Assignment 4

SABIHA MHATARNAIK

10/23/2021

Q.1) ***Solving this transportation problem utilizing R***

Adding the LP Objective function and constraint

We will be using "lpSolveAPI" package in R

```r
#Import the lpSolveAPI package
library(lpSolveAPI)
```

```r
#Creating an lp object named lptrans with 0 constraints and 6 decision variables.
lptrans <- make.lp(0,6)

#Set the objective function.Since,minimization problem is the default so we don't
#have to change the direction.
set.objfn(lptrans, c(622,614,630,
                     641,645,649))

#Add the 9 constraints based on the number of units made on each plant and
#shipped to different warehouses.

add.constraint(lptrans, c(1,1,1,0,0,0), "<=", 100)
add.constraint(lptrans, c(0,0,0,1,1,1), "<=", 120)

add.constraint(lptrans, c(1,0,0,1,0,0), "=", 80)
add.constraint(lptrans, c(0,1,0,0,1,0), "=", 60)
add.constraint(lptrans, c(0,0,1,0,0,1), "=", 70)
```

Setting the bounds for the variables. Also, it is important to note that all variables have to be non-negative. There is no need to do this as this is by default,but we can set the bounds explicitly.

```r
#Set bounds for variables.
set.bounds(lptrans ,lower =c(0,0,0,0,0,0),
           columns= c(1:6)) # not really needed

#To identify the variables and constraints, we can
# set variable names and name the constraints
RowNames <- c ("PrdtnCapPlantA" ,"PrdctnCapPlantB",
               "DemandWh1","DemandWh2","DemandWh3")

ColNames <- c("PlantAWh1","PlantAWh2","PlantAWh3",
              "PlantBWh1","PlantBWh2","PlantBWh3")
```

```r
dimnames(lptrans) <- list(RowNames,ColNames)

#Now print out the model
lptrans
```

```
## Model name:
##                 PlantAWh1  PlantAWh2  PlantAWh3  PlantBWh1  PlantBWh2  PlantBWh3
## Minimize              622        614        630        641        645        649
## PrdtnCapPlantA          1          1          1          0          0          0  <=  100
## PrdctnCapPlantB         0          0          0          1          1          1  <=  120
## DemandWh1               1          0          0          1          0          0   =   80
## DemandWh2               0          1          0          0          1          0   =   60
## DemandWh3               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
#Saving to a file
write.lp(lptrans,filename="AED.lp",type="lp")
```

Now solve the lp problem

```r
solve(lptrans)
```

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

The output above indicate that the result is 0, means that there was a successful solution We now output the value of the objective function, and the variables.

```r
#It will give us the minimum value for the optimal solution.
get.objective(lptrans)
```

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

Hence \$132790 is the minimum combined cost of production and shipping for the optimal solution.

```r
#It will give us the optimal number of units produced on each plant and shipped to each warehouse.
get.variables(lptrans)
```

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

In order to get the minimum combined cost of production and shipping each plant should produce and ship the following amounts: Plant A should produce 100 units. the company should ship 60 units of plant A to warehouse 2 and 40 units to warehouse 3. Plant B should produce 110 units .The company should ship 80 units of plant B to warehouse 1 and 30 units to warehouse 3

```r
#It will give us the optimal constraints
get.constraints(lptrans)
```

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

This output is important that shows the correct values in order to satisfy the equality assumption.The equality assumption in a transportation problem says that the given problem will have a feasible solution iff total production= total demand. Hence, the constraints on plant B must be reduced from 120 to 110 to have a feasible solution.

```
get.sensitivity.obj(lptrans) #get Reduced Costs
```

```
## $objfrom
## [1] -1.00e+30 -1.00e+30  6.18e+02 -1.00e+30  6.33e+02  6.49e+02
##
## $objtill
## [1] 6.22e+02 6.26e+02 6.30e+02 6.41e+02 1.00e+30 6.61e+02
```

```
get.sensitivity.rhs(lptrans) #get Shadow Price
```

```
## $duals
##  [1] -19   0 641 633 649   0   0   0   0  12   0
##
## $dualsfrom
##  [1]  9e+01 -1e+30  0e+00  3e+01  4e+01 -3e+01 -1e+30 -1e+30 -1e+30 -4e+01
## [11] -1e+30
##
## $dualstill
##  [1] 1.3e+02 1.0e+30 9.0e+01 7.0e+01 8.0e+01 4.0e+01 1.0e+30 1.0e+30 1.0e+30
## [10] 3.0e+01 1.0e+30
```

