Integer\_Programming

Dutt Thakkar

2023-11-16

library(lpSolveAPI)

# Objective Function: Minimize Total Wage Expenses

$$ MIN \hspace{0.3mm} Z = 775X\_1 + 800X\_2 + 800X\_3 + 800X\_4 + 800X\_5 + 775X\_6 + 750X\_7$$

IP <- make.lp(0, 7)  
lp.control(IP, sense="min", verbose = "neutral")

## $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.objfn(IP, c(775,800,800,800,800,775,750))

Constraints:

Sunday:-

Monday:-

Tuesday:-

Wednesday:-

Thursday:-

Friday:-

Saturday:-

Non-negative integers

add.constraint(IP, c(0,1,1,1,1,1,0), ">=", 18)  
add.constraint(IP, c(0,0,1,1,1,1,1), ">=", 27)  
add.constraint(IP, c(1,0,0,1,1,1,1), ">=", 22)  
add.constraint(IP, c(1,1,0,0,1,1,1), ">=", 26)  
add.constraint(IP, c(1,1,1,0,0,1,1), ">=", 25)  
add.constraint(IP, c(1,1,1,1,0,0,1), ">=", 21)  
add.constraint(IP, c(1,1,1,1,1,0,0), ">=", 19)

set.type(IP,1:7,"integer")  
RowNames <- c("Sun,Mon", "Mon,Tue", "Tue,Wed", "Wed,Thu", "Thu,Fri", "Fri,Sat", "Sat,Sun")  
ColNames <- c("x1", "x2","x3","x4","X5","X6","X7")  
dimnames(IP) <- list(RowNames, ColNames)  
IP

## Model name:   
## x1 x2 x3 x4 X5 X6 X7   
## Minimize 775 800 800 800 800 775 750   
## Sun,Mon 0 1 1 1 1 1 0 >= 18  
## Mon,Tue 0 0 1 1 1 1 1 >= 27  
## Tue,Wed 1 0 0 1 1 1 1 >= 22  
## Wed,Thu 1 1 0 0 1 1 1 >= 26  
## Thu,Fri 1 1 1 0 0 1 1 >= 25  
## Fri,Sat 1 1 1 1 0 0 1 >= 21  
## Sat,Sun 1 1 1 1 1 0 0 >= 19  
## Kind Std Std Std Std Std Std Std   
## Type Int Int Int Int Int Int Int   
## Upper Inf Inf Inf Inf Inf Inf Inf   
## Lower 0 0 0 0 0 0 0

solve(IP)

## [1] 0

# What was the total cost?

get.objective(IP)

## [1] 25675

cat("The minimum total wage expanse is $", get.objective(IP), "\n")

## The minimum total wage expanse is $ 25675

# How many workers are available each day?

get.variables(IP)

## [1] 2 4 5 0 8 1 13

cat("Total numbers of workers available on Sunday is", sum(get.variables(IP)[c(2,3,4,5,6)]), "\n")

## Total numbers of workers available on Sunday is 18

cat("Total numbers of workers available on Monday is", sum(get.variables(IP)[c(3,4,5,6,7)]), "\n")

## Total numbers of workers available on Monday is 27

cat("Total numbers of workers available on Tuesday is", sum(get.variables(IP)[c(4,5,6,7,1)]), "\n")

## Total numbers of workers available on Tuesday is 24

cat("Total numbers of workers available on Wednesday is", sum(get.variables(IP)[c(5,6,7,1,2)]), "\n")

## Total numbers of workers available on Wednesday is 28

cat("Total numbers of workers available on Thursday is", sum(get.variables(IP)[c(6,7,1,2,3)]), "\n")

## Total numbers of workers available on Thursday is 25

cat("Total numbers of workers available on Friday is", sum(get.variables(IP)[c(7,1,2,3,4)]), "\n")

## Total numbers of workers available on Friday is 24

cat("Total numbers of workers available on Saturday is", sum(get.variables(IP)[c(1,2,3,4,5)]), "\n")

## Total numbers of workers available on Saturday is 19