Assignment 3

Ananth Kumar

06/10/2021

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

.libPaths("C:\Users\Ananth\OneDrive\Desktop\MSBA Kent\Fall 2021\Quantittative management modelling\Module 4")

This notebook contains the code for the examples in Chapter 5. Specifically, for postoptimality analysis.

```
library(lpSolveAPI)
```

```
## Warning: package 'lpSolveAPI' was built under R version 4.0.3
```

---

getwd()

```
rm(list=ls())

x <- read.lp("Assignment3.lp") # create an lp object x
print(x)                          # display x
```

```
## Model name:
##   a linear program with 9 decision variables and 11 constraints
```

1. Solve the problem using lpsolve, or any other equivalent library in R.

```
solve(x) # solving the given lp equation ,'0' optimal solution exists
```

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

```
get.objective(x)  # get objective value of x which is profit =  696000 $
```

```
## [1] 696000
```

```r
get.variables(x)  # get values of decision variables of x which is 9 decision variables
```

```
## [1] 516.6667 177.7778   0.0000   0.0000 666.6667 166.6667   0.0000   0.0000
## [9] 416.6667
```

```r
get.constraints(x)  # get constraint LHS  values
```

```
##  [1]  6.944444e+02  8.333333e+02  4.166667e+02  1.300000e+04  1.200000e+04
##  [6]  5.000000e+03  5.166667e+02  8.444444e+02  5.833333e+02 -2.037268e-10
## [11]  0.000000e+00
```

2. Identify the shadow prices, dual solution, and reduced costs

```r
get.dual.solution(x) # Dual values for reduced cost.
```

```
##  [1]    1.00    0.00    0.00    0.00   12.00   20.00   60.00    0.00    0.00
## [10]    0.00   -0.08    0.56    0.00    0.00  -24.00  -40.00    0.00    0.00
## [19] -360.00 -120.00    0.00
```

```r
get.sensitivity.rhs(x)  # get shadow prices of the constraints the shadow prices are 12.00 , 20.00 and 
```

```
## $duals
##  [1]    0.00    0.00    0.00   12.00   20.00   60.00    0.00    0.00    0.00
## [10]   -0.08    0.56    0.00    0.00  -24.00  -40.00    0.00    0.00 -360.00
## [19] -120.00    0.00
## 
## $dualsfrom
##  [1] -1.000000e+30 -1.000000e+30 -1.000000e+30  1.122222e+04  1.150000e+04
##  [6]  4.800000e+03 -1.000000e+30 -1.000000e+30 -1.000000e+30 -2.500000e+04
## [11] -1.250000e+04 -1.000000e+30 -1.000000e+30 -2.222222e+02 -1.000000e+02
## [16] -1.000000e+30 -1.000000e+30 -2.000000e+01 -4.444444e+01 -1.000000e+30
## 
## $dualstill
##  [1] 1.000000e+30 1.000000e+30 1.000000e+30 1.388889e+04 1.250000e+04
##  [6] 5.181818e+03 1.000000e+30 1.000000e+30 1.000000e+30 2.500000e+04
## [11] 1.250000e+04 1.000000e+30 1.000000e+30 1.111111e+02 1.000000e+02
## [16] 1.000000e+30 1.000000e+30 2.500000e+01 6.666667e+01 1.000000e+30
```

```r
get.sensitivity.obj(x)  # get reduced cost The reduced costs are expressed here from $objfrom and $objt
```

```
## $objfrom
## [1]  3.60e+02  3.45e+02 -1.00e+30 -1.00e+30  3.45e+02  2.52e+02 -1.00e+30
## [8] -1.00e+30  2.04e+02
## 
## $objtill
## [1] 4.60e+02 4.20e+02 3.24e+02 4.60e+02 4.20e+02 3.24e+02 7.80e+02 4.80e+02
## [9] 1.00e+30
```

```r
get.sensitivity.rhs(x)$duals[1:11] #  shadow price
```

```
##  [1]  0.00  0.00  0.00 12.00 20.00 60.00  0.00  0.00  0.00 -0.08  0.56
```

```r
get.sensitivity.rhs(x)$duals[12:20] # reduced price
```

```
## [1]    0    0  -24  -40    0    0 -360 -120    0
```

