Data Mining Project 4: Implementation of the K-Means Algorithm

Lalithanjana Kollipara (G01386376)
Mail ID: lkollipa@gmu.edu
Miner username: nocturnal

Aim:
To implement the K-Means algorithm, to deal with image data (processed and stored in vector format), to explore methods for dimensionality reduction and to think about metrics for evaluating clustering solutions.

Datasets:

1. **The Iris Dataset:** There are 4 features for each of the 150 instances. It is a simple dataset to perform the k-means algorithm. The algorithm, subject to correct implementation, will assign the cluster IDs for each instance. Cluster ID can either be 1, 2 or 3.
2. **The Images Dataset:** This dataset consists of 10,000 images of handwritten digits (0-9). The format of the input data consists on 10,000 rows, each is a record (image), which contains 784 comma-delimited integers. The k-means algorithm on this dataset, subject to correct implementation, will assign the cluster IDs for each instance, ranging between 1 to 10.

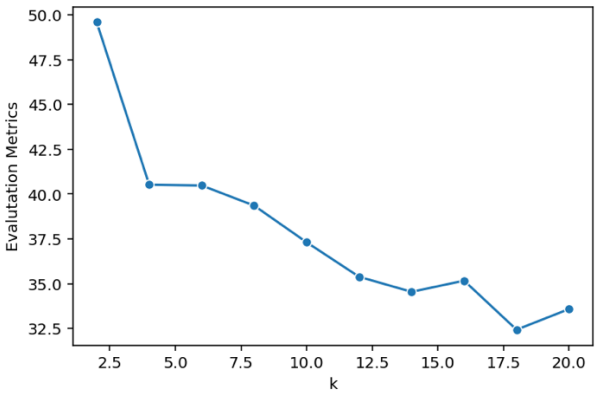
Implementation:

1. **Extracting the Data:** We read the test data files provided into dataframes on which we perform any further operations.
2. **Removing Missing Values:** We check the dataframes for any missing values and remove them.
3. **K-Means Algorithm:** The implementation of the K-Means Algorithm involves the following steps:
    a. Initialize **k** to the number of clusters we need the data to be categorized into.
    b. Assign k random centroids for each of the k-clusters.
    c. Calculate the cosine similarity distance each between data point and centroids and assign the point to the cluster whose centroid is closest to the data point.
    d. Next, we recalculate the centroids and observe the difference between the new centroids and the old ones.
    e. Repeat steps **c** and **d** until the algorithm congregates i.e., the difference between the new centroids and the old centroids is 0.
4. **Internal Evaluation Metric:** Making the use of Sum of Squared Errors as an internal metric evaluation. Sum of Squared Error is the sum of the squared distances between the cluster's centroid and each member. This is plotted on the y-axis with value of **k** increasing from 2 to 20 in steps of 2 for the data.
5. **Part A – Iris:**
    a. Apply the k-means algorithm on the Iris dataset with k = 3. The data gets divided into three clusters with 1, 2 and 3 as cluster IDs.
    b. The predictions are then stored into a .dat file and the V-measure is calculated on miner by uploading this file. The result shows an accuracy of **0.95** when uploaded on miner.
    c. The graph is plotted for all the values of k ranging from 2 to 20, with a 2-step increase and the internal metric for all these values. The values of k are on the x-axis and the sum of squared errors for the respective k values is on the y-axis.
6. **Part B – Image:**
    a. Standardize the data so that the mean is approximately 0 and the variance of all the features in the Image dataset is 1.
    b. Apply the k-means algorithm on the Image dataset with k = 10 after standardising it. The data gets divided into ten clusters with cluster IDs that are between 1 to 10.
    c. The predictions are then stored into a .dat file and the V-measure is calculated on miner by uploading this file. The result shows an accuracy of **0.53** when uploaded on miner.

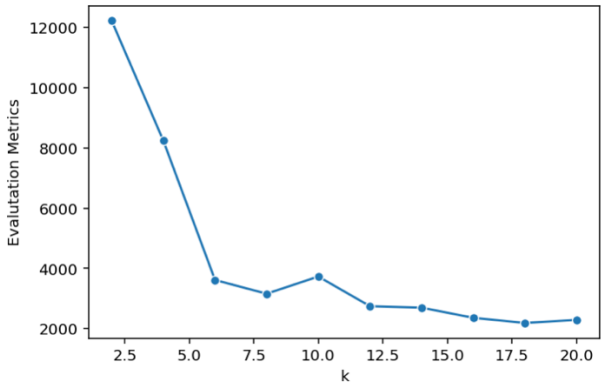
d. Perform dimensionality reduction to increase the accuracy of the model. We use two methods of dimensionality reduction.
    i. **Principal Component Analysis (PCA):** The purpose of this strategy is to reduce the dimensionality of densely connected data by translating the original collection of vectors into a new set known as the principle components and finding the projection that captures the most variance in data. For improved predictions, the enormous number of characteristics is reduced to a small number.
    ii. **t-distributed Stochastic Neighbourhood Embedding (t-SNE):** It embeds points from a higher dimension to a lower dimension while attempting to retain the point's neighbourhood. This method is also used to reduce the amount of features.
e. Apply the k-means algorithm on the Image dataset that was produced after applying both the dimensionality reduction techniques, with k = 10 after standardising it. The data gets divided into ten clusters with cluster IDs that are between 1 to 10.
f. The predictions are then stored into a .dat file and the V-measure is calculated on miner by uploading this file. The result shows an accuracy of **0.66** when uploaded on miner.
g. The graph is plotted for all the values of k ranging from 2 to 20, with a 2-step increase and the internal metric for all these values. The values of k are on the x-axis and the sum of squared errors for the respective k values is on the y-axis.

Graphs:

Graph Plot for the Iris Dataset



Graph Plot for the Image Dataset



- We can conclude from the above graphs that as the value of k increases, the SSE error decreases.
- The graph shows an **elbow effect,** which gives us the optimal value of **k** for the k-means algorithm.

Summarizing the Results:

	Iris Dataset	Image Dataset
V-Measure on Miner	0.95	0.66
Optimal k Value from Graph	Between 4 and 6	Between 6 and 8
Rank	36	281

Conclusion:
Learned how to implement unsupervised learning and its capability of classifying data that is unlabelled into a predetermined number of clusters based on similarities (k), avoiding the curse of dimensionality using dimensionality reduction measures, and evaluating clusters using internal metrics.