CS 6220 Data Mining — Assignment 5

Clustering: K-Means

This assignment requires you to apply and interpret few K-means clustering techniques introduced in class. An IPython Notebook with possible predefined functions to assist you with this assignment is available in the file M05-A01-functions.ipynb. The code in this notebook is meant to help you get started, it is not guaranteed to work or be the latest or the only way to implement the functions. But it's a good place to start and get things working. Keep in mind that the main objective of this assignment is to highlight the insights that we can derive from applying these techniques—the coding aspect is secondary. Accordingly, you are welcome to consult any online documentation and/or code that has been posted to the course website, so long as all references and sources are properly cited. You are also encouraged to use code libraries, but be sure to acknowledge any source code that was not written by you by mentioning the original author(s) directly in your submission (comment or header).

Objectives:

1. Apply clustering to an unlabeled dataset
2. Evaluate the clustering method

Submission:

Submit your ipynb file on the Assignment submission portal.

Grading Criteria:

Follow the instructions in the pdf, and complete each task. You will be graded on the application of the modules' topics, the completeness of your answers to the questions in the assignment notebook, and the clarity of your writing and code.

Assignment Description

The Data The given dataset contains information of customers related to their shopping trends. The dataset contains information on their gender, age, annual income and spending score. For the clustering we will work only with annual income and spending score of each person. These can be visually explored using scatter plots.

The Idea: Choosing k for k-means Your objective here will be to assess the performance of k-means clustering on the provided shopping dataset. Recall that the number of clusters, k, is an input parameter to the k-means algorithm. A variety of measurements are available for estimating the optimal value of k. For this assignment, you will look at the sum of squared deviation (SSQ) and the gap statistic. Both of these criteria make use of the intuition that k-means tries to minimize variance (the distance or deviation of each point from each of the k clusters) by iteratively assigning points to their nearest clusters.

Choosing k with SSQ
The SSQ criterion is a direct application of the intuition that k -means tries to minimize variance. Recall that the SSQ criterion sweeps over a range of possible k values, with each value of k associated with a degree of deviation (the distance of each point from each of the k clusters). These deviations can be squared and summed to arrive at the "sum of squared deviation" (SSQ) for each value of k. Larger values of k are expected to continue to reduce the SSQ (because there are more clusters for points to be near, reducing their deviation). However, one could expect a leveling in the SSQ once the value of k exceeds the true number of clusters, which would result in true clusters (that is, clusters actually present in the data) being separated. If, then, one plots the SSQ over a range of k values, this leveling point may produce a noticeable "elbow" in the plot. By this criterion, the estimated optimal value of k is that which occurs at this elbow point. While simple, the difficulty with this criterion is that often the elbow point is not distinctive or well-defined.

Choosing k with the Gap Statistic
The gap statistic provides a criterion that produces a quantifiable estimate of the optimal value of k over a range of possible k values. The intuition here is that there is an expected degree of deviation associated with clustering any given dataset. We want the number of clusters, k, that displays the largest "gap" between the deviation we expect, given the dataset and the number of clusters, and the deviation we estimate or observe. Thus, rather than simply considering estimated deviation by itself, we can standardize the estimated deviation for a possible value of k by comparing it with the expected deviation under an appropriate null reference distribution of the data (e.g., a uniform distribution). This difference or gap between the expected deviation and the estimated deviation is termed the gap statistic. Maximizing this gap statistic then corresponds to minimizing the estimated deviation relative to what would be expected. To ensure that we do not needlessly posit additional clusters (i.e., larger values of k), we only consider the value k+1 if its gap statistic (minus any measurement error) is higher than that for k. By this criterion, the lowest value of k with a corresponding gap statistic higher than or equal to the gap statistic of k+1 is the estimated optimal value of k.

What to Do
The provided IPython Notebook includes a possible version of functions to compute and plot the gap statistic (`gap_statistics` and `plot_gap_statistics`) and the sum of squared deviation (`ssq_statistics` and `plot_ssq_statistics`).

Load the dataset, and remove features other than Annual Income and Spending Score.

Then the sum of squared deviations (SSQ) can possibly be computed and the results plotted by using the following code snippet:
```
# Generate and plot the SSQ statistics
ssqs = ssq statistics (data , ks=range(1,11+1))
plot ssq statistics(ssqs)
```

Similarly, the gap statistic can possibly be computed and the results plotted by using the following code snippet:
```
# Generate and plot the gap statistics
gaps , errs , difs = gap statistics (data , nrefs=20, ks=range(1,11+1))
plot gap statistics(gaps, errs , difs)
```

Both the SSQ and gap statistic code snippets require a variable ks, which defines the range of k values (the "ks") to evaluate. For example, if you would like to evaluate k values between 1 an 11 (inclusive), you could set ks as range(1,11+1).

The function `plot_gap_statistics` generates two plots. The first plot simply displays the gap statistic for each k value evaluated. The second plot displays the difference between the gap statistic for value k and that computed for value $k + 1$. On the second plot, the first non- negative value, or gap difference, is the optimal number of clusters estimated by the gap statistic criterion.

What to Provide
Your output should contain the following:

- The SSQs computed for k values between 1 and 10 (inclusive). There should be one plot corresponding to the SSQs.
- The gap statistics computed for k values between 1 and 10 (inclusive). Present atleast two plots corresponding to the gap statistics.
- Scatter plot of the data in 2d showing the clusters in different colors. Also show the cluster centers in the plot.

Given this output, respond to the following questions:

1. Where did you estimate the elbow point to be (between what values of k)? What value of k was typically estimated as optimal by the gap statistic? To adequately answer this question, consider generating both measures several (atleast 5) times, as there may be some amount of variation in the value of k that they each estimate as optimal.

2. Based on the scatter plot of the clustered data, what makes most sense? Give logical interpretation from visually inspecting the clusters.

3. Between SSQ and Gap Statistics, does one measure seem to be a consistently better criterion for choosing the value of k than the other? Why or why not?