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.

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

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: 15
Número de objetos: 7

Datos del Problema

Objeto	Costo	Valor	Cantidad
A	3,00	7,00	1
B	4,00	9,00	1
C	2,00	5,00	1
D	6,00	12,00	1
E	7,00	14,00	1
F	3,00	6,00	1
G	5,00	12,00	1

Tabla de Programación Dinámica

Capacidad/Objetos	A	B	C	D	E	F	G
0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
2	0	0	5	5	5	5	5
3	7	7	7	7	7	7	7
4	7	9	9	9	9	9	9
5	7	9	12	12	12	12	12
6	7	9	14	14	14	14	14
7	7	16	16	16	16	16	17
8	7	16	16	17	17	18	19
9	7	16	21	21	21	21	21
10	7	16	21	21	21	22	24
11	7	16	21	24	24	24	26
12	7	16	21	26	26	27	28
13	7	16	21	28	28	28	30
14	7	16	21	28	30	30	33
15	7	16	21	33	33	33	34

Solución Óptima

Valor máximo obtenido: 34
Objetos seleccionados: G, F, B, A
Capacidad utilizada: 15