CS 484: Introduction to Machine Learning

Fall Semester 2023 Assignment 2 Answer Key

# Question 1 (20 points)

I invited six friends to watch a basketball game at home. They brought the following items along.

|  |  |
| --- | --- |
| Friend | Items |
| Andrew | Cheese, Cracker, Soda, Wings |
| Betty | Cheese, Soda, Tortilla, Wings |
| Carl | Cheese, Ice Cream, Wings |
| Danny | Cheese, Ice Cream, Salsa, Soda, Tortilla |
| Emily | Salsa, Soda, Tortilla, Wings |
| Frank | Cheese, Cracker, Ice Cream, Wings |

I noticed that my friends often brought Cheese, Soda, and Wings together. Since I prefer to spend on food instead of Soda, I studied how likely my friends would bring Soda if they already bought Cheese and Wings. To this end, can you calculate the Lift, Leverage, and Zhang’s metrics of this association rule {Cheese, Wings} ==> {Soda} for me?

We will determine if the itemsets in the association rule are present in each friend's basket.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Friend | Items | {Cheese, Wings} | {Soda} | {Cheese, Wings, Soda} |
| Andrew | Cheese, Cracker, Soda, Wing | Yes | Yes | Yes |
| Betty | Cheese, Soda, Tortilla, Wing | Yes | Yes | Yes |
| Carl | Cheese, Ice Cream, Wing | Yes | No | No |
| Danny | Cheese, Ice Cream, Salsa, Soda, Tortilla | No | Yes | No |
| Emily | Salsa, Soda, Tortilla, Wing | No | Yes | No |
| Frank | Cheese, Cracker, Ice Cream, Wing | Yes | No | No |

**Step 1**. Calculate

1. Support of Antecedent {Cheese, Wings} = 4/6,
2. Support of Consequent {Soda} = 4/6, and
3. Support of Rule {Cheese, Wings, Soda} = 2/6.

**Step 2**. Calculate Confidence of {Cheese, Wings} 🡺 {Soda} = Support of Rule / Support of Antecedent = (2/6) / (4/6) = 1/2.

**Step 3**. Calculate Expected Confidence of the rule is Support of Consequent = 4/6 = 2/3.

**Step 4**. Calculate Lift as Confidence divided by Expected Confidence = (1/2) / (2/3) = 3/4 = 0.75.

**Step 5**. Calculate Leverage as Support of Rule minus the product of Support of Antecedent and Support of Consequent = 2/6 – 4/6 × 4/6 = -4/36 = -1/9 = -0.1111.

**Step 6**. Calculate the Zhang’s metric as Leverage divided by (1 - Support of {Cheese, Wings} × Support of {Soda}) = (-1/9) / (1 - 4/6 × 4/6) = (-1/9) / (5/9) = -1/5 = -0.2.

Grading Notes: Students are allowed to solve this question without any Python codes.

# Question 2 (40 points)

This question walks you through the typical process of discovering association rules. We will use the market basket data in the **Chinese\_Bakery.csv** file to discover association rules. Here are the data contents.

1. Customer: Customer Identifier
2. Item: Name of Product Purchased

For your information, we have sorted the observations in ascending order first by Customer and then by Item. Also, we have removed duplicated items for each customer.

1. (10 points) What is the number of items in the Universal Set? What is the maximum number of itemsets that we can find in theory from the data? What is the maximum number of association rules that we can generate in theory from the data?

The Universal Set consists of all the unique items in the market basket data. We found that items in the Universal Set. The theoretical maximum number of items is . In our case, this number is 65,535. The theoretical maximum number of association rules is . In our case, this number is 42,915,650.

1. (10 points) We are interested in the itemsets that can be found in the market baskets of at least one hundred (100) customers. How many itemsets did we find? Also, what is the largest number of items, i.e., , among these itemsets?

The maximum number of items per customer is 14. Therefore, we set the max\_len argument in the apriori function to 14. There are 995 customers. Therefore, we set the min\_support argument in the aprior function to 100/995. Based on these specifications, we found 571 frequent itemsets. The maximum number of items among all itemsets is .

1. (10 points) We will use up to the largest value we found in Part (b) and then generate the association rules whose Confidence metrics are greater than or equal to 1%. How many association rules can we find? Next, we plot the Support metrics on the vertical axis against the Confidence metrics on the horizontal axis for these association rules. We will use the Lift metrics to indicate the size of the marker. We will add a color gradient legend to the chart for the Lift metrics.

