Aplicaciones de la programación lineal en planificación de proyectos

Ignacio García-Jurado

Departamento de Matemáticas
Universidade da Coruña

Xornada de formación para elaborar proyectos de estatística: aplicación á incubadora

Qué es un proyecto

- Un proyecto es un conjunto de tareas o actividades que deben realizarse a lo largo del tiempo para crear un producto o servicio único.
- El uso de herramientas matemáticas para la planificación de proyectos comienza a finales de los años cincuenta con el desarrollo de los métodos CPM y PERT.
- En 1969 se crea el Project Management Institute (PMI), una organización profesional internacional sin ánimo de lucro que promueve el conjunto de habilidades y cuerpo de conocimientos que son necesarios para la gestión de proyectos.

El ciclo de vida de un proyecto

1. **Formulación y selección.** En esta etapa los gestores definen el proyecto y su alcance y valoran su impacto en el marco del plan estratégico de la empresa. En base a ello, deciden si desarrollarlo o no.
2. **Planificación.** Si el proyecto ha sido seleccionado para ser desarrollado, los gestores realizarán una programación detallada. Definirán las tareas o actividades de las que consta, estudiarán las relaciones entre las tareas, estimarán los recursos necesarios para desarrollarlas y las duraciones de las tareas en base a los recursos que se les destinarán, estimarán también los costes y la duración del proyecto.
3. **Control.** En esta fase, los gestores auditarán el desarrollo del proyecto para que pueda ser terminado en el tiempo y coste previsto.
4. **Implantación y entrega.** En esta fase el proyecto se implanta y se entrega a los usuarios.

CPM y PERT

- CPM son las iniciales de Critical Path Method. El CPM y el PERT (Program Evaluation and Review Technique) son métodos de planificación de proyectos que fueron desarrollados independientemente a finales de los años cincuenta en la empresa DuPont y en la Marina americana, respectivamente.
- Ambos métodos eran muy parecidos y se basaban en modelizar el proyecto como un grafo y buscar el camino más largo de tal grafo.
- El CPM consideraba que todas las actividades tenían duraciones deterministas, mientras que el PERT hacía algunas consideraciones probabilísticas sobre la duración de las actividades.
- Hoy en día se designa indistintamente como CPM o PERT al método que describimos aquí.

El grafo

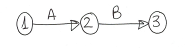
- El método CPM parte de la descomposición del proyecto en un conjunto de actividades, entendiendo por actividad la ejecución de una tarea que exige para su realización la utilización de recursos.
- Un suceso es un acontecimiento que no consume recursos y que indica el principio o fin de una o varias actividades.
- Para representar un proyecto se utiliza un grafo, en el que los arcos representan las actividades y los nodos representan los sucesos.

Instrucciones para construir el grafo

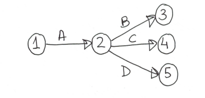
- Cada actividad se representa por un arco (i, j). i es el suceso inicio de la actividad y j es el suceso fin.
- El grafo debe tener un único suceso inicio del proyecto, que representa el inicio de una o más actividades, pero no representa el fin de ninguna. También debe tener un único suceso fin, que representa el fin de una o más actividades, pero no representa el inicio de ninguna.
- Las precedencias entre las actividades deben representarse en el grafo.
- La numeración de los nodos debe ser tal que no exista ningún arco (i, j) con $i > j$. Esto siempre es posible en un grafo CPM.
- Antes de construir el grafo CPM es recomendable construir el cuadro de prelaciones que contiene la lista de las actividades del proyecto junto con sus actividades precedentes.

Cómo se representan las precedencias

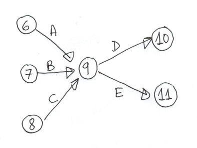
- A precede a B o, lo que es lo mismo, para que comience la actividad B es necesario que haya terminado la A.



- A precede a B, C y D.



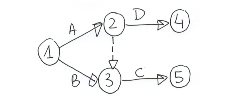
- A, B y C preceden a D y E.



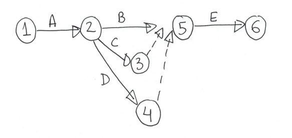
Actividades ficticias

A veces será necesario utilizar actividades ficticias para representar las precedencias entre las actividades.

- Por ejemplo, si queremos representar que A y B preceden a C y que A precede a D necesitaremos usar una actividad ficticia.



