Project 4: Graph Algorithms

1. Stock Market

In this part of the project, we study data from stock market. The data is available on this Dropbox Link. The goal of this part is to study correlation structures among fluctuation patterns of stock prices using tools from graph theory. The intuition is that investors will have similar strategies of investment for stocks that are effected by the same economic factors. For example, the stocks belonging the transportation sector may have different absolute prices, but if for example fuel prices change or are expected to change significantly in the near future, then you would expect the investors to buy or sell all stocks similarly and maximize their returns. Towards that goal, we construct different graphs based on similarities among the time series of returns on different stocks at different time scales (day vs a week). Then, we study properties of such graphs. The data is obtained from Yahoo Finance website for 3 years. You're provided with a number of csv tables, each containing several fields: Date, Open, High, Low, Close, Volume, and Adj Close price. The files are named according to Ticker Symbol of each stock. You may find the market sector for each company in Name sector.csv.

1. Return correlation
In this part of the project, we will compute the correlation among log-normalized stock-return time series data. Before giving the expression for correlation, we introduce the following notation:

- pi(t) is the closing price of stock i at the tth day
- qi(t) is the return of stock i over a period of [t−1,t]
  qi(t) =pi(t)−pi(t−1)/pi(t−1)
- ri(t) is the log-normalized return stock i over a period of [t−1,t] ri(t) = log(1 + qi(t))

Then with the above notation, we define the correlation between the log-normalized stock-return time series data of stocks i and j as
ρij = ⟨ri(t)rj(t)⟩−⟨ri(t)⟩⟨rj(t)⟩ √(⟨ri(t)2⟩−⟨ri(t)⟩2)(⟨rj(t)2⟩−⟨rj(t)⟩2)where ⟨·⟩ is a temporal average on the investigated time regime (for our data set it is over 3years).

Preprocessing:
We downloaded the data and calculated the q and r for each csv, and saved them into a new file correlation.txt.

Question 1

What are upper and lower bounds on ρij? Provide a justification for using log-normalized return (ri(t)) instead of regular return (qi(t)).

The upper bound is +1, which means a perfect relationship and lower bound is -1 indicating a perfect negative relationship. While ρ_{ij} equalls to 0, there is no relationship at all. For example, if ri and rj are independent, $\langle r_i(t) r_j(t)\rangle$ and $\langle r_i(t)\rangle\langle r_j(t)\rangle$ are the same so the numerator is 0 which indicates no relationship. If ri and rj are correlated, we assume ri = a*rj, after simplifying, the ρ_{ij} = a / a^2^(1/2) such that a =1 when positive and a =-1 when negative.

Normal distribution is symmetrical but lognormal distribution is not. Since the values in ri is always positive and they can be skewed to the right. This skewness is important since an investor wants to look at the growth factor of a stock, and the lognormal distribution is the best for them to looking continuously. Since the future stock price must be positive and no one would care about the stock prices below $0.

Question 2

Plot a histogram showing the un-normalized distribution of edge weights.

In this part,we construct a correlation graph using the correlation coefficient computed in the previous section. The correlation graph has the stocks as the nodes and the edge weights are given by the following expression $w_{ij} =\sqrt{2(1-\rho_{ij})}$ Compute the edge weights using the above expression and construct the correlation graph.

We read the processed data and got the values from the above equations and get the following figures:

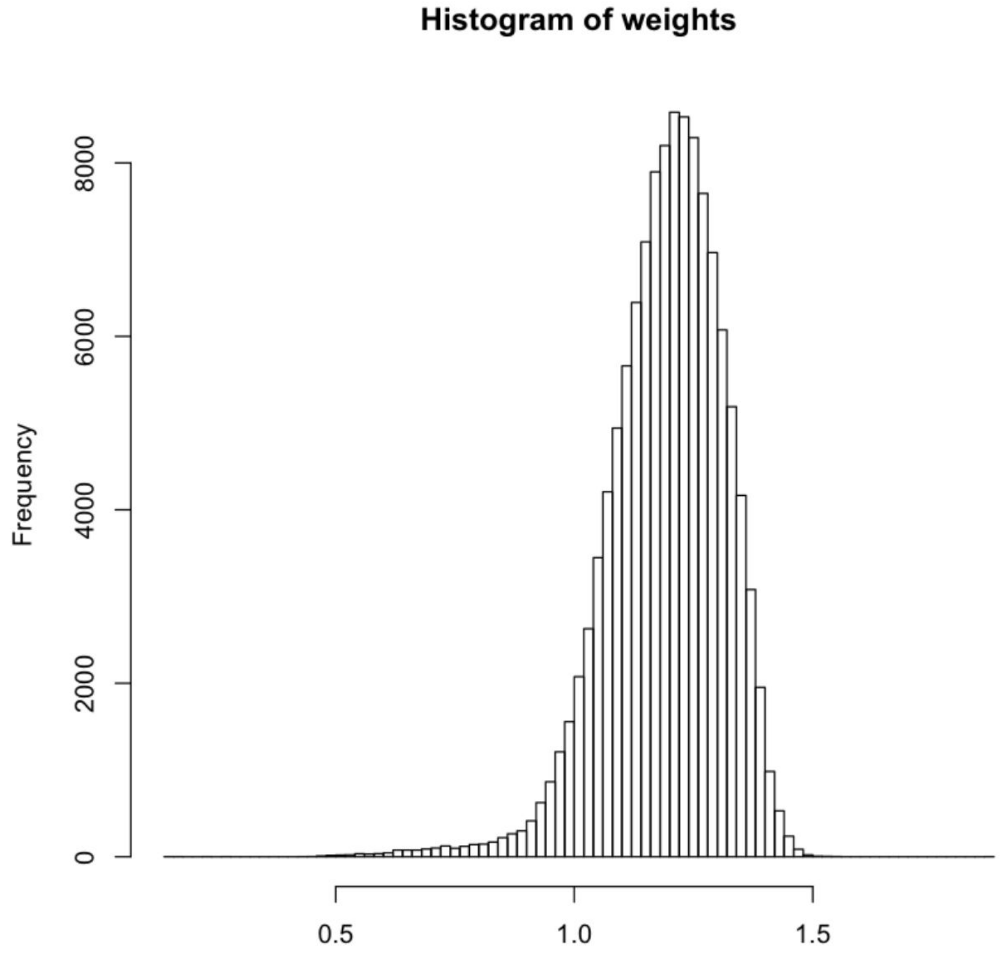


Figure 1: Histogram of Weights

As we can see from the figrue1, the weights at around 1.3 has the highest frequency around 8000. The value skew the right has more weights.

