

Simplex

Dalia Camacho

21 de septiembre de 2018

Se define la función Simplex que recibe como entradas una matriz A, un vector b y un vector c que definen el problema de optimización:

$$\max c^t x \text{ s.t. } Ax \leq b.$$

Se tiene como resultado un vector que indica las variables que dan la solución del problema y el valor de la función objetivo. El código imprime los arreglos intermedios

```
Simplex <- function(A,b,c,.print=TRUE){
  # Generar arreglo inicial
  nrest <- nrow(A) # Número de variables
  nvar <- ncol(A) # Número de restricciones y de variables de holgura

  #Definir tamaño del arreglo
  Arreglo <- matrix(0 ,ncol = (nvar+nrest+1), nrow = (nrest+1))

  # Agregar valores al arreglo inicial
  Arreglo[1:nrest,1:nvar] <- A
  Arreglo[nrest+1,1:nvar] <- -c
  diag(Arreglo[1:nrest,(nvar+1):(nrest+nvar)]) <- 1
  Arreglo[1:nrest,nrest+nvar+1] <- b

  # Definir nombres del arreglo inicial
  colnames(Arreglo) <- c(unlist(lapply(1:nvar, function(i){paste0("x_",i)})),
                        unlist(lapply(1:nrest, function(i){paste0("h_",i)})), "solución")
  rownames(Arreglo) <- c(unlist(lapply(1:nrest, function(i){paste0("h_",i)})), "z")

  # Imprimir arreglo inicial
  if(.print){
    print("El arreglo inicial es:")
    print(Arreglo)
  }

  # Contador de iteraciones
  iter <- 0
  while(any(Arreglo[nrest+1,]<0)){
    iter <- iter+1
    # Definir columna pivote (valor negativo mínimo en la columna z)
    col.piv <- which.min(Arreglo[nrest+1,])

    # Definir renglón pivote (variable de entrada)
    # (El renglón donde se encuentre el mínimo (positivo) de dividir
    # la columna de soluciones entre la columna pivote)
    aux <- Arreglo[(1:nrest), (nvar+nrest+1)]/Arreglo[1:nrest,col.piv]

    if(any(aux<0)){
      aux[which(aux<0)] <- Inf
    }
  }
}
```

```

if(any(aux!=Inf)==FALSE){
  stop("El problema pudo haberse definido incorrectamente.")
}

reng.piv <- which.min(aux)

# Cambiar variables dentro de la solución
rownames(Arreglo)[reng.piv] <- colnames(Arreglo)[col.piv]

#Dividir el renglón pivote entre el pivote operacional
Arreglo[reng.piv,] <- Arreglo[reng.piv,]/Arreglo[reng.piv,col.piv]
col.aux <- Arreglo[,col.piv]
for(i in 1:(nrest+1)){
  if(i!=reng.piv){
    for(j in 1:(nvar+nrest+1)){
      Arreglo[i,j] <- Arreglo[i,j]-col.aux[i]*Arreglo[reng.piv,j]
    }
  }
}
if(.print){
  print(paste("El arreglo en la iteración", iter, "es:"))
  print(Arreglo)
}
}

return(list(Valor=Arreglo[nrest+1,nrest+nvar+1], Variables=Arreglo[1:nrest,nrest+nvar+1]))
}

```

Problema de la cervecería

Se tienen dos tipos de cerveza **Lite** y **Light**, para producir 100 litros de cada una se requieren los ingredientes dados en la siguiente tabla.

	Malta	Lúpulo	Levadura	Ganancias por 100 lts
Lite	2"	4"	2"	21"
Light	3"	1"	9"	31"
Cantidad	25"	32"	54"	-"

El problema se puede escribir como:

$$\max 21x_1 + 31x_2 \text{ s.t. } 2x_1 + 3x_2 \leq 25, 4x_1 + x_2 \leq 32, 2x_1 + 9x_2 \leq 54, x_1, x_2 \geq 0$$

```

# Definir problema en términos de A, b y c
A <- matrix(c(2,3,4,1,2,9),ncol = 2, byrow = TRUE)
b <- c(25, 32, 54)
c <- c(21, 31)

```

```

# Correr el código
sol <- Simplex(A,b,c)

```

```

## [1] "El arreglo inicial es:"
##      x_1 x_2 h_1 h_2 h_3 solución
## h_1   2   3   1   0   0        25
## h_2   4   1   0   1   0        32

```

```

## h_3  2  9  0  0  1      54
## z   -21 -31  0  0  0      0
## [1] "El arreglo en la iteración 1 es:"
##      x_1 x_2 h_1 h_2      h_3 solución
## h_1  1.3333333  0  1  0 -0.3333333      7
## h_2  3.7777778  0  0  1 -0.1111111     26
## x_2  0.2222222  1  0  0  0.1111111      6
## z   -14.1111111  0  0  0  3.4444444    186
## [1] "El arreglo en la iteración 2 es:"
##      x_1 x_2      h_1 h_2      h_3 solución
## x_1  1  0  0.7500000  0 -0.2500000    5.250000
## h_2  0  0 -2.8333333  1  0.8333333    6.166667
## x_2  0  1 -0.1666667  0  0.1666667    4.833333
## z    0  0 10.5833333  0 -0.0833333 260.083333
## [1] "El arreglo en la iteración 3 es:"
##      x_1 x_2  h_1  h_2 h_3 solución
## x_1  1  0 -0.1  0.3  0      7.1
## h_3  0  0 -3.4  1.2  1      7.4
## x_2  0  1  0.4 -0.2  0      3.6
## z    0  0 10.3  0.1  0     260.7

```

```

# Solución
sol

```

```

## $Valor
## [1] 260.7
##
## $Variables
## x_1 h_3 x_2
## 7.1 7.4 3.6

```

Las ganancias se maximizan cuando se producen 710 litros de Lite, 360 litros de Light, sobran 7.4 unidades de levadura y se obtienen \$ 260.7 de ganancias.