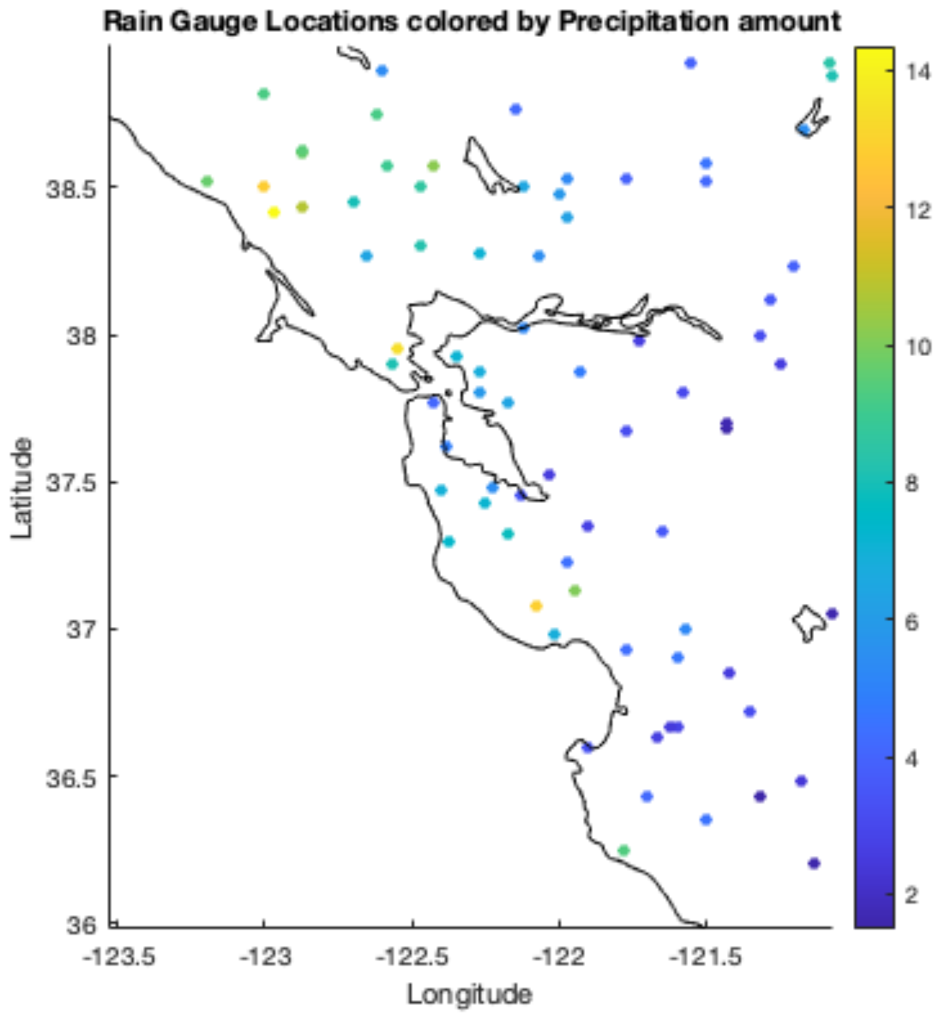
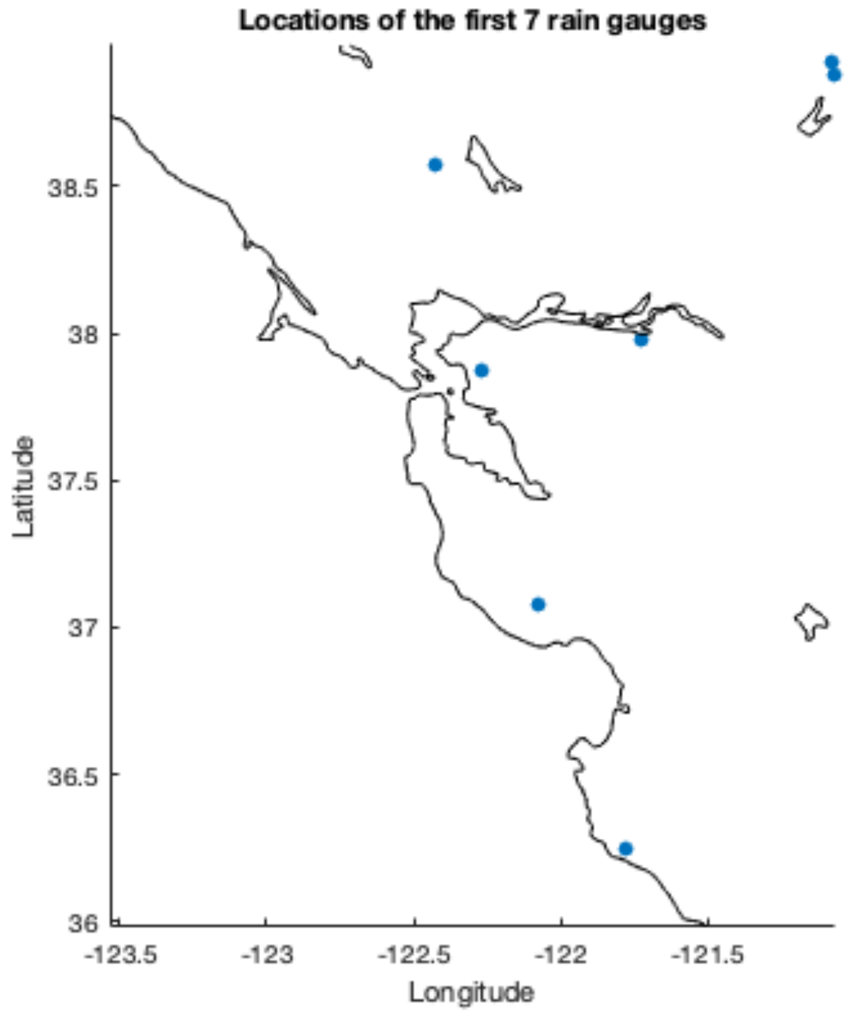
Pre Question 1

```
% Before the first Question
scatter(bayprecip(:,2),bayprecip(:,3),25,bayprecip(:,4),'filled'),colorbar;
hold on;
plot(baycoast(:,1), baycoast(:,2),'k-');
axis image;
hold off;
```



