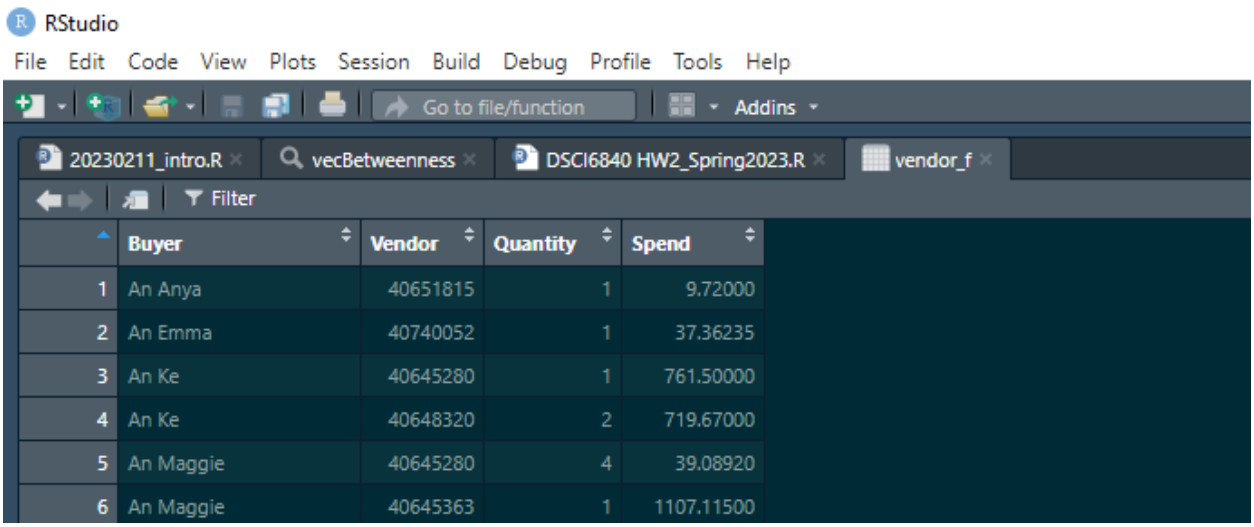
Jonathan Hu
23SP/DSCI-6840-81
HW2
d. 2023-03-06

1. **Task 1: Renaming of column titles**

Having loaded the 6840HW_data.xlsx into dataframe vendor_f, column titles can be renamed by:

$$names(vendor\_f) = c("Buyer","Vendor","Quantity","Spend")$$

Resulting dataframe is View()'d to verify:



	Buyer	Vendor	Quantity	Spend
1	An Anya	40651815	1	9.72000
2	An Emma	40740052	1	37.36235
3	An Ke	40645280	1	761.50000
4	An Ke	40648320	2	719.67000
5	An Maggie	40645280	4	39.08920
6	An Maggie	40645363	1	1107.11500

2. **Task 2 and 3: Degreemost Vendors and Buyers**

One interpretation of centrality/big-ness of nodes in a system is its degree of connectedness: degreeness is, for each node, proportion of (edges to adjacent nodes) over (possible edges to adjacent nodes). It's a little redundant, as adjacency is possible edges between vertices.

In this case, how fully-connected a buyer/vendor is in this system. Using igraph package, the edgelist is built from vendor_f dataframe, and graph g is constructed. The degreeness is calculated, stored in dataframe, and exported to a comma-separated-values (csv) file for further analysis in Excel.

```
# Task 2:
#        using the two columns of "Buyer" and "Vendor" to answer the following questions:
#        Hint: select the two columns as save as a new dataframe

#        1. who are the top 5 most powerful buyers?
#        2. who are the top 5 most powerful vendors?

# we'll create graph based off the vendor data, calculate out degreeness,
# and use excel to find top 5 buyers and top 5 vendors
# the vendors' id's all start with numbers, so a filter and ranking oughta do it
#
# activate igraph package
library(igraph)
edge_df_b = data.frame(vendor_f$Vendor,vendor_f$Buyer)
g = graph_from_data_frame(d=edge_df_b, directed=F)
# plot(g) # for checking
df_degreeness = as.data.frame(degree(g))
write.csv(df_degreeness,"df_degreeness.csv")
```

