Byoungjin Oh

PUID: 0030411229

Late day use: 2

CS373: Homework 4

Due Date: 11:59 PM, Nov 15, 2019

Kmeans (25 points)

Theory (10 points)

1. (2 points) What are some limitations of using k-means algorithm?
   - Terminates at local optimum
   - Applicable only when mean is defined
   - Need to specify k
   - Susceptible to outliers/noise
2. (2 points) Can you always discover structure in data using k-means algorithm? Does the algorithm always generate meaningful clusters (e.g., in uniformly distributed data, in binary data)?

   K-means algorithm does not always discover structure. Basically, k-means algorithm is used to group data instances similar to each other (e.g., clustering). Sometimes, it is hard to find features that best describe the data (Dimensionality reduction), discover useful patterns and rules from data (pattern mining) and detect behavior/pattern that is out of normal (Anomaly detection). The algorithm dose not always generate meaningful clusters. If there is a outliers/noise, then the center of the cluster will be calculated with the outliers then the cluster would not be meaningful.

3. (2 points) What is the theoretical time complexity of the algorithm? Discuss how you can parallelize the algorithm to improve the complexity.

   O(|clusters| x |data-instances| x |iterations|).
   Divide the data-instances by the number of threads and each thread assign the data to a cluster and combine them to get a new center of each cluster and repeat until the center does not change.

4. (2 points) Consider dataset X with p discrete, q continuous attributes and 1 binary class label. Imagine you learnt cluster memberships for each of the data instances using the k-means algorithm on the q continuous attributes. Let approach A represent the Naive Bayes classifier learned only with the p discrete features, and approach B represent the NBC model using the p discrete features and a new cluster membership feature (learnt using k-means). Discuss how the accuracy (or 0/1-loss or squared loss) of approach A is expected to be compared to that of approach B.

   Approach B has better accuracy than approach A because B has more information (p + q features) than A has (only q) and the more information will increase the accuracy.

5. (2 points) Improve the score function. To evaluate the clustering, it is not sufficient to measure only the within-cluster sum of squares (wc) defined above. It is also desired to have each cluster separate from others as much as possible. To improve the resulting clustering, define your own score function that takes into account not only the compactness of the clusters but also the separation of the clusters. Write a formal mathematical expression of your score function and explain why you think your score function is better than the within-cluster sum of squares.

   To improve the score function, I would calculate between-cluster distance ($\sum_{1 \le j \le k \le K} d(r_j, r_k)^2$ *where K is the number of clusters and r is the cluster centroid*) and then divide

it by within-cluster sum of squares (bc/wc). Calculating distance between clusters will be helpful for score function because the farther the groups are, the better it is to distinguish them. So now we have minimizing within-cluster distance and maximizing between-cluster and the ratio bc to wc is the score function.

Analysis (30 points)

You only need to include your plots and discussions in your report. Make sure that the code you submit doesn't include any changes you don't want to be included.

1. (5 points) Cluster the Yelp data using k-means.

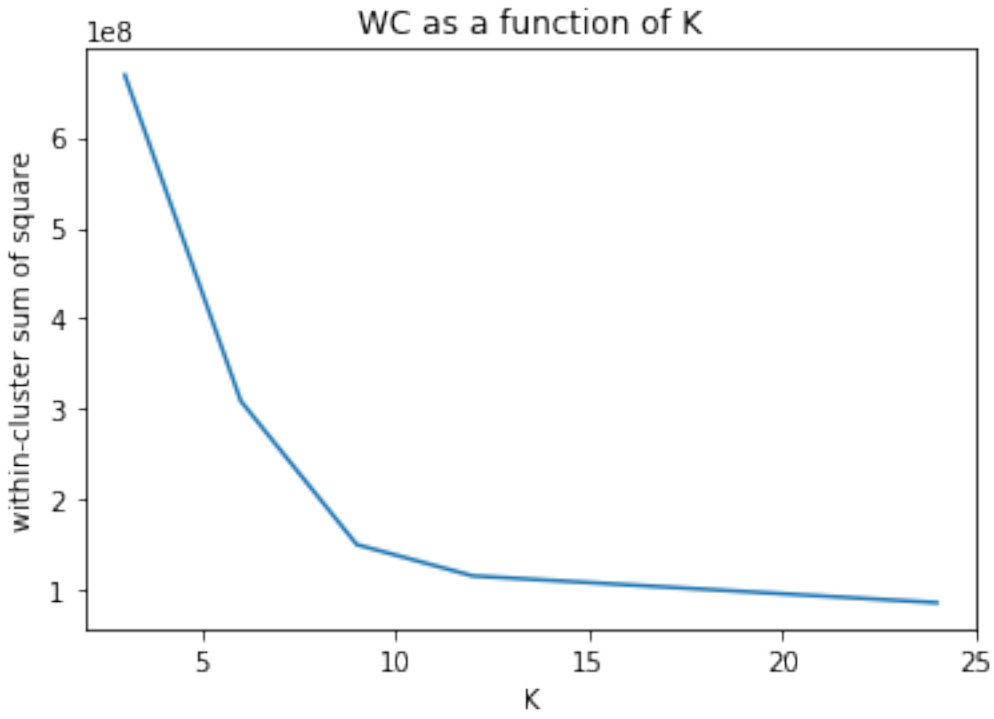
   (a) Use a random set of examples as the initial centroids.