Question 1

```
% Question 1
xy = bayprecip(1:7,2:3); id = bayprecip(1:7,1);
% plot the 7 rain gauge locations
scatter(xy(:,1),xy(:,2),'filled');
hold on;
plot(baycoast(:,1), baycoast(:,2),'k-');
axis image;
hold off;
```




```
% compute the euclidean distance matrix
D = squareform(pdist(xy));
% This distance matrix is size 7 x 7
D_3_6 = D(3,6);
% The distance between station 3 and 6 is 1.587.
D(1,1) = 10;
D(2,2) = 10;
D(3,3) = 10;
D(4,4) = 10;
D(5,5) = 10;
D(6,6) = 10;
D(7,7) = 10;
[distMin, idMin] = min(D(5,:));
```

The station with the closest distance to station 5 is station 6 with a distance of 0.8125.

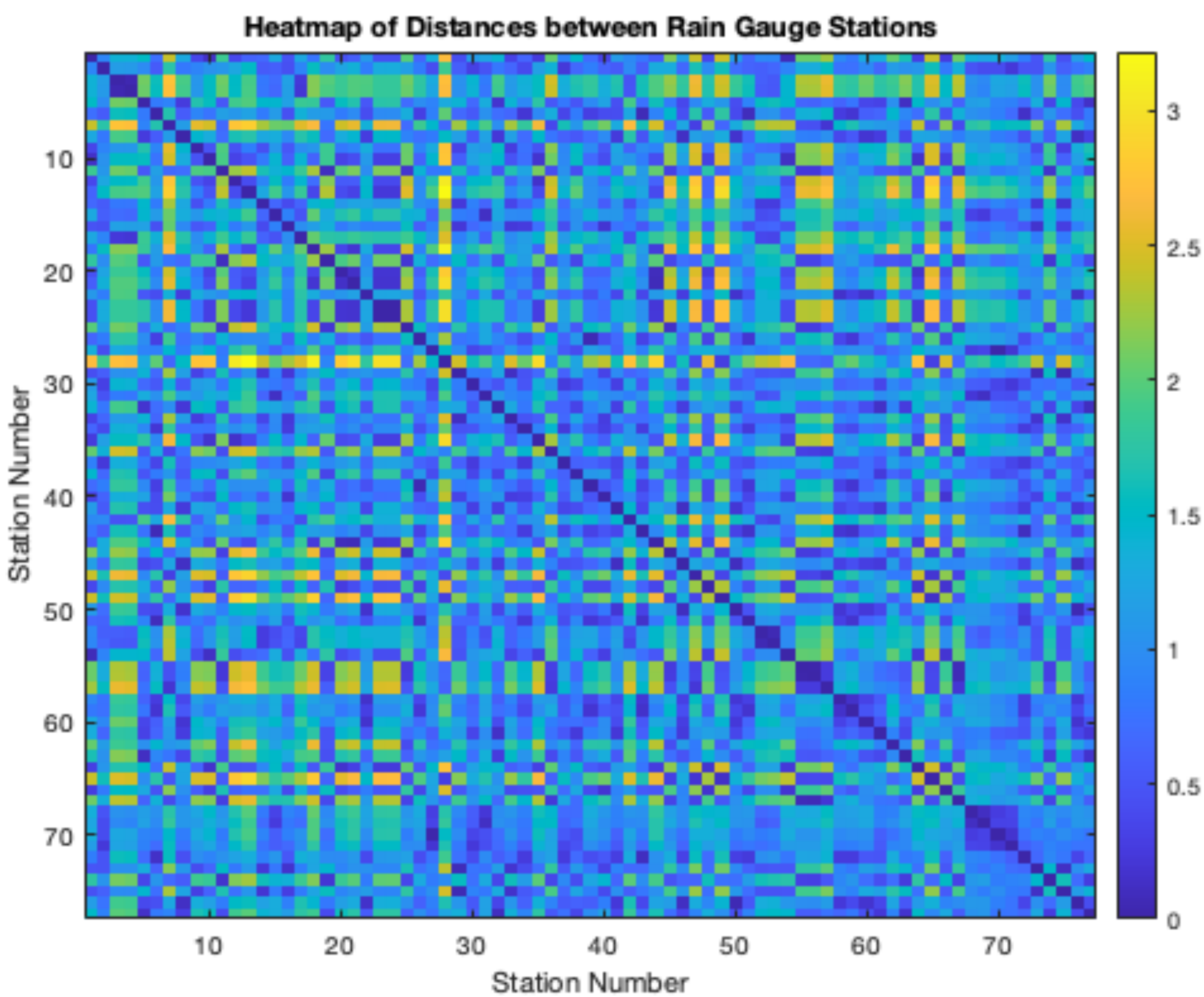
Question 2

```
% Question 2
D2 = pdist2(xy,baycoast);
% D2 is of size 7x2072.
distMin2 = min(D2,[],2);
[dd,ii] = min(distMin2);
```

Station 2 is the closest to the coastline with a distance of 0.0174.

Question 3

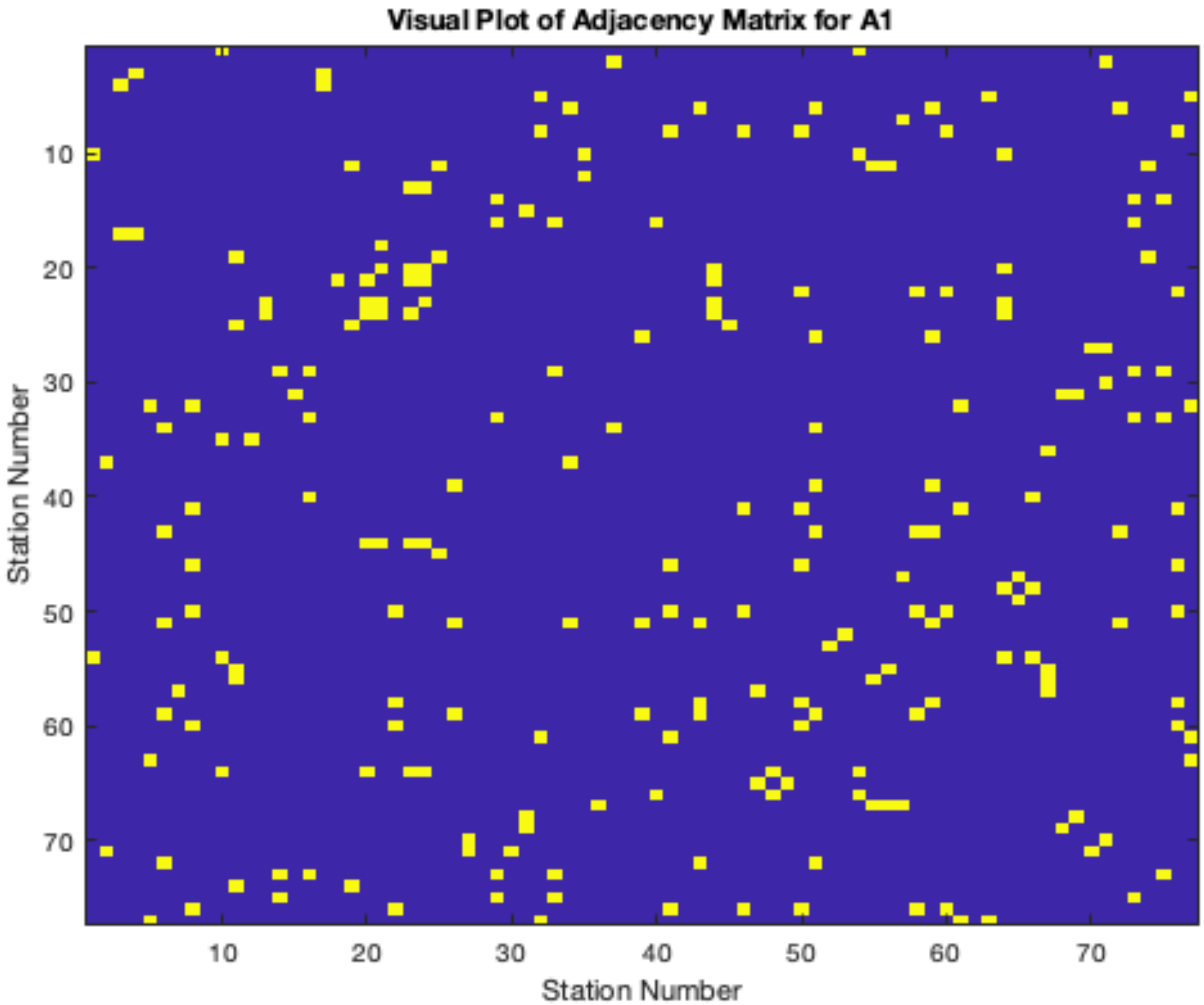
```
% Question 3
D = squareform(pdist(bayprecip(:,2:3)));
imagesc(D);
colorbar;
```



