Assignment4

Xiying Zhao

2023-05-11

Q1. Unsupervised Learning

Use the data on supplier audit and supplier characteristics from the class notes on unsupervised learning and do the following:

```
# load the datasets
d1 = read.csv("C:/Users/zhaox/OneDrive/Desktop/UIUC/BADM 575/Assignment/hw4/SupplierAuditData.csv")
d2 = read.csv("C:/Users/zhaox/OneDrive/Desktop/UIUC/BADM 575/Assignment/hw4/SupplierCharacteristics.csv
```

1. Use the supplier audit dataset to create a data table with the following variables: average detects, standard deviation of defects, average cost, standard deviation of cost, average deliverytime and average flexibility.

```
head(d1)
```

```
##   Month SUP_ID Defects Delivery Unit_Cost Flexibility
## 1     1  SUP_1    16.6        5     25.62         5.5
## 2     2  SUP_1     4.7       11     26.61         1.7
## 3     3  SUP_1     5.1        1     33.81         0.6
## 4     4  SUP_1    17.4        7     33.62         3.2
## 5     5  SUP_1     4.5        5     42.11         2.5
## 6     6  SUP_1    11.0        6     34.38         2.0
```

```
library(sqldf)
```

```
## Warning: package 'sqldf' was built under R version 4.2.3

## Loading required package: gsubfn

## Warning: package 'gsubfn' was built under R version 4.2.3

## Loading required package: proto

## Warning: package 'proto' was built under R version 4.2.3

## Loading required package: RSQLite

## Warning: package 'RSQLite' was built under R version 4.2.3
```

```
dt = sqldf("SELECT SUP_ID,
                   AVG(Defects) AS Avg_Defects,
                   STDEV(Defects) AS SD_Defects,
                   AVG(Unit_Cost) AS Avg_Cost,
                   STDEV(Unit_Cost) AS SD_Cost,
                   AVG(Delivery) AS Avg_Delivery,
                   AVG(Flexibility) AS Avg_Flexibility
            FROM d1
            GROUP BY SUP_ID")
head(dt)
```

```
##   SUP_ID Avg_Defects SD_Defects Avg_Cost  SD_Cost Avg_Delivery Avg_Flexibility
## 1  SUP_1    9.345000   4.578262 30.53417 4.960600     4.750000        2.058333
## 2 SUP_10    9.855000   5.328193 19.74533 4.772947    34.950000        1.825000
## 3 SUP_11   19.711667  11.547113 25.35542 7.257808     4.800000        5.615000
## 4 SUP_12    9.873333   1.013781 35.00600 1.047575     4.933333        1.848333
## 5 SUP_13   31.048333   4.458832 19.96983 2.087725    35.283333        8.913333
## 6 SUP_14   29.780000   5.157184 20.47050 2.093247    35.516667        1.966667
```

2. Use the created dataset to in the question above to create k-means clusters for k = 4. Look at the cluster centers and interpret the meaning of the clusters such as high cost, low quality, etc.

```
# Scale data
dt[,c(2:7)] = apply(dt[,c(2:7)], 2, FUN=scale)
head(dt)
```

```
##   SUP_ID Avg_Defects   SD_Defects   Avg_Cost    SD_Cost Avg_Delivery
## 1  SUP_1   -1.115885 -0.180828434  1.0671727  0.6723130   -1.1479360
## 2 SUP_10   -1.062461  0.061837327 -0.7442456  0.5888993    0.8299220
## 3 SUP_11   -0.029942  2.074182677  0.1976735  1.6934462   -1.1446614
## 4 SUP_12   -1.060540 -1.334238708  1.8179824 -1.0670679   -1.1359291
## 5 SUP_13    1.157612 -0.219474085 -0.7065526 -0.6047105    0.8517527
## 6 SUP_14    1.024750  0.006501509 -0.6224919 -0.6022559    0.8670341
##   Avg_Flexibility
## 1      -0.7050725
## 2      -0.7820210
## 3       0.4678428
## 4      -0.7743262
## 5       1.5555650
## 6      -0.7353023
```

```
x = dt[,c(2:7)]
set.seed(123)
k1 = kmeans(x, 4, nstart = 25)
k1$cluster
```

```
##  [1] 1 3 4 1 2 2 3 4 1 2 2 1 3 4 1 2 2 3 2 2 2 3 4 1 2 2
```

```r
k1$centers
```