Question 3

Extract the MST of the correlation graph. Each stock can be categorized into a sector, which can be found in Name sector.csv file. Plot the MST and color-code the nodes based on sectors. Do you see any pattern in the MST? The structures that you find in MST are called Vine clusters. Provide a detailed explanation about the pattern you observe.

We assigned different color for each sector and created a minimum spanning tree, which can be shown in the following figure. And the pattern we observed is Vine Clusters. Since the MST is a subset of the edges of a connected edge-weighted undirected graph which connects all vertices with no cycle and with minimum edge weight. And in figure 2, the nodes represents each stock tend to cluster lowest weights, so they create such a Vine cluster.

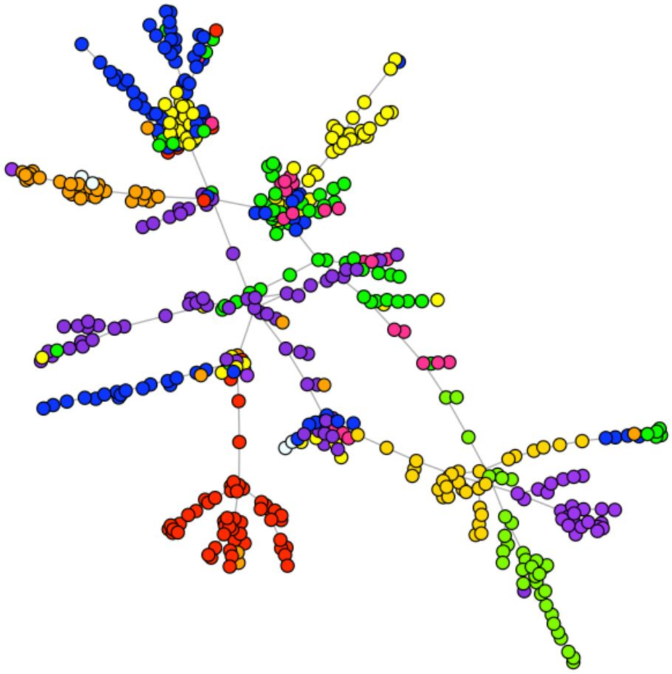


Figure 2: MST of Correlation Graph

Question 4

Report the value of α for the above two cases and provide an interpretation for the difference.

Table 1: Alpha 1 and Alpha 2

Alpha 1(|Qi|/|Ni|)	Alpha 2(|Si|/|V|)
0.11202	0.11397

As we can see, there is a small difference between alpha 1 and alpha 2. The reason is the calculation of alpha 1 only considers the neighbors of each node while the second one calculated more nodes since it needs to consider the probability of different nodes belonging to the same sector. As the second one consider nodes other than the neighbors, its value is slightly larger.

Question 5

Extract the MST from the correlation graph based on weekly data. Compare the pattern of this MST with the pattern of the MST found in Question 3.

In the previous parts, we constructed the correlation graph based on daily data. In this part of the project, we will construct a correlation graph based on weekly data. To create the graph, sample the stock data weekly on Mondays and then calculate ρij using the sampled data. If there is a holiday on a Monday, we ignore that week. Create the correlation graph based on weekly data. The MST is shown in the following figure:

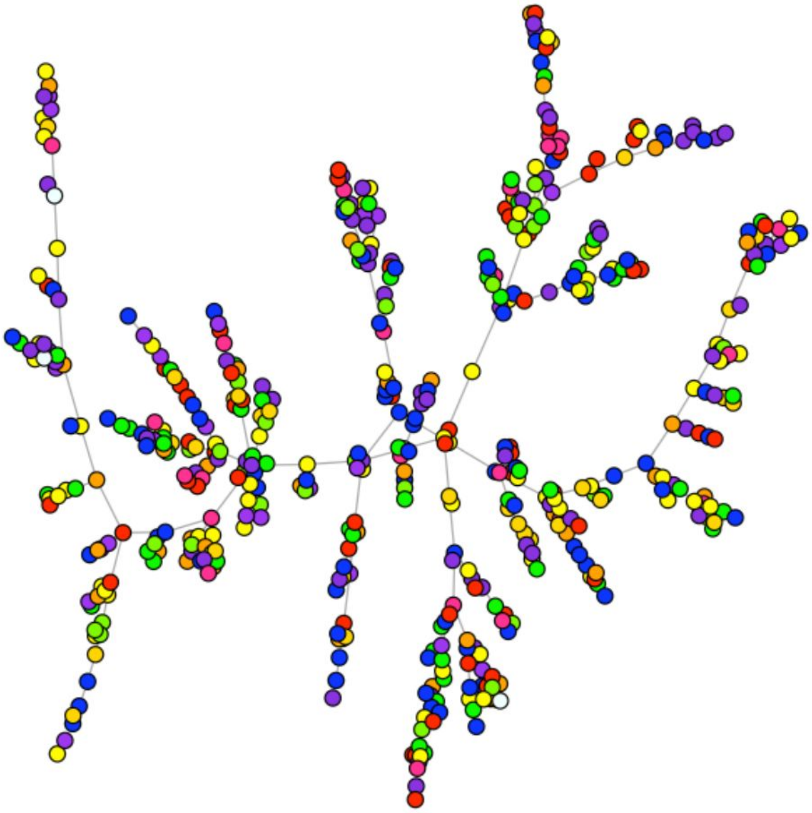


Figure 3: MST of Correlation Graph of Weekly Data

We can see from the figure 3 that it also shows the Vine cluster but less profound than the one in question3. This is because the nodes in weekly data are more separated than the daily data.

2. Let’s Help Santa

Question 6

Report the number of nodes and edges in G.

The graph will contain some isolated nodes (extra nodes existing in the Geo Boundaries JSON file) and a few small connected components. Remove such nodes and just keep the giant connected component of the graph. In addition, merge duplicate edges by averaging their weights 3. We will refer to this cleaned graph as G afterwards.

Table 2: Number of Nodes and Edges of G

Number of Nodes	Number of Edges
1898	321703

Question 7

Build a minimum spanning tree (MST) of graph G. Report the street addresses of the two endpoints of a few edges. Are the results intuitive?