Further, identify the sensitivity of the above prices and costs. That is, specify the range of shadow prices and reduced cost within which the optimal solution will not change.

```r
cbind(get.sensitivity.rhs(x)$duals[1:11], get.sensitivity.rhs(x)$dualsfrom[1:11],get.sensitivity.rhs(x)$
```

```
##         [,1]          [,2]         [,3]
##  [1,]  0.00 -1.000000e+30 1.000000e+30
##  [2,]  0.00 -1.000000e+30 1.000000e+30
##  [3,]  0.00 -1.000000e+30 1.000000e+30
##  [4,] 12.00  1.122222e+04 1.388889e+04
##  [5,] 20.00  1.150000e+04 1.250000e+04
##  [6,] 60.00  4.800000e+03 5.181818e+03
##  [7,]  0.00 -1.000000e+30 1.000000e+30
##  [8,]  0.00 -1.000000e+30 1.000000e+30
##  [9,]  0.00 -1.000000e+30 1.000000e+30
## [10,] -0.08 -2.500000e+04 2.500000e+04
## [11,]  0.56 -1.250000e+04 1.250000e+04
```

```r
# the range of the shadow price and reduced cost is from negative 1 to  positive 1


cbind(get.sensitivity.rhs(x)$duals[12:20], get.sensitivity.rhs(x)$dualsfrom[12:20],get.sensitivity.rhs(
```

```
##       [,1]          [,2]         [,3]
##  [1,]    0 -1.000000e+30 1.000000e+30
##  [2,]    0 -1.000000e+30 1.000000e+30
##  [3,]  -24 -2.222222e+02 1.111111e+02
##  [4,]  -40 -1.000000e+02 1.000000e+02
##  [5,]    0 -1.000000e+30 1.000000e+30
##  [6,]    0 -1.000000e+30 1.000000e+30
##  [7,] -360 -2.000000e+01 2.500000e+01
##  [8,] -120 -4.444444e+01 6.666667e+01
##  [9,]    0 -1.000000e+30 1.000000e+30
```

```r
dual<-read.lp ("dual1.lp")
dual
```

```
## Model name:
##   a linear program with 11 decision variables and 9 constraints
```

```r
# '0' optimal solution exists

set.bounds(dual, lower = c(-Inf,-Inf), columns = 10:11)

solve(dual)
```

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

```r
get.objective(dual)#  The objective value for the dual is 698000.4 while the objective in primal is 696
```

```
## [1] 696000
```

```r
get.variables(dual)        #  values of decision variables for dual functions which are 11
```

```
##  [1]  0.00  0.00  0.00 12.00 20.00 60.00  0.00  0.00  0.00 -0.08  0.56
```

```r
get.constraints(dual) #  9 constraints of the dual function.
```

```
## [1] 420 360 324 460 360 300 780 480 300
```

```r
get.sensitivity.rhs(dual)$duals#  shadow price of the dual is equal to the variables in primal.
```

```
##  [1] 516.66667 177.77778   0.00000   0.00000 666.66667 166.66667   0.00000
##  [8]   0.00000 416.66667  55.55556  66.66667  33.33333   0.00000   0.00000
## [15]   0.00000 383.33333 355.55556 166.66667   0.00000   0.00000
```

get.sensitivity.obj(dual) # the vales of the sensitive

“‘ [1] 516.6667 177.7778 0.0000 0.0000 666.6667 166.6667 0.0000 0.0000 416.6667

516.6667 177.7778 0.0000 0.0000 666.6667 166.6667 0.0000 0.0000 416.6667 get.variables(dual) # values of decision variables