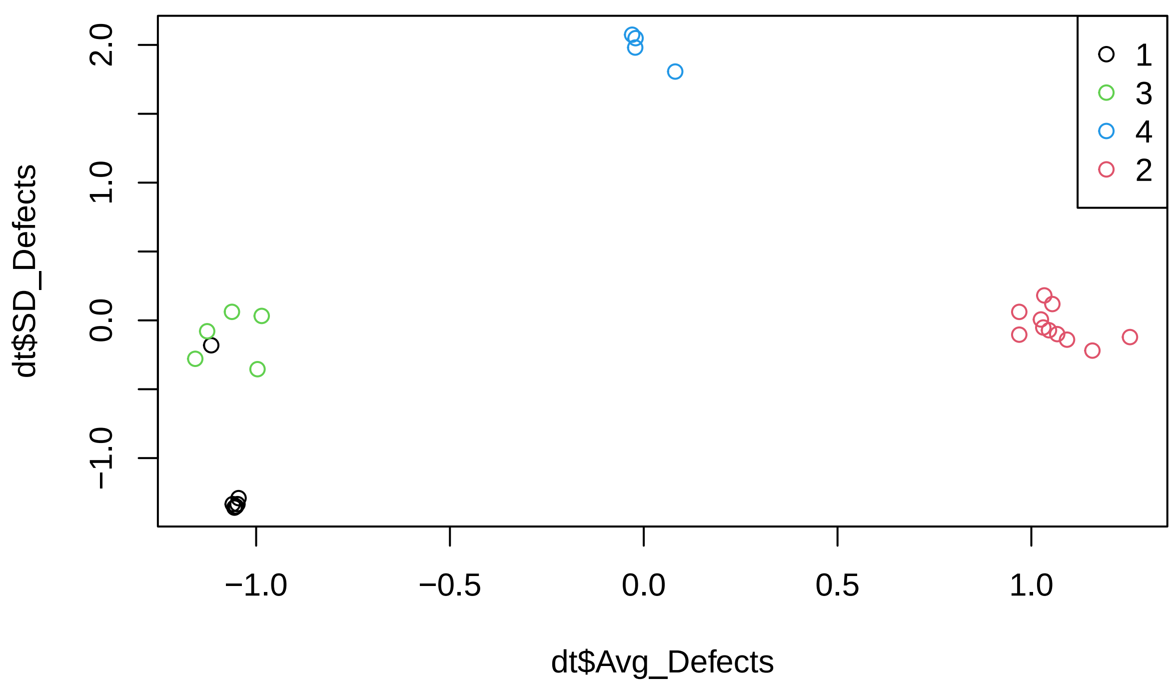
```
##   Avg_Defects  SD_Defects   Avg_Cost    SD_Cost Avg_Delivery Avg_Flexibility
## 1 -1.06298457 -1.14189286  1.6954126 -0.7801029   -1.1417506      -0.7162484
## 2  1.06342898 -0.04006645 -0.6643721 -0.5265891    0.6380106       0.5634288
## 3 -1.06560341 -0.12374102 -0.6763534  0.7672678    0.8714002      -0.7227707
## 4  0.00205142  1.97769830  0.1293462  1.6591897   -1.1311536       0.4284066
```

Interpretation: Cluster 1: High cost with high quality and fast delivery with low flexibility. Cluster 2: Low cost with low quality and slow delivery and high flexibility. Cluster 3: Low cost but high quality and slow delivery with low flexibility. Cluster 4: Average cost with average quality and fast delivery with high flexibility.