- También conviene usar actividades ficticias para evitar que varias actividades tengan los mismos sucesos inicio y fin. Por ejemplo: A precede a B, C y D; B, C y D preceden a E.



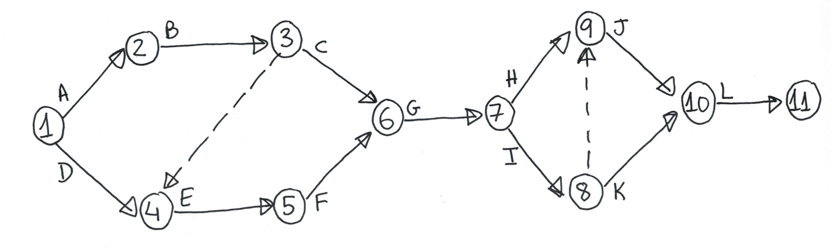
Proyecto: Organización y estudio del impacto de la semana de la lucha contra el hambre en un instituto de enseñanza secundaria.

- A. Diseño de las encuestas (2 semanas).
- B. Realización y análisis estadístico de la primera encuesta (2 semanas).
- C. Confección de la primera parte de la memoria, incluyendo estructura general y análisis de la primera encuesta (2 semanas).
- D. Documentación sobre el problema del hambre en el mundo (3 semanas).
- E. Diseño actividades a realizar (2 semanas).
- F. Organización de las actividades (6 semanas).
- G. Semana de la lucha contra el hambre (1 semana).
- H. Confección de la segunda parte de la memoria incluyendo los datos relativos a la semana de la lucha contra el hambre (1 semana).
- I. Realización y análisis estadístico de la segunda encuesta (2 semanas).
- J. Finalización de la memoria (2 semanas).
- K. Preparación de la presentación pública (1 semana).
- L. Presentación pública de los resultados (1 semana).

- A precede a B.
- B precede a C.
- B,D preceden a E.
- E precede a F.
- C,F preceden a G.
- G precede a H,I.
- H precede a J
- I precede a J,K.
- J,K preceden a L.

- A precede a B.
- B precede a C.
- B,D preceden a E.
- E precede a F.
- C,F preceden a G.
- G precede a H,I.
- H precede a J
- I precede a J,K.
- J,K preceden a L.

Actividad	A	B	C	D	E	F	G	H	I	J	K	L
Precedentes Inmediatas	-	A	B	-	B,D	E	C,F	G	G	H,I	I	J,K
Duración	2	2	2	3	2	6	1	1	2	2	1	1



Duración de las actividades

Una vez que hemos contruido el grafo que representa el proyecto debemos asignarle una duración a cada una de las actividades. En muchos casos las duraciones de las actividades serán aleatorias y se tratará de estimarlas haciendo uso de técnicas estadísticas.

Supongamos que (N, M) es el grafo que representa el proyecto. $N = \{1, \ldots, n\}$ es el conjunto de nodos (que representan a los sucesos) y $M \subset N \times N$ es el conjunto de arcos (que representan a las actividades).

Un procedimiento heurístico que se recomienda en muchas ocasiones para estimar tales duraciones es el siguiente. Si para una cierta actividad $(i, j) \in M$ la experiencia nos indica que, si todo marcha especialmente bien, su duración será a_{ij}, si todo se complica su duración será b_{ij} y lo más probable es que su duración sea m_{ij}, es conveniente estimar su duración media t_{ij} con la expresión siguiente:

$$\frac{a_{ij} + b_{ij} + 4m_{ij}}{6}.$$

Tiempo early de un suceso

El tiempo early del suceso j, que denotamos por t_j, nos da el tiempo que, como mínimo, es necesario para llegar hasta j. Formalmente,

$$t_j = \begin{cases} 0 & \text{si } j = 1, \\ \max\limits_{i \in N \mid (i,j) \in M} (t_i + t_{ij}) & \text{en otro caso.} \end{cases}$$

- El proceso de cálculo de los tiempos early es iterativo.
- El tiempo early del suceso fin de proyecto nos da el tiempo mínimo necesario para terminar el proyecto.
- En términos de grafos, es fácil comprobar que el tiempo early de un suceso nos da la longitud del camino más largo desde el suceso inicio hasta él.

