

Tarea 4: Knapsack 01

Dalia Camacho

Código en R

Se creó una función que utiliza el algoritmo visto en clase para obtener el valor máximo de los objetos que se pueden meter en la mochila sin sobrepasar un peso máximo W . Además se agrega una matriz en la que se van guardando los objetos seleccionados en cada paso del algoritmo.

```
knapsack01 <- function(w,v,W){  
  # Se encuentran y n  
  W  
  n <- length(v)  
  
  # Se define m, la matriz que va obteniendo el valor de los objetos  
  # en la mochila  
  m <- matrix(0, ncol = (W+1), nrow = (n+1))  
  
  # Se define Res, la matriz donde se van  
  # guardando los objetos dentro de la mochila  
  Res <- matrix(0, nrow = (n+1)*(W+1), ncol = n)  
  
  # Los índices comienzan en 1, por lo  
  # que j va desde 1 hasta W+1  
  for(i in 1:(n)){  
    for(j in 1:(W+1)){  
      # Se compara el peso del objeto i con j-1  
      # Con la resta se tiene que (j-1) va de 0 a W  
      if(w[i]>(j-1)){  
        # Si el peso del objeto es mayor se toma  
        # el valor previo para el peso j y los  
        # elementos previos para ese peso  
        m[i+1,j] <- m[i,j]  
        Res[(i)*(W+1)+j,] <- Res[(i-1)*(W+1)+j,]  
      }else{  
        # Si el peso del objeto es menor o igual a j  
        # el valor de la función objetivo se toma como  
        # el máximo entre el valor de los objetos previamente seleccionados  
        # y valor del objeto i más el valor máximo calculado para  
        # objetos que pesen menos de j-w[i]  
        # Los objetos se seleccionan de la misma forma que el valor  
        if(m[i,j] > m[i,j-w[i]]+v[i]){  
          m[i+1,j] <- m[i,j]  
          Res[(i)*(W+1)+j,] <- Res[(i-1)*(W+1)+j,]  
        }else{  
          m[i+1,j] <- m[i,j-w[i]]+v[i]  
          Res[(i)*(W+1)+j,] <- Res[(i-1)*(W+1)+j,]  
        }  
      }  
    }  
  }  
}
```

```

    Res[(i)*(W+1)+j,] <- Res[(i-1)*(W+1)+j,]
  }else{
    m[i+1,j] <- m[i,j-w[i]]+v[i]
    Res[(i)*(W+1)+j,] <- Res[(i-1)*(W+1)+j-w[i],]
    Res[(i)*(W+1)+j,i] <- 1
  }
}
}
}
}
return(list(Value = m[n+1,W+1], Objects = Res[(n+1)*(W+1),]))
}

```

Ejercicio 1

Se considera una mochila que puede llevar 10 kg y los siguientes objetos:

	X1	X2	X3	X4	X5
Peso	6	5	3	2	2
Valor	10	9	8	7	3

```

w <- c(6, 5, 3, 2, 2)
v <- c(10, 9, 8, 7, 3)
W <- 10

sol <- knapsack01(w=w,v=v,W=W)
sol

```

```

## $Value
## [1] 24
##
## $Objects
## [1] 0 1 1 1 0

```

El valor máximo es 24 y se logra tomando los objetos 2, 3, 4.

Ejercicio 2

Se considera una mochila que puede llevar 20 kg y los siguientes objetos:

	tent	canteen	clothes	camp stove	sleeping bag	dried food	first aid kit	repellent	flashlight	novel	rain gear	water purifier
Peso	11	7	5	4	3	3	3	2	2	2	2	1
Valor	20	10	11	5	25	50	15	12	6	4	5	10

```

w <- c(11, 7, 5, 4, 3, 3, 3, 2, 2, 2, 2, 1)
v <- c(20, 10, 11, 5, 25, 50, 15, 12, 6, 4, 5, 10)
W <- 20

sol <- knapsack01(w=w,v=v,W=W)

sol

## $Value
## [1] 129
##
## $Objects
## [1] 0 0 1 0 1 1 1 1 1 0 0 1

```

El valor máximo es 129 y se logra tomando los objetos clothes, sleeping bag, dried food, first aid kit, repelent, flashlight, water purifier.