EE 232E

Graphs and Network Flows

Homework 4 Report

Pingyuan Yue: 504737715

Ke Xu: 604761427

Yuanyi Ding: 404773978

6/12/2017

Table of Contents

1. Why Log Returns

To estimate the auto regression for stock prices, it is often the case that people use log returns instead of stock prices. After searching relative resources online, we found that there are a couple of reason behind this:

The first is that log return could make the data more smooth and stable while the relation between dataset remains the same.

The second reason is that log return is convenience. Log returns behave much better and are often more convenient to work with in an algorithm. For example, you can easily add the values when calculating compounding returns.

2. Constructing Correlation Graphs

In this part we write a function to compute the correlation coefficient and log return just as the mathematics formula expresses for each tuple read from the .csv file and implement the graph.data.frame function in R to construct the correlation graphs which has 505 nodes and 127260 edges.

Figure 2.1 represents the histogram of d_{ij} 's in which dij is $\sqrt{2(1-\rho_{ij})}$

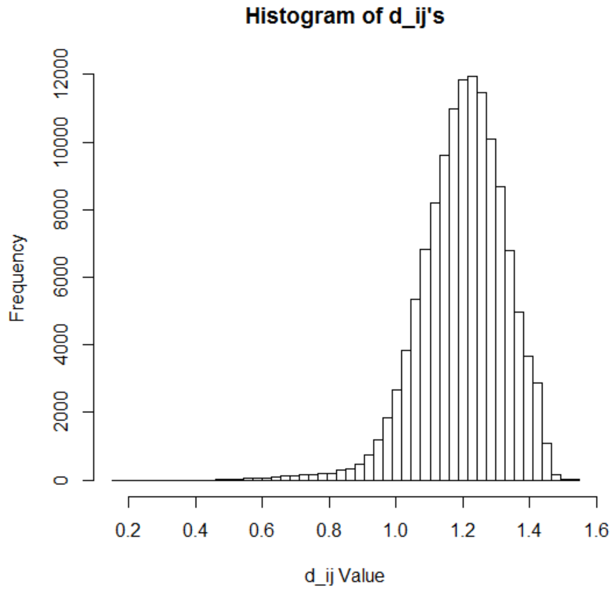


Figure 2.1 the histogram of d_{ij} 's

Figure 2.2 represents the correlation graph but it has too many nodes and edges so there may be some of them not shown in the graph.

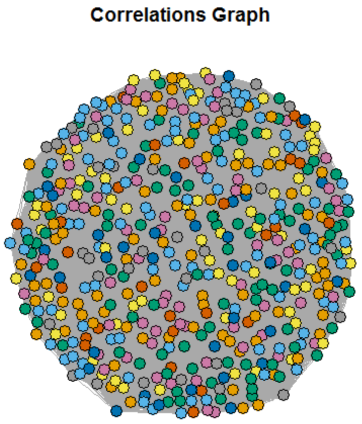


Figure 2.2 the correlation graph

3. Minimum Spanning Trees

In this part we just use a unique set to store all the sectors read from the file and then add colors for them. Then we implement mst() function to construct the minimum spanning tree whose weight is the same as the correlation graph.

After building the tree, we can find that it has 505 nodes and 504 edges, which has the same number of nodes as the graph and no circles.

Figure 3.1 represent the structure of minimal spanning tree.

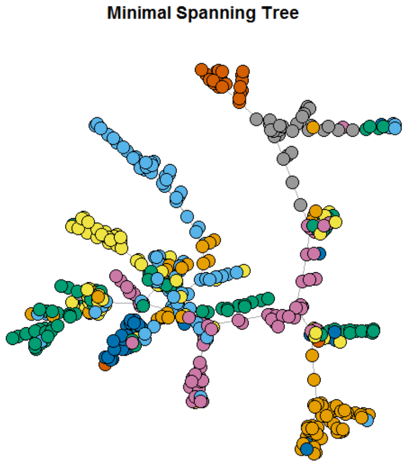


Figure 3.1 minimal spanning tree

From the above MST we can easily see that there are some “clusters” in it with each of which almost all the nodes share the same color (sector). So if we regard the correlation value as the weight of the graph, the nodes of same sectors turn out to be linked together.

4. Construct the Movie Graph

In this part, we just do for every node in the MST that finding all of its neighbors and get the each node's sector and summarize them. Then we just divide them by the total number of neighbors of certain nodes to get the probability.

