Quantative Management - Assignment-4

Heart Start produces automated external defibrillators (AEDs) in each of two different plants (A and B). The unit production costs and monthly production capacity of the two plants are indicated in the table below. The AEDs are sold through three wholesalers. The shipping cost from each plant to the warehouse of each wholesaler along with the monthly demand from each wholesaler are also indicated in the table. How many AEDs should be produced in each plant, and how should they be distributed to each of the three wholesaler warehouses so as to minimize the combined cost of production and shipping?

Category	warehouse_1	Warehouse_2	Warehouse_3	Product Cost	Product Capacity
Plant A	22	14	30	600	100
Plant B	16	20	24	625	120
Monthly Deposit	80	60	70		

Here we would be applying 2 methods approach:-

1.Using inequalities

2.Using dummy Variables

Solution 1:Using Inequalities

Assuming that the plant A is producing X11, X12 & X13 units of AEDs for Warehouse 1, Warehouse 2 and Warehouse 3 respectively. And plant B is producing X21, X22 & X23 units of AEDs for Warehouse 1, Warehouse 2 and Warehouse 3 respectively.

So, the total cost will be calculated as sum of Unit Shipping cost to each Warehouse and the Unit production cost:

= 600X11 + 600X12 + 600X13 + 625X21 + 625X22 + 625X23 + 22X11 + 14X12 + 30X13 + 16X21 + 20X22 + 24X23

Where, Z denotes the total cost.

As mentioned above, plant A has monthly production capacity of 100 AEDS. And plant B has monthly production capacity of 120 AEDs. Therefore,

X11 + X12 + X13 + X14<=100 X21 + X22 + X23 <=120

Also, monthly demand for each warehouse is as follows :

X11 + X21 = 80

X12 + X22 = 60

X13 + X23 = 70

Hence, the Linear programming model should be defined as:

The decision variables are:

```
X11 = units of AEDs shipped from Plant A to Warehouse 1
X12 = units of AEDs shipped from Plant A to Warehouse 2
X13 = units of AEDs shipped from Plant A to Warehouse 3
X21 = units of AEDs shipped from Plant B to Warehouse 1
X22 = units of AEDs shipped from Plant B to Warehouse 2
X23 = units of AEDs shipped from Plant B to Warehouse 3
```

Objective function:

```
Minimize   Z = 622X11 + 614X12 + 630X13 + 640X21 + 645X22 + 649X23

**subject to :**
                      X11  +   X12 +   X13 <=100
                  X21 +  X22 +  X23 <=120
                           X11 +   X21 =   80
                           X12 +   X22 =   60
                           X13 +   X33 =   70
                           X14 +   X24 =   10
```

```
And             X11, X12, X13, X21, X22, X23    >=  0
```

