RESOLVER PROBLEMAS

En el capítulo anterior hemos visto que, a pesar de ser una ciencia relativamente joven y aún en fase temprana en comparación con otros campos, la lA es ya hoy una herramienta que nos ayuda a afrontar la resolución de problemas usando un enfoque muy distinto al de la computación clásica. No obstante, no puede verse la lA como la receta mágica capaz de resolver cualquier problema, sino como un conjunto de técnicas que por sí solas o en combinación con otras nos ayudarán a encontrar una solución (no necesariamente la mejor) a un problema cuya resolución es compleja e incluso inabordable por una persona humana.
No vamos a definir lo que es un "Problema", ya que esto entra más dentro del campo de la Filosofía que de la Informática. Aun así, vamos a quedarnos con la idea intuitiva de que un problema es una cuestión difícil de solucionar, es decir, que no tiene una solución trivial.

Todos recordamos de nuestro paso por el colegio aquellos ejercicios que teníamos que resolver en clase de Matemáticas. Vamos a proponer un par de problemas de nivel de primaria e infantil.

Problema 1: Juan tiene 5 caramelos. De camino a casa pierde 1 caramelo que cae del bolsillo. Cuando sube las escaleras de su portal pierde de nuevo otro caramelo. Finalmente, al llegar, se encuentra 3 caramelos que se ha dejado olvidados sobre la mesa su hermano Víctor. ¿Cuántos caramelos tiene Juan?

Problema 2: Ordene en orden descendente la siguiente lista de números: 3, 6, 1, 3,9, 4, 2.

Son dos problemas muy simples, pero ¿cómo se enfrenta nuestro cerebro a ellos? Si ha intentado buscar la solución de ambos, habrá observado que las técnicas que ha usado para resolverlos son bien distintas a pesar de ser de una dificultad similar.

En el **primer problema**, seguramente, ha tenido que contextualizar las situaciones. Seguramente se ha imaginado a Juan perdiendo un caramelo mientras subía la escalera. Finalmente, ha reducido el enunciado a una simple operación con sumas y restas. En definitiva, todo el párrafo que describía el problema ha quedado reducido a la operación;
Caramelos = 5-1-1+3 = 6
Evidentemente, una vez reducido el problema a una operación matemática habrá tenido que aplicar unas herramientas de las que ya disponía desde la infancia, como son las operaciones matemáticas.

El acercamiento al **segundo problema** ha debido ser algo diferente. En este caso no ha tenido que hacer operaciones matemáticas (al menos directamente), sino que la tarea consistía en comparar los valores de dos números para saber cuál debía colocar en primer lugar. Repitiendo esta operación, habrá acabado ordenando todos los números. El cerebro, en este segundo caso, ha usado unas herramientas diferentes a las utilizadas para el primero. En todo caso, al igual que en el primero se conceptualizó todo el enunciado en una sola operación matemática, en el segundo también se ha reducido un problema complejo (ordenar una lista de números), a una secuencia de operaciones más simples (ordenar sucesivamente un par de números hasta tener toda la lista ordenada).

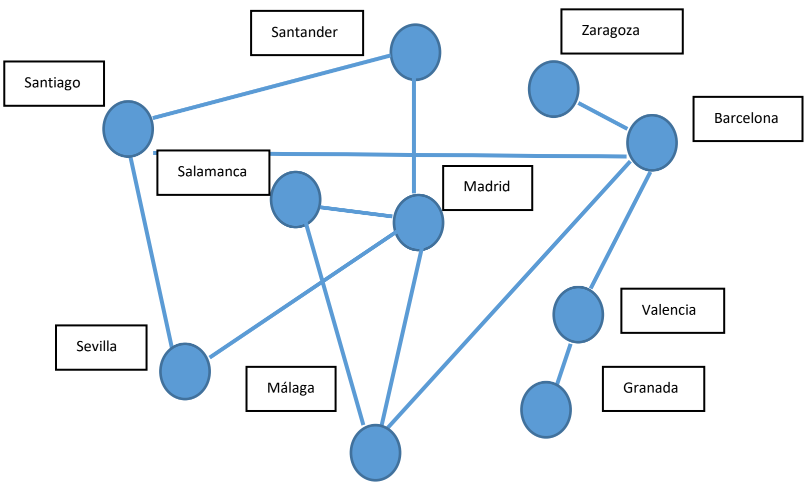
En **ambos casos**, el cerebro ha recogido la información que necesitaba dejando de lado todo aquello que le era superfluo (no importa si el caramelo se cayó subiendo las escaleras o en la calle) y lo ha traducido a un modelo mental que sí es capaz de manejar. Sin embargo, no todos los problemas son tan sencillos. Vamos a plantear algunos problemas algo más complejos y que nos van a servir de base para seguir indagando en la resolución de problemas.

