Large-Scale Data Mining: Models and Algorithms
ECE 232E Spring 2018

Project 5
Graph Algorithms

Qinyi Tang (204888348)
Shuo Bai (505032786)
Jinxi Zou (605036454)
Xuan Hu (505031796)
6/15/2018

1 Stock Market

Answer for Question 1:

In this part, we calculate the return correlation. First, we extracted the closing price from the csv file. For most cases, the stock has the 765 prices of different days, but there are some exceptions that have less than 765 prices. They are “CFG”, “CSRA”, “FTV”, “HPE”, “KHC”, “PYPL”, “QRVO”, “SYF”, “UA”, “WLTW”, “WRK”. We discard these stocks. Thus, there are 494 stock in total.

After we load the closing price of the stocks, we calculated the return based on the defination in the statement.Although there are some gaps in the dates, but we still can assume the data is continuous, because the missing dates are weekends, contributing nothing to the price of the stock. After normalized the return, we get the correlation in every two stocks.

Theoretically, the lower and upper bound of ρ_{ij} is -1 and 1. If ri(t)=rj(t), then the correlation is 1, and if ri(t)=rj(t), then the correlation is -1. The negative correlation means that the return pattern in these two stock has little in common. We validate the lower and upper bound using our data. The minimum correlation is -0.19857, and the maximum correlation is 0.98842.

The reason for using the log-normalized return instead of regular return is that log transformation can decrease the variability of data and make data conform more closely to the normal distribution. We used the log(1+x) instead of log(x) because in the return data, there are some values that is negative.

Answer for Question 2:

In this part, we used the equation in the statement to calculate the weight from the correlation. To build the graph, we used the networkx package in python. We add the nodes as the stock names, and add the edges with the weight we calculated. There are 494 nodes, and 121771 edges in the graph. We tried to plot the graph as shown in Figure 1, but with density of edges, it is not suitable to view.

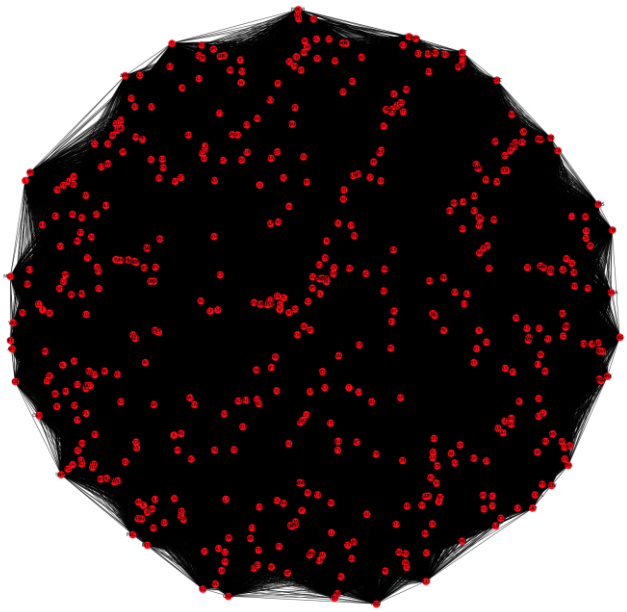


Figure 1. The correlation graph

Figure 2 is the degree distribution of the correlation graph. From the definition of the correlation graph, there an edge between every two nodes. Thus, for each node, it has the degree of 504. Correspondingly, in the degree histogram, there is only one bar at degree of 504 with frequency of 1.0.