Now, we will solve this transportation problem using lpsolve in R.Below is the solution based on the above formulations.

```r
# Solution 1 using inequalities

# Let us set up the Linear problem. Note that we have 6 decision variables and 5 constraints.

library(lpSolveAPI)

lprec <- make.lp(5, 6)

# Set the minimization objective function

set.objfn(lprec, c(622, 614, 630, 641, 645, 649))
lp.control(lprec,sense='min')
```

```
## $anti.degen
## [1] "fixedvars" "stalling"
##
## $basis.crash
## [1] "none"
##
## $bb.depthlimit
## [1] -50
##
## $bb.floorfirst
## [1] "automatic"
##
## $bb.rule
## [1] "pseudononint" "greedy"       "dynamic"      "rcostfixing"
```

```
##
## $break.at.first
## [1] FALSE
##
## $break.at.value
## [1] -1e+30
##
## $epsilon
##       epsb       epsd      epsel     epsint epsperturb   epspivot
##      1e-10      1e-09      1e-12      1e-07      1e-05      2e-07
##
## $improve
## [1] "dualfeas" "thetagap"
##
## $infinite
## [1] 1e+30
##
## $maxpivot
## [1] 250
##
## $mip.gap
## absolute relative
##    1e-11    1e-11
##
## $negrange
## [1] -1e+06
##
## $obj.in.basis
## [1] TRUE
##
## $pivoting
## [1] "devex"    "adaptive"
##
## $presolve
## [1] "none"
##
## $scalelimit
## [1] 5
##
## $scaling
## [1] "geometric"   "equilibrate" "integers"
##
## $sense
## [1] "minimize"
##
## $simplextype
## [1] "dual"   "primal"
##
## $timeout
## [1] 0
##
## $verbose
## [1] "neutral"
```

```r
# Set values for the rows (set the Left hand side constraints)

set.row(lprec, 1, c(1, 1, 1), indices = c(1, 2, 3))
set.row(lprec, 2, c(1, 1, 1), indices = c(4, 5, 6))
set.row(lprec, 3, c(1, 1), indices = c(1, 4))
set.row(lprec, 4, c(1, 1), indices = c(2, 5))
set.row(lprec, 5, c(1, 1), indices = c(3, 6))

# Set the right hand side values

rhs <- c(100, 120, 80, 60, 70)
set.rhs(lprec, rhs)

# Set constraint type and set variable types and bound


set.constr.type(lprec, c("<=", "<=", "=", "=", "="))
set.bounds(lprec, lower = rep(0, 6))

# Finally, name the decision variables (column) and constraints (rows)


lp.rownames <- c("CapacityA", "CapacityB", "DemandW1", "DemandW2", "DemandW3")
lp.colnames <- c("PlantAW1", "PlantAW2", "PlantAW3","PlantBW1", "PlantBW2", "PlantBW3")
dimnames(lprec) <- list(lp.rownames, lp.colnames)

# View the linear problem object to make sure it's correct

lprec
```

```
## Model name:
##             PlantAW1  PlantAW2  PlantAW3  PlantBW1  PlantBW2  PlantBW3
## Minimize         622       614       630       641       645       649
## CapacityA          1         1         1         0         0         0  <=  100
## CapacityB          0         0         0         1         1         1  <=  120
## DemandW1           1         0         0         1         0         0   =   80
## DemandW2           0         1         0         0         1         0   =   60
## DemandW3           0         0         1         0         0         1   =   70
## Kind             Std       Std       Std       Std       Std       Std
## Type            Real      Real      Real      Real      Real      Real
## Upper            Inf       Inf       Inf       Inf       Inf       Inf
## Lower              0         0         0         0         0         0
```

```r
# Save this into a file
write.lp(lprec, filename = "AED_firstSolution.lp", type = "lp")
solve(lprec) # Now, solve the model
```

```
## [1] 0
```

```r
# Show the value of objective function, variables, constraints and slack

get.objective(lprec)
```

```
## [1] 132790
```

```
get.variables(lprec)
```

```
## [1]  0 60 40 80  0 30
```

```
get.constraints(lprec)
```

```
## [1] 100 110  80  60  70
```

```
get.constraints(lprec) - rhs
```

```
## [1] -2.842171e-14 -1.000000e+01 -1.421085e-14  0.000000e+00  0.000000e+00
```

Observations:

1. We were **successfully able to solve** the LP problem using inequalities.

2.The minimized combined cost of production and shipping is **132790**.

3.The AED should be produced for each plant based on each of the **3 warehouses are 0,60,40,80,0,30**.for Plant1 warehouse 1,Plant1 warehouse 2,Plant 1 warehouse 3,Plant 2 warehouse 1, Plant2 warehouse 2,Plant2 warehouse 3 respectively.

Now let's take look on the dummy variable approach and its outcome.

Solution 2: Using Dummy Variables

Since, the monthly demand is less than the monthly production capacity. Let's create dummy Warehouse (Warehouse 4) which can accommodate 10 units. The unit shipping cost to the warehouse 4 is $0 since it is dummy warehouse.

Category	warehouse_1	Warehouse_2	Warehouse_3	Warehouse_4	Product Cost	Product Capacity
Plant A	22	14	30	0	600	100
Plant B	16	20	24	0	625	120
Monthly Deposit	80	60	70	10		

So now Assume that the plant A produces X11, X12, X13 and X14 units of AEDs for Warehouse 1, Warehouse 2, Warehouse 3 and Warehouse 4 respectively. And plant B produces X21, X22, X23 & X24 units of AEDs for Warehouse 1, Warehouse 2, Warehouse 3 & Warehouse 4 respectively.

