k-Means and Expectation Maximization

Joe Shymanski

April 10, 2022

Abstract

This paper seeks to report over implementations of the k-means and expectation maximization models for clustering data. These models are trained and applied to a number of different data sets, such as one-dimensional Gaussian distributions and image compression. Some clustering performance measurements are also utilized to compare the models.

1 Introduction

The k-means and expectation maximization (EM) algorithms both aim to cluster potentially unlabeled data. This data must be continuously-valued and the desired number of clusters must be pre-specified. Though their goals and inputs are the same, these two models can perform quite differently on the same data.

2 Data Sets

The data for the first four experiments came from a number of Gaussian distributions in one dimension with specified means and standard deviations (SDs). Then the k-means algorithm is used to compress various images into a few colors. These images are digital versions of famous paintings, such as Mona Lisa or The Scream. Lastly, the geographical coordinates of urban road accidents in Great Britain are clustered according to major urban centers, demonstrated by Figure 1.

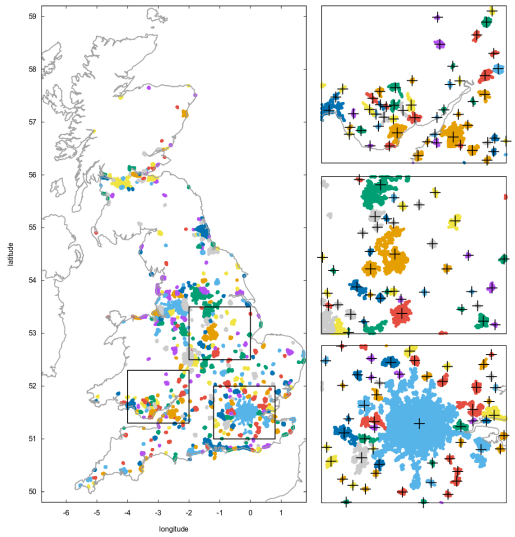


Figure 1: Original clusters for urban road accident data set.

3 Models

3.1 k-Means

The k-means clustering model is quite simple. It accepts the unlabeled data set and a value for k, the number of clusters, and it returns an array c of cluster indices for each data point and a list of means, one for each cluster.

The generating algorithm starts by initializing the means to k distinct random points from the data set. It then enters a loop with three sections. First, each data point is assigned the index of the nearest mean. Then each mean is recalculated using all the points assigned to its index. Finally, the algorithm checks to see how much these new means changed from the old means. If the maximum change is less than a certain tolerance level, the loop is finished.

3.2 Expectation Maximization

The EM model is slightly more complex than k-means. It still takes in the unlabeled data and a value for the number of clusters, but this algorithm returns a list of probabilities, means, and covariance matrices for each cluster. The goal is to estimate the parameters for k different Gaussian distributions,

along with soft probabilities for each distribution instead of distances as in k-means.

To start, the algorithm randomly initializes the soft probabilities, chooses k random data points as the first means, and creates k identical covariance matrices which are equal to the overall covariance of the data. Then it loops with three sections that are very similar to those of k-means. The first step, called the E-step, creates a weight vector w for each cluster k which is obtained from the equations given by Andrew Ng [2]. These weights are then used to re-compute the parameter estimates in the next step, called the M-step. The equations for these updates are also given by Ng [2]. Finally, the same check as in k-means is employed to determine when to exit the loop and return the estimates.

4 Experimental Results

The results for the first four experiments were averaged over 50 runs. Thus, for these experiments only, the tolerance for both algorithms was set to .01 in order to speed up the computations. The columns named Pairwise Accuracy represent the fraction of pairs belonging to the same output cluster that were generated from the same source. Every data set contains sources starting at 1 and then subsequent sources are spaced out increasingly. So if there are three sources with spacing 1.5, then the resulting sources will be at [1, 2.5, 4].

4.1 Experiment 1

For the first experiment, the number of points per source was 1000, the number of sources was varied between 3, 5, and 8 with either .5, 1, 1.5, or 2 units of spacing between each of them, and the SD of each source was 1. The trials with 3 sources tried 2, 3, 6, and 8 clusters, while the other trials simply used the same number of clusters as sources. Since there is no good way to tabulate the results, the trends will instead be discussed later.

4.2 Experiment 2

For the second experiment, the number of points per source was 1000, there were 5 sources with 3 units of spacing between each of them, and the SDs for

Model	Means	SD	Pairwise Accuracy
k-Means	[0.91, 4.00, 7.06, 10.07, 13.09]	0.85	0.78
k-Means	[0.35, 3.62, 6.87, 10.10, 13.46]	0.99	0.63
k-Means	[-.10, 3.49, 6.99, 10.43, 14.09]	0.63	0.35
EM	[0.90, 4.17, 7.41, 9.79, 13.10]	2.96	0.59
EM	[1.11, 4.62, 7.29, 9.31, 12.89]	4.97	0.51
EM	[1.38, 4.55, 7.11, 9.32, 12.80]	7.79	0.44

Table 1: Results for the second experiment.