Otros problemas más complejos

Si ha hecho algún viaje largo alguna vez en avión, seguramente ha intentado viajar al destino directamente y sin escalas. Este es un problema clásico y con una gran aplicación práctica. El problema podría enunciarse de la siguiente manera: Dada una ciudad de origen y otra de destino, **encontrar una combinación de vuelos que nos permita cubrir la distancia con el mínimo número de trasbordos**. Por ahora vamos a obviar tiempos de vuelo, distancias, horarios de los vuelos y todo lo demás.
Vamos a enunciar otro problema similar. En esta ocasión se trata de ir de una ciudad a otra pero esta vez por carretera y **tratando de realizar el menor número de kilómetros posible**. Parece que es un problema similar al de los aviones, solo que en lugar de pasar por aeropuertos pasamos por diversas ciudades en el camino que va del origen al destino. Sin embargo, hay un pequeño detalle que lo hace un problema muy diferente. Y es que los objetivos son distintos.

En el primer problema teníamos un objetivo: minimizar el número de trasbordos. En el segundo problema lo que tratamos de minimizar es la distancia por carretera. Veámoslo con un ejemplo.

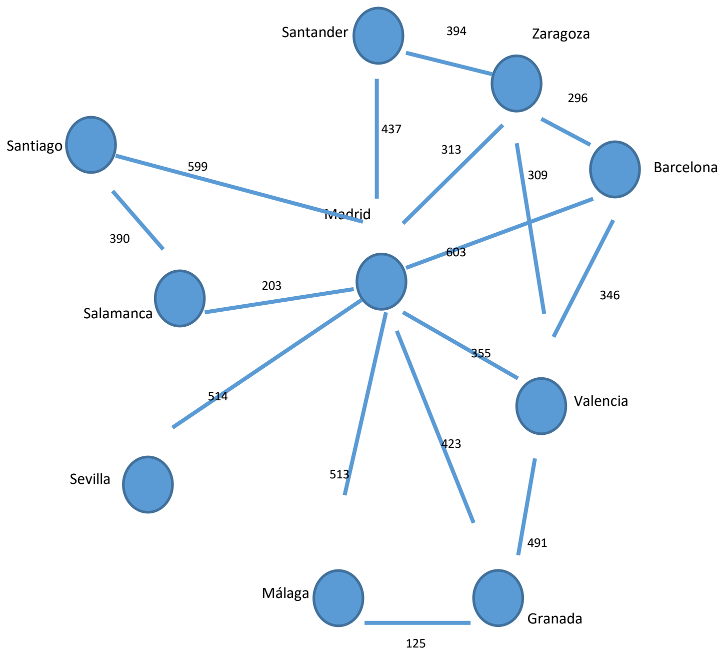
En la figura siguiente vemos un esquema con algunos aeropuertos españoles y unas conexiones ficticias a través de vuelos para un día concreto.



Si quisiéramos viajar desde Málaga a Santiago, podríamos optar por las siguientes rutas:
Málaga > Barcelona > Santiago (3 aeropuertos)
Málaga > Madrid > Santander > Santiago (4 aeropuertos)
Málaga > Madrid > Sevilla > Santiago (4 aeropuertos)

Si quisiéramos viajar desde Málaga a Granada, a pesar de que las ciudades son cercanas, necesitamos visitar nada menos que cuatro aeropuertos:
Málaga > Barcelona > Valencia > Granada

Pero ¿qué ocurre si nos planteamos esos mismos viajes por carretera y con el objetivo de minimizar la distancia recorrida?
En el siguiente esquema vemos las mismas ciudades y sus conexiones por carretera junto con sus distancias.