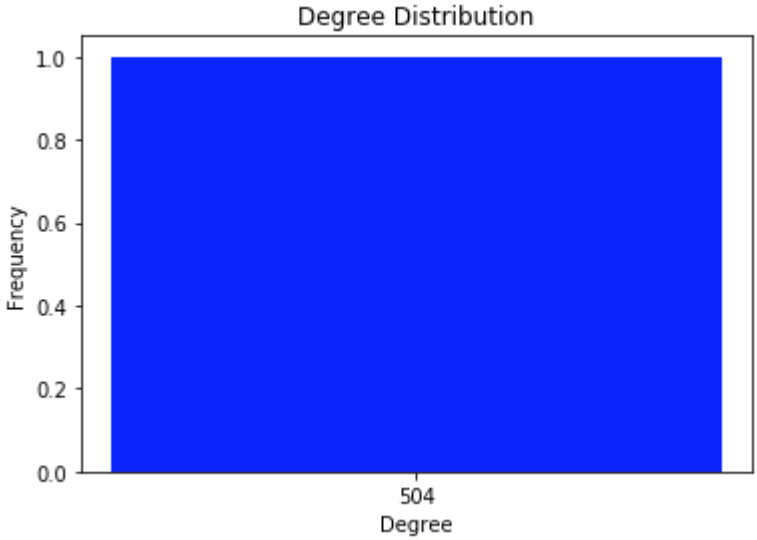


Figure 2. Degree distribution of the correlation graph

Figure 3 is the distribution of the edge weights.

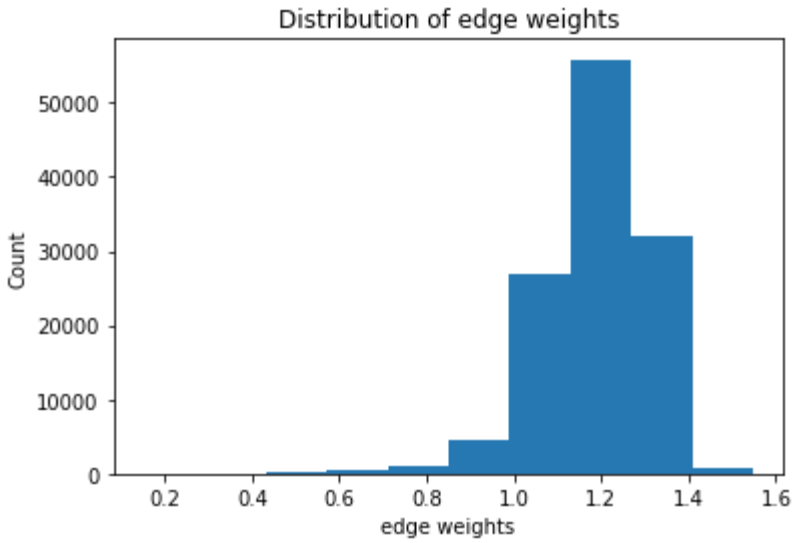


Figure 3. Distribution of un-normalized edge eights

Answer for Question 3:

We add the sector information as the properties to the nodes. To extract MST, we used the function provided by the networkx package. (T = nx.minimum_spanning_tree(G)). Then we plot the MST with the nodes color-coded. To color-code the nodes with the sector information, we mapped the sector to the following number. The color each number represent can be found in the color bar in Figure 5.

```
{'Consumer Discretionary': 0,
 'Consumer Staples': 1,
 'Energy': 2,
 'Financials': 3,
 'Health Care': 4,
 'Industrials': 5,
 'Information Technology': 6,
 'Materials': 7,
 'Real Estate': 8,
 'Telecommunication Services': 9,
 'Utilities': 10}
```

The following two figure is the same MST. Figure 4 is with labels on its nodes, we removed the labels on its nodes in Figure 5 for a better and clearer view.

From the Figure 5, we can easily see that there are some patterns in the MST. The nodes with the same sector are tend to be connected together. This is what is called vine cluster. In some nodes with the same sector, they are grouped as a cluster. Only two nodes in this cluster is connected to other nodes with other sectors, like the nodes with color 4 (light blue) and color 8(orange).

