Assignment_6

Purna Sai Kiran, G

11/21/2021

Q1: Formulate and solve the binary integer programming (BIP) model for this problem using library lpsolve or equivalent in R

Formulating an LP function. We set the function to maximization because we are trying to find the longest route.

```
library(lpSolveAPI)
lproute <- make.lp(0,12)
lp.control(lproute, sense="max")
```

```
## $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] "maximize"
##
## $simplextype
## [1] "dual"   "primal"
##
## $timeout
## [1] 0
##
## $verbose
## [1] "neutral"
```

Setting the objective function

```
t_value <- c(5, 3, 4, 2, 3, 1, 4, 6, 2, 5, 4, 7)
set.objfn(lproute, 1*t_value)
set.type(lproute, 1:12, "binary")
```

Setting the constraints

```
add.constraint(lproute,c(1,1),"=",1,indices = c(1,2))
add.constraint(lproute,c(1,-1,-1),"=",0,indices = c(1,3,4))
add.constraint(lproute,c(1,-1),"=",0,indices = c(2,5))
add.constraint(lproute,c(1,-1,-1),"=",0,indices = c(3,6,7))
add.constraint(lproute,c(1,1,-1,-1),"=",0,indices = c(4,5,8,9))
add.constraint(lproute,c(1, -1),"=",0,indices = c(6,10))
add.constraint(lproute,c(1,1,-1),"=",0,indices = c(7,8,11))
```

```r
add.constraint(lproute,c(1,-1),"=",0,indices = c(9,12))
add.constraint(lproute,c(1,1,1),"=",1,indices = c(10,11,12))
```

Solving the LP Problem

```r
solve(lproute)

## [1] 0

get.objective(lproute)

## [1] 17

get.variables(lproute)

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

get.constraints(lproute)

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

Mapping input to the output

```r
arc <- c("X12", "X13", "X24", "X25", "X35", "X46", "X47", "X57", "X58",
"X69", "X79", "X89")
vars<-get.variables(lproute)
output<-data.frame(arc,vars)
output

##    arc vars
## 1  X12    1
## 2  X13    0
## 3  X24    0
## 4  X25    1
## 5  X35    0
## 6  X46    0
## 7  X47    0
## 8  X57    1
## 9  X58    0
## 10 X69    0
## 11 X79    1
## 12 X89    0
```

Formulate and solve the binary integer programming (BIP) model for this problem using library lpsolve or equivalent in R. From the variables, the longest route will be X12 - X24 - X57 - X79. The maximum objective function is 17

Q2: 1) Determine the maximum return on the portfolio. What is the optimal number of shares to buy for each of the stocks? What is the corresponding dollar amount invested in each stock? 2) Compare the solution in which there is no integer restriction on the number of shares invested. By how much (in percentage terms) do the integer restrictions alter the value of the optimal objective function? By how much (in percentage terms) do they alter the optimal investment quantities?

formulating the lp function. Setting the function to maximization

```
library(lpSolveAPI)
lpstock<-make.lp(0,8)
lp.control(lpstock,sense="max")
```

```
## $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] "maximize"
##
## $simplextype
## [1] "dual"   "primal"
##
## $timeout
## [1] 0
##
## $verbose
## [1] "neutral"
```

Since we are trying a interger programming we will set the type to integer.

```
set.objfn(lpstock,c(4,6.5,5.9,5.4,5.15,10,8.4,6.25))
set.type(lpstock,c(1:8),type = "integer")
```

Adding the constraints

```
add.constraint(lpstock,c(40,50,80,60,45,60,30,25),"<=",2500000,indices =
c(1:8))
add.constraint(lpstock,1000,">=",0,indices = 1)
add.constraint(lpstock,1000,">=",0,indices = 2)
add.constraint(lpstock,1000,">=",0,indices = 3)
add.constraint(lpstock,1000,">=",0,indices = 4)
add.constraint(lpstock,1000,">=",0,indices = 5)
add.constraint(lpstock,1000,">=",0,indices = 6)
add.constraint(lpstock,1000,">=",0,indices = 7)
add.constraint(lpstock,1000,">=",0,indices = 8)
add.constraint(lpstock,40,">=",100000,indices = 1)
add.constraint(lpstock,50,">=",100000,indices = 2)
add.constraint(lpstock,80,">=",100000,indices = 3)
```

```r
add.constraint(lpstock,60,">=",100000,indices = 4)
add.constraint(lpstock,45,">=",100000,indices = 5)
add.constraint(lpstock,60,">=",100000,indices = 6)
add.constraint(lpstock,30,">=",100000,indices = 7)
add.constraint(lpstock,25,">=",100000,indices = 8)
add.constraint(lpstock,c(40,50,80),"<=",1000000,indices = c(1,2,3))
add.constraint(lpstock,c(60,45,60),"<=",1000000,indices = c(4,5,6))
add.constraint(lpstock,c(30,25),"<=",1000000,indices = c(7,8))
```

solving the lp problem

