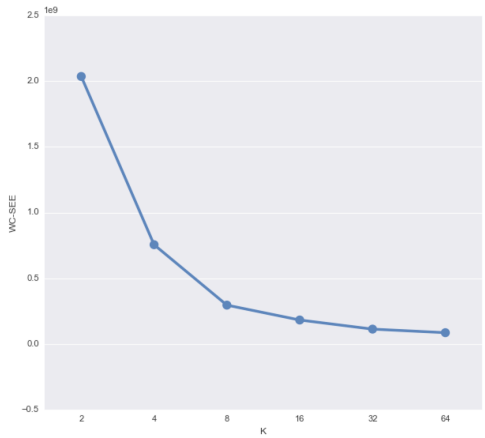
HW2 - Answers

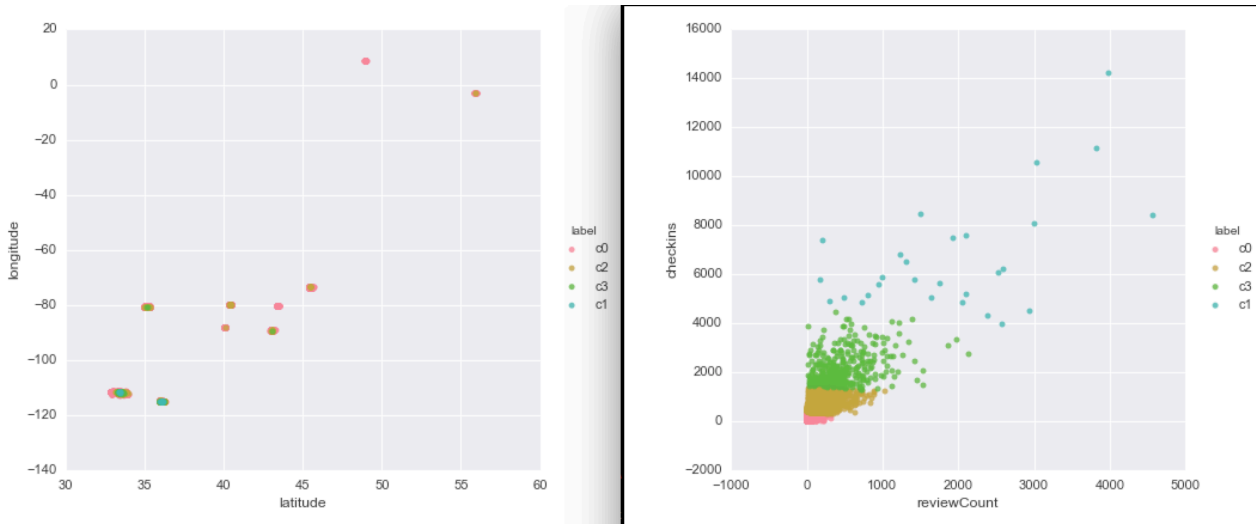
1. **30 Points:**
   - **AC - 15 Points:**
     - ✳ Correct Output (5 Points)
     - ✳100 sample run (5 Points)
     - ✳1000 sample run (5 Points)
   - **KM - 15 Points:**
     - ✳ Correct Output (5 Points)
     - ✳100 sample run (5 Points)
     - ✳1000 sample run (5 Points)

2.a - 9 Points



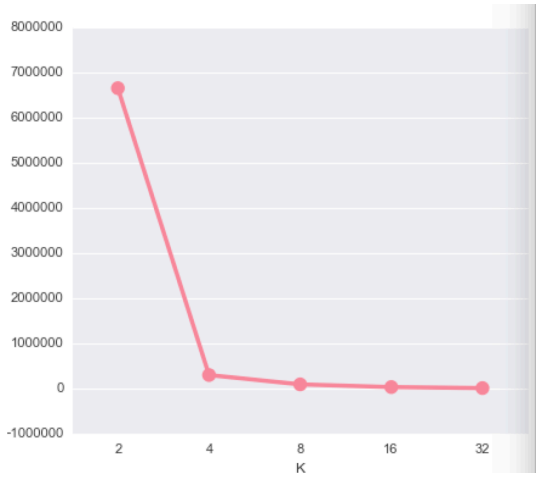
- K can be any point between 8 and 32
- Because after those points the gain in WC-SEE does not seem to improve significantly.