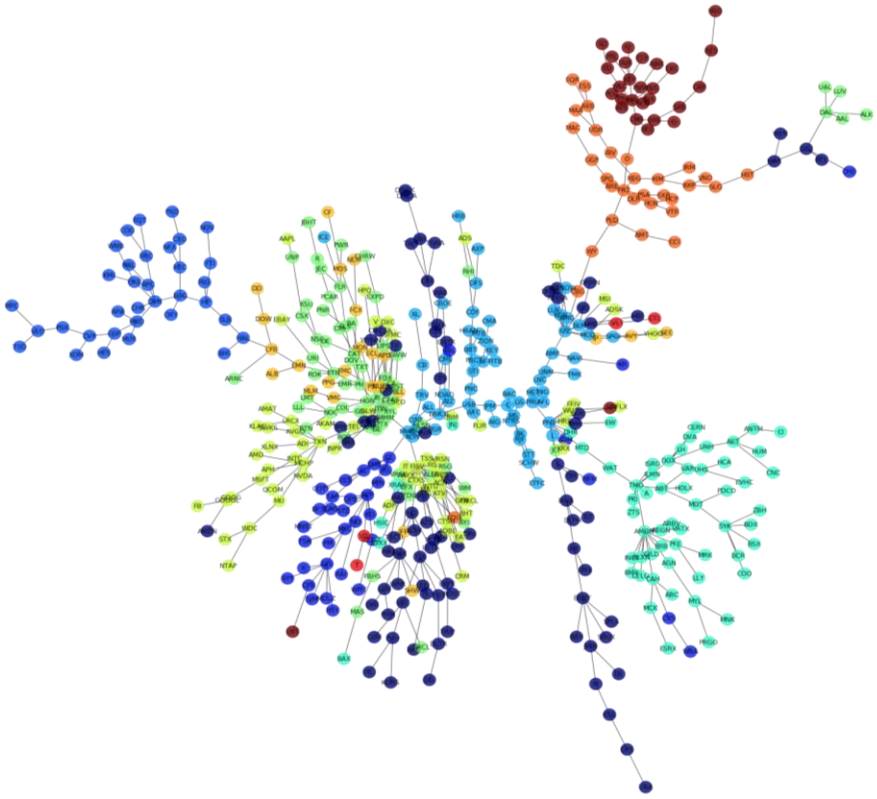


Figure 4. MST with labels on the nodes.

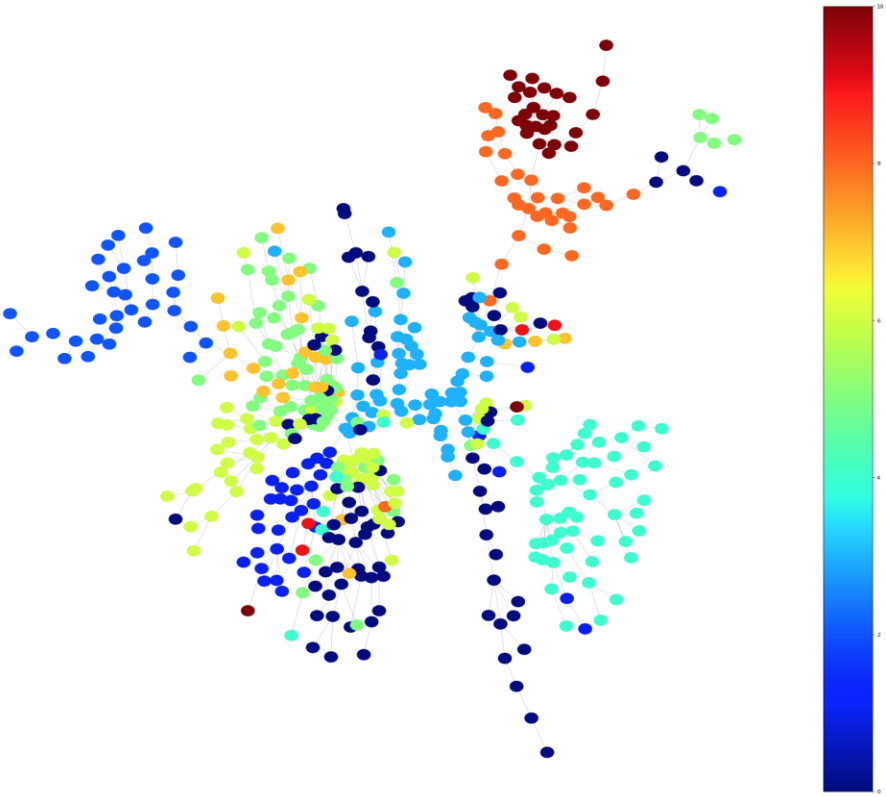


Figure 5. MST without the labels on the nodes

The reason for this kind of pattern is the nodes in the same sectors have more in common, leading to a larger correlation and smaller weight. Intuitively, the stocks in the same sector are probably affected by the same economic effects, and the return of them are probably has larger correlation. From the definition of the weight in the correlation, the larger the correlation is, the smaller the weight is. For MST algorithm, the edges with smaller weights are tend to be picked in the MST. Thus, MST gives a tree that the nodes with the same sector are connected and clustered.