Jonathan Hu
23SP/DSCI-6840-81
HW2
d. 2023-03-06

Opening the saved df_betweenness.csv file in excel, column headers are cleaned up, and a new column, isNumber(), is created. This column helps sort out buyers from vendors, since from original data, vendors are identified using a number, while buyers use names.

Sorting by degreeness, and filtering by numbertude, top 5 buyers and sellers by degree (thus by fully-connected-ness) are found:

	A	B	C
1	Buyer/vendor	degree(	isNumb
2	40645280	1164	TRUE
3	40740052	883	TRUE
4	40645363	690	TRUE
5	40648320	551	TRUE
6	40651815	544	TRUE
7	40643113	221	TRUE
8	40654222	138	TRUE
9	40676072	54	TRUE
10	40604408	53	TRUE
11	40737080	48	TRUE
12	40685451	46	TRUE
13	40741244	43	TRUE
14	Guo Luna	36	FALSE
15	40687271	35	TRUE
16	40687354	32	TRUE
17	Li Coco	31	FALSE
18	Wang Gary	30	FALSE
19	40660628	28	TRUE
20	40623232	24	TRUE
21	40733285	24	TRUE
22	40640250	23	TRUE
23	Cui Rui	22	FALSE
24	40645710	21	TRUE
25	Pan Jeff	21	FALSE
26	Xu George	21	FALSE

Ranked by degree, but combined

	A	B	C	D
1	Buyer/vendor	degree(	isNumb	
2	40645280	1164	TRUE	
3	40740052	883	TRUE	
4	40645363	690	TRUE	
5	40648320	551	TRUE	
6	40651815	544	TRUE	

Task 2.2: Ranked by degree, filtered to top 5 Vendors

	A	B	C
1	Buyer/vendor	degree(	isNumb
14	Guo Luna	36	FALSE
17	Li Coco	31	FALSE
18	Wang Gary	30	FALSE
23	Cui Rui	22	FALSE
25	Pan Jeff	21	FALSE
26	Xu George	21	FALSE

Task 2.1 : Ranked by degree, filtered to top 5 Buyers

Jonathan Hu
23SP/DSCI-6840-81
HW2
d. 2023-03-06

3. Task 2.3 & Task 2.4 – Betweenness vendors/buyers

To determine who controls most information flow in the network, is the concept of betweenness. Then, betweenness for a nodes is how many paths between pairs of nodes/vertices pass through them. Essentially, who are the biggest middle-persons in the network.

For this is similar to task 2.1 and task 2.2: graph is set up from the edgelist, and betweenness is calculated and exported as csv. Re-loading library and re-creating graph not actually necessary, since should already exist from Task 2.{1,2}

```
#         3. who are the top 5 buyers that control the most information flow in the network?
#         4. who are the top 5 vendors that control the most information flow in the network??

# same as above except controlling information flow is betweenness

# set up graph
# activate igraph package
library(igraph)
edge_df = data.frame(vendor_f$Buyer,vendor_f$vendor)
g = graph_from_data_frame(d=edge_df, directed=F)

# turn into dataframe and save
df_betweenness = as.data.frame(betweenness(g))
write.csv(df_betweenness,"df_betweenness.csv")

# just print reminder of where we are
getwd()
```