We found 5,424 association rules with Confidence of 1% or higher. Below is the graph of plotting the Support metrics versus the Confidence metrics.

A graph of a graph showing a number of points

Description automatically generated with medium confidence

1. (10 points) Among the rules that you found in Part (c), list the rules whose Confidence metrics are greater than or equal to 85%. Please show the rules in a table that shows the Antecedent, the Consequent, the Support, the Confidence, the Expected Confidence, and the Lift. Please sort the rows in descending order of the Lift.

I added a column of count to indicate the number of customers. Here is the table of rules.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| count | antecedents | consequents | antecedent support | consequent support | support | confidence | lift |
| 113 | {'Plain Dinner Rolls', 'Sponge Cake'} | {'Bean Paste Bun'} | 0.1286 | 0.6422 | 0.1136 | 0.8828 | 1.3746 |
| 116 | {'Coconut Tart'} | {'Ham & Egg Bun'} | 0.1296 | 0.6995 | 0.1166 | 0.8992 | 1.2855 |
| 111 | {'Coconut Twist Bun', 'BBQ Pork Bun', 'Pineapple Sweet Top Bun', 'Plain Dinner Rolls'} | {'Ham & Egg Bun'} | 0.1296 | 0.6995 | 0.1116 | 0.8605 | 1.2301 |
| 110 | {'Plain Dinner Rolls', 'Sponge Cake'} | {'Ham & Egg Bun'} | 0.1286 | 0.6995 | 0.1106 | 0.8594 | 1.2286 |
| 102 | {'Coconut Twist Bun', 'Egg Custard Tart', 'Plain Dinner Rolls'} | {'Ham & Egg Bun'} | 0.1206 | 0.6995 | 0.1025 | 0.8500 | 1.2152 |

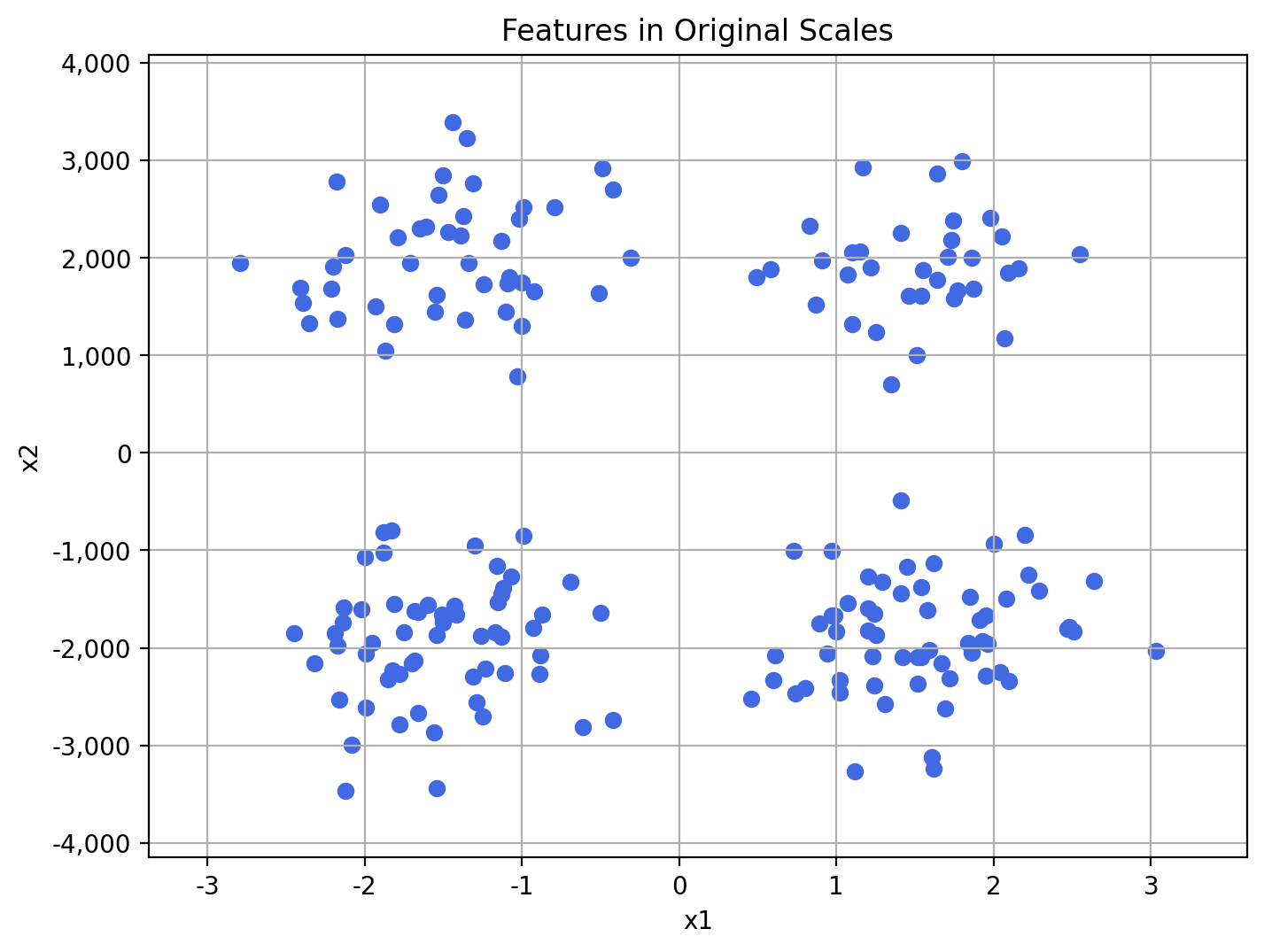
# Question 3 (40 points)