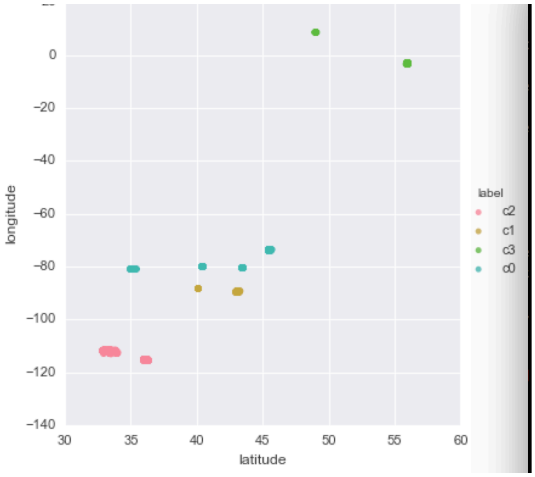
2.b - 9 Points

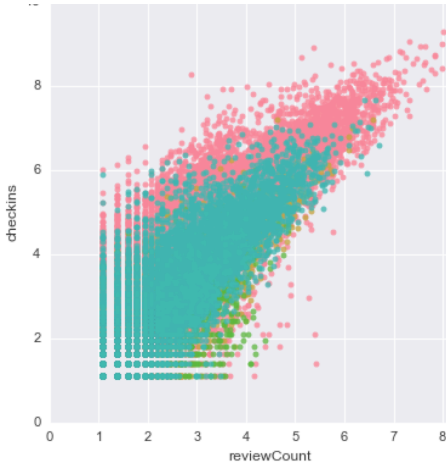


- Checkins is the one having most influence on the distance metric because it has the highest ranges of values (greater scale), thus it is driving the clustering algorithm.

2.c - 16 Points

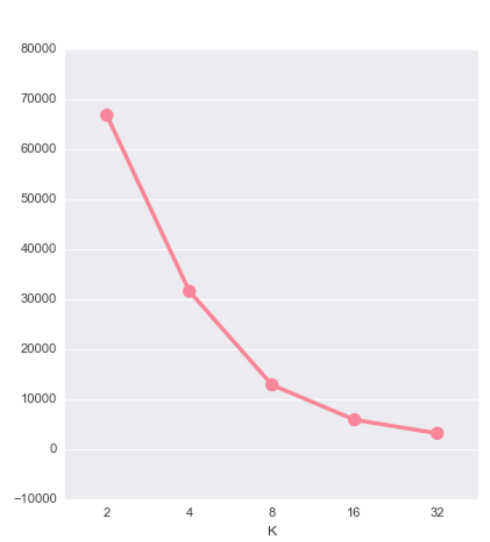


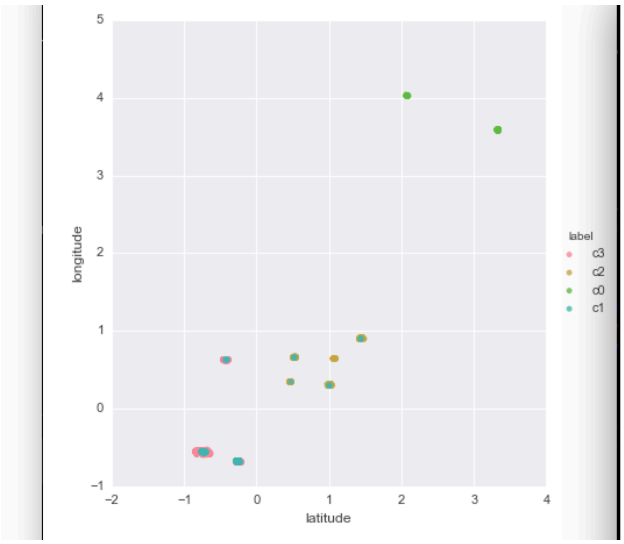




- The dimension driving the algorithm is longitude.
- The new value for K is any number in the interval [4, 6] the reason being the same as in 2.a
- The range of values and shape in the first plot (Error vs. K) changed because *Checkins* and *Review Count* are contributing less than longitude and latitude to the distance measure.
- The new dimensions driving the algorithm changed because we scaled the two dimensions with greater order of magnitude. This leads to longitude being the most influential in the distance metric.
- Now a couple of clusters are clearly visible and make more visual sense.

