Assignment 9 Austin Frownfelter

# Problem 1

## Part a

A: (0,2.5) B: (6.5, 3.5)

A: {(0,0),(0,5)} B: {(6,7),(7,0)}

## Part b

A: (2,4) B: (7,0)

A: {(0,0),(0,5),(6,7)} B: {(7,0)}

## Part c

I chose to express the score in terms of the mean Euclidean distance between all data points and the cluster center. This lead to the following scores:

Part A: A: 2.5 B: 3.536

Part B: A: 4.212 B: 0

By this, we could argue part A was better, since B has a high variability in its cluster scores.

# Problem 2

## Part a

|  |  |
| --- | --- |
| 341.584633981219 | 578.648069832888 |
| 368.794566956760 | 349.075868598825 |
| 351.565605583060 | 338.150269960246 |
| 410.509494678876 | 346.638958150111 |
| 342.435108860697 | 429.596274640447 |
| 346.892036584773 | 359.493975761507 |
| 341.415156960650 | 735.115913386599 |
| 357.967516379631 | 326.567512155382 |
| 346.750703063115 | 560.791809808526 |
| 497.332803273314 | 341.495102009820 |
| 349.190661025391 | 339.864769320957 |
| 420.494622524663 | 345.216563296163 |
| 346.744344059110 | 414.241179431855 |
| 352.708513690241 | 340.865170209132 |
| 341.447277416296 | 631.623947816436 |
| 372.987656369841 | 340.936488403785 |
| 367.403921595743 | 349.859160538508 |
| 343.454626243984 | 565.092929865253 |
| 342.887215625346 | 564.915411452494 |
| 364.392137418244 | 335.197362389279 |
| 352.382433439432 | 342.527421972374 |
| 348.578014879485 | 340.328791672998 |
| 341.554250578052 | 561.632402384369 |
| 344.503589385257 | 478.627199079344 |
| 343.452458424311 | 363.512275117667 |
| 349.540666926441 | 355.566409224802 |
| 355.660505182178 | 350.080656144220 |
| 350.040163105803 | 361.723952432449 |
| 538.127783813139 | 341.262340500204 |
| 344.269796089177 | 424.793028353764 |

My score decided the best clustering was at index 3, with sizes 351.57 and 338.15, which appears to be the smallest when both are considered.

## Part b

My seed generation technique is as follows:

1. Select n random data points without replacement, where .
2. Place centroid at the center of mass (average) of those n data points.
3. Repeat step 1 until k clusters are created.

This “initial” seed was better (having a lower score) 79% of the time, where the default, kmeans++ seed was better 21% of the time.

## Part c

In every case, the best-of-30 for kmeans++ was perfect classification. Mine was perfect for 10% of those bests. In general, my best-of-30 classification score was around (mean) 98.52%.

To change this, I did a mean instead of max, in which case my average classification score was ~88%, with kmeans++ at 93%. Never did my initial seed beat kmeans++.

## Part d

When I ran my initial seed, the resulting classification scores were the same, in all cases. I checked the permutations, and all were “shuffled”. Based on this result, the class labels and clusterings are 100% random. I believe I am not calculating something right, but assuming my scoring is correct, the permute and check are correct. This should look more “normal”, but it’s a constant.