This question demonstrates the effect of rescaling input variables on the cluster results. We will discover clusters using all the observations in the **TwoFeatures.csv** file with the following specifications.

* The input interval variables are x1 and x2
* The metric is the Manhattan distance
* The minimum number of clusters is 1
* The maximum number of clusters is 8
* Use the Elbow value for choosing the optimal number of clusters

Since the sklearn.cluster.KMeans class works only with the Euclidean distance, you will need to develop custom Python codes to implement the K-Means algorithm with the Manhattan distance.

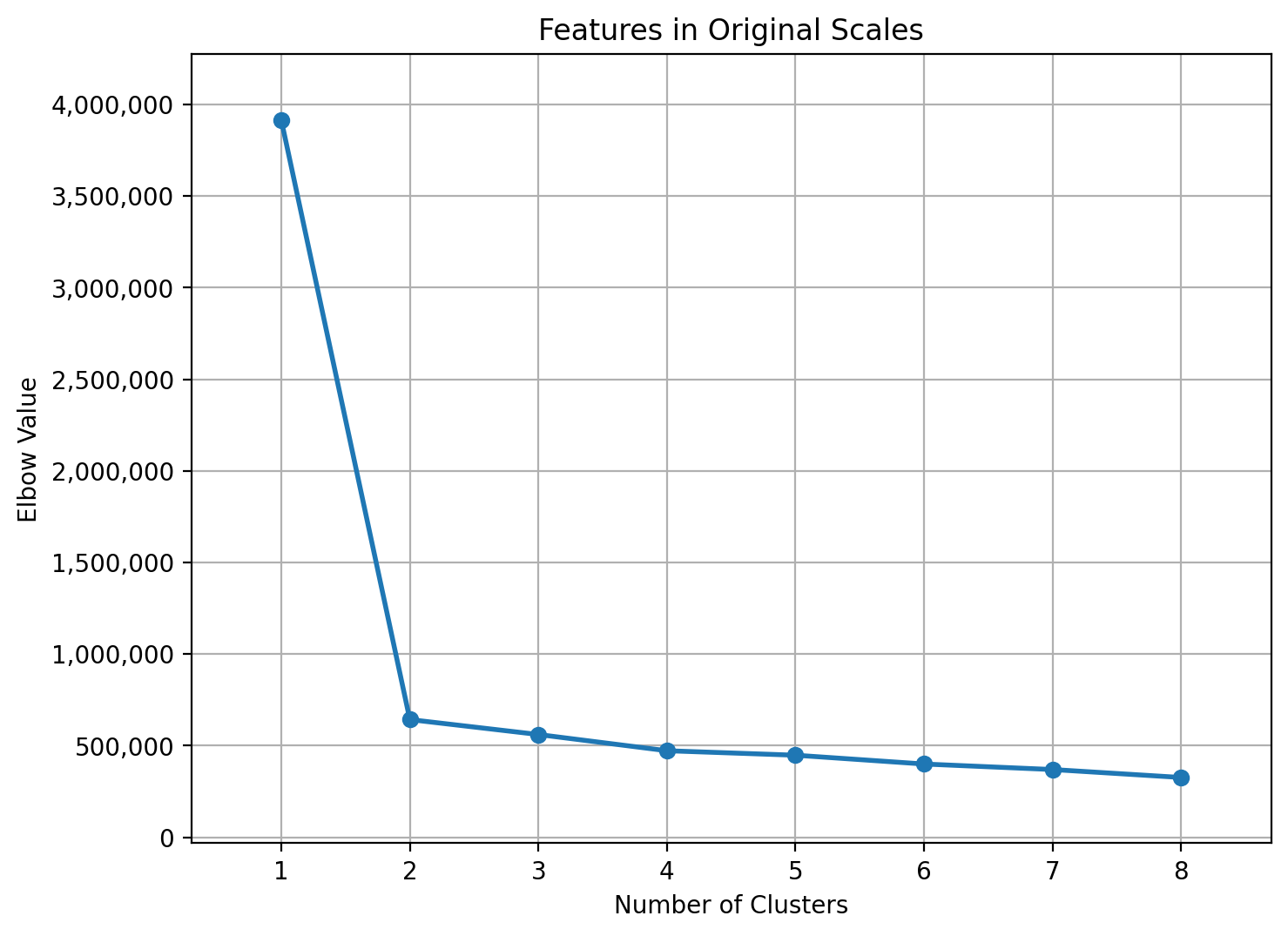
1. (10 points) Plot x2 (vertical axis) versus x1 (horizontal axis). Add gridlines to both axes. Let the graph engine choose the tick marks. How many clusters do you see in the graph?