So, the total cost will be calculated as sum of Unit Shipping cost to each Warehouse and the Unit production cost:

$= 600X11 + 600X12 + 600X13 + 625X21 + 625X22 + 625X23 + 22X11 + 14X12 + 30X13 + 0X14 + 16X21 + 20X22 + 24X23 + 0X24$

Where, Z denotes the total cost. Now, plant A has monthly production capacity of 100 AEDS. And plant B has monthly production capacity of 120 AEDs. Therefore, $X11 + X11 + X13 + X14 = 100$ $X21 + X22 + X23 + X24 = 120$

Also, monthly demand for each warehouse is as follows

X11 + X21 = 80

X12 + X22 = 60

X13 + X23 = 70

X14 + X24 = 10

Hence, the Linear programming model should be defined as:

The decision variables are:

```
  X11 = units of AEDs shipped from Plant A to Warehouse 1
  X12 = units of AEDs shipped from Plant A to Warehouse 2
  X13 = units of AEDs shipped from Plant A to Warehouse 3
  X14 = units of AEDs shipped from Plant A to Warehouse 4
  X21 = units of AEDs shipped from Plant B to Warehouse 1
  X22 = units of AEDs shipped from Plant B to Warehouse 2
  X23 = units of AEDs shipped from Plant B to Warehouse 3
  X24 = units of AEDs shipped from Plant B to Warehouse 4
```