We built the MST of the new graph G which can be seen in the figure 4:

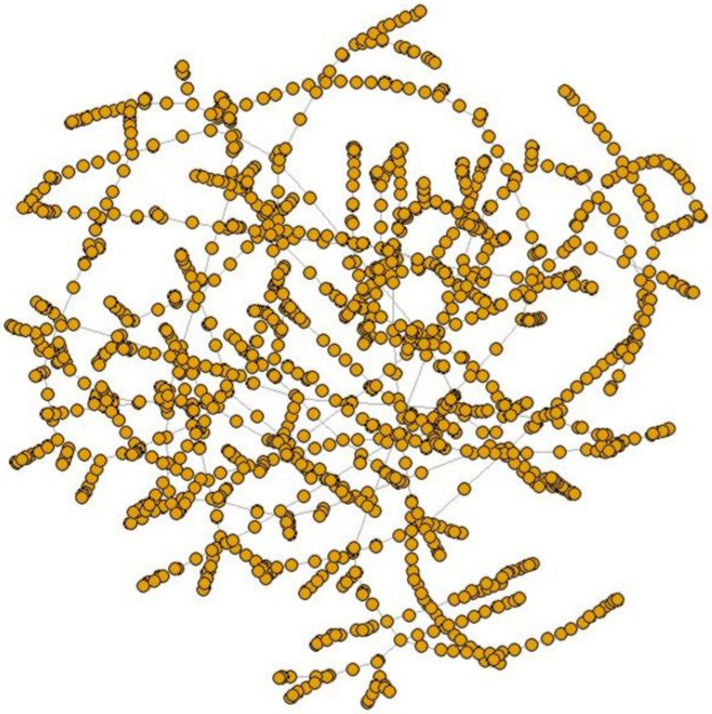
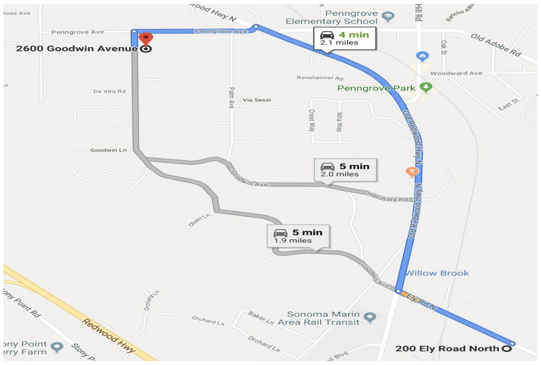
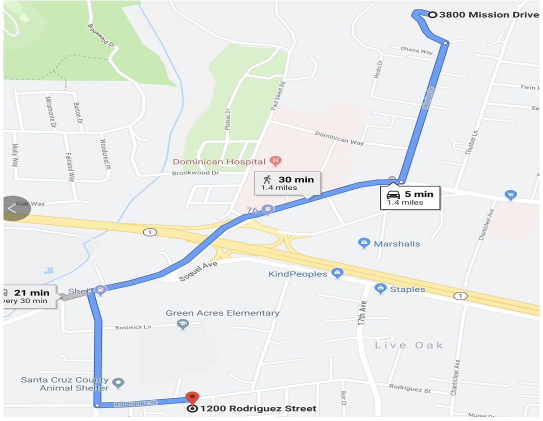
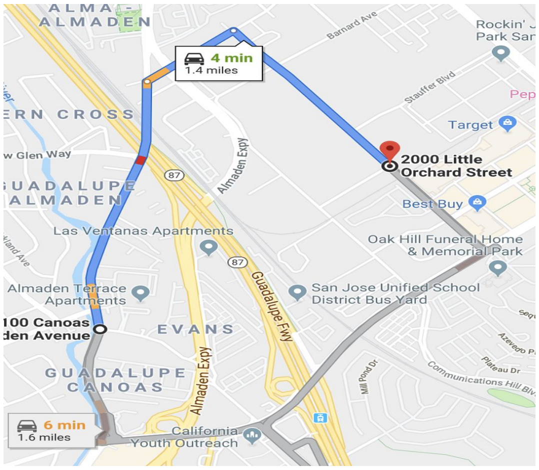


Figure 4: MST of G with December only Data

We randomly chose a few of the endpoint pairs to test and they were all in the same city, which indicates that the data makes sense. To make sure the data we created is correct, we used Google Map to help check whether the endpoints are intuitive, as the results are shown in the table 3.

Table 3. Checking with Google Map

Starting endpoint	Ending endpoint	Distance On Google Map

200 Ely Road North, Petaluma	2600 Goodwin Avenue, Penngrove	 2600 Goodwin Avenue 4 min 2.1 miles 5 min 2.0 miles 5 min 1.9 miles Penngrove Park Willow Brook Sonoma Marin Area Rail Transit Redwood Hwy 200 Ely Road North 4 min 2.1miles
3800 Mission Drive, Santa Cruz	1200 Rodriguez Street, Santa Cruz	 3800 Mission Drive Dominican Hospital 30 min 1.4 miles 5 min 1.4 miles Marshalls KindPeoples Staples Green Acres Elementary Live Oak 21 min Santa Cruz County Animal Shelter 1200 Rodriguez Street 5min 1.4 miles
2100 Canoas Garden Avenue, South San Jose, San Jose	2000 Little Orchard Street, South San Jose, San Jose	 ALMADEN 4 min 1.4 miles Rockin' Park Target 2000 Little Orchard Street Best Buy Oak Hill Funeral Home & Memorial Park Las Ventanas Apartments Almaden Expy Guadalupe Fwy Almaden Terrace Apartments San Jose Unified School District Bus Yard 100 Canoas den Avenue EVANS GUADALUPE CANOAS 6 min 1.6 miles California Youth Outreach 4min 1.4miles

As the endpoints are very close to each other, we can conclude that the results are intuitive.

