dependent variable *class*) will be similar, but TA's file may contain different number of data points than the file supplied to you. Please ensure you take this into account, and do not hard code any dimensions.

iii. In this exercise, you will apply an Support Vector Machine (SVM) classification method to the dataset supplied to you:

A. **Support Vector Machine:** You will use the `e1071` library for training and testing an SVM model. In this exercise, you will use `tune` function to tune svm models under the selected kernel. After tuning, you will report the best model and the prediction results on the test set generated by the best model. Implement the function `alda_svm()` and tune each hyperparameter as instructed in the comments (Note: some of the hyperparameters are kernel-specific). Detailed requirements for implementation, along with relevant functions and allowed packages have been provided in `hw4.R` and `hw4_checker.R`.

B. **Compare the kernels:** After getting the predictions for each kernels with hyperparameter tuning, you will first use the plotting functions to visualize the prediction results, and compare these results. Which kernel(s) do you think worked best for this dataset? Report the best kernel you identified and your reason in your PDF. Also, you are required to implement a function `classification_compare_accuracy`, to calculate the accuracy of every kernels. The details are also included in `hw4.R` and `hw4_checker.R`.

NOTE: Your entire solution `hw4.R` should not take more than 2 minutes to run. Any solution taking longer will be awarded a zero.