3. Plot the clusters with different colors on the following plots, Comment on the graphs.

(i) average quality versus standard deviation of quality,

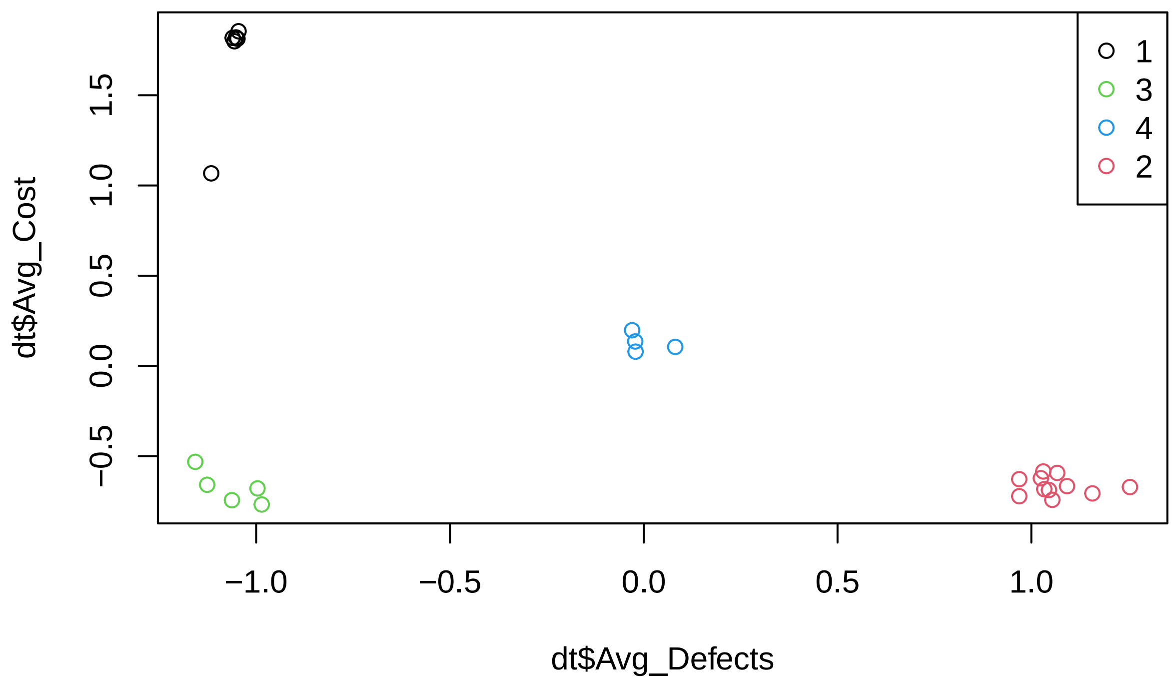
```r
plot(dt$Avg_Defects, dt$SD_Defects, col = k1$cluster)
legend("topright", legend = unique(k1$cluster), col = unique(k1$cluster), pch = 1)
```



Comments: Both Cluster1 and Cluster3 exhibit high quality, with a low number of defects. But Cluster3 has higher variance in quality compared to Cluster1. Cluster4 represents average quality, but with high variance in quality.Cluster2 represents low quality with a variance of quality similar to that of Cluster3.