We saw four clusters of points in the above graph.

1. (10 points) Discover the optimal number of clusters without any transformations. List the number of clusters, the Total Within-Cluster Sum of Squares (TWCSS), and the Elbow values in a table. Plot the Elbow Values versus the number of clusters. How many clusters do you find? What are the centroids of your optimal clusters?

| N Clusters | Total WCSS | Elbow Value |
| --- | --- | --- |
| 1 | 782,891,013.0239 | 3,914,455.0651 |
| 2 | 65,089,654.3897 | 642,801.0915 |
| 3 | 39,336,456.7379 | 561,190.5233 |
| 4 | 23,904,953.9075 | 472,686.2983 |
| 5 | 17,021,316.9630 | 448,255.0986 |
| 6 | 12,920,965.1758 | 400,209.8419 |
| 7 | 8,962,132.2437 | 370,061.8569 |
| 8 | 6,979,380.0268 | 326,694.5269 |

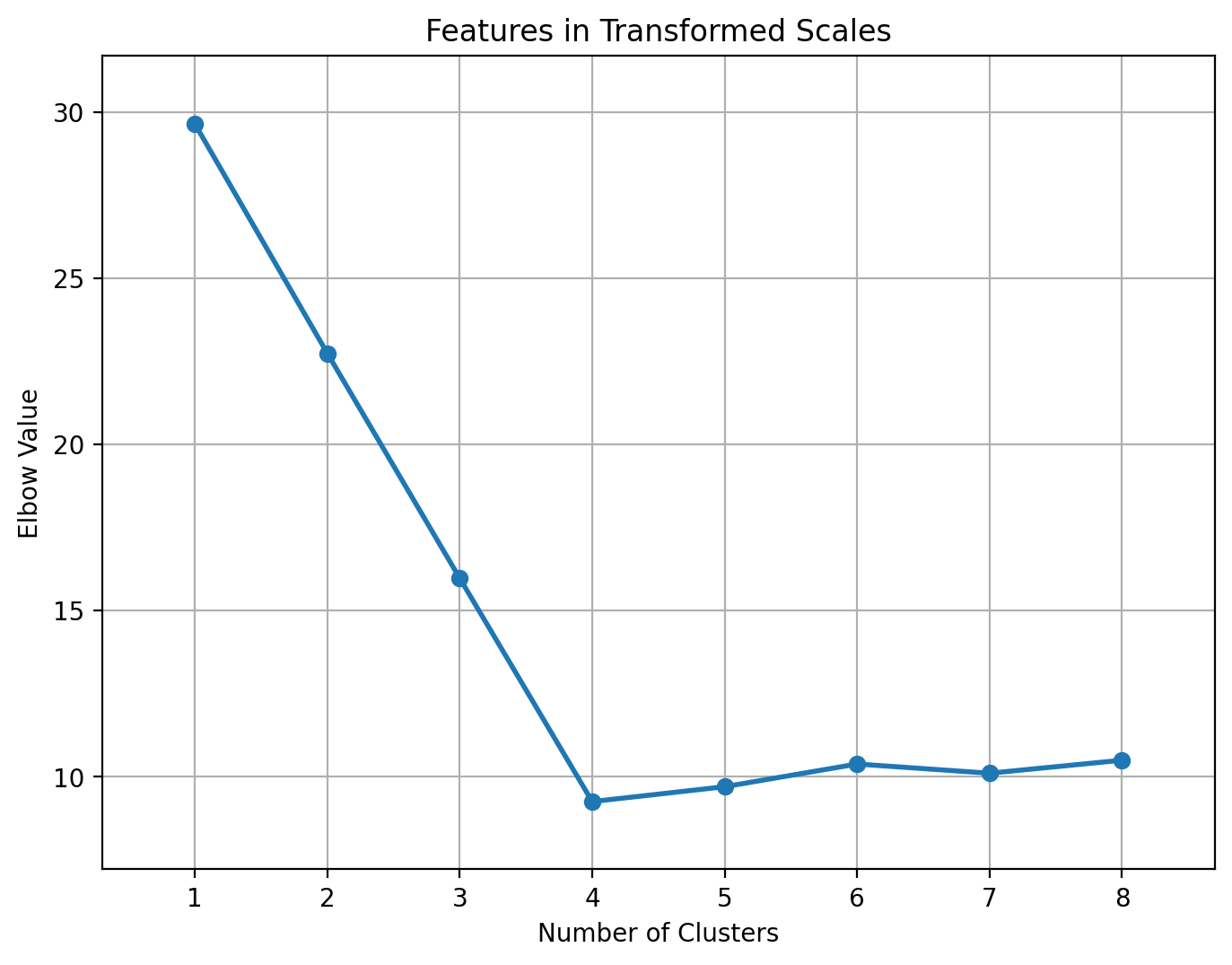


Based on the Elbow chart, we discovered two clusters. Below is the 2-cluster solution.

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Size | Centroid | |
| x1 | x2 |
| 0 | 79 | -0.1948 | 1,967.8835 |
| 1 | 121 | 0.0147 | -1,905.1967 |

1. (10 points) Linearly rescale x1 such that the resulting variable has a minimum of zero and a maximum of ten. Likewise, rescale x2. Discover the optimal number of clusters from the transformed observations. List the number of clusters, the Total Within-Cluster Sum of Squares (TWCSS), and the Elbow values in a table. Plot the Elbow Values versus the number of clusters. How many clusters do you find? What are the centroids of your optimal clusters in the original scale of x1 and x2?