Minimize = 622X11 + 614X12 + 630X13 + 0X14 + 640X21 + 645X22 + 649X23 + 0X24

subject to X11 + X12 + X13 + X14 = 100 X21 + X22 + X23 + X24 = 120 X11 + X21 = 80 X12 + X22 = 60 X13 + X33 = 70 X14 + X24 = 10

```
And            X11, X12, X13, X14, X21, X22, X23, X24  >=  0

Now , lets solve the dummy variable solution using lpsolve in R.
```

```r
# Let us set up the Linear problem. Note that we had 8 decision variables and 6 constraints.

lprec_dummy <- make.lp(6, 8)

# Setting the minimization objective function

set.objfn(lprec_dummy, c(622, 614, 630, 0, 641, 645, 649, 0))
lp.control(lprec_dummy, sense='min')
```

```
## $anti.degen
## [1] "fixedvars" "stalling"
##
## $basis.crash
## [1] "none"
##
## $bb.depthlimit
## [1] -50
##
## $bb.floorfirst
## [1] "automatic"
##
## $bb.rule
## [1] "pseudononint" "greedy"       "dynamic"      "rcostfixing"
##
```

```
## $break.at.first
## [1] FALSE
##
## $break.at.value
## [1] -1e+30
##
## $epsilon
##       epsb       epsd      epsel     epsint epsperturb   epspivot
##      1e-10      1e-09      1e-12      1e-07      1e-05      2e-07
##
## $improve
## [1] "dualfeas" "thetagap"
##
## $infinite
## [1] 1e+30
##
## $maxpivot
## [1] 250
##
## $mip.gap
## absolute relative
##    1e-11    1e-11
##
## $negrange
## [1] -1e+06
##
## $obj.in.basis
## [1] TRUE
##
## $pivoting
## [1] "devex"    "adaptive"
##
## $presolve
## [1] "none"
##
## $scalelimit
## [1] 5
##
## $scaling
## [1] "geometric"   "equilibrate" "integers"
##
## $sense
## [1] "minimize"
##
## $simplextype
## [1] "dual"   "primal"
##
## $timeout
## [1] 0
##
## $verbose
## [1] "neutral"
```

```r
# Setting values for the rows (set the Left hand side constraints)
set.row(lprec_dummy, 1, c(1, 1, 1, 1), indices = c(1, 2, 3, 4))
set.row(lprec_dummy, 2, c(1, 1, 1, 1), indices = c(5, 6, 7, 8))
set.row(lprec_dummy, 3, c(1, 1), indices = c(1, 5))
set.row(lprec_dummy, 4, c(1, 1), indices = c(2, 6))
set.row(lprec_dummy, 5, c(1, 1), indices = c(3, 7))
set.row(lprec_dummy, 6, c(1, 1), indices = c(4, 8))

# Setting the right hand side values

rhs <- c(100, 120, 80, 60, 70, 10)
set.rhs(lprec_dummy, rhs)

# Setting constraint type and set variable types and bound

set.constr.type(lprec_dummy, c("=", "=", "=", "=", "=", "="))
set.bounds(lprec_dummy, lower = rep(0, 8))

# setting the decision variables (column) and constraints (rows)names

lp.rownames <- c("CapacityA", "CapacityB", "DemandW1", "DemandW2", "DemandW3", "DemandW4")
lp.colnames <- c("PlantAW1", "PlantAW2", "PlantAW3", "PlanntAW4", "PlantBW1", "PlantBW2", "PlantBW3", "I
dimnames(lprec_dummy) <- list(lp.rownames, lp.colnames)

# View the linear program object to make sure it's correct

lprec_dummy
```

```
## Model name:
##              PlantAW1   PlantAW2   PlantAW3  PlanntAW4   PlantBW1   PlantBW2   PlantBW3   PlantBW4
## Minimize          622        614        630          0        641        645        649          0
## CapacityA           1          1          1          1          0          0          0          0  =
## CapacityB           0          0          0          0          1          1          1          1  =
## DemandW1            1          0          0          0          1          0          0          0  =
## DemandW2            0          1          0          0          0          1          0          0  =
## DemandW3            0          0          1          0          0          0          1          0  =
## DemandW4            0          0          0          1          0          0          0          1  =
## Kind              Std        Std        Std        Std        Std        Std        Std        Std
## Type             Real       Real       Real       Real       Real       Real       Real       Real
## Upper             Inf        Inf        Inf        Inf        Inf        Inf        Inf        Inf
## Lower               0          0          0          0          0          0          0          0
```

```r
# solving the model

solve(lprec_dummy)
```

```
## [1] 0
```

```r
# Show the value of objective function, variables, constraints


get.objective(lprec_dummy)
```

```
## [1] 132790
```

```r
get.variables(lprec_dummy)
```

```
## [1]  0 60 40  0 80  0 30 10
```

```r
get.constraints(lprec_dummy)
```

```
## [1] 100 120  80  60  70  10
```

```r
get.constraints(lprec_dummy) - rhs
```

```
## [1]  0.000000e+00  0.000000e+00  0.000000e+00 -7.105427e-15  0.000000e+00
## [6]  3.552714e-15
```

```r
#saving into lp file
write.lp(lprec_dummy, filename = "AED_dummySOlution.lp", type = "lp")


# Read from file and solve it

x <- read.lp("AED_dummySOlution.lp")     # create an lp object x
x                          # display x
```

```
## Model name:
##              PlantAW1  PlantAW2  PlantAW3  PlantBW1  PlantBW2  PlantBW3  PlanntAW4  PlantBW4
## Minimize          622       614       630       641       645       649          0         0
## CapacityA           1         1         1         0         0         0          1         0  =
## CapacityB           0         0         0         1         1         1          0         1  =
## DemandW1            1         0         0         1         0         0          0         0  =
## DemandW2            0         1         0         0         1         0          0         0  =
## DemandW3            0         0         1         0         0         1          0         0  =
## DemandW4            0         0         0         0         0         0          1         1  =
## Kind              Std       Std       Std       Std       Std       Std        Std       Std
## Type             Real      Real      Real      Real      Real      Real       Real      Real
## Upper             Inf       Inf       Inf       Inf       Inf       Inf        Inf       Inf
## Lower               0         0         0         0         0         0          0         0
```

```r
solve(x)                   # Solution
```

```
## [1] 0
```

```r
get.objective(x)           # get objective value
```

```
## [1] 132790
```

```r
get.variables(x)           # get values of decision variables
```

```
## [1]  0 60 40 80  0 30  0 10
```

```r
get.constraints(x)          # get constraints
```

```
## [1] 100 120  80  60  70  10
```

Observations:

1. We were **successfully able to solve** the LP problem using dummy variables.

2.The minimized combined cost of production and shipping is **132790**.

3.The AED should be produced for each plant based on each of the 4 warehouses are 0,60,40,80,0,30,0,10 for Plant1 warehouse 1,Plant1 warehouse 2,Plant 1 warehouse 3,Plant 1 warehouse 4,Plant 2 warehouse 1, Plant2 warehouse 2,Plant2 warehouse 3 ,Plant 2 warehouse 4 respectively.