(ii) average quality versus average costs,

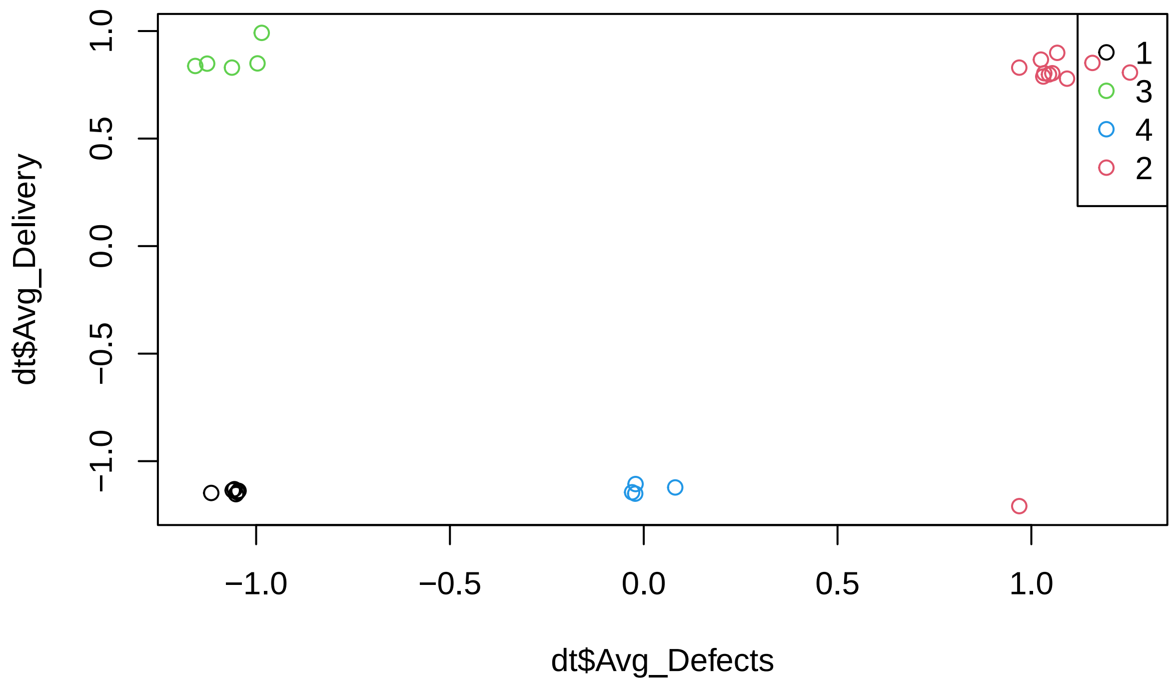
```
plot(dt$Avg_Defects, dt$Avg_Cost, col = k1$cluster)
legend("topright", legend = unique(k1$cluster), col = unique(k1$cluster), pch = 1)
```



Comments: Both Cluster1 and Cluster3 exhibit high quality, with a low number of defects. but they differ in terms of their average costs, with Cluster1 having the highest average cost and Cluster3 having very low average cost. Cluster4 represents average quality with an average cost. Finally, Cluster2 has low average cost similar to that of Cluster3, but exhibits low quality.

(iii) average quality versus average delivery time,

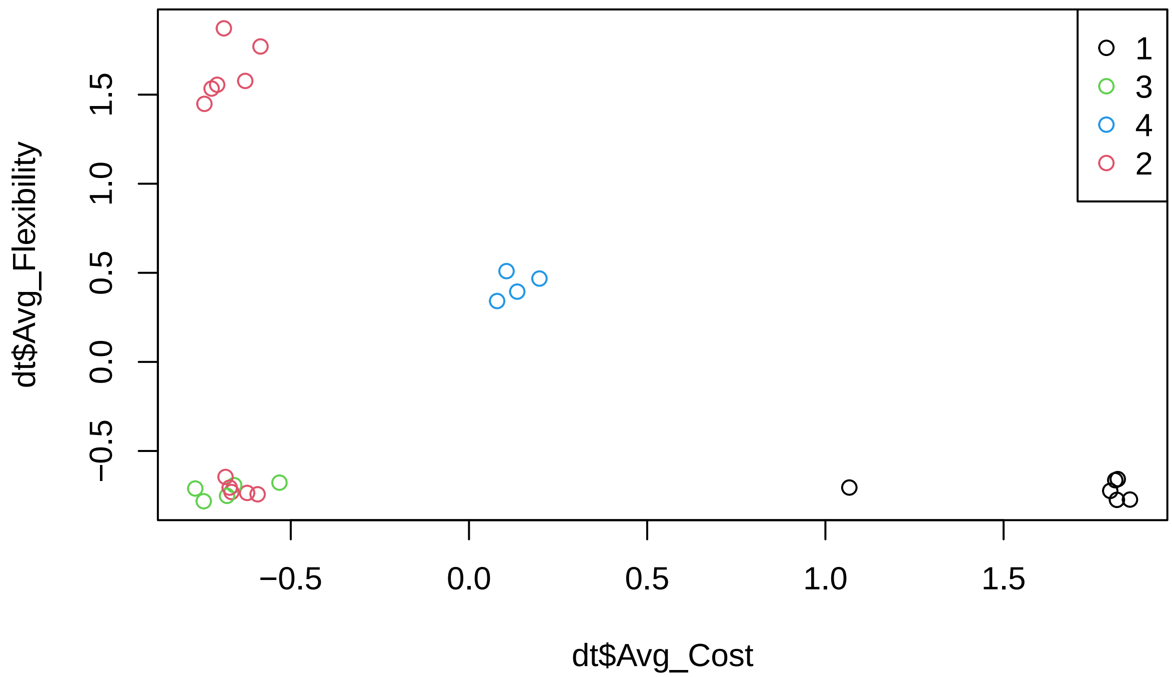
```
plot(dt$Avg_Defects, dt$Avg_Delivery, col = k1$cluster)
legend("topright", legend = unique(k1$cluster), col = unique(k1$cluster), pch = 1)
```