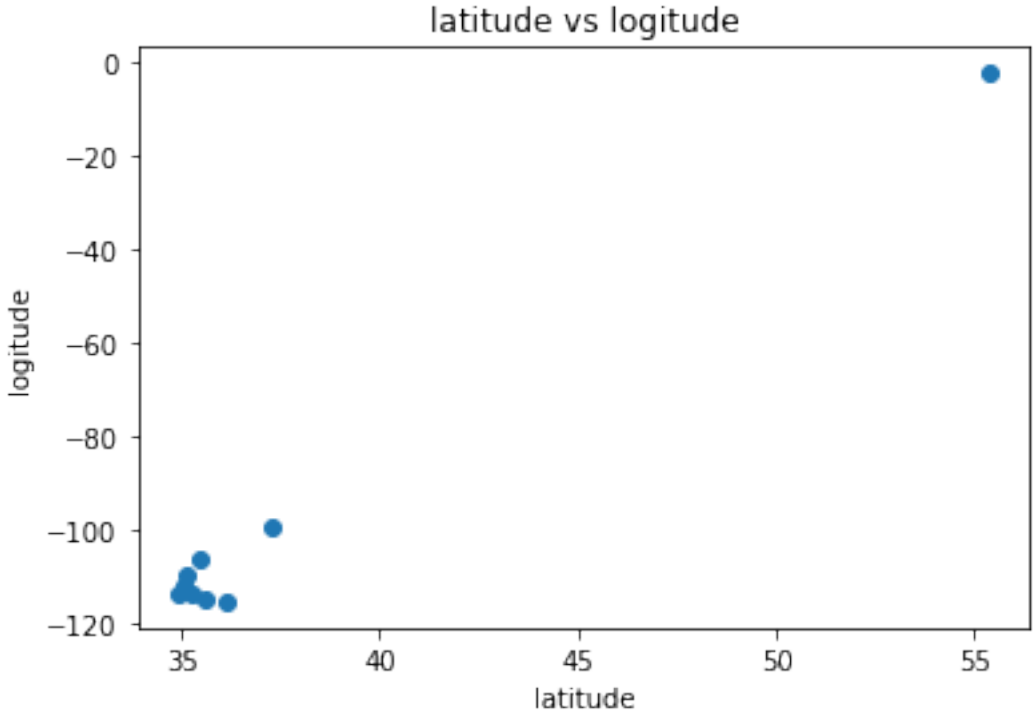
   (b) Use values of K = [3,6,9,12,24].

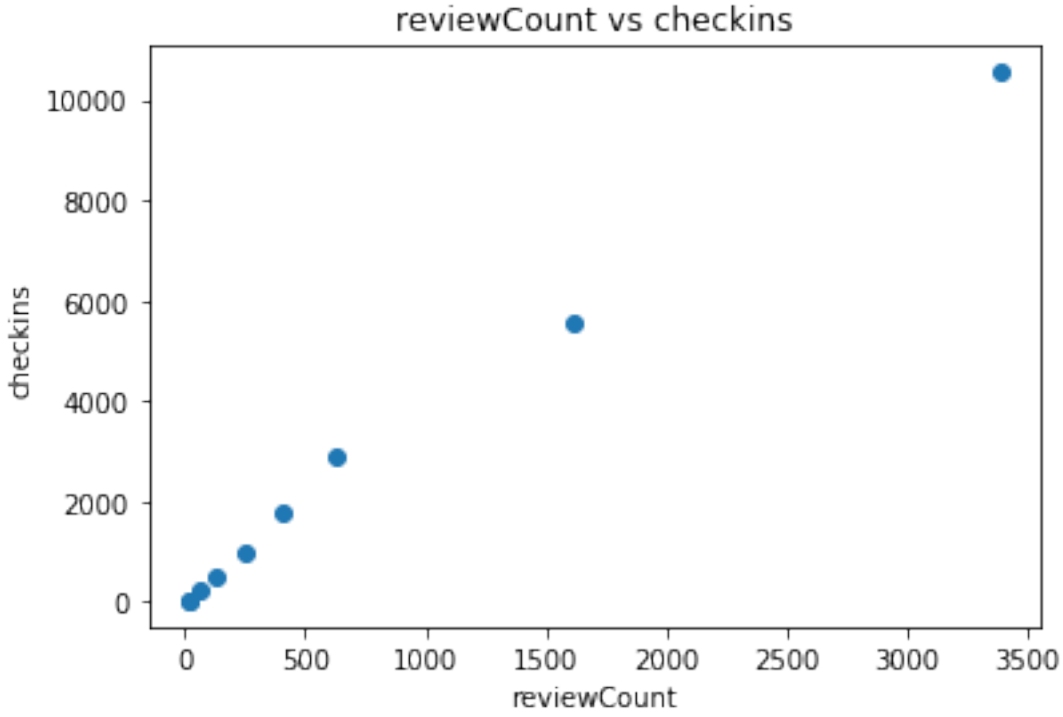
   (c) Plot the within-cluster sum of squares (wc) as a function of K.

   (d) Choose an appropriate K from the plot and argue why you choose this particular K.

   (e) For the chosen value of K, plot the clusters with their centroids in two ways: first using latitude vs. longitude and second using reviewCount, checkins. Discuss whether any patterns are visible.