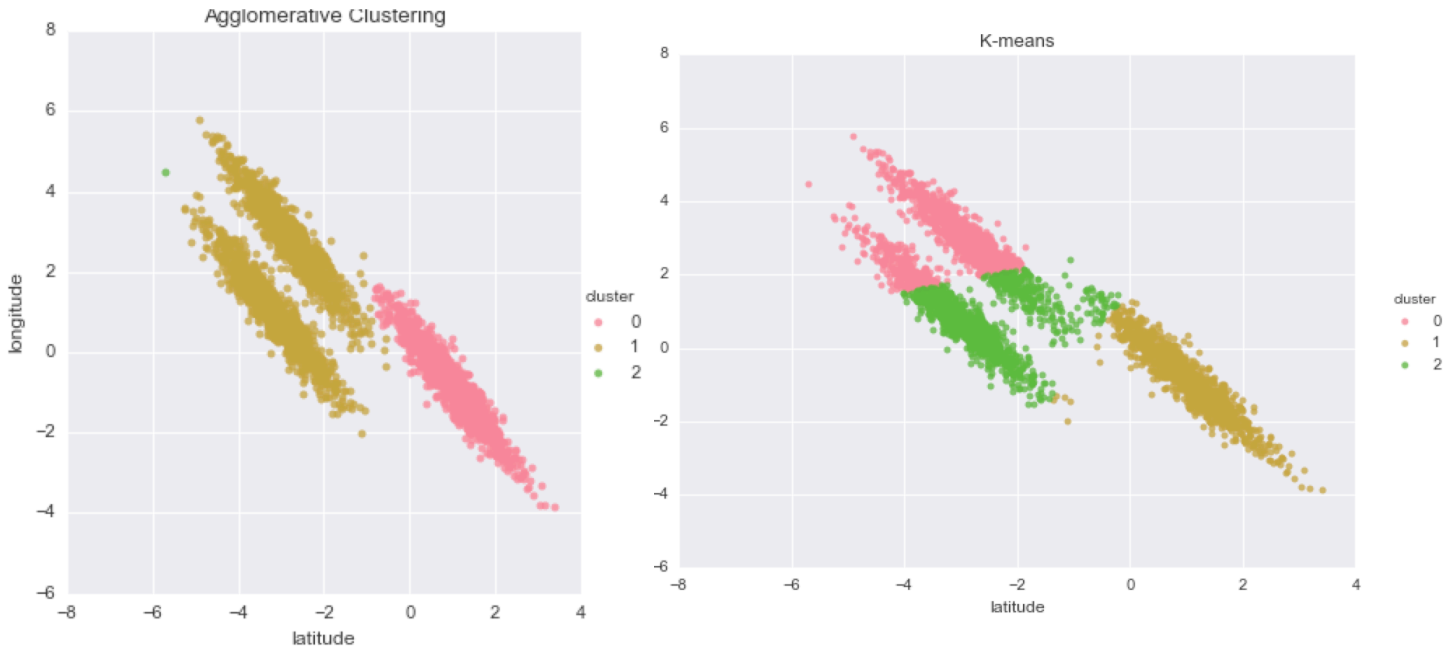
Q2.d - 16 Points





- Now there seems like every feature is contributing a little bit to the model
- The selection of K should change to the interval between [8, 32] the reason being the same as in 2.a. Also, the range of values and shape of the first plot changed
- The new dimensions driving the algorithm changed because we scaled everything. Even if now every dimension is contributing to the decision, Checkins and Review counts are distributed in a different way (more spread) than Lat and Long. The former two dimensions managed to have greater influence than the latter two even after the scaling.
- Even if we scaled all features we are not able to give them the same weight, thus another method should be tested as part of the data analysis
- Dimensions without clear clusters can drive the model without letting us obtain clear (visible) ones

3.a - 12 Points



- Subjectively (looking at the plot), Agglomerative clustering performs better because K-means creates a clusters that splits clusters into several parts.
- The reason being that the k-means, model using the euclidean distance, will put together points that are in a radius (circumference) to the centroid. As the shapes are not circles, but ellipsis, K-means is not able to produce a good clustering.
- On the other hand, agglomerative clustering will take points that in average are near. As result, AC leaves the cluster from the bottom right perfectly separated (nothing is nearer to them than themselves). Moreover, even if it is not able to separate the top clusters, it does not divide them.

3.b - 4 Points

- K means does not always give the same answer because it has a random initialization
- AC should always yield the same result because it is a greedy algorithm that takes deterministic decisions.

3.c - 4 Points

Depends on the value of K. Example: when k=n-1 where n is the number of samples agglomerative clustering just needs to merge the two nearest points. On the other hand K means would need to update all clusters and converge. If the number of iterations is not fixed and the tolerance is very small it will take more time for K-means to give a answer.