Model	Means	SD	Pairwise Accuracy
k-Means	[-.14, 1.84, 3.55, 5.13, 7.34]	0.63	0.35
EM	[1.18, 2.75, 3.68, 4.28, 5.30]	2.88	0.30

Table 2: Results for the third experiment.

each source were varied between 1, 2, and 3. Table 1 reports on the results of this experiment.

4.3 Experiment 3

For the third experiment, the number of points per source was 1000, there were 5 sources with 1.25 units of spacing between each of them, and the SDs were sampled from the uniform distribution over [0.75, 2], thus averaging 1.375. Table 2 reports on the results of this experiment.

4.4 Experiment 4

For the fourth experiment, the number of points per source was varied between 500, 1000, and 2000 (5000 took far too long to calculate pairwise accuracy), there are 5 sources with 1.25 units of spacing between each of them, and the SD for each source was .75. Table 3 reports on the results of this experiment.

4.5 Image Compression

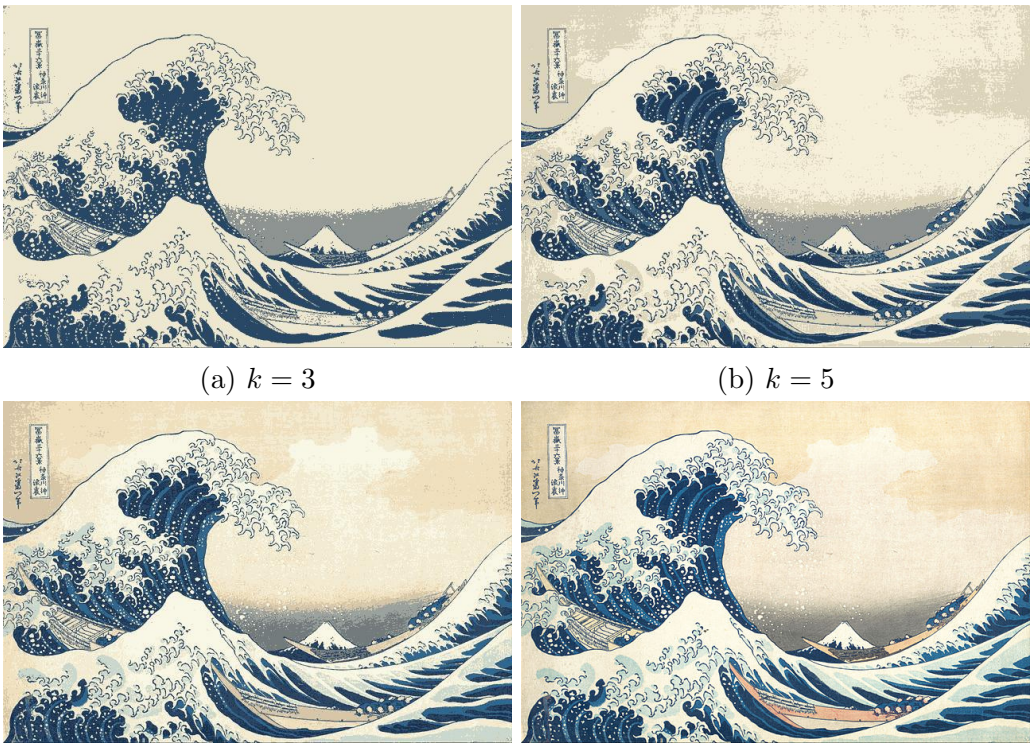
Four different famous paintings were compressed into three different numbers of colors, and the resulting images are presented in Figures 2, 3, 4, and 5.

Model	Means	SD	N	Pairwise Accuracy
k-Means	[0.65, 1.99, 3.35, 4.92, 6.50]	0.45	500	0.50
k-Means	[0.65, 2.02, 3.40, 4.89, 6.39]	0.45	1000	0.50
k-Means	[0.67, 2.07, 3.45, 4.91, 6.38]	0.45	2000	0.50
EM	[1.18, 2.66, 3.66, 4.36, 5.72]	1.28	500	0.43
EM	[1.35, 2.61, 3.60, 4.40, 5.70]	1.31	1000	0.42
EM	[1.37, 2.63, 3.62, 4.41, 5.67]	1.33	2000	0.42

Table 3: Results for the fourth experiment.