I see the largest yellow patch (i.e. the parts where the distance between the stations is the farthest) between stations 40 – 60 and the stations 1-about 20 (bottom left of the graph). Although it isn't consistently yellow, it just has the largest yellow pathc. Similarly, the largest dark patch happens with the stations 60-70 with 60-70 (i.e. the bottom right of the graph. These indicate that the points are closer to each other.

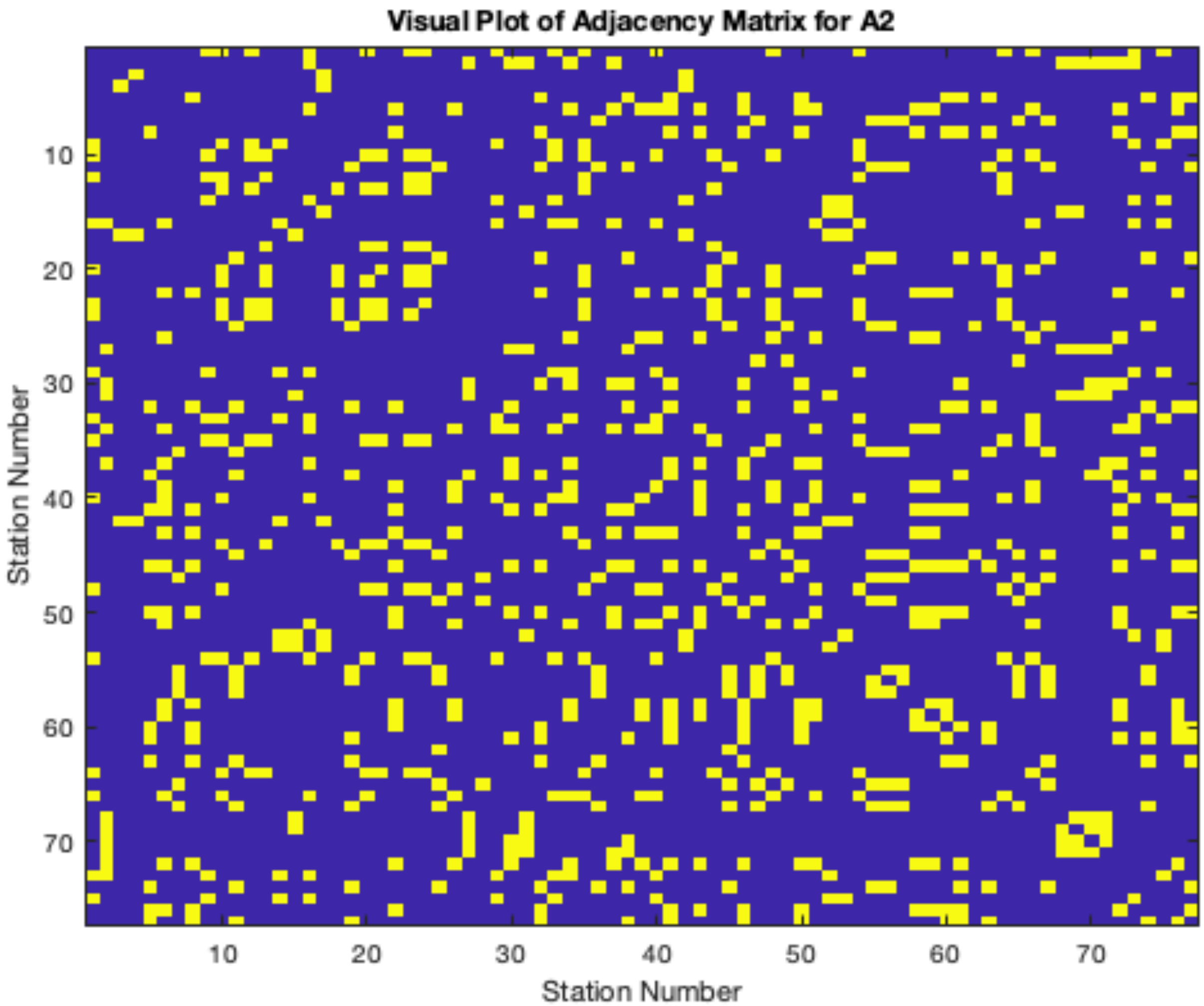
Question 4: A1

```
% Question 4
N = length(D);
indDiag = 1:N+1:N*N;
A1 = D < 0.25;
A1(indDiag) = 0;
imagesc(A1);
```



