

Caso 01 - Problema de la Mochila (Knapsack) para cargar un camión

Enunciado

Dispones de un único camión con capacidad limitada. Debes seleccionar qué paquetes cargar para maximizar el valor total transportado.

Cada paquete tiene un peso y un volumen. La carga elegida no puede exceder la capacidad del camión en ninguno de los dos recursos.

Datos y parámetros

- Capacidad máxima de peso: 15
- Capacidad máxima de volumen: 20
- Objetivo: maximizar el valor total de los paquetes cargados

Dataset de muestra (pequeño)

Este dataset es intencionalmente pequeño para poder validar a mano la evaluación de soluciones.

item	peso	volumen	valor
A	4	6	8
B	2	3	4
C	6	7	11
D	5	4	7
E	3	5	6
F	7	9	13
G	1	2	2
H	4	5	9

Pistas para un Algoritmo Genético

Representación (cromosoma)

- Vector binario de longitud n (n = número de items): 1 si el item se carga, 0 si no.

Fitness (evaluación)

- Valor = suma(valor_i * x_i).
- Si se excede peso o volumen, aplicar penalización: fitness = valor - alpha*(exceso_peso + exceso_volumen), o marcar como inválido.

Mutación (ejemplos)

- Bit-flip: elegir una posición al azar y cambiar 0→1 o 1→0.

- Mutación guiada: si hay exceso, apagar algunos items aleatorios hasta ser factible.

Cruce (ejemplos)

- Cruce de 1 punto o 2 puntos sobre el vector binario.
- Uniform crossover: por cada gen, elegir el del padre A o B con probabilidad 0.5.

Selección (ejemplos)

- Torneo ($k=3$): elegir 3 individuos al azar y quedarse con el mejor (mayor fitness).
- Ruleta (proporcional): seleccionar con probabilidad proporcional al fitness (siempre que sea no-negativo).