(a) $k = 3$ (b) $k = 5$ (c) $k = 10$ (d) Original

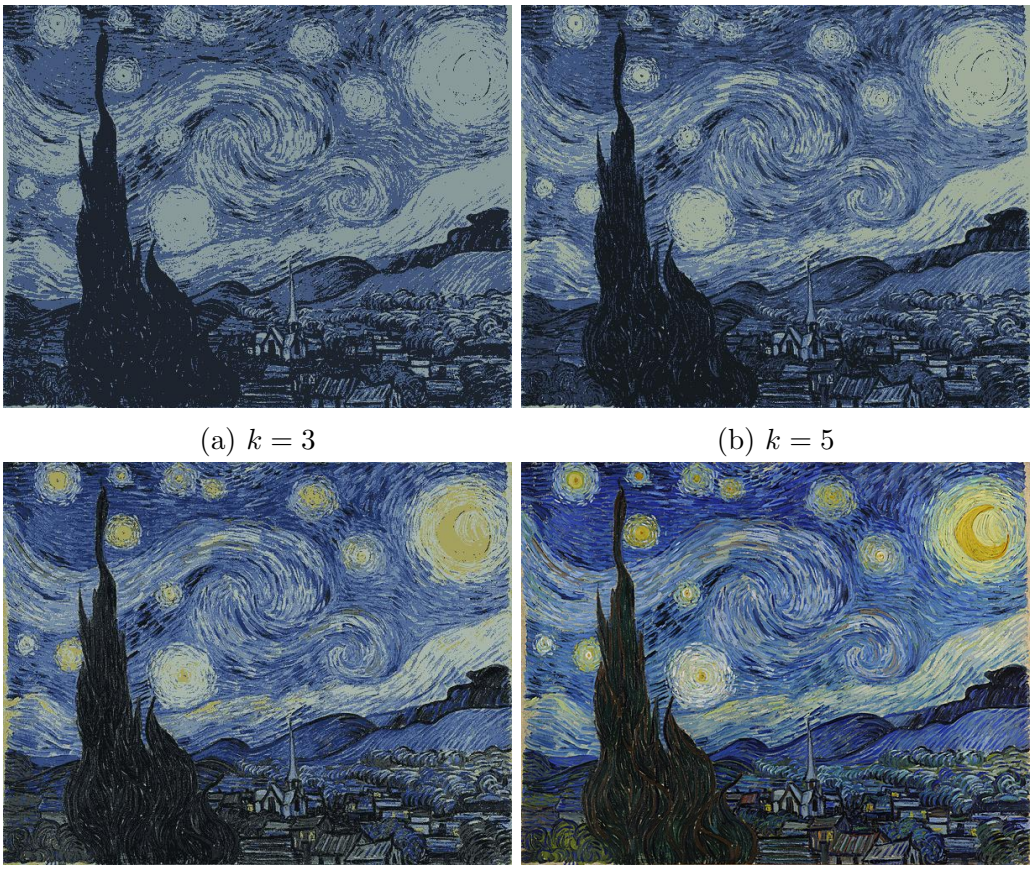
Figure 2: Different k-means clustering of Mona Lisa.



(a) $k = 3$ (b) $k = 5$

(c) $k = 10$ (d) Original

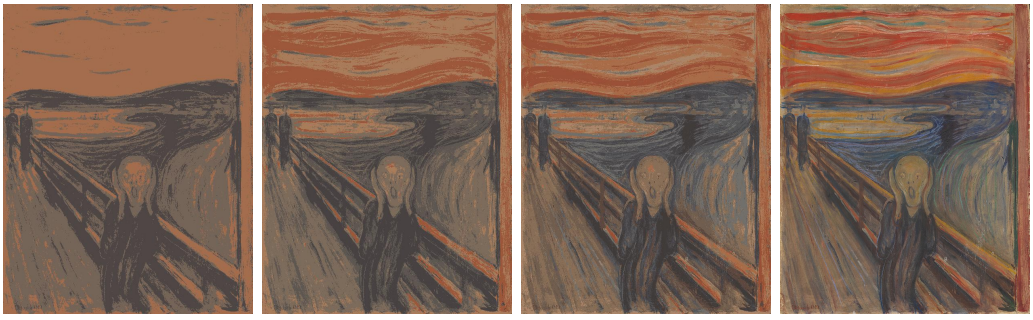
Figure 3: Different k-means clustering of The Great Wave off Kanagawa.



(a) $k = 3$ (b) $k = 5$

(c) $k = 10$ (d) Original

Figure 4: Different k-means clustering of The Starry Night.