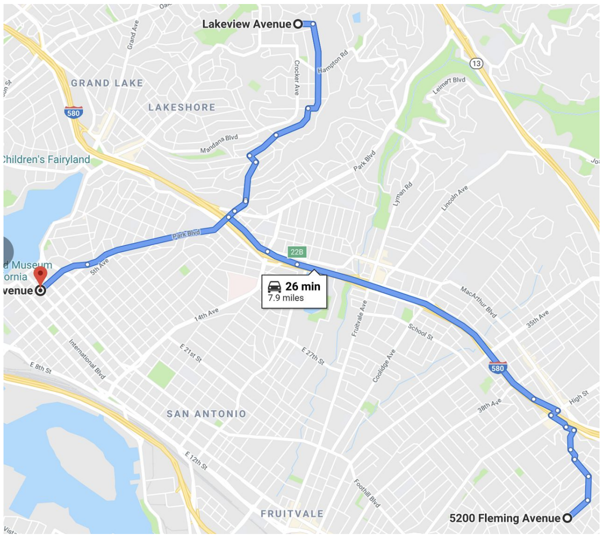
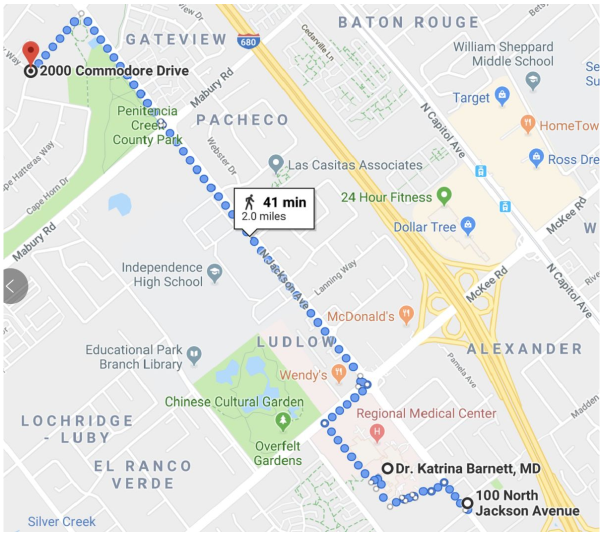
Question 8

Determine what percentage of triangles in the graph (sets of 3 points on the map) satisfy the triangle inequality. You do not need to inspect all triangles, you can just estimate by random sampling of 1000 triangles.

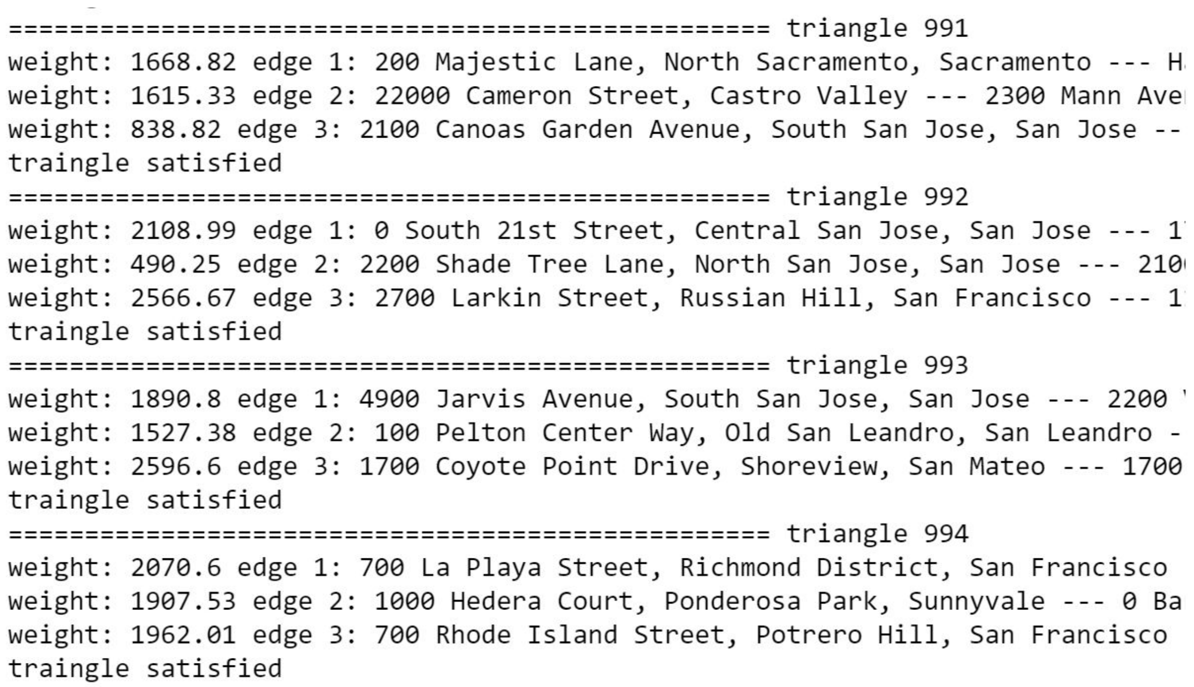
We randomly selected 1000 sets of 3 points on the map and checked whether they satisfy the triangle inequality and the results show that 95.7% of the triangles pass the test.

Now, we want to find an approximation solution for the traveling salesman problem (TSP) on G. Apply the 1-approximate algorithm described in the class4. Inspect the sequence of street addresses visited on the map and see if the results are intuitive. The result is shown in the table 4:

Table 4: Triangle Inequality Check with Google Map

Sequential Address	Google Map
5200 Fleming Ave, Oakland, Lakeview Ave, Piedmont, 1500 3rd Ave, Oakland,	 Lakeview Avenue GRAND LAKE LAKESHORE 580 13 Children's Fairyland 26 min 7.9 miles SAN ANTONIO FRUITVALE 5200 Fleming Avenue
100 North Jackson Avenue, East San Jose, San Jose 500 North Jackson Avenue, East San Jose, San Jose 2000 Commodore Drive, East San Jose, San Jose	 BATON ROUGE GATEVIEW 680 2000 Commodore Drive William Sheppard Middle School Target HomeTow Ross Dre Mabury Rd Penitencia Creek County Park PACHECO Las Casitas Associates 41 min 2.0 miles 24 Hour Fitness Dollar Tree N Capitol Ave McKee Rd Independence High School N Jackson Ave McDonald's LUDLOW ALEXANDER Educational Park Branch Library Wendy's Chinese Cultural Garden Regional Medical Center LOCHRIDGE - LUBY Overfelt Gardens EL RANCO VERDE Dr. Katrina Barnett, MD 100 North Jackson Avenue Silver Creek