Question 4: A2

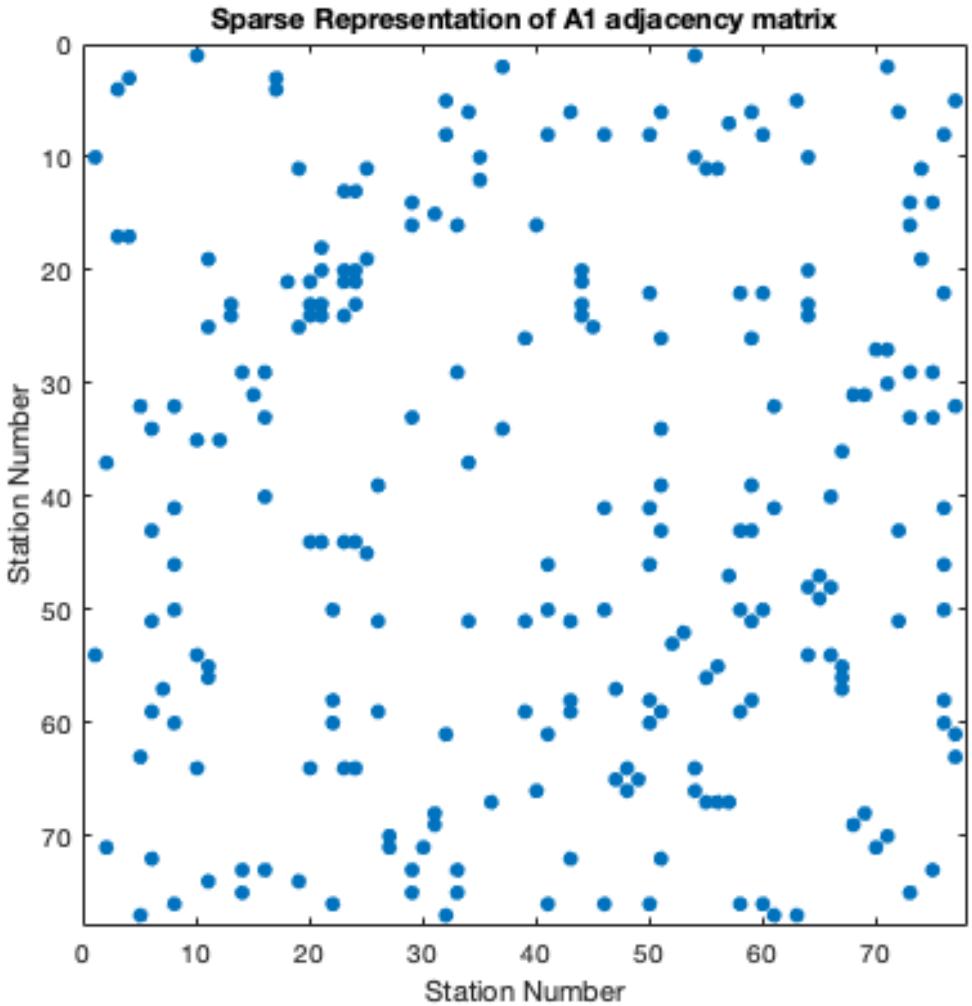
```
A2 = D < 0.5;
A2(indDiag) = 0;
imagesc(A2);
```



Just as I would expect, there would be more adjacent points in A1 than in A2 because the criteria for A1 was more filtered than that of A2. For A1, we see one main cluster of adjacencies between stations 20 and 30. However for 2, there are much more sporadic clusters throughout the heatmap, one of the denser ones on the bottom right representing the adjacency between stations 65–70.