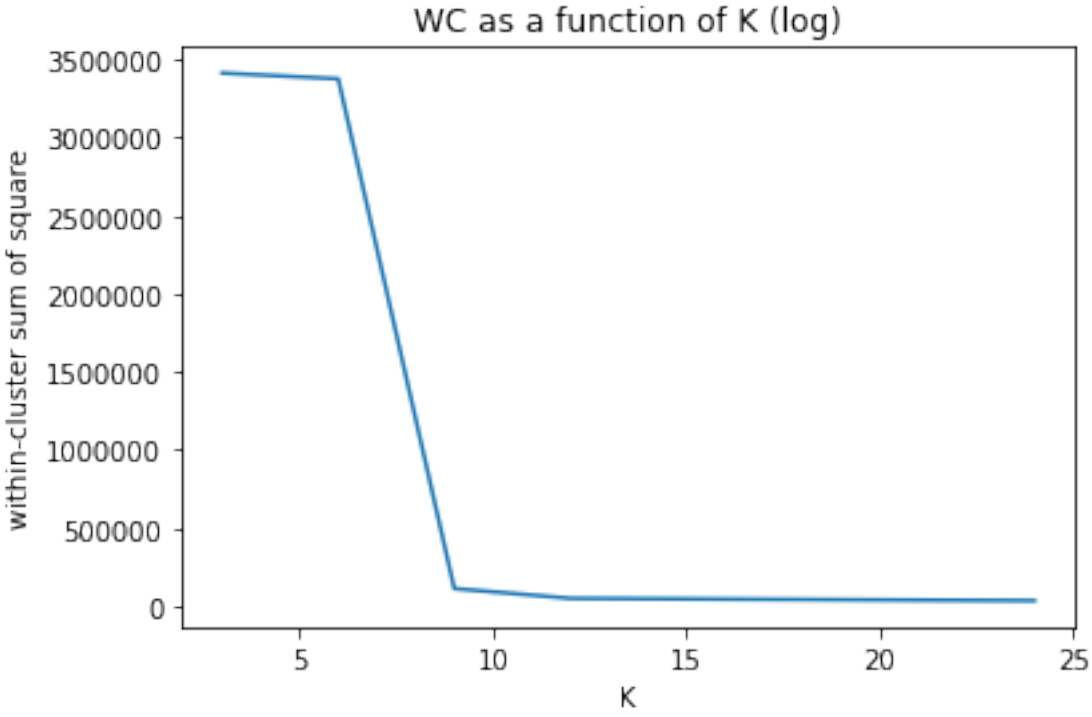


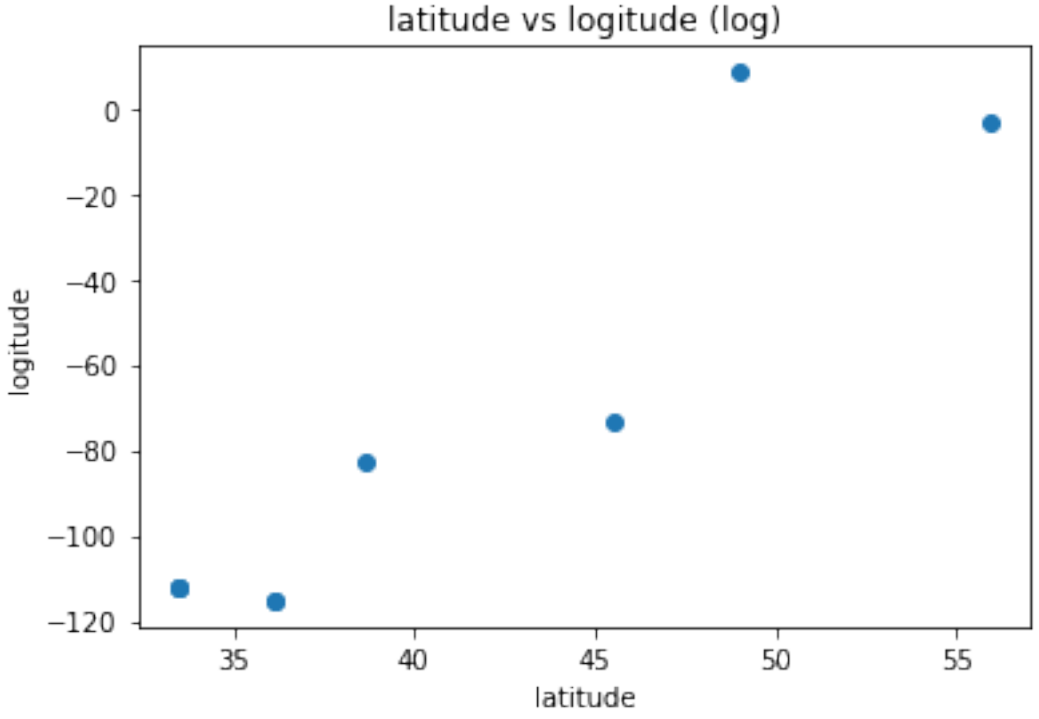


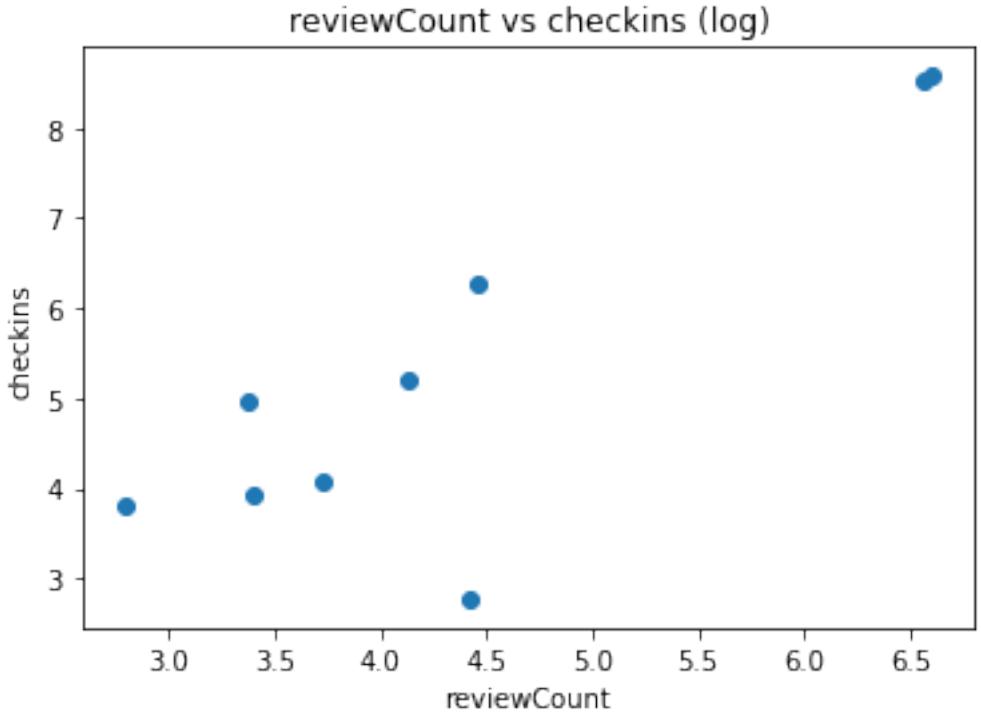


I chose 9 for the K value because, the point at 9 declines the most. For the latitude vs longitude plot, I see 8 point are very close but 1 point a little far away from the others. For the reviewCount vs checkins plot, I see there is a linear relationship. As reviewCount increases, checkins increases.

2. (5 points) Do a log transform of reviewCount, checkins. Describe how you expect the transformation to change the clustering results. Then repeat the analysis (1). Discuss any differences in the results.

