Julius Ahenkora

CS 484

**K-Means Report**

SSE Table

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Approach # | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| Iris  K = 3 | 230 | 243.3 | 130.6 | 128.5 | 126.3 | 108 | 105 |
| Image  K =10 | 2.9\*1e10 | 3.4\*1e10 | 400,000 | 390,000/10 | 2.5\*1e10 | 2.35\*1e10 | 2.30\*1e10 |

Miner name: jahenkor1

Approach #1-2:

Selection: Input of data

Preprocessing: None

Transformation: None

Data Mining:

K-Means with random centroids

Distance Metric: Euclidean Distance

Clustering Method: K-Means

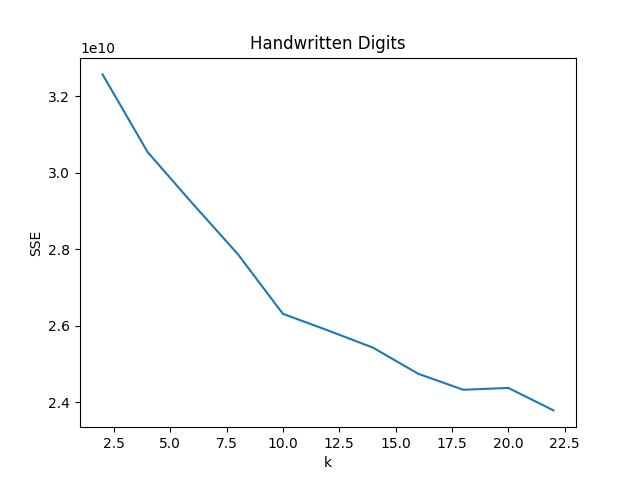
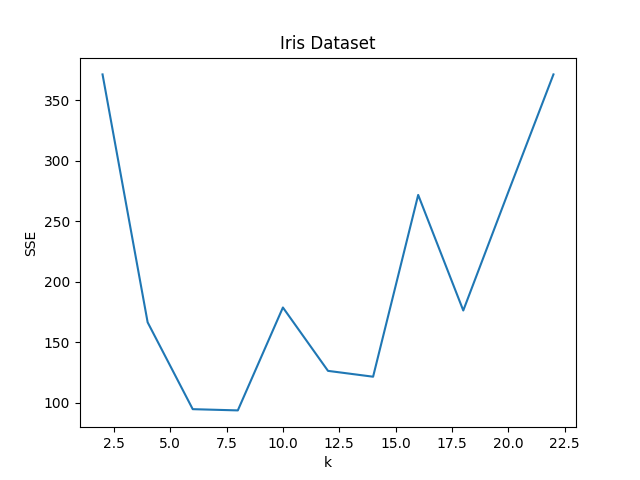
Interpretation/Evaluation:

To interpret the datasets I began with a k-means clustering algorithm which chose random points in the dataset as centroids up to a specified k value. I used euclidean distance as my distance metric and SSE as my evaluation metric which calculates the squared sum of errors from a data point to its nearest centroid. Human visualization is an important factors in k-means clustering and to assess the optimum K-values for my dataset I plotted SSE v K-value for my datasets and using the elbow criterion method, we should see an arm shape appear and select the “elbow” or K value in which there appears to be a major decrease in SSE, but is a relatively small value so our clustering algorithm has some meaning.

Choosing random points for my centroids led to SSE values with varying results, in some instances, SSE would rise dramatically and in other runs it would decrease in both datasets at the start of the algorithm. There was no consistency in SSE for a certain K value.

Iris- Optimal Clusters: 3, Handwritten-Image: 10

Elbow Method:



Approach #3-4

Selection: Input of data [Iris/Handwritten data]

Preprocessing: Standard Scaler – Iris, MinMaxScaler – Image

Transformation:

Data Mining:

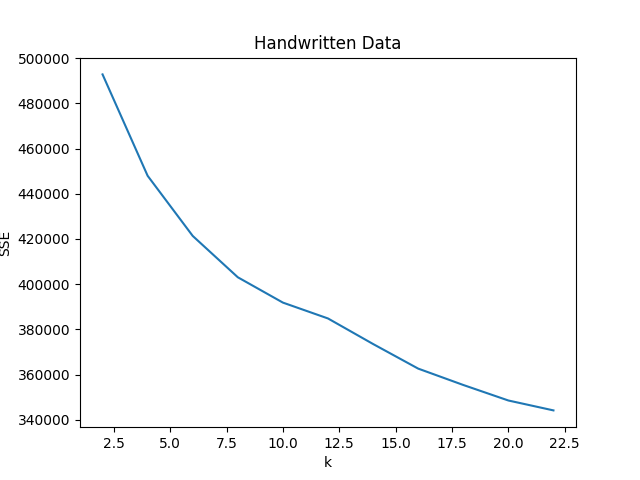
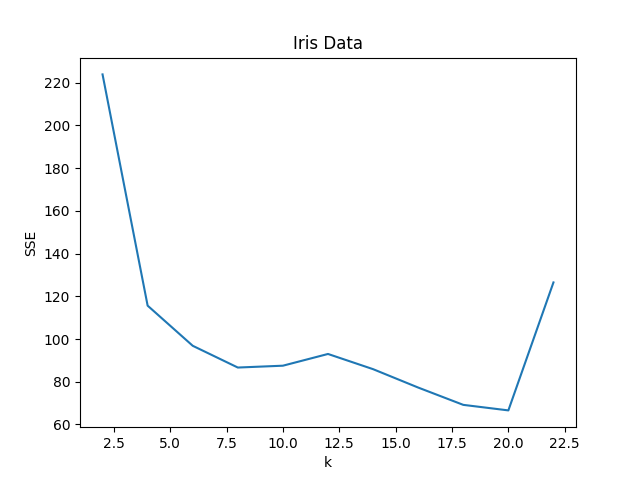
K-Means with k-means++ centroids

