

Proyecto 2: El Problema de la Mochila

Emily Sanchez
Viviana Vargas

Curso: Investigación de Operaciones
II Semestre 2025

September 19, 2025

1 Problema de la Mochila (Knapsack Problem)

El **problema de la mochila** es un clasico de la *optimizacion combinatoria*. Se dispone de una **mochila** con una **capacidad maxima** W y un conjunto de n objetos. Cada objeto i tiene un **peso** w_i y un **valor** v_i . El objetivo es seleccionar los objetos de manera que:

- La suma total de los pesos no exceda la capacidad W .
- Se maximice el valor total de los objetos elegidos.

1.1 Variantes principales

0/1 Knapsack Cada objeto puede elegirse una sola vez o no elegirse: decision binaria.

Bounded Knapsack Cada objeto puede seleccionarse un numero limitado de veces.

Unbounded Knapsack Se permite una cantidad ilimitada de cada objeto.

1.2 Solucion

0/1 Knapsack Se resuelve comunmente con **programacion dinamica**. Sea $dp[i][w]$ el valor maximo al considerar los primeros i objetos y capacidad w .

$$dp[i][w] = \begin{cases} dp[i-1][w] & \text{si } w_i > w, \\ \max(dp[i-1][w], v_i + dp[i-1][w - w_i]) & \text{si } w_i \leq w. \end{cases}$$

Bounded Knapsack Similar al 0/1 pero puede tener uno o más cantidades por objeto. Es limitado, por lo que no puede ser infinito.

$$dp[i][w] = \max_{0 \leq k \leq c_i, k w_i \leq w} (dp[i-1][w - k w_i] + k v_i).$$

Unbounded Knapsack Similar al bounded pero permitiendo repeticiones sin limite de cantidades (infinito).

$$dp[w] = \max(dp[w], v_i + dp[w - w_i]).$$

Tipo de problema: 0/1 Knapsack
Capacidad máxima: 17
Número de objetos: 5

Formulación Matemática

Función objetivo:

$$\text{Maximizar } Z = 30x_A + 15x_B + 6x_C + 18x_D + 10x_E$$

Restricción:

$$13x_A + 5x_B + 9x_C + 10x_D + 4x_E \leq 17$$

Restricciones de variables:

$$x_i \in \{0, 1\} \quad \forall i \in \{A, B, C, D, E\}$$

Datos del Problema

Objeto	Costo	Valor	Cantidad
A	13,00	30,00	1
B	5,00	15,00	1
C	9,00	6,00	1
D	10,00	18,00	1
E	4,00	10,00	1

Tabla de Programación Dinámica

Capacidad/Objetos	A	B	C	D	E
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	10
5	0	15	15	15	15
6	0	15	15	15	15
7	0	15	15	15	15
8	0	15	15	15	15
9	0	15	15	15	25
10	0	15	15	18	25
11	0	15	15	18	25
12	0	15	15	18	25
13	30	30	30	30	30
14	30	30	30	30	30
15	30	30	30	33	33
16	30	30	30	33	33
17	30	30	30	33	40

Solución Óptima

Valor máximo obtenido: 40
Objetos seleccionados: E, A
Capacidad utilizada: 17