Answer for Question 4:

In this question, we are asked to evaluate the performance of sector predicting using the MST clustering. We iterated all the nodes in the graph to calculate the P score of the each node, and take the average. For P score, we used two method to calculate. The first is using the majority of the neighbors' sector to predict the unknown stock's sector. The P score is calculated as following equation:

$$P(v_i \in S_i) = \frac{|Q_i|}{|N_i|}$$

The second is using the majority of the entire dataset to predict the unknown stock's sector. The P score is calculated as following equation.

$$P(v_i \in S_i) = \frac{|S_i|}{|V|}$$

The alpha we get using the first method is 0.8289, and the alpha we get using the second method is 0.1142. Apparently, the first method is better than the second method.

Because we have mentioned above question, the MST provides clusters of sectors. Thus, a node in the MST is tend to have the same sectors with the majority of its neighbors, it takes the advantage of the pattern of correlation. In contrast, the second method predicts the sector just based on the majority of the entire dataset, without exploring the correlation between the unknown stock with other stocks.

Answer for Question 5:

In this question, the steps are similar with the previous questions, only different with the data we used. We sampled the data by only keeping the data on Mondays and calculated the correlation with the weekly data. First, we explored the data. We find that some prices on Monday are missing because some Mondays are holidays. Thus, we sampled the data on the first day of the week. In practice, we sampled some data are on Tuesdays, but it is still the price of the first day in that week.

After we sampling the data, we build the graph in the same way and extracted the MST as shown in Figure 6. Obviously, there are some differences between this MST and the MST of daily data. Although we can still observe the vine cluster, the vine cluster is not that obvious compared to

the MST of daily data, some nodes with the some sectors are seperated. The possible reason for it is that the weekly closing prices have not that big correlation in the same sector. When the data is downsampled by week, the correlation between two stocks in the same sectors may not be that strong.