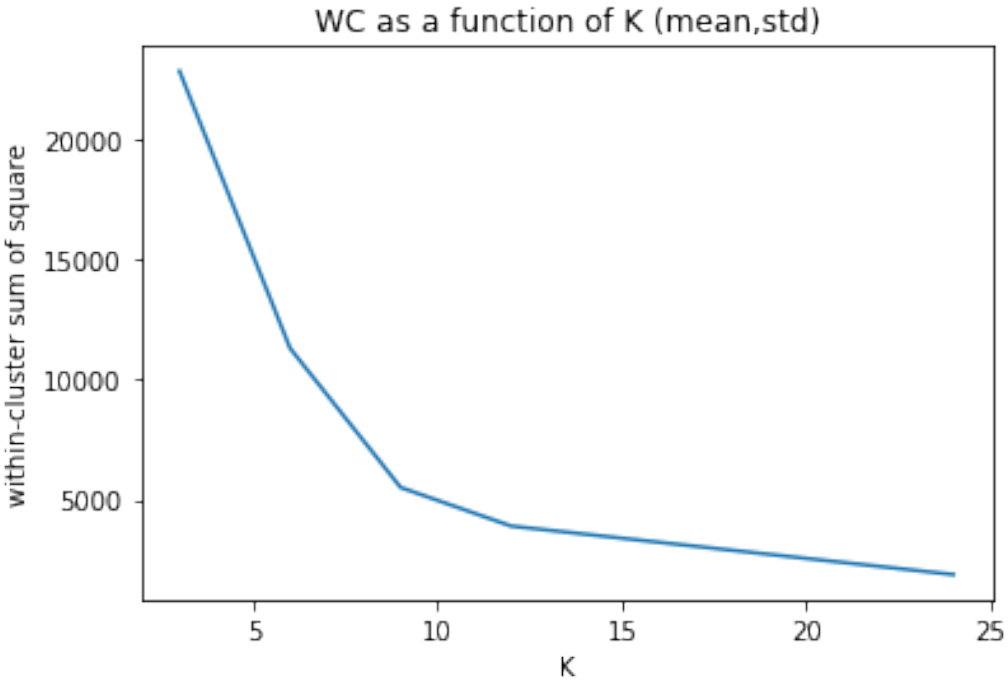


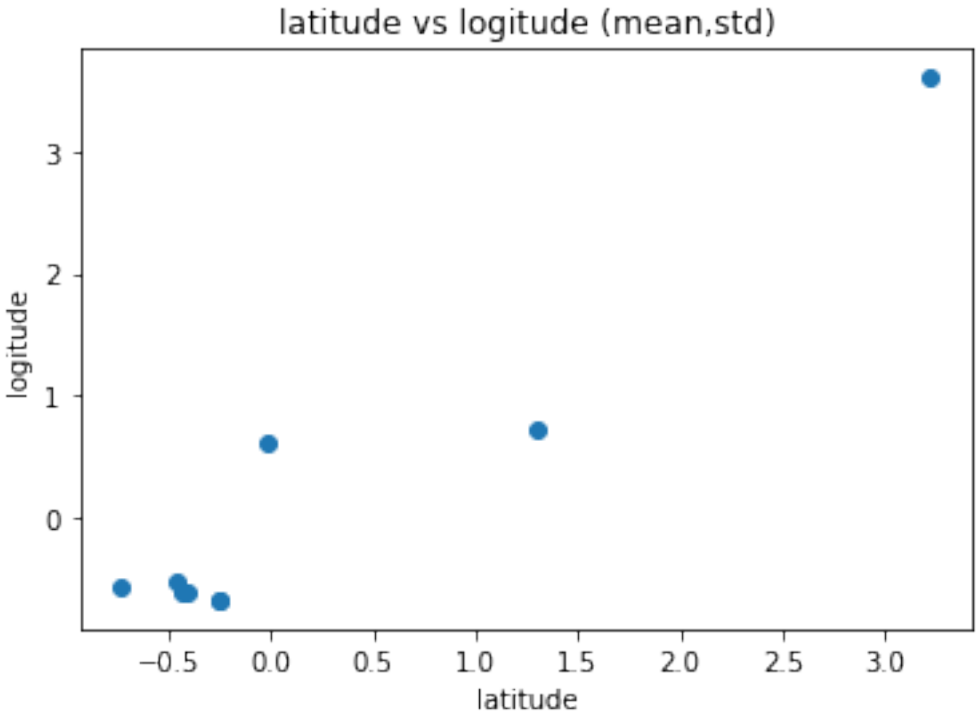


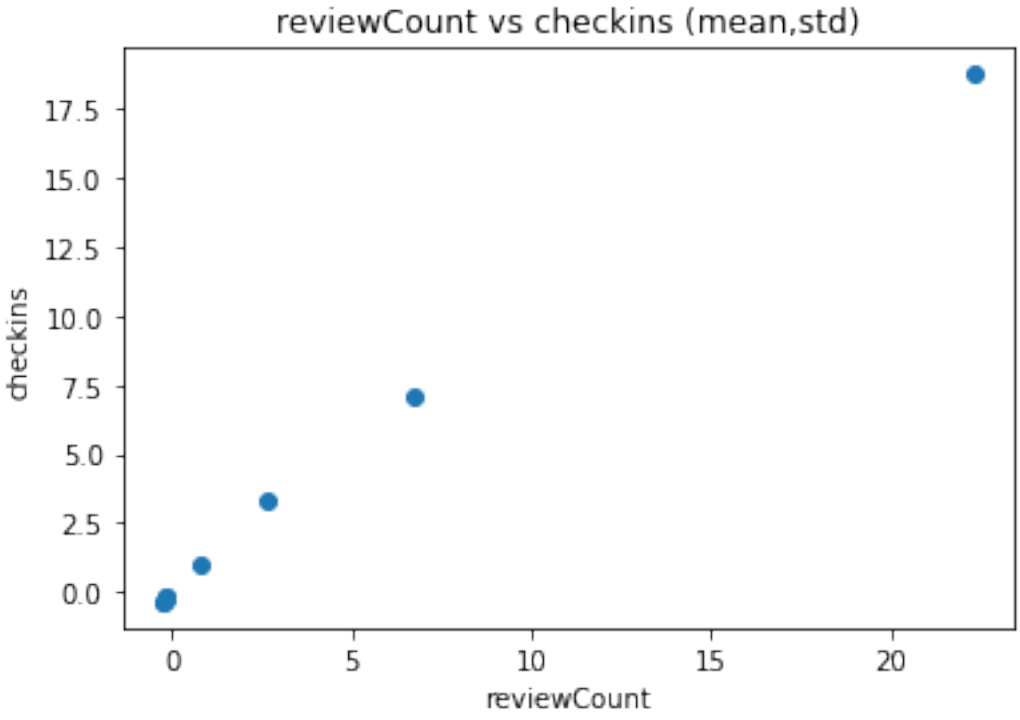


I expected the transformation will change the slope of WC vs K plot and will not change the pattern for the other plots. I chose 9 for the K value because, the point at 9 declines the most. I see the plots have a little bit linear x axis vs y axis. The results are little bit different from 3.1 results. For the first plot, the slope is steeper than 3.1 plot. For the second and third plots, they are more scattered than 3.1 plots.

3. (5 points) Transform the four original attributes so that each attribute has mean = 0 and stdev = 1. You can do this with the numpy functions, numpy.mean() and numpy.std() (i.e., subtract mean, divide by stdev). Describe how you expect the transformation to change the clustering results. Then repeat the analysis (1). Discuss any differences in the results.