Code to generate

	A	B	C
1	buyer/vendc	betwee(	isNumb
2	40645280	1938699	TRUE
3	40740052	1447224	TRUE
4	40645363	890681.1	TRUE
5	40648320	625794.9	TRUE
6	40651815	615897.7	TRUE
7	40643113	185187.4	TRUE
8	40654222	67264.76	TRUE
9	Guo Luna	62421.46	FALSE
10	40685451	50569.91	TRUE
11	40737080	49220.62	TRUE
12	Wang Gary	47553.85	FALSE
13	40676072	37688.1	TRUE
14	Zhang Zhi Qi	36346.09	FALSE
15	Lue Koko	35919.61	FALSE

Betweenness.csv, ranked by betweenness, with isNumber() added to distinguish between buyer/vendor

Jonathan Hu
23SP/DSCI-6840-81
HW2
d. 2023-03-06

	A	B	C
1	buyer/vendc	betwee	isNumb
9	Guo Luna	62421.46	FALSE
12	Wang Gary	47553.85	FALSE
14	Zhang Zhi Qi	36346.09	FALSE
15	Lue Koko	35919.61	FALSE
16	Li Coco	34693.81	FALSE
17	Pan Jeff	33895.57	FALSE

Task 2.3 : top 5 buyers by betweenness (control of information (middlemanliness)) (rank and filter)

	A	B	C
1	buyer/vendc	betwee	isNumb
2	40645280	1938699	TRUE
3	40740052	1447224	TRUE
4	40645363	890681.1	TRUE
5	40648320	625794.9	TRUE
6	40651815	615897.7	TRUE
7	40643113	185187.4	TRUE

Task 2.4 : Top 5 vendors by betweenness (rank and filter)

Jonathan Hu
23SP/DSCI-6840-81
HW2
d. 2023-03-06

Question 2 Task 1 :

Can filter out those spending less than 50 k$ from vendor_f:

```
########## QUESTION 2 ##########

# Task 1: Data Cleaning
#         Only keep spend >= 50000 in vendor_f file

vendor_f_cleaned = vendor_f[vendor_f$"Spend" >= 50000,]
```

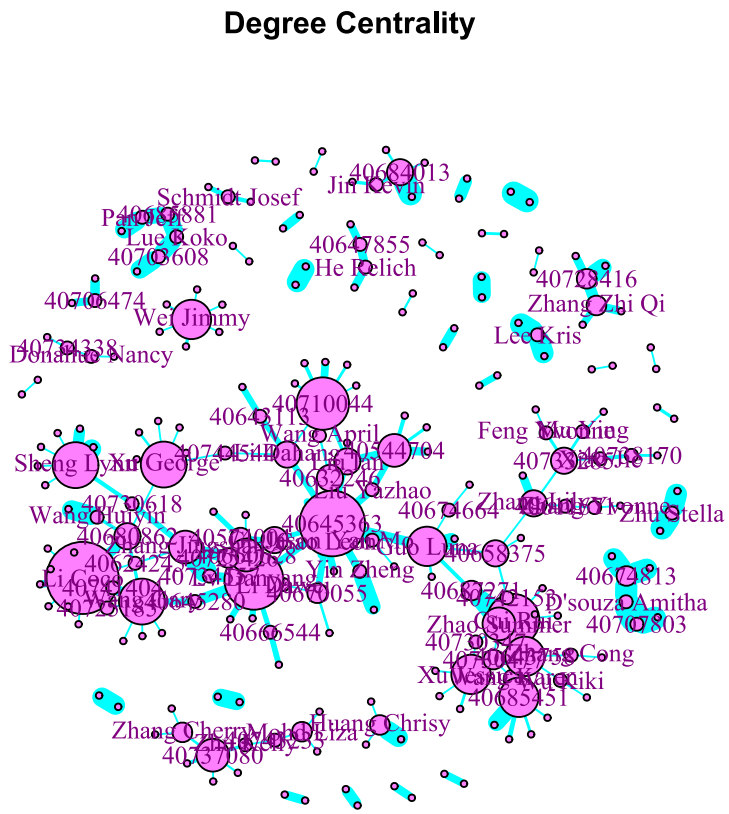
Manually checked in console by using min()

```
> vendor_f_cleaned = vendor_f[vendor_f$"Spend" >= 50000,]
> min(vendor_f$"Spend")
[1] 2.71
> min(vendor_f_cleaned$"Spend")
[1] 50434.02
>
```