Tiempo last de un suceso

El tiempo last del suceso i, que denotamos por t_i^*, nos da el tiempo que, como máximo, podemos emplear en llegar hasta él sin que se retrase la duración del proyecto. Formalmente,

$$t_i^* = \begin{cases} t_i & \text{si } i = n, \\ \min_{j \in N \mid (i,j) \in M} (t_j^* - t_{ij}) & \text{en otro caso.} \end{cases}$$

- El proceso de cálculo de los tiempos last es iterativo.
- El tiempo last del suceso inicio de proyecto es cero.
- En términos de grafos, es fácil comprobar que el tiempo last del suceso i nos da la longitud del camino más largo desde el suceso inicio hasta el fin menos la del camino más largo desde i hasta el fin.

Holgura de una actividad

Sea un proyecto representado por un grafo (N, M). La holgura total de la actividad $(i, j) \in M$, que se denota por H_{ij}^T, se define del siguiente modo:

$$H_{ij}^T = t_j^* - t_i - t_{ij}.$$

- La holgura total de una actividad nos indica el tiempo que puede retrasarse la actividad sin que, por ello, se retrase la duración del proyecto.
- Se dice que una actividad es crítica cuando su holgura total es cero.
- Puede demostrarse que las actividades críticas son las que pertenecen a caminos más largos desde el inicio hasta el fin del proyecto.

Calendario del proyecto

En base a la definición de holgura total y a las definiciones de los tiempos early y last de un suceso, se puede establecer el siguiente calendario para la ejecución de cada actividad $(i, j) \in M$:

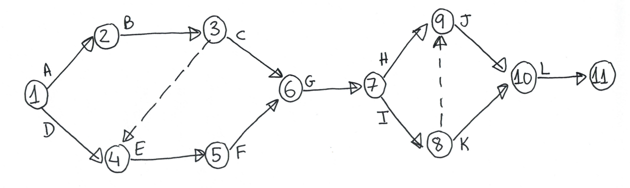
- Δ_{ij}, la fecha de comienzo más temprana de (i, j), es t_i.
- Δ^*_{ij}, la fecha de comienzo más tardía de (i, j), es $t^*_j - t_{ij} = t_i + H^T_{ij}$.
- ∇_{ij}, la fecha de finalización más temprana de (i, j), es $t_i + t_{ij} = t^*_j - H^T_{ij}$.
- ∇^*_{ij}, la fecha de finalización más tardía de (i, j), es t^*_j.

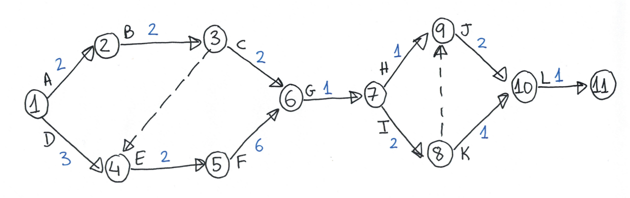
Proyecto: Organización y estudio del impacto de la semana de la lucha contra el hambre en un instituto de enseñanza secundaria.

- A. Diseño de las encuestas (2 semanas).
- B. Realización y análisis estadístico de la primera encuesta (2 semanas).
- C. Confección de la primera parte de la memoria, incluyendo estructura general y análisis de la primera encuesta (2 semanas).
- D. Documentación sobre el problema del hambre en el mundo (3 semanas).
- E. Diseño actividades a realizar (2 semanas).
- F. Organización de las actividades (6 semanas).
- G. Semana de la lucha contra el hambre (1 semana).
- H. Confección de la segunda parte de la memoria incluyendo los datos relativos a la semana de la lucha contra el hambre (1 semana).
- I. Realización y análisis estadístico de la segunda encuesta (2 semanas).
- J. Finalización de la memoria (2 semanas).
- K. Preparación de la presentación pública (1 semana).
- L. Presentación pública de los resultados (1 semana).

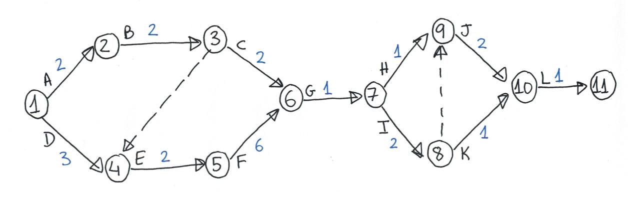
- A precede a B.
- B precede a C.
- B,D preceden a E.
- E precede a F.
- C,F preceden a G.
- G precede a H,I.
- H precede a J
- I precede a J,K.
- J,K preceden a L.