4.Therefore,we can say that the **minimized cost is same using dummy variable approach and inequalities approach**.

Question 2:

Oil Distribution Texxon Oil Distributors, Inc., has three active oil wells in a west Texas oil field. Well 1 has a capacity of 93 thousand barrels per day (TBD), Well 2 can produce 88 TBD, and Well 3 can produce 95 TBD. The company has five refineries along the Gulf Coast, all of which have been operating at stable demand levels. In addition, three pump stations have been built to move the oil along the pipelines from the wells to the refineries. Oil can flow from any one of the wells to any of the pump stations, and from any one of the pump stations to any of the refineries, and Texxon is looking for a minimum cost schedule. The refineries' requirements are as follows.

1) What is the minimum cost of providing oil to the refineries? Which wells are used to capacity in the optimal schedule? Formulation of the problem is enough.

Solution:

The transshipment model is an extension of the transportation model in which intermediate transshipment points are added between the sources and destinations.Therefore ,It seems Texxon Oil distrbuters problem would need to follow an **Transshipment model approach**.

Formulation of the above problem seems to be **similiar to the standard Transportation problem (special type of LPP) but it has two phases**. Firstly oil moves from **three wells to any of the three pumps(Transshipment point)**, second phase is movement of oil **from three pumps to any of the five Refineries**.

Therefore, we need to determine out how much quantity of oil **from which well -> refinery -> which pump station** should be moved so that the total cost of these movements is minimum subject to availability and requirement constraints mentioned above.

Based on the daily cost given in the table in dollars per thousand barrels and considering the company's cost accounting system recognized charges by the segment of pipeline that is used, we can determine the mimimum cost of providing oil to refineries with below formulation:

- **Objective:**

Minimize total cost

= 1.52X14 + 1.6X15 + 1.4X16 + 1.7X24 + 1.63X25 + 1.55X26 + 1.45X34 + 1.57X35 + 1.3X36 + 5.15X47 + 5.69X48 + 6.13X49 + 5.63X410 + 5.8X411 + 5.12X57 + 5.47X58 + 6.05X59 + 6.12X510 + 5.71X511 + 5.32X67 + 6.16X68 + 6.25X69 + 6.17X610 + 5.87X611

- **subject to** :

X14 + X15 + X16 <=93 (Supply at Well 1)

X24 + X25 + X26 <=88 (Supply at Well 2)

X34 + X35 + X36 <=95 (Supply at Well 3)

X47 + X57 + X67 = 30 (Demand at Refinery 1)

X48 + X58 + X68 = 57 (Demand at Refinery 2)

X49 + X59 + X69 = 48 (Demand at Refinery 3)

X4,10 + X5,10 + X6, 10 = 91 (Demand at Refinery 4)

X4, 11 + X5, 11 + X6, 11 = 48 (Demand at Refinery 5)) X14+ X24+ X34= X47+ X48+ X49+ X410+ X411 (Shipping through Pump 1)

X15 + X25 + X35 = X57 + X58 + X59 + X510 + X511 (Shipping through Pump 2)

X16 + X26 + X36 = X67 + X68 + X69 + X610 + X611 (Shipping through Pump 3)

Xij 0 for all i and j (Non negativity constraint)

Now, in order to determine the which well will be used at its capacity , we need to solve the LP problem formulated above:

```
#loading manually created LP file based on the above formulation

Oil_D <- read.lp("C:/Users/khush/Documents/Oil_distribution.lp")      # create an lp object x
Oil_D                              # display x
```

```
## Model name:
##   a linear program with 24 decision variables and 11 constraints
```

```
solve(Oil_D)                       # Solution
```

```
## [1] 0
```

```
get.objective(Oil_D)               # get objective value
```

```
## [1] 1963.82
```

```
get.variables(Oil_D)               # get values of decision variables
```

```
##  [1] 93  0  0  0 86  0 28  0 67 30  0  0 91  0  0 57 29  0  0  0  0 19  0 48
```

```
get.constraints(Oil_D)             # get constraints
```

```
##  [1] 93 86 95 30 57 48 91 48  0  0  0
```

Observation:

Based on the above solution , we can see that **Well 1 and Well 3** will be used at its maximum capacity.(values: 93 and 95 respectively for Well 1 and Well 3). Well 2 is using only 86 from 88 capacity.