Distance Metric: Euclidean Distance

Clustering Method: K-Means

Interpretation/Evaluation:

I found that the k-means++ algorithm, which chooses centroids that are far away from other chosen centroids based on a probability matrix which gives a higher chance that data points at a greater distance from the current centroids are chosen as the next centroid for our clustering algorithm. I found that this algorithm converged faster, and gave me better clustering results in terms of V-measure on the data miner as well as SSE score. I chose to preprocess my data so outliers or features with higher means/variance would not disrupt the results of the clustering. I chose to “standardize features by removing the mean and scaling to unit variance” for the iris dataset and the MinMaxScaler for the image dataset, since values ranged from 0-255, and distance computations would be more performant if the data was normalized and ranged instead, from 0-1.



Approach #5-7

Selection: Input of data [Iris/Handwritten data]

Preprocessing: PCA to find centroids

Transformation:

Data Mining:

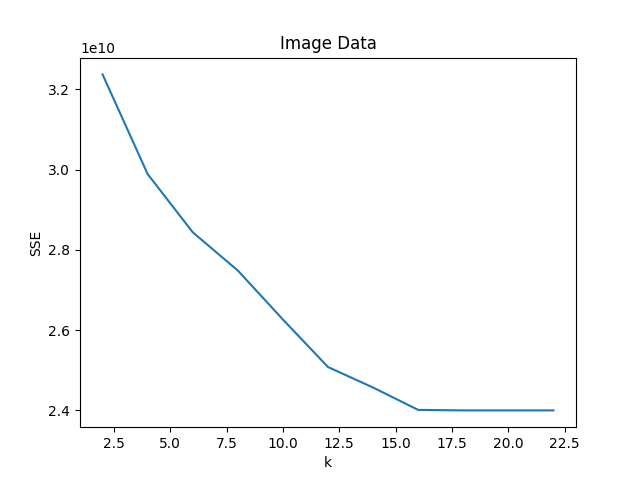
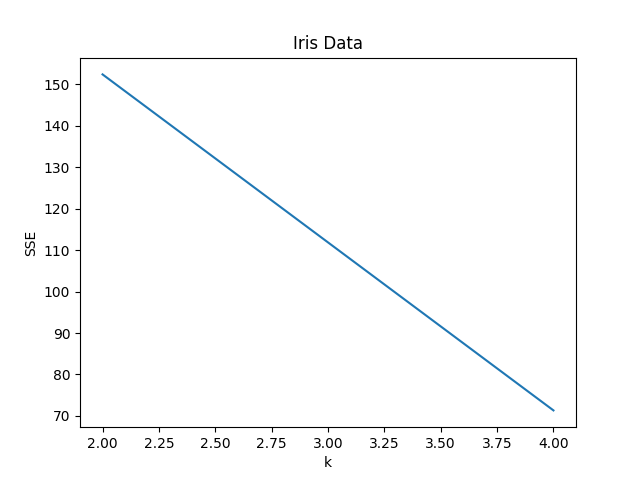
K-Means with k-means++ centroids

Distance Metric: Euclidean Distance

Clustering Method: K-Means

Interpretation/Evaluation:

Finally, for approach 5-7 I employed PCA in order to choose k centroids, with the “principal axes in feature space, representing the directions of maximum variance in the data” (sklearn). This essentially chooses centroids based on maximum variance in k chosen dimensions. Using the component vector as my intial centroids led to the fastest convergence rate that I have had thus far. The SSE, in terms of relative scale, reduced considerably compared to my previous attempts.



Approached used for miner:

Selection: Input of data [Iris/Handwritten data]

Preprocessing: Standard Scaler – Iris, MinMaxScaler – Image

Transformation:

Data Mining:

K-Means with k-means++ centroids

Distance Metric: Euclidean Distance

Clustering Method: K-Means

Evaluation Metric: SSE

Miner Score as of this report:

Iris-Dataset: 73

Image-Digits: 44

Rank: 36

References:

[https://scikit-learn.org/stable/modules/clustering.html#k-means](https://scikit-learn.org/stable/modules/clustering.html" \l "k-means)

<https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html>