KMeans Algorithm

Kruthay Kumar Reddy Donapati, kdonapat@gmu.edu

Part A: IRIS Data

Miner: kanewilliamson

VMeasure: 0.95

Ranking: 23

Abstract:

To implement the K-Means algorithm and use dimensionality reduction techniques to form clusters from IRIS data

Steps:

1. Implemented K-Means basic algorithm with the help of information from lecture slides
2. Initial clustering is solved using the Kmeans++ technique.
3. Various Dimensionality reduction methods, which are given in the homework description, are tried and ended up using t-SNE as it helped increase the VMeasure

4. silhouette_score is used as an internal evaluating clustering solution

Methodology:

Initially, I used the basic KMeans algorithm by taking random values as clusters. The VMeasure was always less than 0.6.
To improve the score, I resorted to the KMeans++. However, to my surprise, 0.72 is the maximum VMeasure that I got from that change.

I imported the sklearn's KMeans++ to evaluate my implementation and both got similar results. I didn't try dimensionality reduction as I assumed that the change would be negligible with only 4 dimensions in the original IRIS data.

After working on the Image data and Dimensional reduction, I used the t-SNE Method on the IRIS data set. Surprisingly, I got 0.95 in the VMeasure. One of the most interesting results from this experiment is that dimensional reduction is useful even if there are less number of dimensions.

KMeans is very fast to converge, I hardly crossed 12 iterations with my implementation. Convergence and Iterations had little to no effect on the VMeasure.
Silhouette Measure is used as it can be imported from Sklearn's library. It helped in estimating the VMeasure.

Observations:

