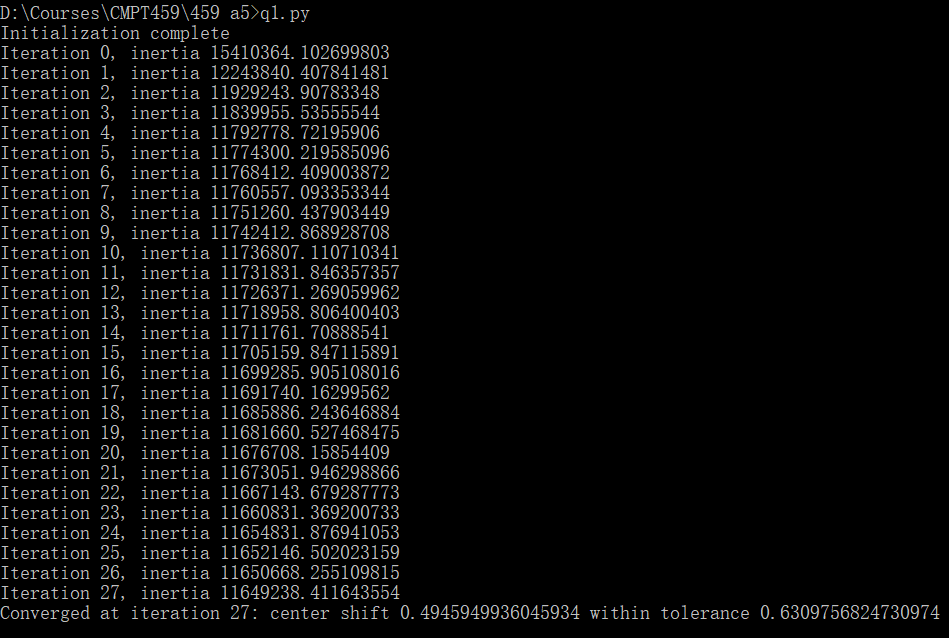
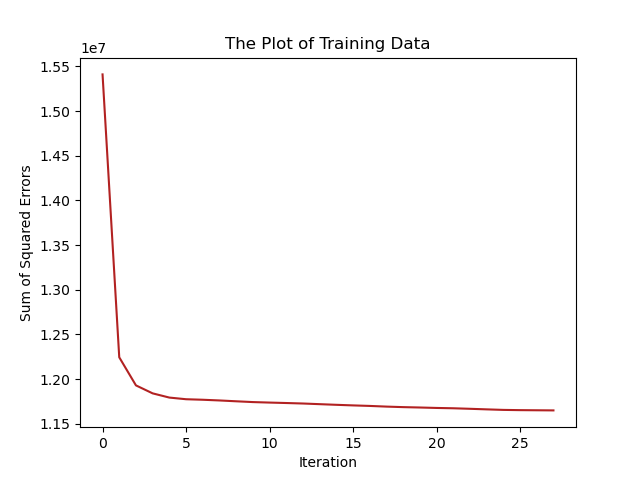
**CMPT 459 Assignment 5**

Huiyi Zou 301355563

**Question 1**

I set the parameter of function to display the SSE for each iteration. And I modified the Scikit-learn k-means source code by adding a list variable to get those SSE.

Here, the inertia is the SSE.



It is obvious that as iteration number increases, the sum of squared errors decreases. When the iteration is large enough, the curve of sum of squared errors become smooth and slow. The result of an iteration will be used for update the model as the initial parameters of the next iteration, which will improve the model accuracy. And the sum of squared errors can be used as a measure of variation. That’s why iteration number increases while the sum of squared errors decreases.