For the random assignment metric, it is easy to implement. Just classify all the nodes by their sectors and then summarize them, dividing them by the total number of nodes to get the probability.

The following data is for the total dataset. We can see from it that the sum of probability using random sector metric is 1 and the probability using neighbor assignment metric is quite large since there are lots of "clusters" in the mst in which the sector is almost the same. So, if we implement the neighbor assignment metric, we can easily predict the sector since they trend to form clusters.

	Neighbor Assignment	Random Sector
Health Care	0.8875000	0.118811881
Industrials	0.7723857	0.128712871
Consumer Discretionary	0.8049020	0.168316832
Information Technology	0.6750000	0.138613861
Consumer Staples	0.8036036	0.073267327
Utilities	0.9196429	0.055445545
Financials	0.8290867	0.130693069
Real Estate	0.9005376	0.061386139
Materials	0.6866667	0.049504950
Energy	0.9901961	0.067326733
Telecommunication Services	0.7500000	0.007920792

5. △-Traveling Salesman Problem

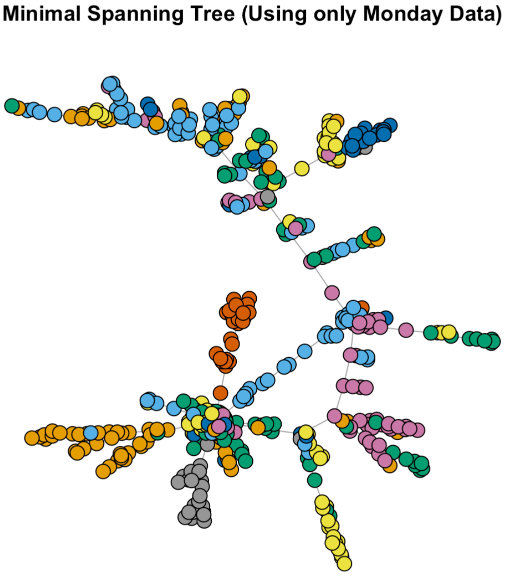
In this part, we first check if there is a delta inequality in the graph G. We simply implement 3 for loops to check every possible group of edges that can form a triangle so that there are totally 505*504*503/6 groups of triangle edges. For each group, we judge whether edge 1 + edge 2 is greater than or equal to edge 3. Moreover, we write a function to find the position of certain d_ijs in the vector with corresponding i and js. We can see that the delta inequality holds for the graph.

Then we implement the approximation traveling salesman problem. Firstly, we duplicate the edges in MST constructed before to create the multigraph and then we write the duplicated edges to a file, using python to find the Eular tour and write back the nodes and edges in Eular tour to R code and compute the sum of edges of TSP solution based on the adjacency matrix of correlation graph. Finally, we got the TSP solution is 477.3086. And the sum of weights of min spanning tree is 432.2612 and it is easily seen that *w(MST) < w(TSP) < 2 w(MST)*. And may not be globally optimality.

6. Constructing Correlation Graphs for Weekly Data

In this part, we are supposed to do something similar with the task in problem 2. In part 2, we used daily closing prices for stocks to compute log returns and created graphs. Now, we sample the stock closing data weekly on Mondays before computing log return arithmetic and creating graph.

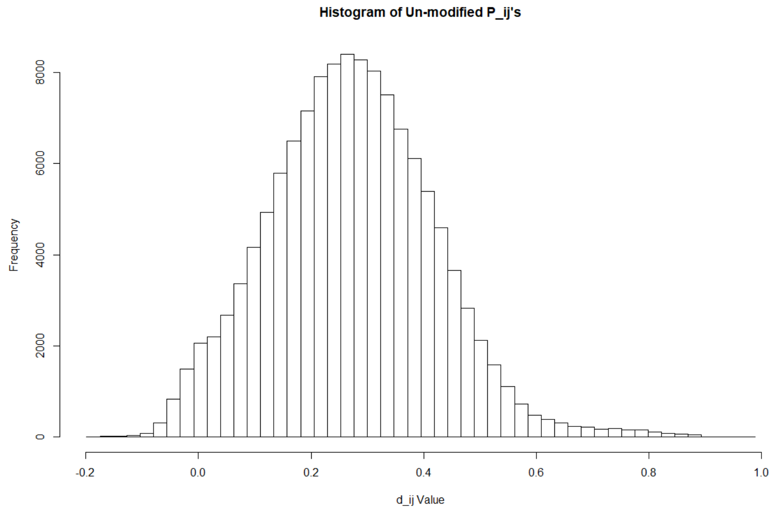
Because the raw .csv file contains a Date column, we can simply convert the date information into weekdays and sample the data on Mondays. After creating graph, we plot the histogram of dij's, MST and color-code the nodes based on sectors. The result is shown as follows.