Jonathan Hu
23SP/DSCI-6840-81
HW2
d. 2023-03-06

Question 2 Task 2 :

Using only buyer, vendor, and quantity columns of the clean vendor_f_cleaned, a graph is made using igraph package, and plotted. Degreeness is represented by size of nodes, and Quantity represented by width of edges. In order to better print, the scales are adjusted: Degreeness*2, and log(Quantity). Log is used for quantity because quantities span a wide range of magnitudes, from 8 figures of 37836995.770 to 1 figure of 1. Labels of nodes with degree below 2 hidden, though it's still little crowded.

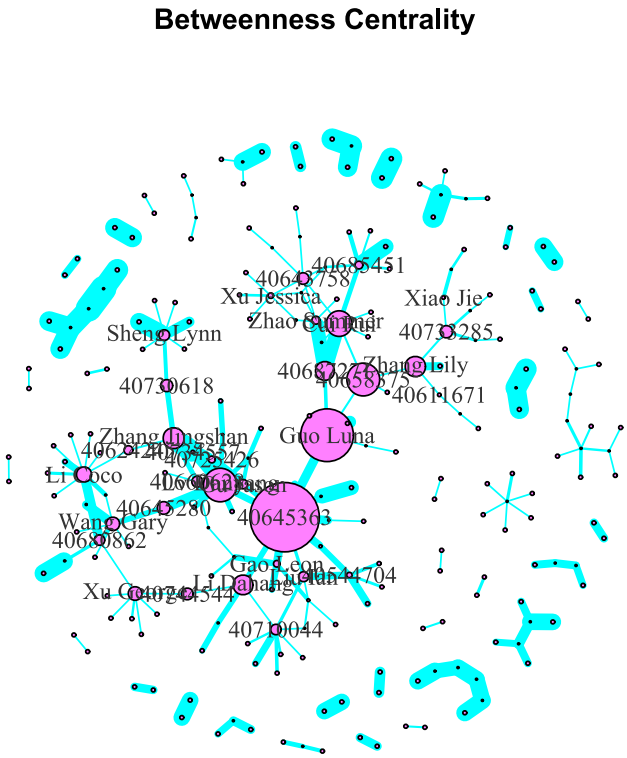


Question 2 Task 2 : part3Plot.svg, plot of vendor_f_cleaned with Degreeness*2 as size of vertices, and log(Quantity) as thickness of nodes.

Conclusions to draw, there are a couple neighborhoods, and some powerful vendors/buyers in the network. The big nodes have high degreeness centrality, and are connected to many as they can. This includes buyers like Li Coco and Sheng Lynn on the left. The big vendors are ones which serve many buyers, such as 40710044 and 40645363 near the middle.

Jonathan Hu
23SP/DSCI-6840-81
HW2
d. 2023-03-06

Question 2 Task 3 :



Question 2 Task 3 : bart3bPlot.svg, plot of vendor_f_cleaned with betweenness*0.005 of node as vertex size, and log(Quantity) of relationship as thickness of edge.

Labels of nodes with betweenness < 200 hidden, for readability's sake.

Conclusions to draw, is the node size represents betweenness of a node, and thus control of information through the network. 40645363 and Luna Guo are very high in centrality, and are significant middle-nodes in buying/vending. Communications-wise, a lot of friend-of-a-friend introductions would pass through these two.

For both graphs, an attempt was made to limit labeling nodes to top quartile (task2) or arbitrary limit (task3) to make the graph a little more legible. Further, svg file format was used over jpg and pdf. Since svg is vector image format, crispness is maintained on zooming in (cf jpg), and it's an image file format (cf pdf).