Comments: Both Cluster1 and Cluster3 exhibit high quality, with a low number of defects. but Cluster3 has the longest delivery time and Cluster1 has very fast delivery. Cluster4 represents average quality with fast delivery.While Cluster2 generally exhibits low quality with slow delivery, it is possible that some suppliers within the cluster may offer fast delivery.

(iv) average cost versus average flexibility.

```
plot(dt$Avg_Cost, dt$Avg_Flexibility, col = k1$cluster)
legend("topright", legend = unique(k1$cluster), col = unique(k1$cluster), pch = 1)
```



Comments: Both Cluster1 and Cluster3 has very low flexibility, but Cluster1 has the highest average cost while Cluster3 has the lowest among 4 clusters. Cluster4 represents average cost with average flexibility. Cluster2 represents low cost similar to that of Cluster3 but with a variance of flexibility.

4. Join the cluster assignments with the supplier characteristics dataset. Create the following tables, Comment on the observations.

```
dw = cbind(dt, Cluster = k1$cluster) #Bind the cluster assignment as a column in the data.
colnames(dw)
```

```
## [1] "SUP_ID"          "Avg_Defects"     "SD_Defects"      "Avg_Cost"
## [5] "SD_Cost"         "Avg_Delivery"    "Avg_Flexibility" "Cluster"
```

```
head(dw)
```

```
##   SUP_ID Avg_Defects   SD_Defects   Avg_Cost    SD_Cost Avg_Delivery
## 1  SUP_1   -1.115885 -0.180828434  1.0671727  0.6723130   -1.1479360
## 2 SUP_10   -1.062461  0.061837327 -0.7442456  0.5888993    0.8299220
## 3 SUP_11   -0.029942  2.074182677  0.1976735  1.6934462   -1.1446614
## 4 SUP_12   -1.060540 -1.334238708  1.8179824 -1.0670679   -1.1359291
## 5 SUP_13    1.157612 -0.219474085 -0.7065526 -0.6047105    0.8517527
## 6 SUP_14    1.024750  0.006501509 -0.6224919 -0.6022559    0.8670341
##   Avg_Flexibility Cluster
```

```
## 1      -0.7050725       1
## 2      -0.7820210       3
## 3       0.4678428       4
## 4      -0.7743262       1
## 5       1.5555650       2
## 6      -0.7353023       2
```

```r
head(d2)
```

```
##   SUP_ID Location Partnership Size
## 1  SUP_1        D           I    L
## 2 SUP_10        G           I    L
## 3 SUP_11        D           I    M
## 4 SUP_12        D           P    L
## 5 SUP_13        G           I    S
## 6 SUP_14        G           I    L
```

```r
#Join the cluster assignments to cluster data
d3 = sqldf("SELECT x.SUP_ID,
                    x.Cluster,
                      d2.Location,
                      d2.Partnership,
                      d2.Size
                FROM d2
              LEFT JOIN
           (SELECT SUP_ID, Cluster FROM dw)x
           ON
            d2.SUP_ID=x.SUP_ID")
head(d3)
```

```
##   SUP_ID Cluster Location Partnership Size
## 1  SUP_1       1        D           I    L
## 2 SUP_10       3        G           I    L
## 3 SUP_11       4        D           I    M
## 4 SUP_12       1        D           P    L
## 5 SUP_13       2        G           I    S
## 6 SUP_14       2        G           I    L
```

```r
d3$Cluster = as.factor(d3$Cluster)
```