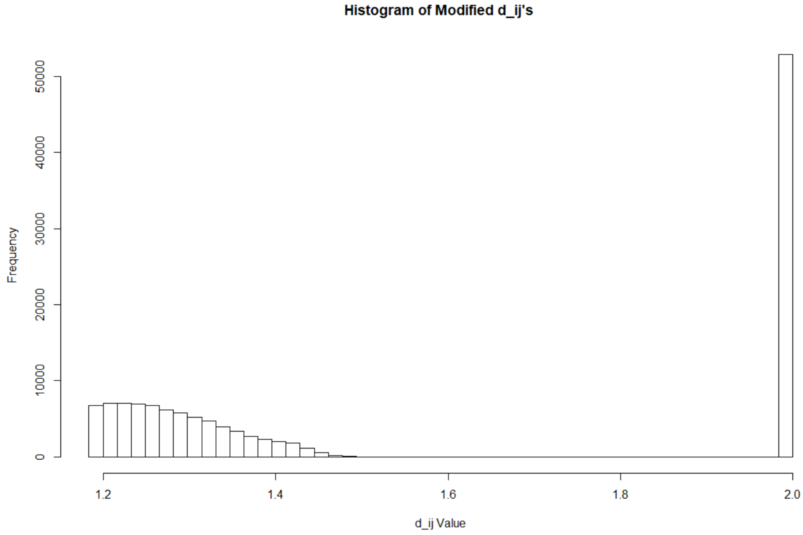
Compared with the MST graph in problem 3, it could be observed that for a shorter time scale (one day), the stocks in the same sector are clustered together which means the correlation between stocks in same sectors are strong. As the time scale increases (one week), the stocks in the same sector gradually departed in MST graph. According to the result in the handout reading material, we could find the opposite conclusion that for longer time period, the data is more likely to cluster together which could also make sense. Due to the reading material focus on time scale shorter than one day, the performance of different stocks could be rather random within a short period of time within one day. However, when the time scale increases to one day, the behavior of stocks with same sector could be easily affected by same economic factors. But when time scale keep increasing to one week, the correlation between stocks within same sector would decrease due to different strategies and situations of different companies.

7. Modifying Correlations

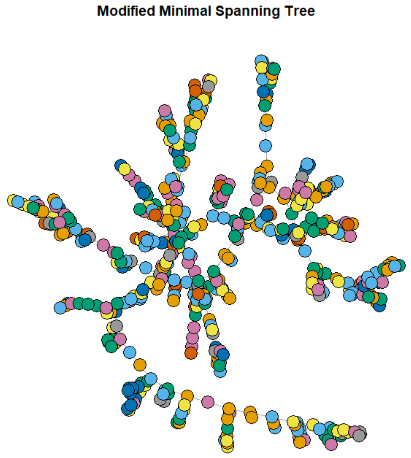
In this problem, we first plot the histogram of ρ_{ij}'s from daily data as shown below:



Then we modify the ρ_{ij}, we set all ρ_{ij}'s larger than 0.3 as -1 and keep the original for ρ_{ij}'s smaller or equal to 0.3. We calculate the $d_{ij} = \sqrt{2(1-\rho_{ij})}$ with the modified ρ_{ij}. We plot histogram of the d_{ij} as following:



Finally, we use the modified matrix to construct the graph and run MST for the modified correlations.



As we know, the nodes within the same economic sector are very close to each other in the vine-cluster structure and the colors of the nodes represent the different economic sectors. Unlike the graph that we plot in problem 3, the Modified Minimal Spanning Tree graph does not show vine-cluster structure.

8. A Generative Model for Vine Cluster Graphs

One generative model for vine cluster graphs I can come up with is the one-factor model. It can produce the star-like minimum spanning tree structures. Furthermore, by modifying the one-factor model, we can fix the problem that the vine-cluster structures of the MST on large scale of data fails to be explained by the correlation with the market mode.