El grafo


1
2
3
4
5
6
7
8
9
10
11
A
B
C
D
E
F
G
H
I
J
K
L


1
2
3
4
5
6
7
8
9
10
11
A 2
B 2
C 2
D 3
E 2
F 6
G 1
H 1
I 2
J 2
K 1
L 1

Cálculo de los tiempos early y last



t_i		1	2	3	4	5	6	7	8	9	10	11
	1		2		3							
	2			2								
	3				0		2					
	4					2						
	5						6					
	6							1				
	7								2	1		
	8									0	1	
	9										2	
	10											1
	11											
	t_i^*											

Cálculo de los tiempos early y last

t_i		1	2	3	4	5	6	7	8	9	10	11
0	1		2		3							
	2			2								
	3				0		2					
	4					2						
	5						6					
	6							1				
	7								2	1		
	8									0	1	
	9										2	
	10											1
	11											
	t_i^*											

Cálculo de los tiempos early y last

t_i		1	2	3	4	5	6	7	8	9	10	11
0	1		2		3							
2	2			2								
	3				0		2					
	4					2						
	5						6					
	6							1				
	7								2	1		
	8									0	1	
	9										2	
	10											1
	11											
	t_i^*											

Cálculo de los tiempos early y last

t_i		1	2	3	4	5	6	7	8	9	10	11
0	1		2		3							
2	2			2								
4	3				0		2					
	4					2						
	5						6					
	6							1				
	7								2	1		
	8									0	1	
	9										2	
	10											1
	11											
	t_i^*											

Cálculo de los tiempos early y last

t_i		1	2	3	4	5	6	7	8	9	10	11
0	1		2		3							
2	2			2								
4	3				0		2					
4	4					2						
	5						6					
	6							1				
	7								2	1		
	8									0	1	
	9										2	
	10											1
	11											
	t_i^*											

Cálculo de los tiempos early y last

t_i		1	2	3	4	5	6	7	8	9	10	11
0	1		2		3							
2	2			2								
4	3				0		2					
4	4					2						
6	5						6					
12	6							1				
13	7								2	1		
15	8									0	1	
15	9										2	
17	10											1
18	11											
	t_i^*											

Cálculo de los tiempos early y last

t_i		1	2	3	4	5	6	7	8	9	10	11
0	1		2		3							
2	2			2								
4	3				0		2					
4	4					2						
6	5						6					
12	6							1				
13	7								2	1		
15	8									0	1	
15	9										2	
17	10											1
18	11											
	t_i^*											18

Cálculo de los tiempos early y last

t_i		1	2	3	4	5	6	7	8	9	10	11
0	1		2		3							
2	2			2								
4	3				0		2					
4	4					2						
6	5						6					
12	6							1				
13	7								2	1		
15	8									0	1	
15	9										2	
17	10											1
18	11											
	t_i^*										17	18

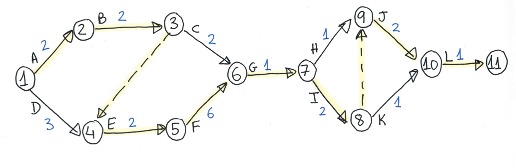
Cálculo de los tiempos early y last

t_i		1	2	3	4	5	6	7	8	9	10	11
0	1		2		3							
2	2			2								
4	3				0		2					
4	4					2						
6	5						6					
12	6							1				
13	7								2	1		
15	8									0	1	
15	9										2	
17	10											1
18	11											
	t_i^*	0	2	4	4	6	12	13	15	15	17	18

Cálculo de las holguras

t_i		1	2	3	4	5	6	7	8	9	10	11
0	1		2		3							
2	2			2								
4	3				0		2					
4	4					2						
6	5						6					
12	6							1				
13	7								2	1		
15	8									0	1	
15	9										2	
17	10											1
18	11											
	t_i^*	0	2	4	4	6	12	13	15	15	17	18

(i,j)	t_{ij}	t_i	t_j^*	H_{ij}^T	¿crítica?
A=(1,2)	2	0	2	0	sí
B=(2,3)	2	2	4	0	sí
C=(3,6)	2	4	12	6	
D=(1,4)	3	0	4	1	
E=(4,5)	2	4	6	0	sí
F=(5,6)	6	6	12	0	sí
G=(6,7)	1	12	13	0	sí
H=(7,9)	1	13	15	1	
I=(7,8)	2	13	15	0	sí
J=(9,10)	2	15	17	0	sí
K=(8,10)	1	15	17	1	
L=(10,11)	1	17	18	0	sí