Question 5

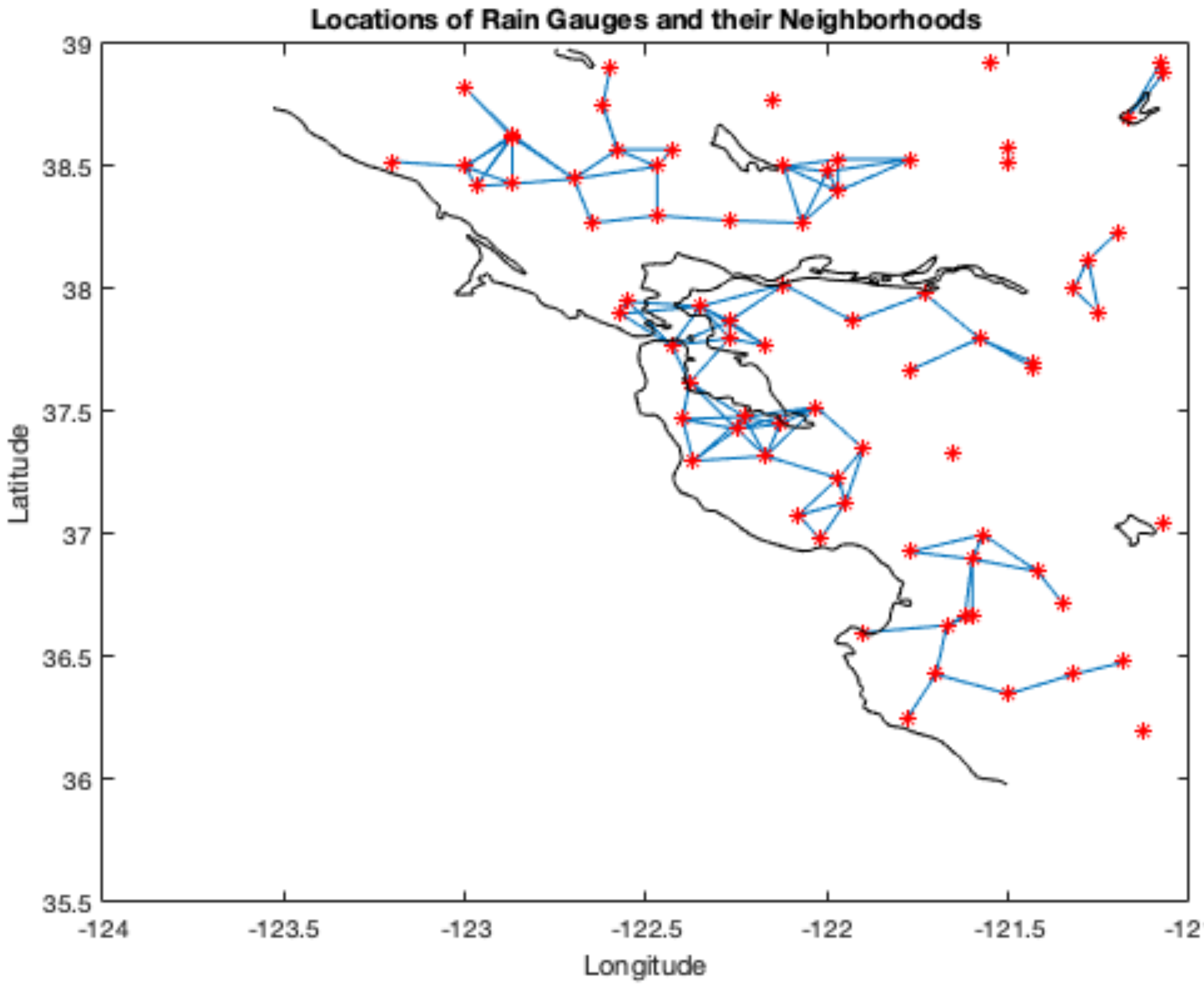
```
% Question 5
A1S = sparse(A1);
spy(A1S);
```



I feel like this representation of adjacency as points and not a heatmap like from the previous question is much easier to detect. We can see approximately how many stations there are in certain clusters as compared to the heatmap where we had to look deeper into the ranges on the axes to estimate this value.