As the sequentially visited locations are close to each other, we concluded that the results are intuitive. And here are more examples:



```
================================================== triangle 991
weight: 1668.82 edge 1: 200 Majestic Lane, North Sacramento, Sacramento --- H
weight: 1615.33 edge 2: 22000 Cameron Street, Castro Valley --- 2300 Mann Ave
weight: 838.82 edge 3: 2100 Canoas Garden Avenue, South San Jose, San Jose --
traingle satisfied
================================================== triangle 992
weight: 2108.99 edge 1: 0 South 21st Street, Central San Jose, San Jose --- 1
weight: 490.25 edge 2: 2200 Shade Tree Lane, North San Jose, San Jose --- 210
weight: 2566.67 edge 3: 2700 Larkin Street, Russian Hill, San Francisco --- 1
traingle satisfied
================================================== triangle 993
weight: 1890.8 edge 1: 4900 Jarvis Avenue, South San Jose, San Jose --- 2200
weight: 1527.38 edge 2: 100 Pelton Center Way, Old San Leandro, San Leandro -
weight: 2596.6 edge 3: 1700 Coyote Point Drive, Shoreview, San Mateo --- 1700
traingle satisfied
================================================== triangle 994
weight: 2070.6 edge 1: 700 La Playa Street, Richmond District, San Francisco
weight: 1907.53 edge 2: 1000 Hedera Court, Ponderosa Park, Sunnyvale --- 0 Ba
weight: 1962.01 edge 3: 700 Rhode Island Street, Potrero Hill, San Francisco
traingle satisfied
```

Figure 5: Examples past the test

Question 9

Find an upper bound on the empirical performance of the approximate algorithm:

ρ =Approximate TSP Cost / Optimal TSP Cost

In this part, we followed three steps:

1. Find the minimum spanning tree T under [dij];
2. Create a multigraph G by using two copies of each edge of T;
3. Find an Eulerian walk of G and an embedded tour.

The first step is easy and we just created a MST. In step2, we constructed a graph that has two copies of each edge. In step 3, we applied Eulerian walk, which is an algorithm that each node appears least once and each edge appears only once. This is also known as Traveller Salesman Problem(TSP). TSP requires some deletion of repeated visits and it recursively go through every node. And our result shows that the total weight is 479842, and the upper bound is 1.6585

Question 10

Plot the trajectory that Santa has to travel!

We plotted the trajectory that Santa has to travel in the following figure:

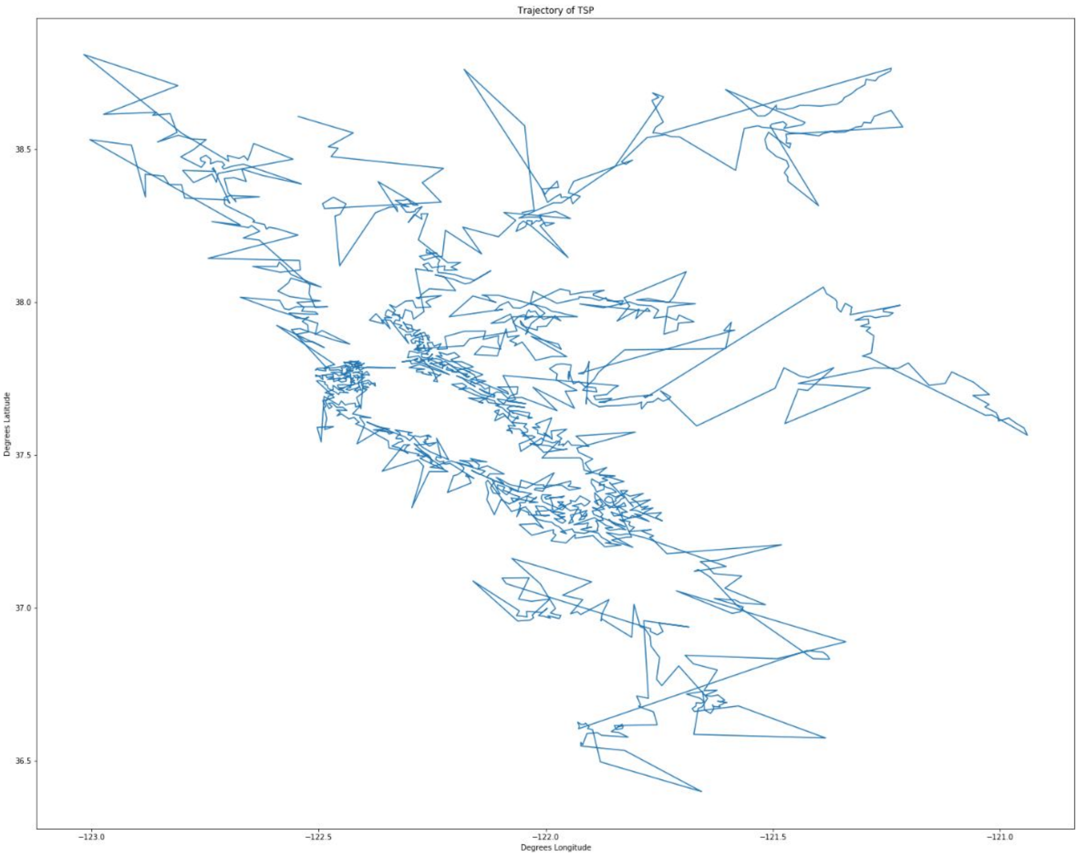


Figure 6: Trajectory that Santa has to travel through the SF Bay.

Question 11

Plot the road mesh that you obtain and explain the result. Create a graph G∆whose nodes are different locations and its edges are produced by triangulation.

Delaunay triangular algorithm is used to discover connecting triangulars. Delaunay triangulation maximizes the minimum angle of the triangles in the triangulation. For this question, we first load the given coordinate data. We use the mean of the boundary to represent the node.Then we take the GCC node and construct the coordinate array. Finally we apply the delaunay triangular algorithm. The result is shown in the figure below. The red spots are the nodes and the blue lines are the edges.

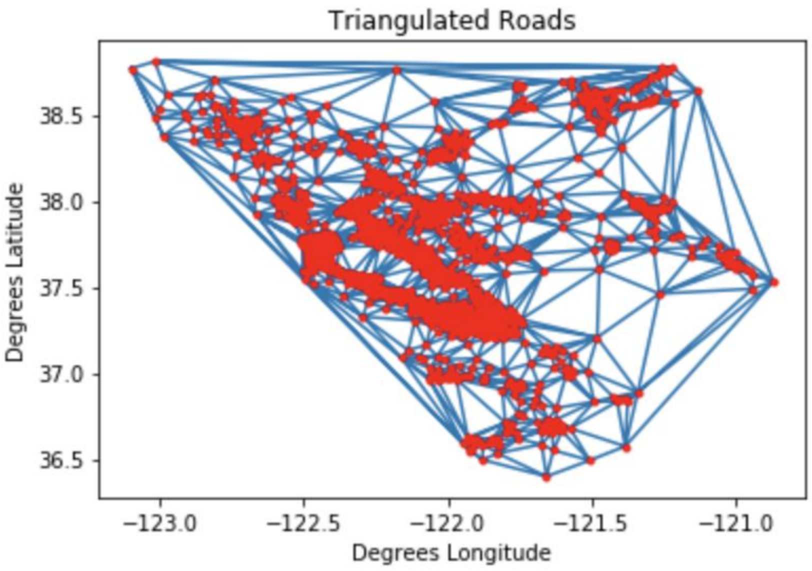


Figure 7 Result of Delaunay Triangulation on Locations in Cleaned Graph

Question 12

Using simple math, calculate the traffic flow for each road in terms of cars/hour. Report your derivation.