| N Clusters | Total WCSS | Elbow Value |
| --- | --- | --- |
| 1 | 5,929.3737 | 29.6469 |
| 2 | 2,294.0635 | 22.7438 |
| 3 | 1,148.4590 | 15.9729 |
| 4 | 472.7691 | 9.2547 |
| 5 | 414.5425 | 9.7021 |
| 6 | 358.3880 | 10.3848 |
| 7 | 323.1894 | 10.1079 |
| 8 | 299.7722 | 10.4994 |



According to the Elbow chart, we discovered four clusters. Below is the 4-cluster solution.

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Size | x1 | x2 |
| 0 | 61 | 1.5297 | -1,879.1787 |
| 1 | 60 | -1.5255 | -1,931.6483 |
| 2 | 45 | -1.4747 | 2,017.7200 |
| 3 | 34 | 1.4991 | 1,901.9235 |

1. (10 points) If you are doing everything correctly, you should discover two different optimal cluster solutions. In your words, how do you explain the difference?

Since the scatterplot in Part (a) clearly shows four clusters of observations, we will accept the 4-cluster solution in Part (c). This exercise shows that when the features are on very different scales, the K-Means algorithm may not find the same number of clusters that we saw. Then rescaling the features may help us determine the cluster solution that is more consistent with our visual experiences.