Dummy Variable Approach

```
#Creating an lp object named lptrans with 0 constraints and 8 decision variables.
lptrans_dummy<- make.lp(0,8)

#Set the objective function.Since,minimization problem is the default so we don't
#have to change the direction.
set.objfn(lptrans_dummy, c(622,614,630,0,
                     641,645,649,0))
```

Add the 8 constraints based on the number of units made on each plant and shipped to different warehouses and the dummy variable that is needed which will be a dummy warehouse 4 to satisfy the equality assumption i.e total production = total demand.

```
add.constraint(lptrans_dummy, c(1,1,1,1,0,0,0,0), "=", 100)
add.constraint(lptrans_dummy, c(0,0,0,0,1,1,1,1), "=", 120)

add.constraint(lptrans_dummy, c(1,0,0,0,1,0,0,0), "=", 80)
add.constraint(lptrans_dummy, c(0,1,0,0,0,1,0,0), "=", 60)
add.constraint(lptrans_dummy, c(0,0,1,0,0,0,1,0), "=", 70)
add.constraint(lptrans_dummy, c(0,0,0,1,0,0,0,1), "=", 10) #Dummy Variable

#Set bounds for variables.
```

```r
set.bounds(lptrans_dummy ,lower =c(0,0,0,0,0,0,0,0),
           columns= c(1:8)) # not really needed

#To identify the variables and constraints, we can
# set variable names and name the constraints
RowNames_dummy <- c ("PrdtnCapPltA" ,"PrdctnCapPltB",
               "DemandWh1","DemandWh2","DemandWh3","DemandWh4")

ColNames_dummy <- c("PltAWh1","PltAWh2","PltAWh3","PltAWh4",
              "PltBWh1","PltBWh2","PltBWh3","PltBWh4")

dimnames(lptrans_dummy) <- list(RowNames_dummy,ColNames_dummy)

#Now print out the model
lptrans_dummy
```

```
## Model name:
##                  PltAWh1  PltAWh2  PltAWh3  PltAWh4  PltBWh1  PltBWh2  PltBWh3  PltBWh4
## Minimize             622      614      630        0      641      645      649        0
## PrdtnCapPltA           1        1        1        1        0        0        0        0  =  100
## PrdctnCapPltB          0        0        0        0        1        1        1        1  =  120
## DemandWh1              1        0        0        0        1        0        0        0  =   80
## DemandWh2              0        1        0        0        0        1        0        0  =   60
## DemandWh3              0        0        1        0        0        0        1        0  =   70
## DemandWh4              0        0        0        1        0        0        0        1  =   10
## 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 the transportation problem
solve(lptrans_dummy)
```

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

The output above indicate that the result is 0, means that there was a successful solution.

```r
#It will give us the minimum value for the optimal solution.
get.objective(lptrans_dummy)
```

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

Same optimal solution as that of the previous approach

```r
#It will give us the optimal number of units produced on each plant and shipped to each warehouse.
get.variables(lptrans_dummy)
```

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

Here, the additional 10 units are logically sent to warehouse 4,which is needed to satisfy th equality assumption.Warehouse 4 is not a physical location, but it means 10 units must be shipped to any warehouse to get a feasible solution.

```r
#It will give us the optimal constraints
get.constraints(lptrans_dummy)
```