(i) clusters against size,

```r
t1 = xtabs(~Cluster+Size, data=d3)
t1
```

```
##        Size
## Cluster L M S
##       1 6 0 0
##       2 5 0 6
##       3 5 0 0
##       4 0 4 0
```

Comments: Cluster4 contains all median-sized suppliers, which offer average cost with fast delivery and high flexibility. In contrast, all small-sized suppliers are in Cluster2, which represents low cost, low quality, slow delivery, and high flexibility. The majority of large-sized suppliers are in Cluster1 and Cluster3, both of which represent high quality, but with potential differences in delivery speed. However, some large-sized suppliers are in Cluster2, which means low cost with low quality and slow delivery.

(ii) clusters against partnerships, and

```
t2 = xtabs(~Cluster+Partnership, data = d3)
t2
```

```
##        Partnership
## Cluster  I  P
##       1  1  5
##       2 11  0
##       3  5  0
##       4  4  0
```

Comments: All suppliers with a partnership agreement with buyer are sorted in Cluster1, meaning that a partnership might lead to high cost with high quality and fast delivery. Most of independent suppliers without partnership agreement with buyer are in Cluster2, which represents low cost, low quality, slow delivery, and high flexibility. However, all clusters contain independent suppliers, which implies that the quality of service provided by each independent supplier could vary.

(iii) clusters against location.

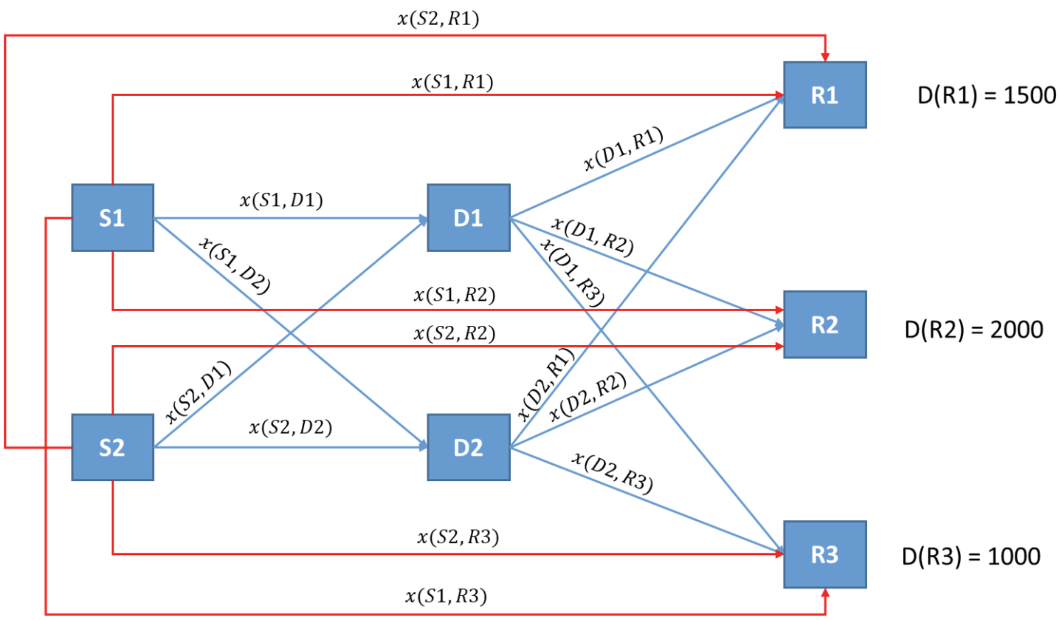
```
t1 = xtabs(~Cluster+Location, data=d3)
t1
```

```
##        Location
## Cluster  D  G
##       1  6  0
##       2  1 10
##       3  0  5
##       4  4  0
```

Comments:

Most of the domestic suppliers are grouped in Cluster1 and Cluster4, indicating that they tend to provide fast delivery services with above average or high quality and cost. On the other hand, global suppliers are mainly located in Cluster2 and Cluster3, both of which represent slow delivery with low cost. It is worth noting that the majority of global suppliers are in Cluster2, indicating that they tend to have low quality.