```r
solve(lpstock)

## [1] 0

get.objective(lpstock)

## [1] 487145.2

get.variables(lpstock)

## [1]  2500  6000  1250  1667  2223 13332 30000  4000

get.constraints(lpstock)

##  [1]  2499975  2500000  6000000  1250000  1667000  2223000 13332000
30000000
##  [9]  4000000   100000   300000   100000   100020   100035   799920
900000
## [17]   100000   500000   999975  1000000
```

1) Determine the maximum return on the portfolio. What is the optimal number of shares to buy for each of the stocks? What is the corresponding dollar amount invested in each stock? With integer restiction, the maximum return on investment is 487145.2. The optimal number of shares to buy are S1 = 2500, S2 = 6000, S3 = 1250, H1 = 1667, H2 = 2223, H3 = 13332, C1 = 30000 and C2 = 4000 and the corresponding dollars to be invested are S1 = 100000, S2 = 300000, S3 = 100000, H1 = 100020, H2 = 100035, H3 = 799920, C1 = 900000, C2 = 100000

formulating lp problem without integer restrictions

```r
lpstock2 <-make.lp(0,8)
lp.control(lpstock2,sense="max")

## $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] "maximize"
```

```
##
## $simplextype
## [1] "dual"   "primal"
##
## $timeout
## [1] 0
##
## $verbose
## [1] "neutral"
```

```
set.objfn(lpstock2,c(4,6.5,5.9,5.4,5.15,10,8.4,6.25))
```

```
add.constraint(lpstock2,c(40,50,80,60,45,60,30,25),"<=",2500000,indices = 
c(1:8))
add.constraint(lpstock2,1000,">=",0,indices = 1)
add.constraint(lpstock2,1000,">=",0,indices = 2)
add.constraint(lpstock2,1000,">=",0,indices = 3)
add.constraint(lpstock2,1000,">=",0,indices = 4)
add.constraint(lpstock2,1000,">=",0,indices = 5)
add.constraint(lpstock2,1000,">=",0,indices = 6)
add.constraint(lpstock2,1000,">=",0,indices = 7)
add.constraint(lpstock2,1000,">=",0,indices = 8)
add.constraint(lpstock2,40,">=",100000,indices = 1)
add.constraint(lpstock2,50,">=",100000,indices = 2)
add.constraint(lpstock2,80,">=",100000,indices = 3)
add.constraint(lpstock2,60,">=",100000,indices = 4)
add.constraint(lpstock2,45,">=",100000,indices = 5)
add.constraint(lpstock2,60,">=",100000,indices = 6)
add.constraint(lpstock2,30,">=",100000,indices = 7)
add.constraint(lpstock2,25,">=",100000,indices = 8)
add.constraint(lpstock2,c(40,50,80),"<=",1000000,indices = c(1,2,3))
add.constraint(lpstock2,c(60,45,60),"<=",1000000,indices = c(4,5,6))
add.constraint(lpstock2,c(30,25),"<=",1000000,indices = c(7,8))
```

```
solve(lpstock2)
```

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

```
get.objective(lpstock2)
```

```
## [1] 487152.8
```

```
get.variables(lpstock2)
```

```
## [1]  2500.000  6000.000  1250.000  1666.667  2222.222 13333.333 30000.000
## [8]  4000.000
```

```
get.constraints(lpstock2)
```

```
##  [1]  2500000  2500000  6000000  1250000  1666667  2222222 13333333 
30000000
##  [9]  4000000   100000   300000   100000   100000   100000   800000
```

```
900000
## [17]   100000   500000  1000000  1000000
```

2) Compare the solution in which there is no integer restriction on the number of shares invested. By how much (in percentage terms) do the integer restrictions alter the value of the optimal objective function? By how much (in percentage terms) do they alter the optimal investment quantities?

The return on investment is 487152.8 The number of stocks to buy are S1 = 2500,S2 = 6000,S3= 1250,H1 = 1666.667,H2= 2222.22,H3= 13333.3,C1=30000,C2=4000 The amounts to be invested are 100000, 300000, 100000, 100000, 100000, 800000, 900000, 100000 respectively.

There is a difference of $7.6 between the lp problems with and without integer restrictions. The value of the objective function differs by 0.00156% The investment quantities are altered as follows: S1 = 0%, S2 = 0%, S3 = 0%, H1 = 0.03996% increased, H2 = 0.03501% decreased, H3 = 75. 01% increased and C1 = C2 = 0%