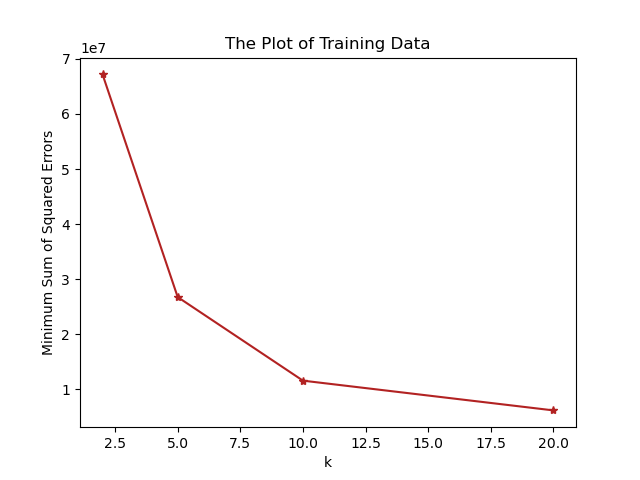
**Question 2**

k = 2, Minimum SSE: 67193645.32656775

k = 5, Minimum SSE: 26724891.416540697

k = 10, Minimum SSE: 11536048.454204438

k = 20, Minimum SSE: 6129890.485943997



The plot shows that the value of the sum of squared errors is declined. As increases, the minimum sum of squared errors decreases. With the number of clusters increasing, the points in each cluster become fewer and the points are closer to the centroid which means the minimum sum of squared errors is reduced.

**Question 3**

k = 10, Average purity: 0.9877818113298724

k = 20, Average purity: 0.9888152287484223

k = 30, Average purity: 0.9862287585276618

k = 50, Average purity: 0.9874526055286303



The average purities are very close. The average purity first increases and declines later and rises finally. When , it has the highest average purity. At this point, the cluster has a good appearance. When , it has the lowest average purity. When , the purity increases, but it does not mean the value is suitable. That is because when the value is too large, a large amount of information is scattered into many clusters, which makes the algorithm useless.

**Question 4**

For this question I modified the return value of the function in the KMeans class in the Scikit-learn k-means source code to get SSD value for each .

k = 10, SSD: 3908476.248153473

k = 11, SSD: 3777783.4772540205

k = 12, SSD: 3688159.3097920525

k = 13, SSD: 3595605.3208252597

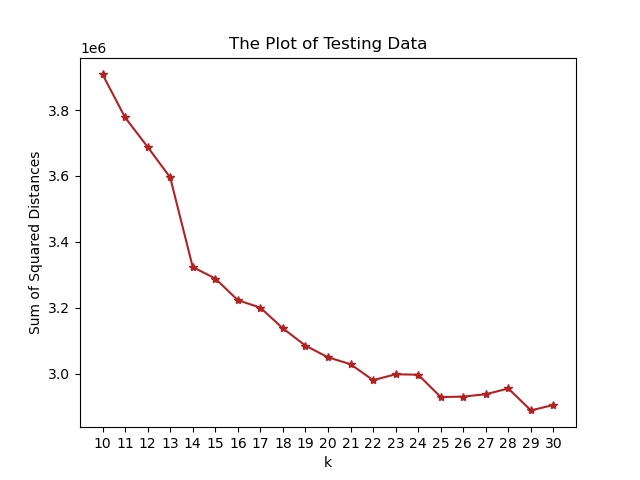
k = 14, SSD: 3323540.875105435

k = 15, SSD: 3288707.340516205

k = 16, SSD: 3222901.4116226397

k = 17, SSD: 3200275.039078066

k = 18, SSD: 3137258.7898213086

k = 19, SSD: 3084990.231775161

k = 20, SSD: 3049447.3537196205

k = 21, SSD: 3028463.1059480985

k = 22, SSD: 2980056.6311656325

k = 23, SSD: 2997876.3018657914

k = 24, SSD: 2996632.044139963

k = 25, SSD: 2928745.0413606092

k = 26, SSD: 2930120.2258050363

k = 27, SSD: 2937364.177074828

k = 28, SSD: 2954729.741762808

k = 29, SSD: 2888231.450473497

k = 30, SSD: 2904736.8478344525

When , it has minimum SSD, . Therefore, the best value of is .s