Calendario

(i,j)	t_{ij}	t_i	t_j^*	H_{ij}^T	¿crítica?
A=(1,2)	2	0	2	0	sí
B=(2,3)	2	2	4	0	sí
C=(3,6)	2	4	12	6	
D=(1,4)	3	0	4	1	
E=(4,5)	2	4	6	0	sí
F=(5,6)	6	6	12	0	sí
G=(6,7)	1	12	13	0	sí
H=(7,9)	1	13	15	1	
I=(7,8)	2	13	15	0	sí
J=(9,10)	2	15	17	0	sí
K=(8,10)	1	15	17	1	
L=(10,11)	1	17	18	0	sí

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
A	A																
		B	B														
				E	E												
						F	F	F	F	F	F						
												G					
													I	I			
															J	J	
																	L
D	D	D	D														
				C	C	C	C	C	C	C	C						
													H	H			
															K	K	

Programación matemática y planificación de proyectos

- Las técnicas de programación matemática resuelven problemas de optimización de funciones con restricciones.
- Son una herramienta de gran importancia en gestión de proyectos, especialmente en temas de planificación, aunque tienen aplicaciones en otros aspectos de la gestión.

Definición

Un problema de programación matemática consiste en lo siguiente. Dada una función $f : \mathbb{R}^n \to \mathbb{R}$ y un conjunto $S \subset \mathbb{R}^n$, se trata de encontrar $\bar{x} \in \mathbb{R}^n$ que resuelva

$$\begin{aligned} max \quad & f(x) \\ s.a: \quad & x \in S. \end{aligned}$$

- Diremos que f es la *función objetivo* y que S es el *conjunto factible*.
- Si $\bar{x} \in \mathbb{R}^n$ diremos que $\bar{x}$ es una *solución*. Si, además, $\bar{x} \in S$ diremos que $\bar{x}$ es una *solución factible*. Si $\bar{x} \in S$ y $f(\bar{x}) \geq f(x)$ para todo $x \in S$ diremos que $\bar{x}$ es una *solución óptima*.
- Si $S = \emptyset$ diremos que el problema es *infactible*. Si existe una sucesión $\{x_k\} \subset S$ tal que $\{f(x_k)\} \to \infty$ cuando $k \to \infty$ diremos que el problema es *no acotado*.
- Si $S \subset \mathbb{Z}^n$ decimos que el problema es entero.

Un ejemplo

Consideremos una empresa de ingeniería que realiza dos tipos de proyectos. En la empresa trabajan cuatro informáticos y dos matemáticos (uno de ellos a tiempo parcial). La siguiente tabla contiene datos que indican la disponibilidad diaria en horas de trabajo de informáticos y matemáticos, el número aproximado de horas de trabajo de informáticos y matemáticos que son necesarias para completar un proyecto de cada tipo, y los beneficios aproximados en miles de euros para cada tipo de proyecto. Se trata de determinar el número de proyectos de cada tipo que debe realizar la empresa diariamente para maximizar los beneficios.

	Tipo 1	Tipo 2	Disponibilidad
Informáticos	6	8	32
Matemáticos	4	2	12
Beneficios	10	4	

Un ejemplo

	Tipo 1	Tipo 2	Disponibilidad
Informáticos	6	8	32
Matemáticos	4	2	12
Beneficios	10	4	

Este problema se puede formular haciendo uso de la programación lineal del siguiente modo.

- x_1 = número de proyectos de tipo 1 que se realizan diariamente.
- x_2 = número de proyectos de tipo 2 que se realizan diariamente.

$$\begin{array}{lll} max & 10x_1 + 4x_2 & \\ s.a: & 6x_1 + 8x_2 & \leq 32 \\ & 4x_1 + 2x_2 & \leq 12 \\ & x_1, x_2 & \geq 0 \end{array}$$

Programación lineal

Un problema de programación lineal es un problema de programación matemática en el que la función objetivo es lineal y el conjunto factible viene definido por una colección finita de igualdades o desigualdades lineales.

- Existen diversos procedimientos para resolver problemas de programación lineal, el más conocido de los cuales es el método símplex.
- Hay en el mercado una amplia variedad de aplicaciones (algunas libres) para resolver problemas de programación lineal. En esta presentación usaremos los paquetes linprog y lpSolve de R.