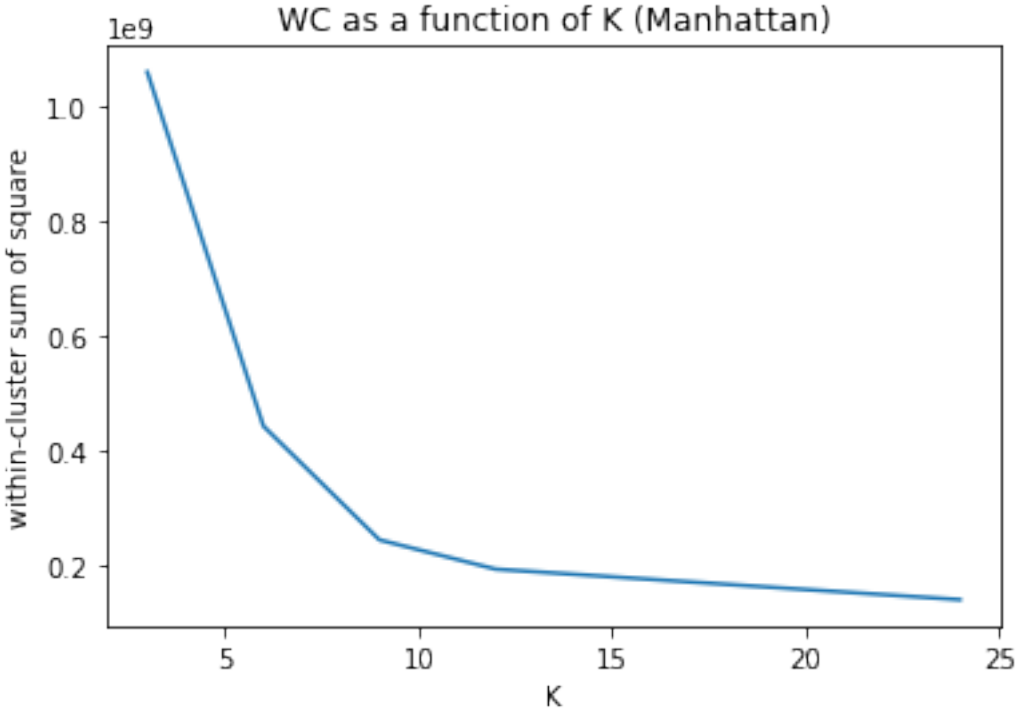


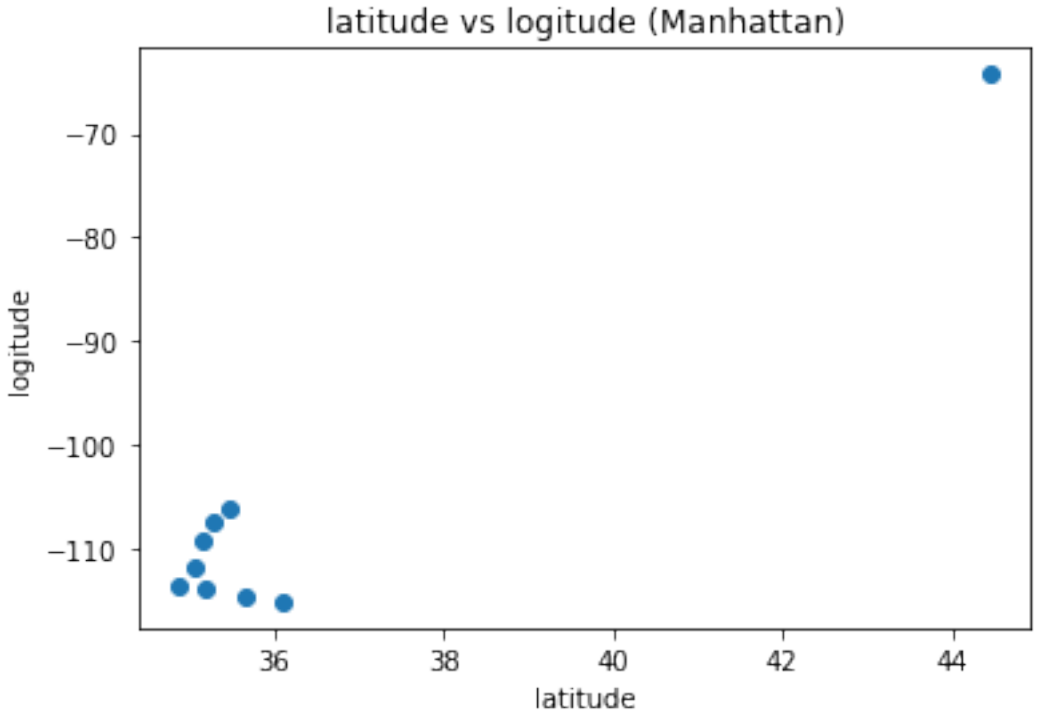


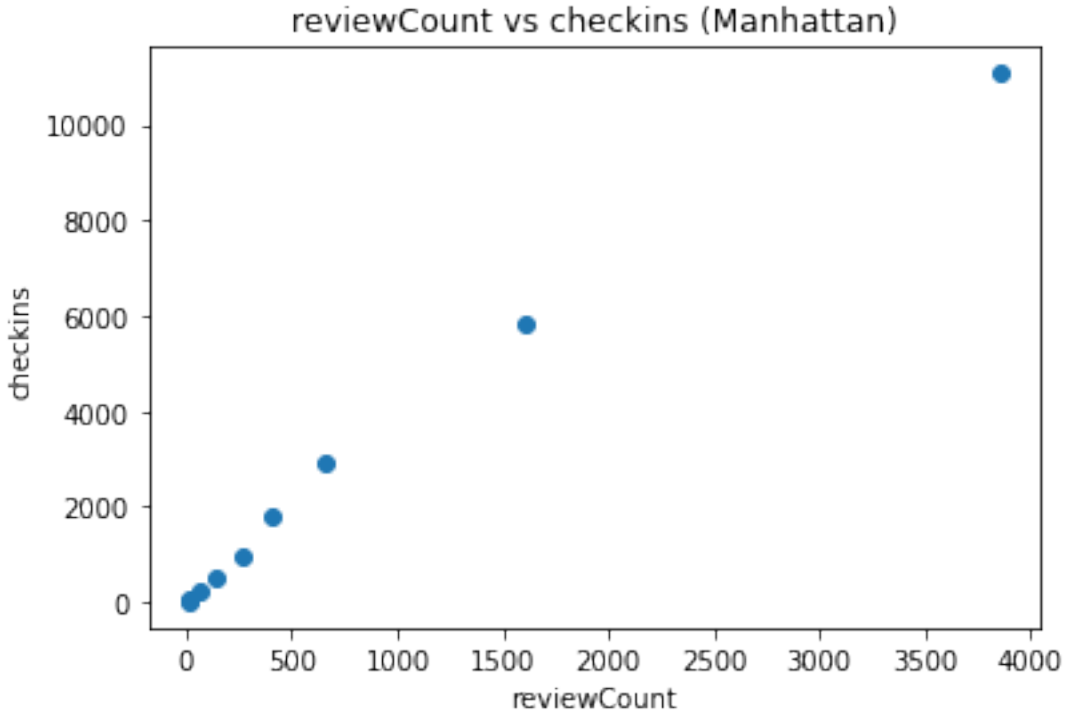


I expected the transformation makes the score function lower and the other result are same. I chose 9 for the K value because, the point at 9 declines the most. The patterns of the plots are same. The score function values are smaller than 3.1.

4. (5 points) Use Manhattan distance instead of Euclidean distance in the algorithm. Describe how you expect the change in the clustering results. Then repeat the analysis (1). Discuss any differences in the results.