We use the following assumptions.

- Each degree of latitude and longitude ≈ 69 miles
- Car length ≈ 5m = 0.003mile
- Cars maintain a safety distance of 2 seconds to the next car
- Each road has 2 lanes in each direction

Also, we assume no traffic jam and consider the calculated traffic flow as the max capacity of each road.

Denote the velocity of car as v mile/hour, which can be derived from the road length, l mile, and passing time, t hour, by taking $v = \frac{l}{t}$.

The safety distance is 2 second of var speed. Therefore, the distance is $\frac{v}{3600} * 2 = \frac{v}{1800}$ mile. Then the number of cars remaining on the road is $\frac{l}{0.003+\frac{v}{1800}}$. Then we divide it by the time to get the traffic flow of one lane, which is $\frac{l/t}{0.003+\frac{v}{1800}}$ = $\frac{v}{0.003+\frac{v}{1800}}$. Then multiply it by 2 and the traffic flow is $\frac{2v}{0.003+\frac{v}{1800}}$ cars/hour.

In our data, we can take the given distance and time to get the velocity, and then use the formula we got above to get the result.

Question 13

Calculate the maximum number of cars that can commute per hour from Stanford to UCSC. Also calculate the number of edge-disjoint paths between the two spots. Does the number of edge-disjoint paths match what you see on your road map?

We can apply “max_flow” function and “edge_disjoint_paths” functions to maximum number of cars and edge-disjoint-paths respectively. Before that, we first find the node of Stanford and UCSC. Stanford is node 2607, and USCS is node 1968. After programming in R, we obtain the solution that maximum number of cars that can commute per hour from Stanford to UCSC is 14866.44 cars/hour, and the number of edge-disjoint paths between two spots is 5.

Then we try to verify that the number of edge-disjoint paths match with the road map. Here we take two zoomed in road maps focusing on Stanford and UCSC respectively. The road maps are in Figure 8 and Figure 9.