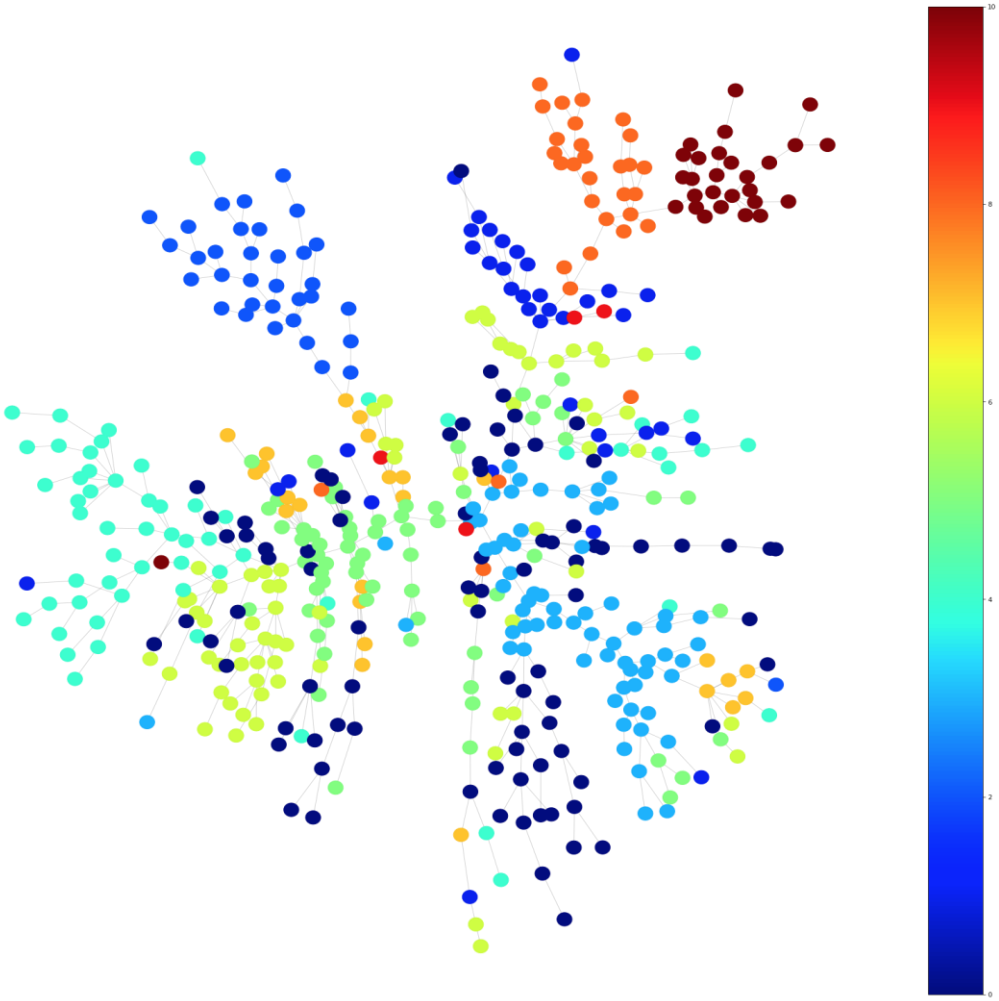


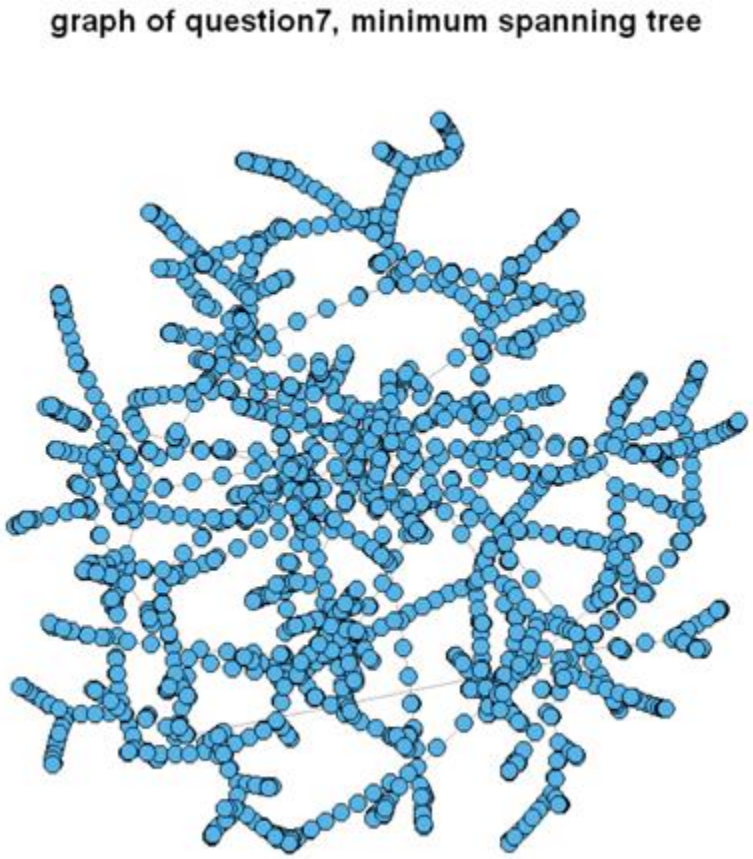
Figure 6. MST without labels for weekly data

2 Let's Help Santa!

Answer for Question 6:

The number of nodes in G is 1880, the number of edges in G is 311802.

Answer for Question 7:



End Point 1	End Point2	Mean Travel Time
3300 Brodie Drive, South San Jose, San Jose	4300 La Torre Avenue, South San Jose, San Jose	132.59
3300 Brodie Drive, South San Jose, San Jose	3700 McLaughlin Avenue, South San Jose, San Jose	126.24
3300 Brodie Drive, South San Jose, San Jose	400 Ginkgo Court, South San Jose, San Jose	109.625
1700 Coyote Point Drive, Shoreview, San Mateo	1800 Helene Court, East San Mateo, San Mateo	80.985

1700 Coyote Point Drive, Shoreview, San Mateo	600 Lexington Way, Oak Grove Manor, Burlingame	111.885

The results of street address is intuitive. First all street address are related to San Jose or San Mateo, which means they are in the same area. Second, the mean travel time of these two end points are all around 100, so the time it takes from one end point to another end point is similar(only one mean travel time is below 100).