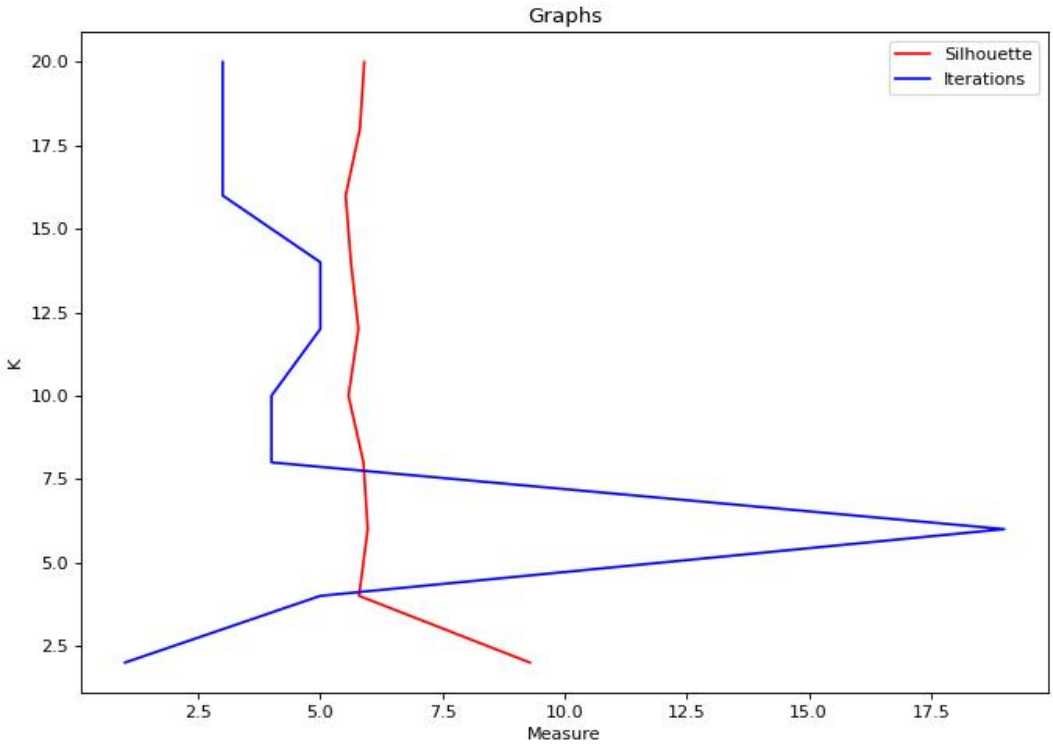
Let x = Silhouette Measure and

For -1<x<0.06, the Vmeasure is always less than 0.5

For 0.1<x<0.3, the Vmeasure is around 0.5-0.7

For x>0.3, the VMeasure is always greater than 0.7

As the VMeasure is not available for any k ≠ 3, the Silhouette Measure acted as a good reference.



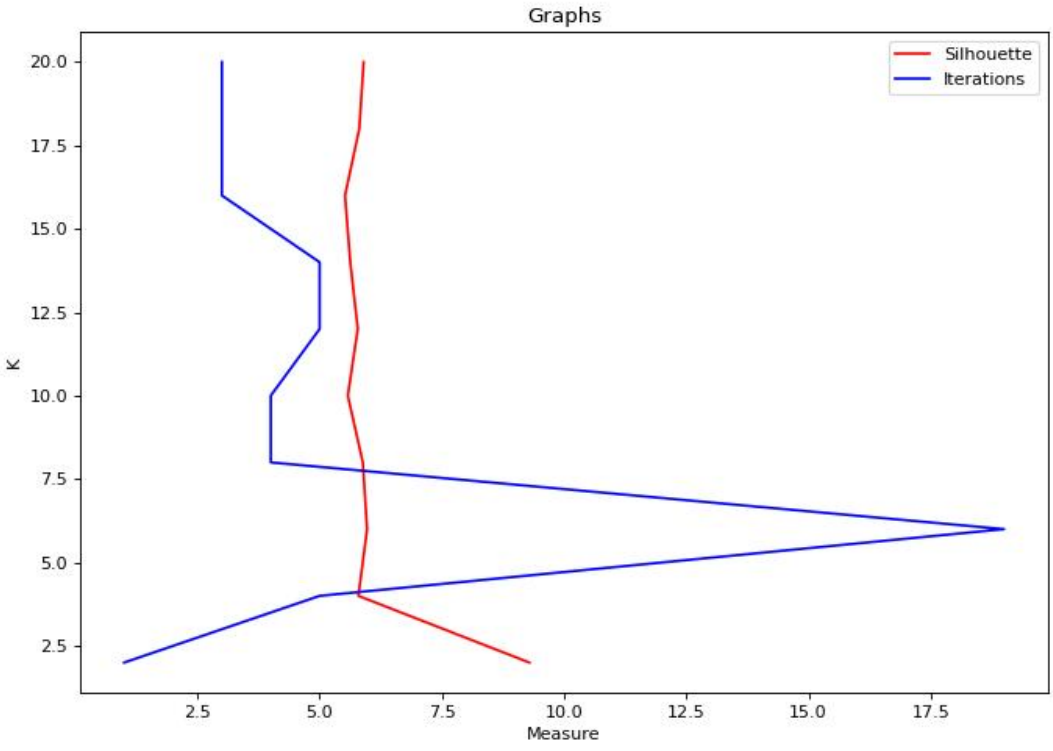
The above graph is the comparison between silhouette Measure and iterations for k values from 2 to 20. The Silhouette measure varies a lot as the k values change

Part B: IMAGE Data
Miner: kanewilliamson
VMeasure: 0.82
Ranking: 26

Methodology:

When I started working on Image data, the initial VMeasure that I got was 0.49. So I worked on dimensionality reduction, The PCA and SVD were not improving the VMeasure. However, the t-SNE improved the VMeasure to 0.82. An interesting observation is that the dimensions are reduced to just 2 components and it solved the data which has 28*28 dimensions.

Most of the observations of IRIS data are valid to the Image data



References:

1. Sklearn, NumPy, Pandas, Matplotlib's official documentation
2. Lecture slides.