

Question 10

c) With K-means clustering of K=3, the algorithm

```
> print(km3$cluster)
[1] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2
[46] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
> print(km)
K-means clustering with 3 clusters of sizes 20, 20, 20
```

correctly identified all 3 clusters of data with ease. They are not numbered the same according to the labels I assigned the data, but the first 20 observations were all classified together, and same for the other 2 sets of 20.

d) With K=2, the split was less even. The data was split

```
> print(km2$cluster)
[1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
[46] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
> km2
K-means clustering with 2 clusters of sizes 20, 40
```

20:40, since the original middle cluster was closer to the cluster on the right than the left, so K=2 identified them as the same cluster

e) With K=2, and nstart = 20, the

```
> print(km4$cluster)
[1] 3 1 3 1 3 3 1 3 1 3 1 3 1 1 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 4 4 4 4 4
[46] 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
> km4
K-means clustering with 4 clusters of sizes 10, 20, 10, 20
```

algorithm identified 2 sub-clusters in the first set of 20 observations. I am sure with a different starting assignment of clusters, the algorithm would choose to split up one of the other clusters. From my plot from Monday, the cluster on the left is more separated than the others, so it may still choose to separate that one.