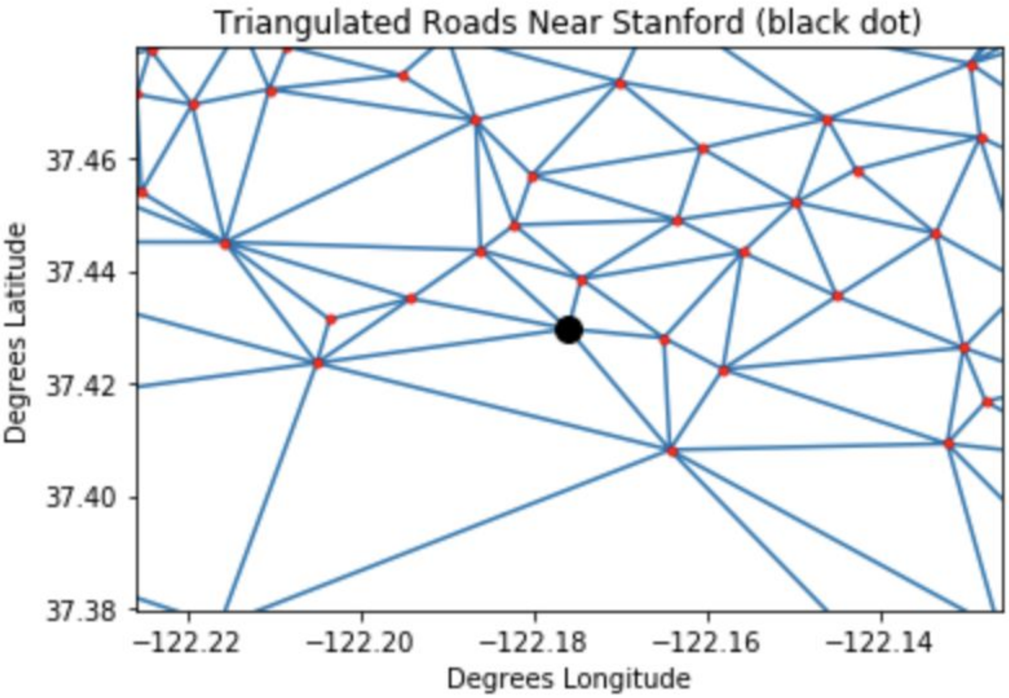


Figure 8 Triangulated Roads Near Stanford

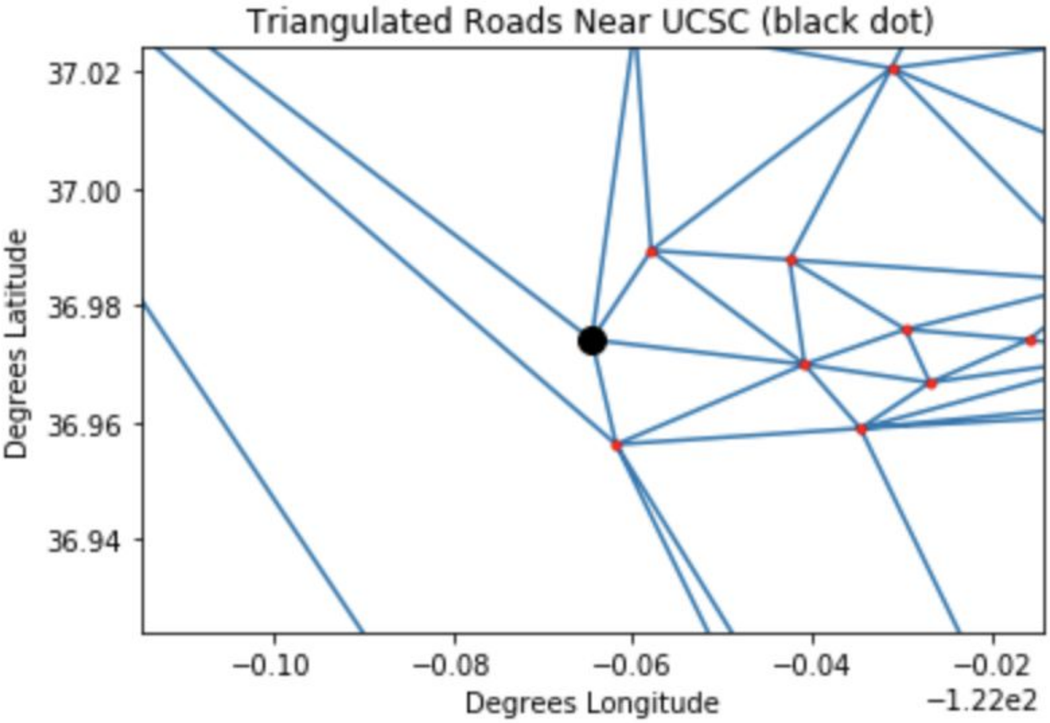


Figure 9 Triangulated Roads Near UCSC

From these two figures, we can see that there are 6 roads out of stanford and 5 roads in UCSC. We must take the minimum between these two values as there are only 5 roads leading to UCSC though we have more combinations. Therefore, the number we obtained before is correct.

Question 14

Plot $\tilde{G}\Delta$ on actual coordinates. Are real bridges preserved?

In this question, we added some threshold on the travel time of the roads in order to remove most of the faking bridges and preserving the real ones. Here we consider following five bridges:

- Golden Gate Bridge: [[-122.475, 37.806], [-122.479, 37.83]]
- Richmond, San Rafael Bridge: [[-122.501, 37.956], [-122.387, 37.93]]
- San Mateo Bridge: [[-122.273, 37.563], [-122.122, 37.627]]
- Dambarton Bridge: [[-122.142, 37.486], [-122.067, 37.54]]
- San Francisco - Oakland Bay Bridge: [[-122.388, 37.788], [-122.302, 37.825]]

From the dataset, we first learned the top 2 longest time to pass the bridges above and then we decide our threshold as 1300 at first. However, the result is below our expectation as there are still many fake bridges though it preserves all real bridges. The result is shown in Figure 11. So we then decrease the threshold trying to removing more fake bridges. When decreasing the threshold, we find that it is hard to remove more fake bridges but also preserve all real ones. Therefore, we make a compromise here and pick 720 as the threshold. The pruned graph is shown in figure x. From this figure, we can see that most of the fake bridges are removed at the cost of removing San Mateo Bridge.