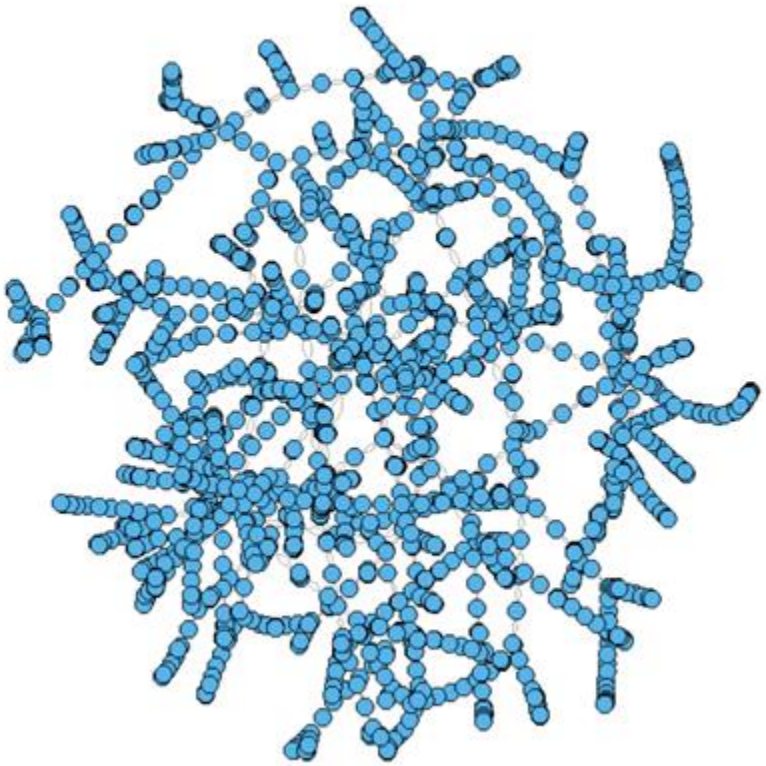
Answer for Question8:

By randomly sampling 1000 triangles, the percentage of triangles in the graph which satisfy triangle inequality is 95.9%

Answer for Question9:

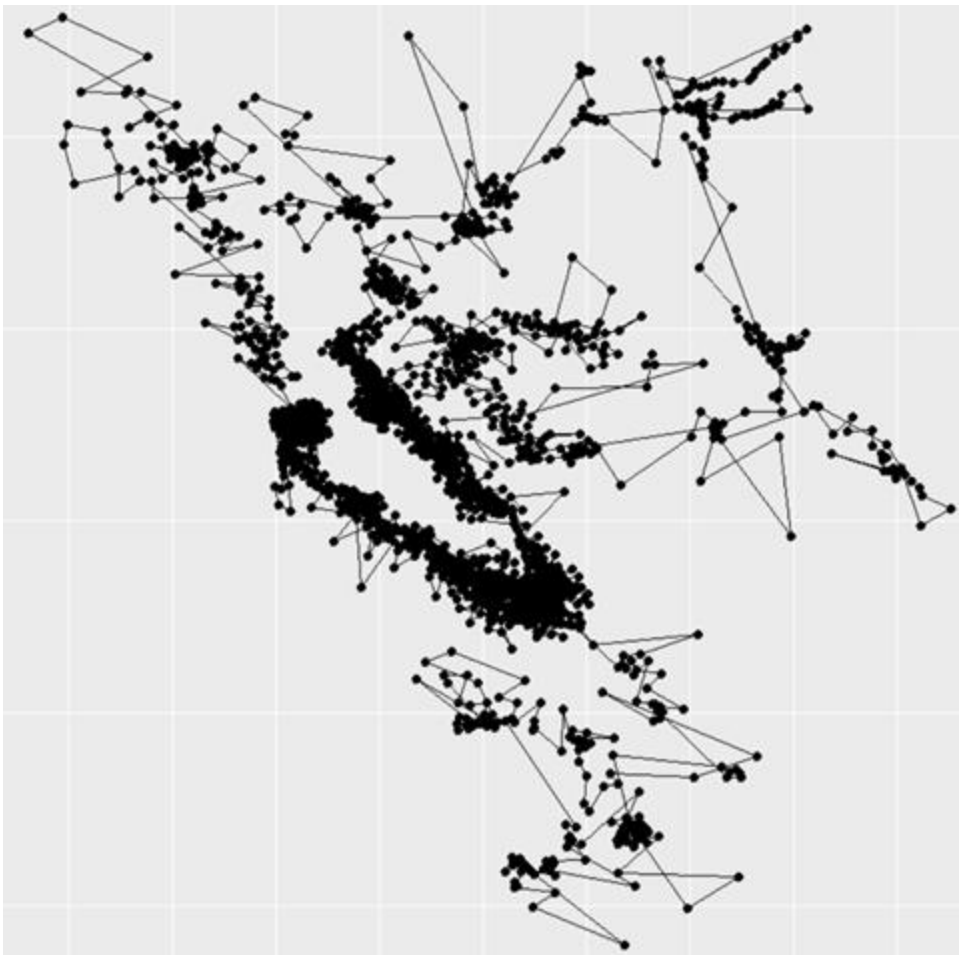
The empirical performance of approximate algorighm is 1.66.