Question 6

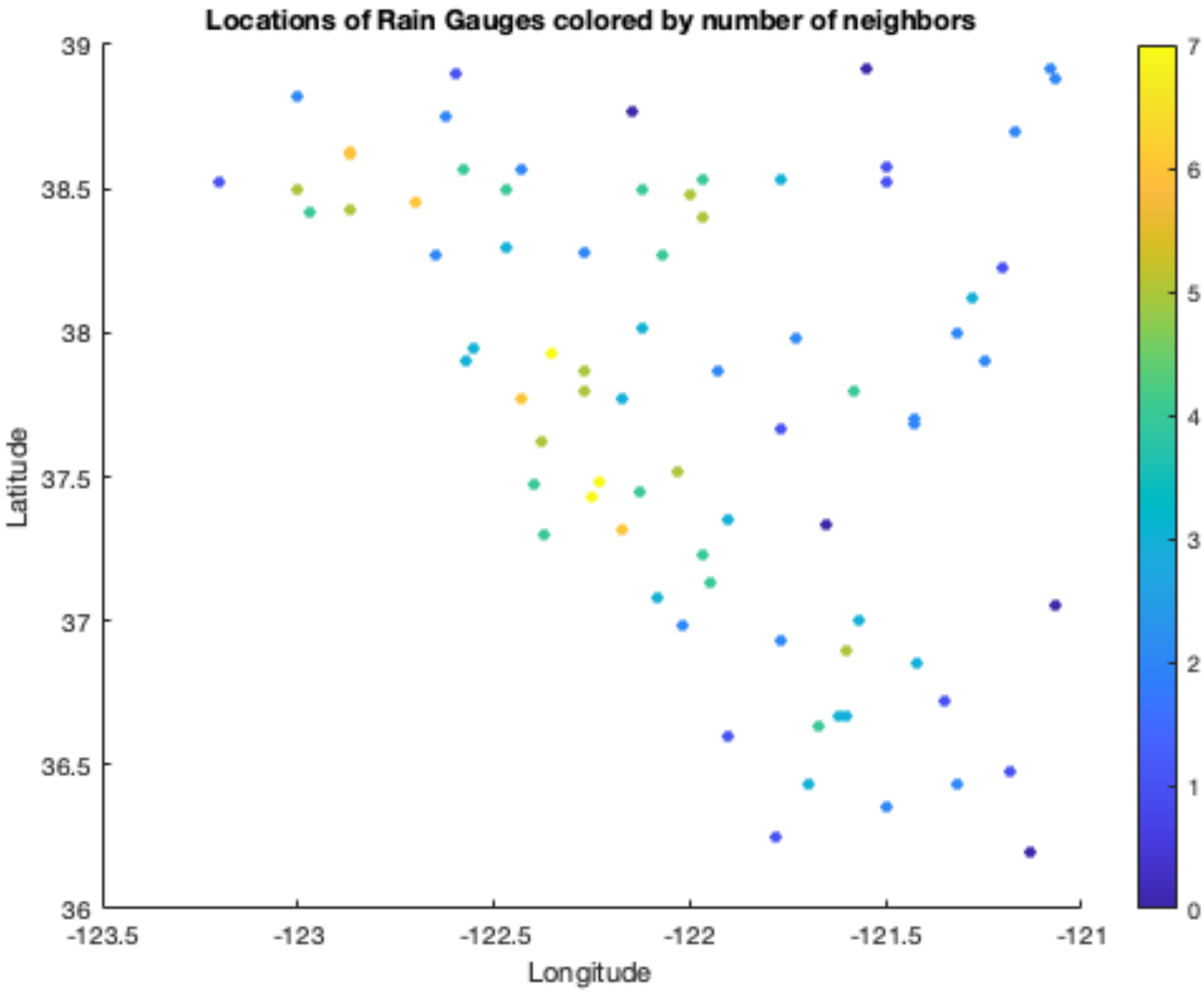
```
% Question 6
x = bayprecip(:,2);
y = bayprecip(:,3);
gplot(A1S,[x y]);
hold on;
plot(x,y,'r*');
plot(baycoast(:,1), baycoast(:,2),'k-');
hold off;
```



It looks like this plot is connecting the neighboring rain gauges by lines. Most of the classified neighborhoods are closer and bigger towards the coast. Meanwhile the points more inland either have smaller neighborhoods or stan on their own. I think this is representative with population as the Coastal Bay Area is more densely populated than the more inland parts of the Bay Area. I think this graph is a bit useful; however, I wish the neighborhoods were represented as polygons like clusters instead of lines because it looks a little messy.