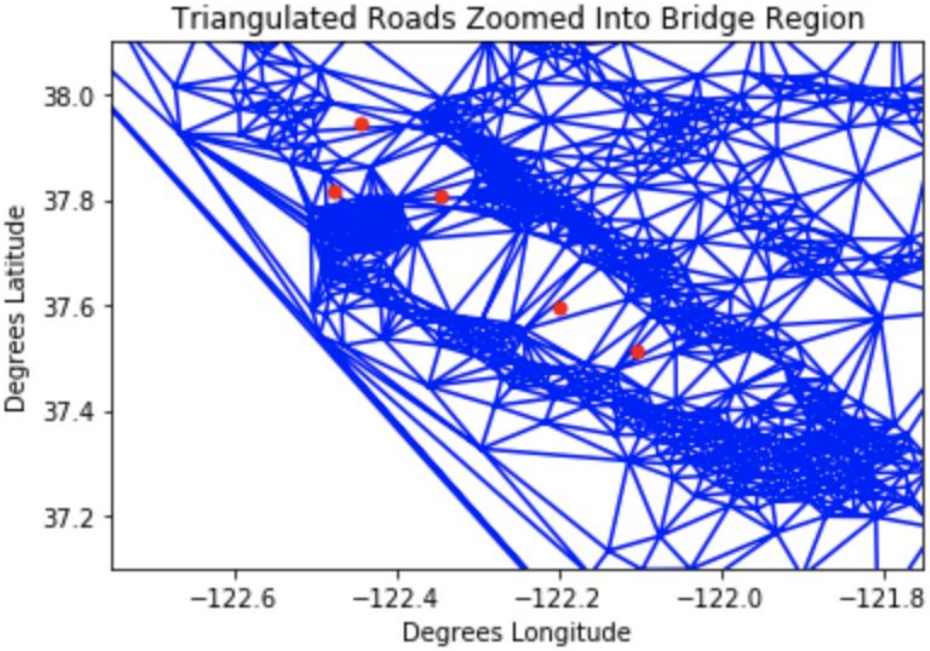


Figure 10 Triangulated Roads Zoomed in Bridge Region

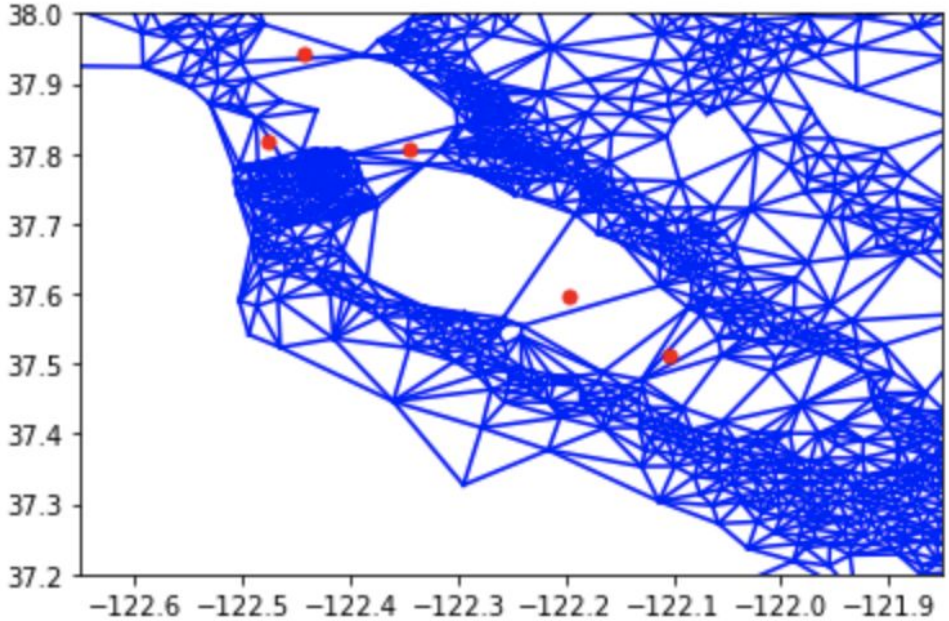


Figure 11 Defoliated Graph with Threshold 1300

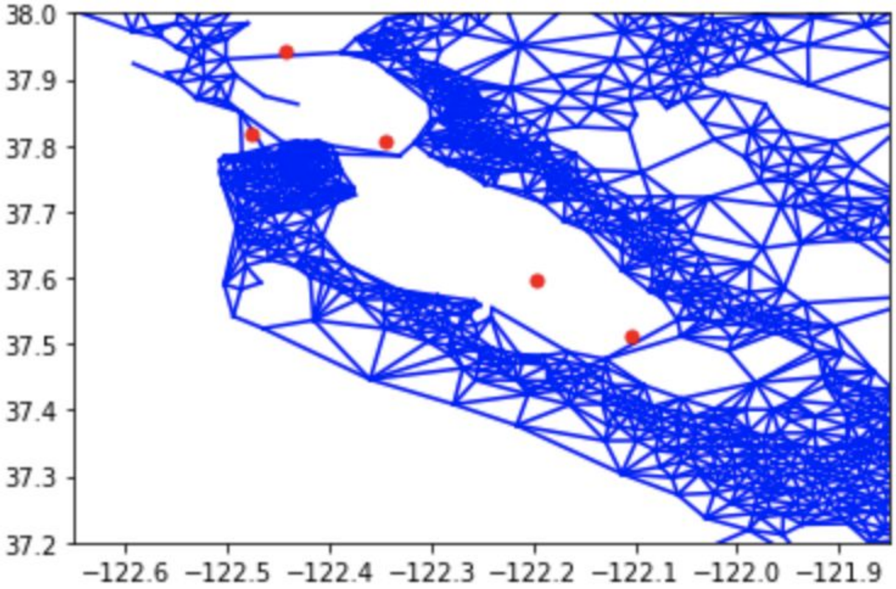


Figure 12 Defoliated Graph with Threshold 720

Question 15

Now, repeat question 13 for $\tilde{G}_{\Delta}$ and report the results. Do you see any significant changes?

In this question, we use the pruned graph from question 14 preserving four bridges. Then we re-compute maximum number of cars that can commute per hour from Stanford to UCSC and number of edge-disjoint paths between the two spots. The results do not change compared to what we obtained in question 13. The maximum number of cars per hour is 14866.44 and the number of edge-disjoint paths is 5.

Therefore, we re-taken two zoom-in maps focusing on Stanford and UCSC. From the two figures below, we can see that the pruning process does not affect roads near Stanford. There are still 6 roads leading out of Stanford. As for UCSC, some neighboring roads are removed but 5 roads leading into UCSC still remained.

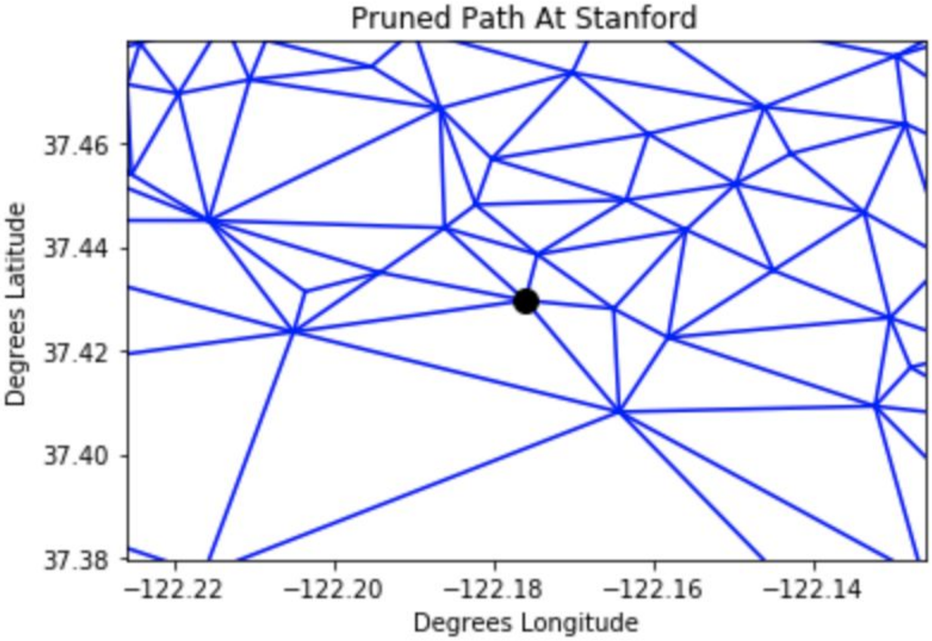


Figure 13 Road Map after Pruning at Stanford

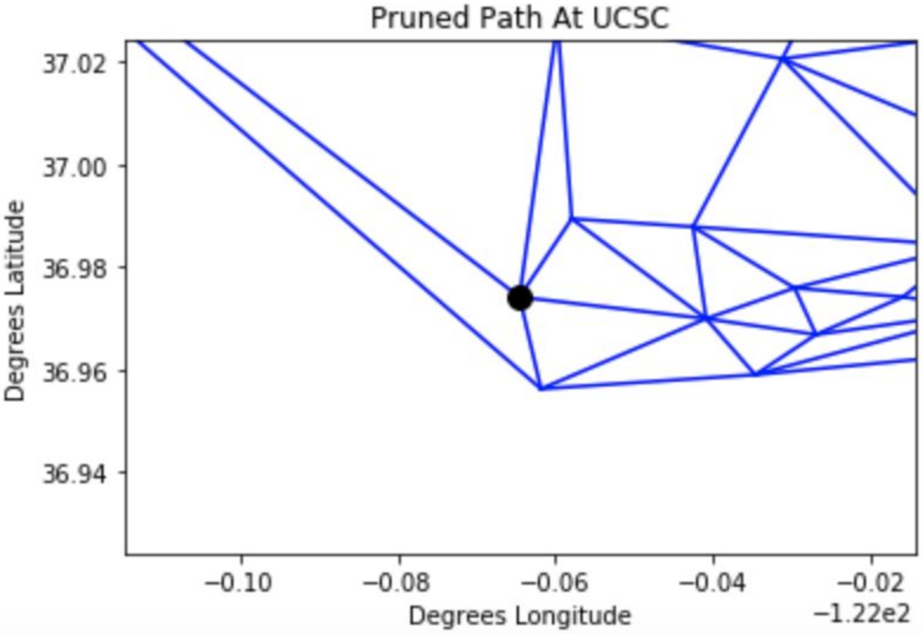


Figure 14 Road Map after Pruning at UCSC