2) Show the network diagram corresponding to the solution in (a). That is, label each of the arcs in the solution and verify that the flows are consistent with the given information.

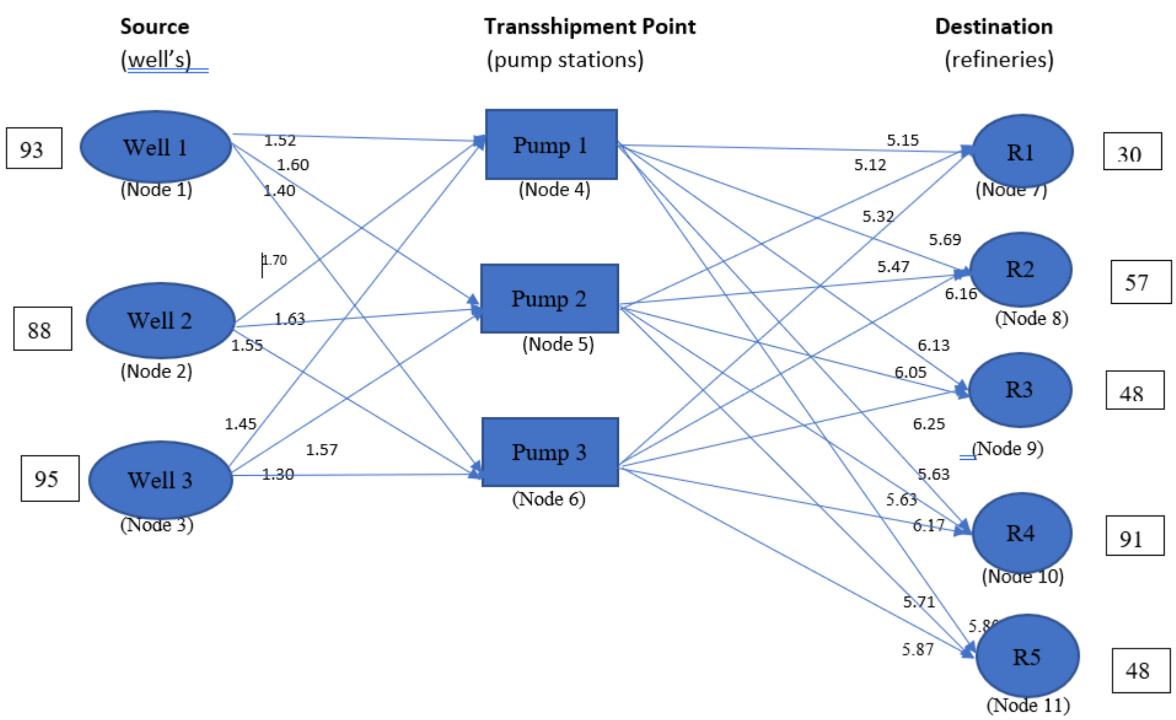
Pump stations(Transshipment Points) and the number of units shipped from each stations to each final destination (refinery plants), as these are the decisions management must make. Assuming the decision variables as number of units shipped from location (node) i to location (node) j we get,