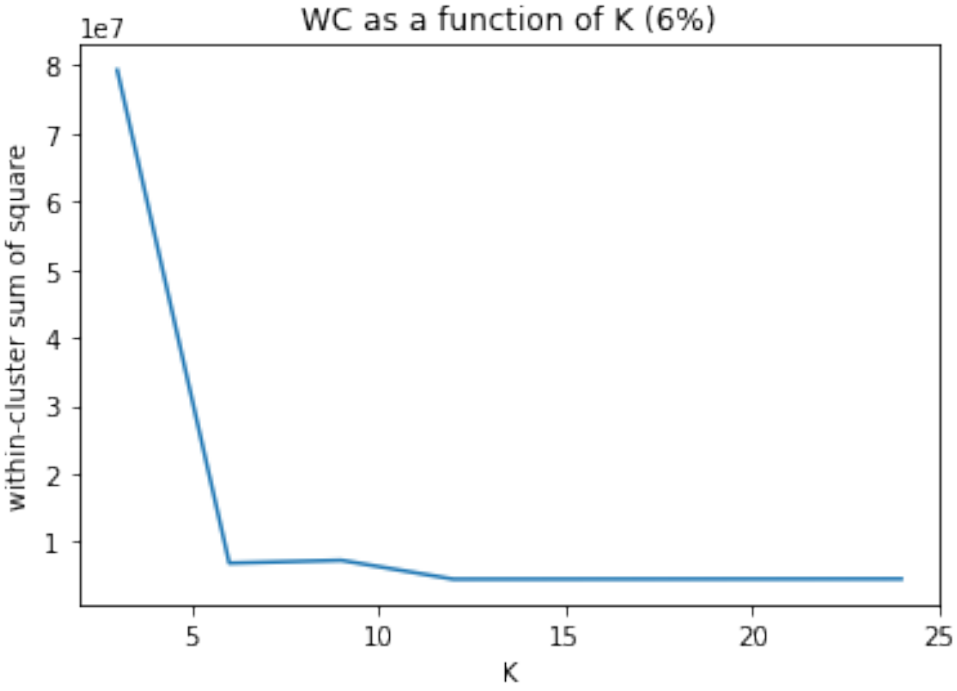






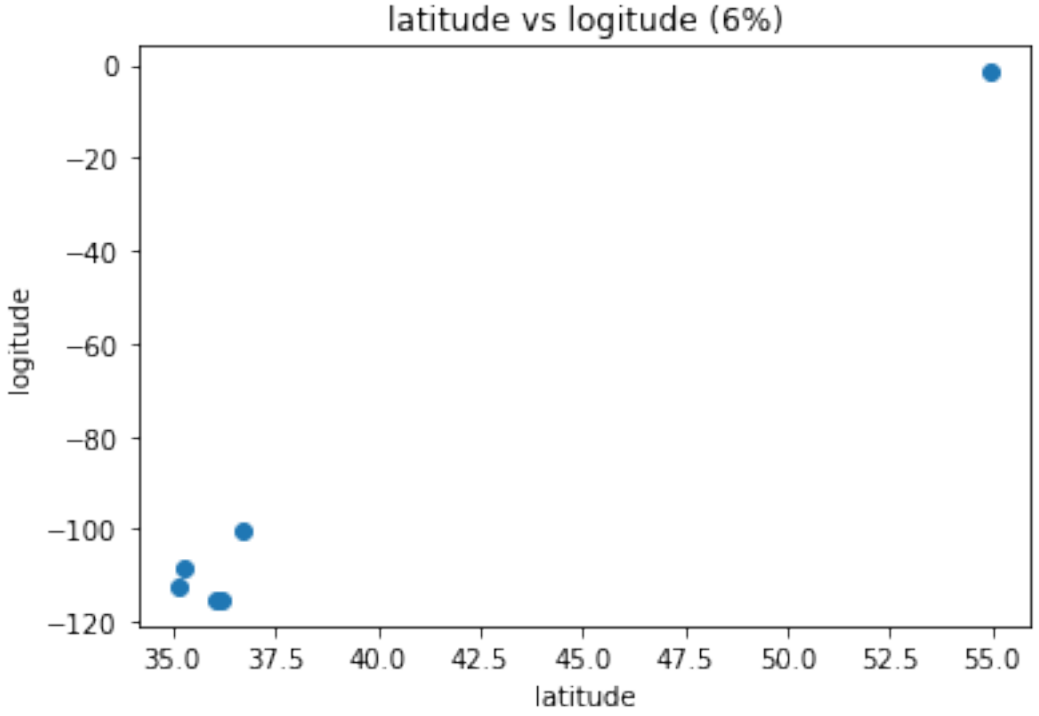
I expected the score function value will be bigger than 3.1 and otherwise the plots are almost same with 3.1. I chose 9 for the K value because, the point at 9 declines the most. The plots are almost same but the score function value is bigger than 3.1.