graph of question9, minimum spanning tree



Answer for Question 10:

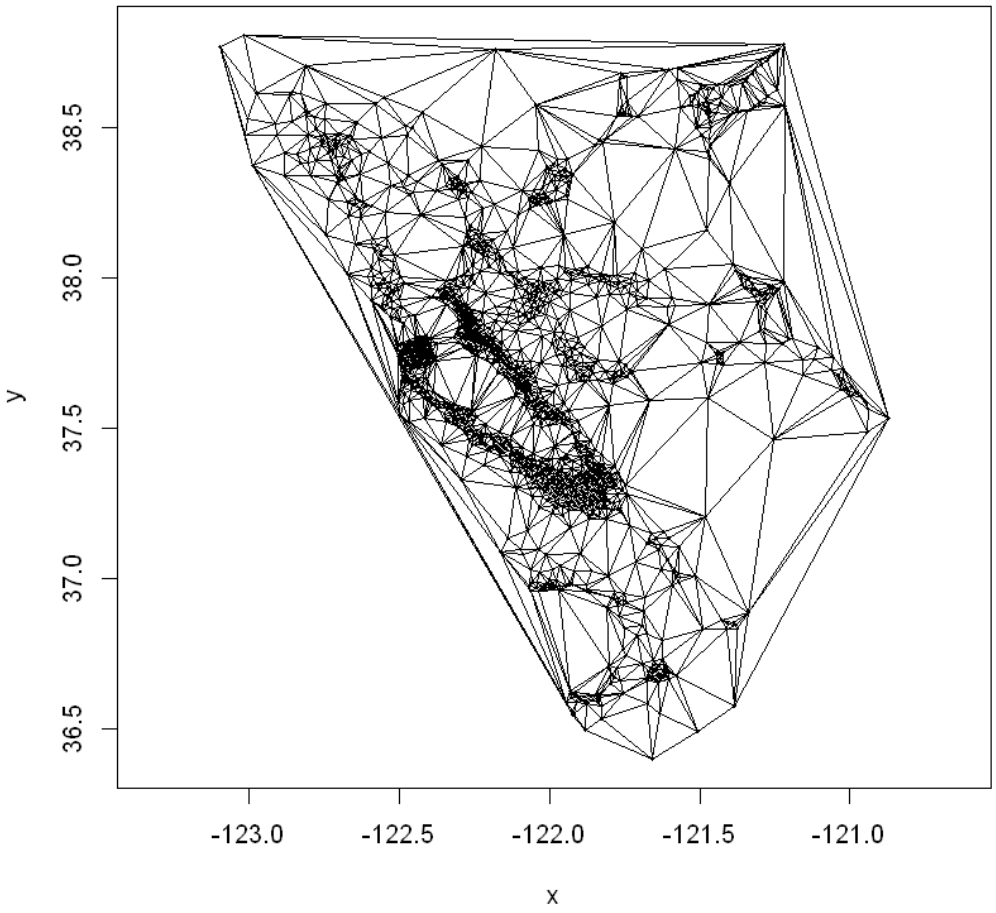
The trajectory that Santa has to travel is showing as below.



3 Analysing the Traffic Flow

Answer for question 11:

The road mesh is shown in following figure. In addition, its subgraph G delta was created by triangulation called g_d.