Supongamos de nuevo que queremos ir de Málaga a Santiago. Estas son las posibles opciones:
Málaga > Madrid > Santiago (1.112 km)
Málaga > Madrid > Salamanca > Santiago (1.106 km)

¿Cuál es el mejor recorrido? Hemos de definir un **criterio** que nos diga cómo de bueno es un recorrido. En este caso hemos escogido la **suma total de kilómetros**, que arroja que el mejor recorrido es el primero porque permite llegar al destino cubriendo menos distancia.

A la vista del breve análisis de los problemas que hemos planteado, podemos intuir que hay tres conceptos importantes que definen un problema.
Por un lado, hemos visto cómo necesitamos crear un **modelo** simplificado que represente al problema real. Por ejemplo, reduciendo el enunciado del problema de Juan y los caramelos a simples operaciones matemáticas.

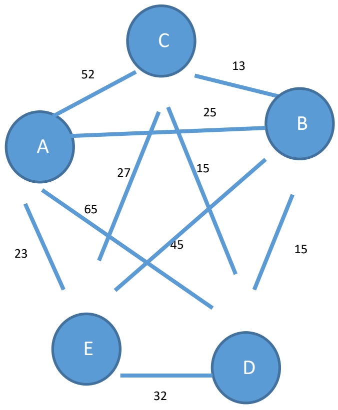
También hemos llegado a la conclusión de que hay que tener muy claro el **objetivo** que buscamos. Es decir, ¿cuándo damos el problema por resuelto? ¿Cuáles serán las soluciones válidas? En los ejemplos del viaje en avión y por carretera, ¿cuáles eran los objetivos? Para el primero, encontrar las conexiones con el número de trasbordos, y para el segundo, encontrar el recorrido con el menor número de kilómetros.

Finalmente, hemos tenido que definir un criterio que valore cómo de buena es cada una de las soluciones del problema, para poder compararla con las demás y quedarnos con la mejor. En el caso de los viajes en avión hemos tomado el criterio del número de aeropuertos visitados. En el problema de los viajes por carretera nuestro criterio ha sido la suma total de kilómetros. Este criterio que nos da información de la calidad de una solución lo llamaremos a partir de ahora función de evaluación.
Por lo tanto, para definir un problema necesitamos un modelo, un objetivo y una **función de evaluación**.

El problema del viajante de comercio

El primer problema que vamos a tratar es bastante conocido dentro de la lA y es conocido como "El problema del viajante de comercio" o también por sus siglas en inglés TSP *(Travelling Salesman Problem).* La formulación del problema es sencilla: Sean N ciudades, todas conectadas por carretera entre sí y cuyas distancias entre ellas es conocida. Nuestro objetivo es encontrar una ruta que, empezando en una ciudad y acabando en la misma, pase exactamente una vez por cada una de las ciudades (excepto por la primera, que será visitada dos veces) y minimice la distancia recorrida.
Tras este simple enunciado se esconde un problema tremendamente difícil. Tanto que, incluso con los ordenadores más potentes que existen hoy, somos incapaces de resolverlo. Incluso para un número de ciudades moderado. Aun así, podemos conseguir aproximaciones muy buenas utilizando diferentes técnicas de lA, algunas de las cuales van a ser estudiadas en los siguientes capítulos.

Vamos a comenzar por construir **un modelo** que el ordenador pueda manejar. Empecemos por el mapa en sí. Hay varias opciones. Por ejemplo, podemos optar por representar las ciudades mediante un dígrafo ponderado. En este caso, cada nodo del grafo representará una ciudad, y cada arista corresponderá a una carretera con su distancia (peso) asociada. Debe tenerse en cuenta que para el caso del TSP, la distancia a cubrir desde una ciudad A hasta otra B no tiene por qué ser la misma que de B hasta A. De todas formas, para simplificar el ejemplo vamos a suponer que las distancias entre dos ciudades son iguales con independencia del sentido en que vayamos.
En la figura siguiente se ha representado un esquema de un hipotético mapa de carreteras para cinco ciudades, y su representación como grafo.