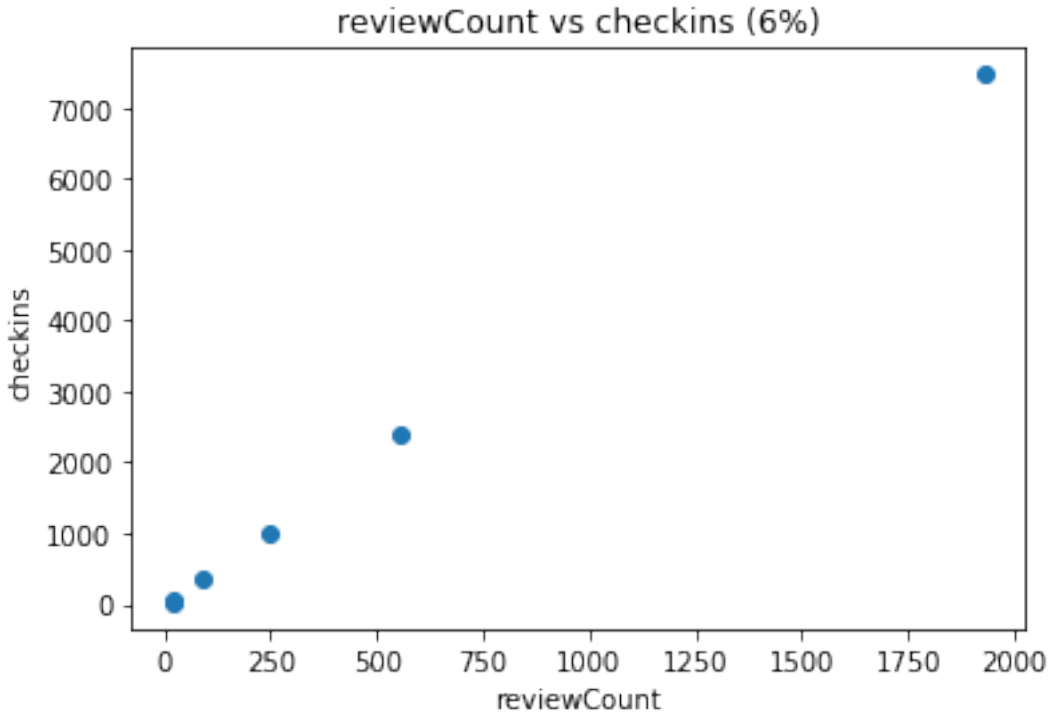
5. (5 points) Take a 6% random sample of the data. Describe how you expect the downsampling to change the clustering results. Then run the analysis (i) five times and report the average performance. Specially, you should use a single random 6% sample of the data. Then run 5 trials where you start k-means from different random choices of the initial centroids. Report the average wc when you plot wc vs. K. For your chosen K, determine which trial had performance closest to the reported average. Plot the centroids from that trial. Discuss any differences in the results and comment on the variability you observe.



Closest to average trial.

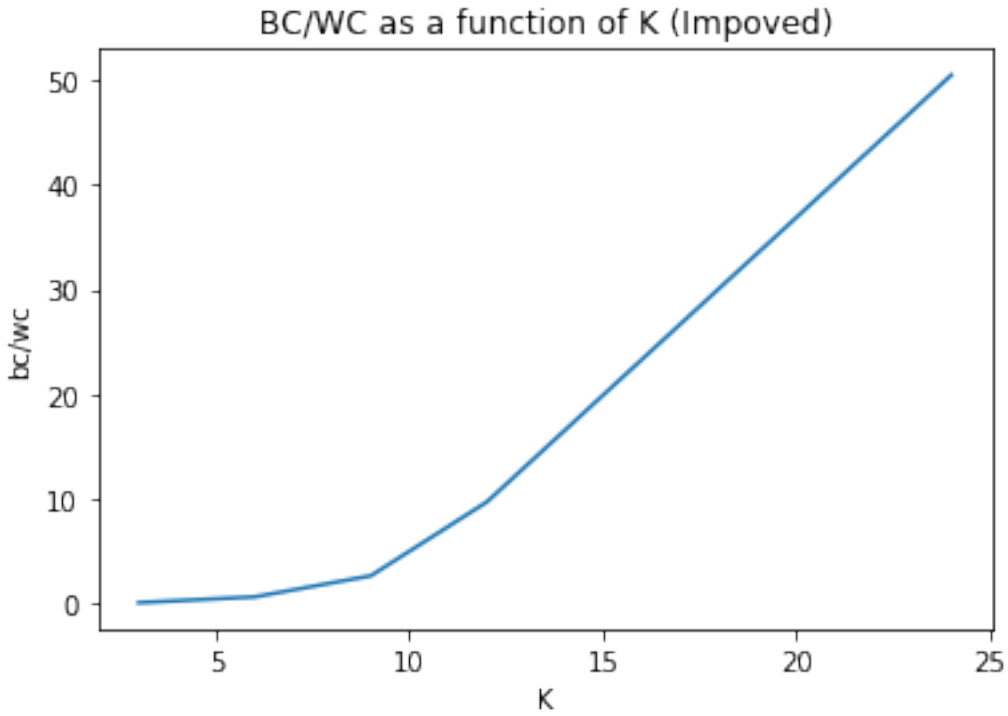
```
WC-SSE=7243929.389874441
Centroid1=[35.271701156296295, -108.13164245212963, 91.49074074074075, 353.9074074074074]
Centroid2= [35.139262172, -112.6177728792, 247.72, 1014.6]
Centroid3= [54.95877598342857, -1.5441710164, 17.942857142857143, 23.65714285714286]
Centroid4= [36.04279791, -115.153225, 1930.0, 7476.0]
Centroid5= [36.72303766852113, -100.28798563044015, 20.714788732394368, 51.651408450704224]
Centroid6= [36.15718139714286, -115.21834014285716, 552.7142857142857, 2411.4285714285716]
```





I expected the downsampling will make the results have different behavior plot of wc vs K. I chose 6 for the K value because, the point at 6 declines the most. The patterns are same as 3.1 plots but the chosen K value is different from 3.1.

6. (5 points) Improved score function. In this case, you will use the score function you proposed in Theory (2.1) question 5. Using the best configuration from Questions 1-5, plot the results of your score function for K = [3, 6, 9, 12, 24], and compare the results to the appropriate algorithm from Question 1-5.



I used Q3 configuration as the best one because by transforming the data mean to 0 and std to 1, the data is well distributed and optimized to find similarity each instance.
The plot has positive linear relationship while the other plots have negative linear relationship. Also, the score function value is much smaller than the other plots.