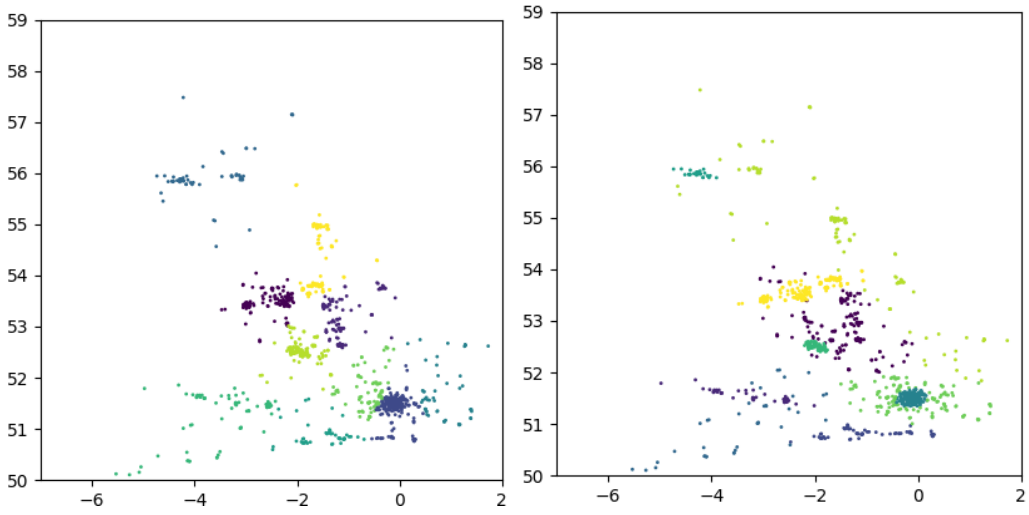


(a) $k = 3$ (b) $k = 5$ (c) $k = 10$ (d) Original

Figure 5: Different k-means clustering of The Scream.

Model	SSE (↓)	Sil (↑)	DBI (↓)
k-Means	77546.45	.47	.79
EM	80021.45	.24	6.74

Table 4: Results for clustering British urban road accidents.



(a) k-Means (b) EM

Figure 6: Different clustering of British urban road accidents.

4.6 Urban Road Accidents

In this experiment, the performance of the two models is compared using three difference measures: the sum of squared errors (SSE), the silhouette index (Sil) and the Davies-Bouldin index (DBI) [1]. The arrows next to each measure indicated whether higher (↑) or lower (↓) values denote better performance. The first 1000 data points were taken as the data set for clustering in this experiment in order to compute these performance measures in a timely manner. The number of clusters in the original data file was in the hundreds, but only a handful of centers had a significant amount of points within them. Thus, the number of clusters found by both models was 10. Table 4 reports these results and Figure 6 shows the different clusters generated.

5 Discussion

In the first experiment, as the spacing between sources increased, the pairwise accuracy increased significantly. The standard deviation also increased. For k-means, the average means were always more spread out than the original means, while the average EM means tended to be less spread out. As the number of sources and clusters increased, the pairwise accuracy decreased. Unsurprisingly, the best pairwise accuracy for the trials with three sources performed best with three desired clusters.

Table 1 shows that, in the second experiment, increasing the SD decreased the pairwise accuracy for both models. The k-means centers became more spread out and the EM centers got tighter; the former's SDs shrunk and the latter's SDs grew.

Table 2 demonstrates that neither model obtained a very good average SD compared to the SDs of the original clusters. Both models performed equally poorly given the varying SDs.

In the final experiment, the increase in data points had little effect on any of the averages or measurements. However, there may be some evidence to show that it slightly decreases accuracy.

Each image is clearly human-recognizable no matter the level of compression. Some of the images keep nearly all of their important features with only three colors used, such as Figure 3a. Most of the images look nearly identical to the original with just 10 colors used.

Finally, the performance measures in 4 show that k-means outperformed EM unilaterally when clustering the data into 10 clusters. The EM clusters are fairly good at identifying major cities, but clumps together streaks of data points that most certainly belong to all different centers. k-means seems to do a decent job of identifying the same huge urban areas and then clusters nearby smaller centers together.

6 Conclusion

The two models clearly have different tendencies given the same data. Thus, it is important to understand the domain of the clustering problem in order to make the best decision of which algorithm to use. If this is overly difficult, several performance measures exist to compare the results of the models and choose which is superior for the data.

References

[1] Olatz Arbelaitz, Ibai Gurrutxaga, Javier Muguerza, Jesús M Pérez, and Iñigo Perona. An extensive comparative study of cluster validity indices. *Pattern recognition*, 46(1):243–256, 2013.

[2] Andrew Ng. Mixtures of gaussians and the em algorithm.