Otra posible representación es utilizar una tabla (también llamada matriz de adyacencia) como la siguiente.

	A	B	C	D	E
A	0	25	52	65	23
B	25	0	13	15	45
C	52	13	0	15	27
D	65	15	15	0	32
E	23	45	27	32	0

Siempre que nos sea posible elegiremos la representación más simple e intuitiva, a no ser que en términos de eficiencia sea más conveniente utilizar otra estructura de datos más compleja.

Una vez modelada la información del problema, necesitamos una forma de representar la solución, es decir, nuestro **objetivo**. Una forma simple y directa de representarlo es como una lista de ciudades que cumpla las siguientes premisas:

- La lista comienza y termina por la misma ciudad.
- En la lista aparecen todas las ciudades.
- No se repite ninguna ciudad, excepto la primera y la última.

Cualquier lista de ciudades que cumpla las tres premisas anteriores es una posible solución. Sin embargo, sabemos que no todas las soluciones son iguales. De hecho buscamos la ruta con menor distancia. Necesitamos pues una **función de evaluación** que nos informe de la calidad de una solución para poder compararla con otras soluciones. Para el TSP usaremos la suma de las distancias de cada una de las ciudades por las que vamos pasando. Por ejemplo, estas son posibles soluciones y el valor obtenido por la función de evaluación.
A-B-E-D-C-A (función de evaluación: 169)
A-E-D-C-B-A (función de evaluación: 108)
A-C-D-E-B-A (función de evaluación: 169)

¿Cómo encontrar la mejor solución? En teoría podríamos proceder de la siguiente manera:

1. Generamos todas las posibles soluciones.
2. Comparamos unas con otras y nos quedamos con la que arroje un valor menor de la función de evaluación.

Analicemos con detalle el primer paso. ¿Cómo generamos todas las posibles soluciones? ¿Cuántas soluciones posibles hay?
Una forma de generar las posibles soluciones es partir de una lista de ciudades cualquiera e ir permutándolas entre ellas. Cada permutación será una posible solución al problema. Vamos a suponer que la lista de ciudades es la siguiente:

A-B-E-D-C-A (función de evaluación: 169)

Si hacemos el cambio entre las ciudades B y E obtenemos:

A-E-B-D-C-A (función de evaluación: 150)

Si ahora efectuamos el cambio de B y D sobre la primera ruta, obtenemos:

A-D-E-B-C-A (función de evaluación: 207)

Y asi sucesivamente hasta obtener todas las combinaciones posibles. ¿Pero cuántas hay? Si tenemos N ciudades, tendremos N! combinaciones. N! es igual a:

$n! = n \times (n-1) \times (n-2) \times \ldots \times 1$

A cada posible solución la llamaremos **estado** y al conjunto de todas las posibles soluciones lo denominaremos **espacio de estados**. En nuestro ejemplo con cinco ciudades tenemos 120 posibles soluciones o estados. Un número manejable por cualquier ordenador personal.

$5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$

Con 7 ciudades tendríamos 5040 posibles soluciones.

Con 10 ciudades tendríamos 3.628.800 posibles soluciones.

Con 20 ciudades tendríamos 2.432.902.008.176.640.000 posibles soluciones.

Luego si cada permutación le ocupara al ordenador 1 microsegundo, con 20 ciudades necesitaríamos 77.147 años para generar todo el espacio de estado.

A este tipo de problemas los denominamos intratables y decimos que pertenecen a la **clase NP**, que sin entrar en detalle diremos que son una clase de problemas cuya complejidad hace que no sean resolubles en un tiempo razonable por un ordenador. Por lo tanto, queda claro que con métodos computacionales clásicos no es posible resolver este problema.