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

Here, all constraints are satisfied.

```r
get.sensitivity.obj(lptrans_dummy) # Reduced Cost
```

```
## $objfrom
## [1]  6.22e+02 -1.00e+30  6.18e+02 -1.90e+01 -1.00e+30  6.33e+02  6.49e+02
## [8] -1.00e+30
##
## $objtill
## [1] 1.00e+30 6.26e+02 6.30e+02 1.00e+30 6.41e+02 1.00e+30 6.61e+02 1.90e+01
```

```r
#It will give us the optimal constraints
get.sensitivity.rhs(lptrans_dummy) # Shadow Price
```

```
## $duals
##  [1]  614  633    8    0   16 -633    0    0    0   19    0   12    0    0
##
## $dualsfrom
##  [1]  1.0e+02  1.2e+02  8.0e+01 -1.0e+30  7.0e+01  1.0e+01 -3.0e+01 -1.0e+30
##  [9] -1.0e+30 -3.0e+01 -1.0e+30 -4.0e+01 -1.0e+30 -1.0e+30
##
## $dualstill
##  [1] 1.0e+02 1.2e+02 8.0e+01 1.0e+30 7.0e+01 1.0e+01 4.0e+01 1.0e+30 1.0e+30
## [10] 1.0e+01 1.0e+30 3.0e+01 1.0e+30 1.0e+30
```

Also, we can read the lp formulation using an lp file and solve it.

```r
x <- read.lp("AED.lp")
x
```

```
## Model name:
##                   PlantAWh1  PlantAWh2  PlantAWh3  PlantBWh1  PlantBWh2  PlantBWh3
## Minimize                622        614        630        641        645        649
## PrdtnCapPlantA            1          1          1          0          0          0  <=  100
## PrdctnCapPlantB           0          0          0          1          1          1  <=  120
## DemandWh1                 1          0          0          1          0          0   =   80
## DemandWh2                 0          1          0          0          1          0   =   60
## DemandWh3                 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
solve(x)
```

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

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

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

```r
get.variables(x) #get the values f decision variables
```

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

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

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

Q.2) ***Oil Distribution***

1)

Solving this transportation problem utilizing R

```r
#Creating an lp object named lptrans with 0 constraints and 27 decision variables.
lptransship<-make.lp(0,27)
lp.control(lptransship,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"
```

```
#Set the objective function.
set.objfn(lptransship,c(1.52,1.60,1.40,1.70,1.63,1.55,1.45,1.57,1.30,5.15,5.12,5.32,5.69,5.47,6.16,6.13
#Adding the constraints
add.constraint(lptransship,c(1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0),"=",93)
add.constraint(lptransship,c(0,0,0,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0),"=",88)
add.constraint(lptransship,c(0,0,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0),"=",95)
add.constraint(lptransship,c(0,0,0,0,0,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0),"=",30)
add.constraint(lptransship,c(0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0),"=",57)
add.constraint(lptransship,c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0),"=",48)
add.constraint(lptransship,c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,0,0,0,0,0,0),"=",91)
add.constraint(lptransship,c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,0,0,0),"=",48)
add.constraint(lptransship,c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1),"=",2)
add.constraint(lptransship,c(1,0,0,1,0,0,1,0,0,-1,0,0,-1,0,0,-1,0,0,-1,0,0,-1,0,0,-1,0,0),"=",0)
add.constraint(lptransship,c(0,1,0,0,1,0,0,1,0,0,-1,0,0,-1,0,0,-1,0,0,-1,0,0,-1,0,0,-1,0),"=",0)
```

```r
add.constraint(lptransship,c(0,0,1,0,0,1,0,0,1,0,0,-1,0,0,-1,0,0,-1,0,0,-1,0,0,-1,0,0,-1),"=",0)
solve(lptransship)
```

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

```r
get.objective(lptransship)
```

```
## [1] 1966.68
```

```r
get.constraints(lptransship)
```

```
##  [1] 93 88 95 30 57 48 91 48  2  0  0  0
```

```r
get.variables(lptransship)
```

```
##  [1] 93  0  0  0 88  0 28  0 67 30  0  0  0 57  0  0 31 17 91  0  0  0  0 48  0
## [26]  0  2
```