i = 1, 2, 3, 4, 5, 6

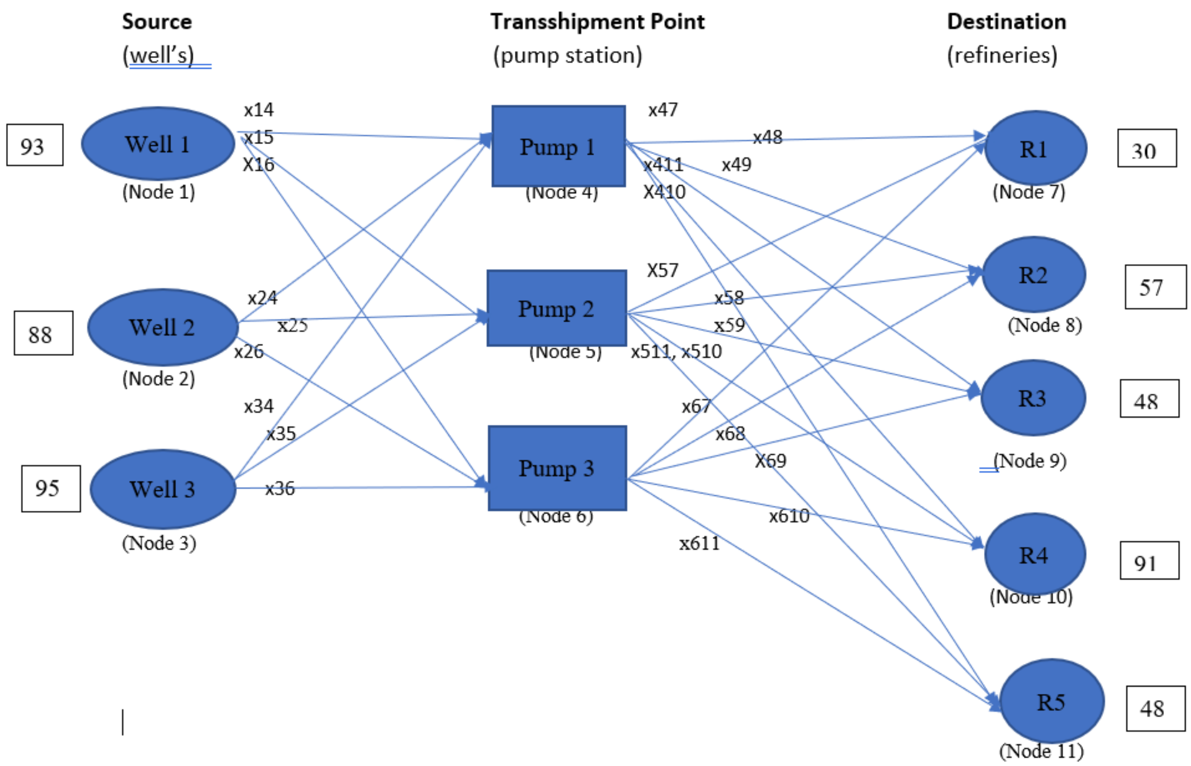
j = 4, 5, 6, 7, 8, 9, 10, 11

In the below figure we can see nodes as the numbers and assigned one variable for each arc:

```
#  install.packages("BiocManager")
#  BiocManager::install("EBImage")

library("EBImage")

#f = system.file("images", "abc.PNG", package="EBImage")
img = readImage("C:/Users/khush/Documents/images/Question2_Network_Daigram_v1.PNG")
img2 = readImage("C:/Users/khush/Documents/images/Question2_Network_Daigram_final.PNG")



display(img)
```



```
display(img2)
```



Based on the second network Diagram we can see derivation of below equations(constraints):

- **Supply at Well**

X14 + X15 + X16   93 (Supply at Well 1 [**node 1**])

X24 + X25 + X26   88 (Supply at Well 2 [**node 2**])

X34 + X35 + X36   95 (Supply at Well 3 [**node 3**])

- **Demand at Refinery**

X47 + X57 + X67 = 30 (Demand at Refinery 1 [**node 7**])

X48 + X58 + X68 = 57 (Demand at Refinery 2 [**node 8**])

X49 + X59 + X69 = 48 (Demand at Refinery 3 [**node 9**])

X4,10 + X5,10 + X6, 10 = 91 (Demand at Refinery 4 [**node 10**])

X4, 11 + X5, 11 + X6, 11 = 48 (Demand at Refinery 5 [**node 11**])

- **Shipping through Pump stations**

$X14 + X24 + X34 = X47 + X48 + X49 + X4,10 + X4,11$ (Shipping through Pump 1 [**node 4**])

$X15 + X25 + X35 = X57 + X58 + X59 + X5,10 + X5,11$ (Shipping through Pump 2 [**node 5**])

$X16 + X26 + X36 = X67 + X68 + X69 + X6,10 + X6,11$ (Shipping through Pump 3 *[**node 6**])

Xij   0 for all i and j (Non negativity constraint)