Q2. Linear Programming.



The suppliers can supply to the distributors, which in turn can supply to the retailers, or the suppliers can also directly supply to any of the retailers. The decision variables are denoted as $x(Origin, Destination)$ indicating the quantity of items transported from the origin to destination. Example: $x(S1, D1)$ indicates the quantity of items delivered from S1 to D1. The suppliers can make any quantity required. The manufacturing cost of the item at S1 is $2000 per unit and the manufacturing cost at S2 is $2400 per unit. The transportation cost per unit of item between any two nodes is shown in Table below.

```
include_graphics("C:/Users/zhaox/OneDrive/Desktop/UIUC/BADM 575/Assignment/hw4/question_table.png")
```

Table 1. Cost of transportation between any two nodes of the network in $/unit.

	Warehouse 1 (W1)	Warehouse 2 (W2)	Retailer 1 (R1)	Retailer 2 (R2)	Retailer 3 (R3)
Supplier 1 (S1)	150	200	325	260	390
Supplier 2 (S2)	400	350	585	650	520
Warehouse 1 (W1)			65	65	97.5
Warehouse 2 (W2)			97.5	32.5	32.5

```
library(lpSolve)
```

1. Write down the total manufacturing cost in terms of the decision variables and the data provides.

The manufacturing cost of the item at S1 is $2000 per unit and the manufacturing cost at S2 is $2400 per unit.

```
Manufacturing <- c(2000,2000,2000,2000,2000,2400,2400,2400,2400,2400,0,0,0,0,0,0)
```

2. Write down the total transportation cost in terms of the decision variables and the data.

```
Transportation <- c(150,200,325,260,390,400,350,585,650,520,65,65,97.5,97.5,32.5,32.5)
```

3. Write down the total cost of manufacturing and transportation in terms of the decision variables.

```
Objective <- Manufacturing+Transportation
```

4. Write down all the constraints for the problem. (Note that there are no supply constraints)

a. Demand constraints.

```
#Constraint matrix LHS
Constraints <- matrix(nrow = 21, ncol = 16)
Constraints[1,] <- c(0,0,1,0,0,0,0,1,0,0,1,0,0,1,0,0)
Constraints[2,] <- c(0,0,0,1,0,0,0,0,1,0,0,1,0,0,1,0)
Constraints[3,] <- c(0,0,0,0,1,0,0,0,0,1,0,0,1,0,0,1)
```

b. Flow balance constraints.

```
Constraints[4,] <- c(1,0,0,0,0,1,0,0,0,0,-1,-1,-1,0,0,0)
Constraints[5,] <- c(0,1,0,0,0,0,1,0,0,0,0,0,0,-1,-1,-1)
```

c. Non-negativity constraints.

```
Constraints[6,] <- c(1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)
Constraints[7,] <- c(0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0)
Constraints[8,] <- c(0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0)
Constraints[9,] <- c(0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0)
Constraints[10,] <- c(0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0)
Constraints[11,] <- c(0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0)
Constraints[12,] <- c(0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0)
Constraints[13,] <- c(0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0)
Constraints[14,] <- c(0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0)
Constraints[15,] <- c(0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0)
Constraints[16,] <- c(0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
Constraints[17,] <- c(0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0)
Constraints[18,] <- c(0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0)
Constraints[19,] <- c(0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0)
Constraints[20,] <- c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0)
Constraints[21,] <- c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1)
```

5. Solve the problem as a linear programming and draw the final optimal network design.

```
#Constraint relationships
Relations <- c(">=",">=",">=","=","=",">=",">=",">=",">=",">=",">=",">=",">=",">=",">=",">=",">=",">=",
#Constraint RHS
Boundaries <- c(1500,2000,1000,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)
#LP formulation - Minimize objective subject to constraints.
lp = lp("min",Objective,Constraints,Relations,Boundaries)
lp$solution
```

```
##  [1] 3500 1000    0    0    0    0    0    0    0    0 1500 2000    0    0    0
## [16] 1000
```

```
include_graphics("C:/Users/zhaox/OneDrive/Desktop/UIUC/BADM 575/Assignment/hw4/Network.png")
```