Question 7

```
% Question 7
nNeigh1 = sum(A1,2);
scatter(x,y,25,nNeigh1,'filled');
colorbar;
```



use a code like what we used in question 1 to find the station with the greatest number of neighbors

```
[neighhigh, idneigh] = max(nNeigh1);
neighhigh; % 7 neighbors
idneigh; %50
```

Using the code above to find the station that has the greatest number of neighbors, I found that station 50 is the most important rain gauge. however, when I look at the plot, there are 3 stations that also have the highest number of neighbors (which in this case it's 7).

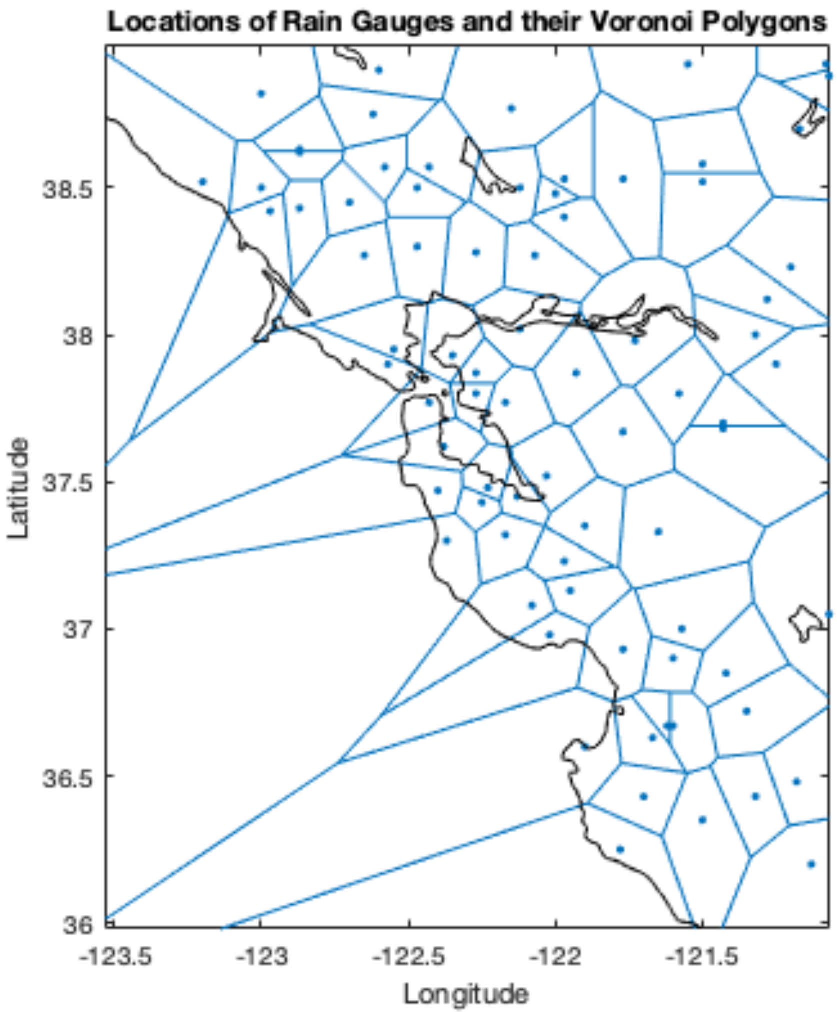
Question 8

```
% Question 8
N = 77;
rowSum2 = sum(A2,2);
T = repmat(rowSum2,1,N);
A2std = A2./T;
verify = sum(A2std, 2);
```

Using the code on the last row, we can verify that the row totals is 1. row-standardized adjacency is useful because it sorts of acts like an adjacency matrix, but each positive value is relative according to it's adjacency.

Question 9

```
% Question 9
voronoi(x,y);
hold on;
plot(baycoast(:,1), baycoast(:,2),'k-');
axis image;
hold off;
```



The polygons closer to the Bay delta (i.e. the region where the water extends inland) have the smallest polygons. As the distance from the delta increases, so does the size of the Voronoi polygons. The problem with Voronoi polygons is that since they are partitioned in the entire region of the image/map, lot of the points on the coastlines are included in polygons that extend out into the ocean. This is unrealistic because there aren't any rain gauges in the ocean. The area covered by these polygons can be a good indication of adjacency because we can regard points within these regions as being closer to each other than any other point in the samples space.