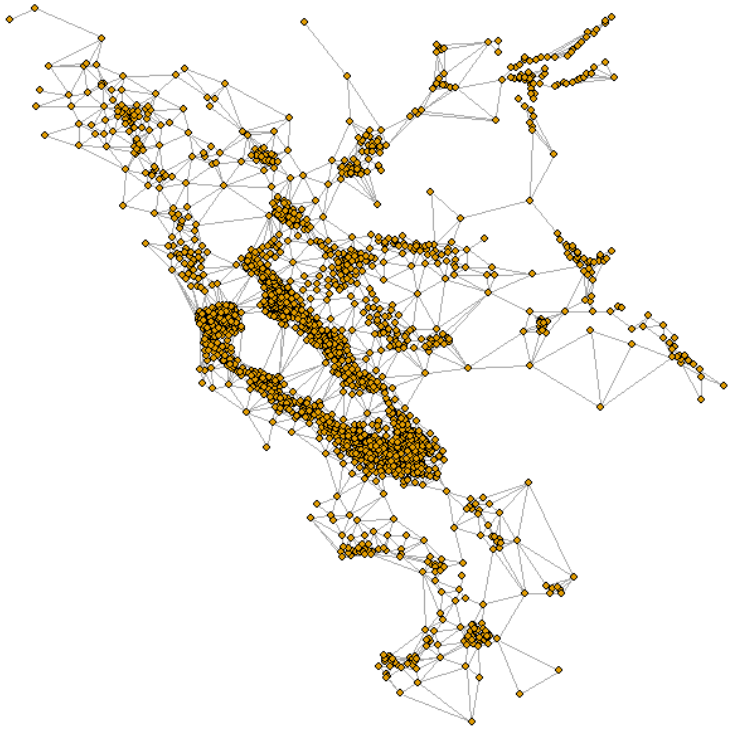


Answer for question 12:

The traffic flow for each road are shown in the code result. Most traffic flows are in range of 2000 to 3500 cars/hour.

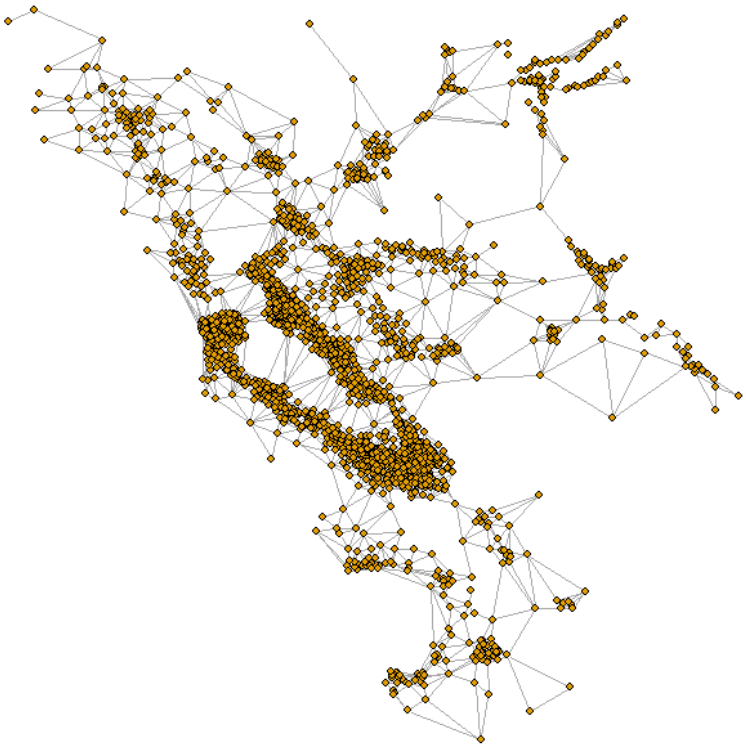
Answer for question 13:

Then we calculated maximum number of cars can commute per hour from Stanford to UCSC, the result is 14891.56. There are 4 dis-joined path between those two points. This matched what I see on my road map.



Answer for question 14:

Then we removed all route that having too large mean travel time. The threshold was set to 870. Then the trimmed plot is like following. All real bridges are preserved. The graph has 5267 edges before trimmed fake roads, and 5145 after removing fake roads.



Answer for question 15:

Then we repeated question 13 with trimmed graph. The new max flow is still 14891.56.
There were still 4 dis-joined paths between Stanford and UCSC. This is because there were only 112 fake roads being removed, which wouldn''t affect paths between Stanford and UCSC too much.