La semana de la lucha contra el hambre como un problema de programación lineal

Actividad	A	B	C	D	E	F	G	H	I	J	K	L
Precedentes Inmediatas	-	A	B	-	B,D	E	C,F	G	G	H,I	I	J,K
Duración	2	2	2	3	2	6	1	1	2	2	1	1

- $x_i =$ instante de comienzo de la actividad i ($i \in \{A, \ldots, L\}$).
- $x_{fin} =$ instante de finalización del proyecto.

minimizar x_{fin}, sujeto a:

$$
\begin{array}{llllll}
x_A + 2 \leq x_B & x_D + 3 \leq x_E & x_F + 6 \leq x_G & x_H + 1 \leq x_J & x_J + 2 \leq x_L & x_L + 1 \leq x_{fin} \\
x_B + 2 \leq x_C & x_E + 2 \leq x_F & x_G + 1 \leq x_H & x_I + 2 \leq x_J & x_K + 1 \leq x_L & x_A \geq 0 \\
x_B + 2 \leq x_E & x_C + 2 \leq x_G & x_G + 1 \leq x_I & x_I + 2 \leq x_K & & x_D \geq 0
\end{array}
$$

Código de R

```
c1<-c(rep(0,12),1)
b1<-c(rep(-2,3),-3, rep(-2,2),-6, rep(-1,3), rep(-2,3), rep(-1,2))
restmat1=matrix(c(
1,-1, rep(0,11),
0,1,-1, rep(0,10),
0,1,0,0,-1, rep(0,8),
0,0,0,1,-1, rep(0,8),
0,0,0,0,1,-1, rep(0,7),
0,0,1,0,0,0,-1, rep(0,6),
rep(0,5),1,-1, rep(0,6),
rep(0,6),1,-1, rep(0,5),
rep(0,6),1,0,-1, rep(0,4),
rep(0,7),1,0,-1, rep(0,3),
rep(0,8),1,-1, rep(0,3),
rep(0,8),1,0,-1, rep(0,2),
rep(0,9),1,0,-1,0,
rep(0,10),1,-1,0,
rep(0,11),1,-1
),nrow=15,byrow=TRUE)
dirv<-c(rep("<=",15))
res1<-solveLP(c1,b1,restmat1,maximum=FALSE,const.dir=dirv,lpSolve=TRUE)
res1
```

Resultado

	x_k
A	0
B	2
C	4
D	0
E	4
F	6
G	12
H	13
I	13
J	15
K	15
L	17
fin	18

Función objetivo: 18

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
A	A																
		B	B														
				E	E												
						F	F	F	F	F	F						
												G					
													I	I			
															J	J	
																	L
D	D	D	D														
				C	C	C	C	C	C	C	C						
													H	H			
															K	K	

Fechas de comienzo más tempranas como un problema de programación lineal

Actividad	A	B	C	D	E	F	G	H	I	J	K	L
Precedentes Inmediatas	-	A	B	-	B,D	E	C,F	G	G	H,I	I	J,K
Duración	2	2	2	3	2	6	1	1	2	2	1	1

- $x_i =$ instante de comienzo de la actividad i ($i \in \{A, \ldots, L\}$).
- $x_{fin} =$ instante de finalización del proyecto.

minimizar $\sum_{i=A}^{L} x_i$, sujeto a:

$$
\begin{array}{llllll}
x_A + 2 \leq x_B & x_D + 3 \leq x_E & x_F + 6 \leq x_G & x_H + 1 \leq x_J & x_J + 2 \leq x_L & x_L + 1 \leq 18 \\
x_B + 2 \leq x_C & x_E + 2 \leq x_F & x_G + 1 \leq x_H & x_I + 2 \leq x_J & x_K + 1 \leq x_L & x_A \geq 0 \\
x_B + 2 \leq x_E & x_C + 2 \leq x_G & x_G + 1 \leq x_I & x_I + 2 \leq x_K & & x_D \geq 0
\end{array}
$$

Fechas de comienzo más tardías como un problema de programación lineal

Actividad	A	B	C	D	E	F	G	H	I	J	K	L
Precedentes Inmediatas	-	A	B	-	B,D	E	C,F	G	G	H,I	I	J,K
Duración	2	2	2	3	2	6	1	1	2	2	1	1

- $x_i =$ instante de comienzo de la actividad i ($i \in \{A, \ldots, L\}$).
- $x_{fin} =$ instante de finalización del proyecto.

maximizar $\sum_{i=A}^{L} x_i$, sujeto a:

$$\begin{array}{llllll}
x_A + 2 \leq x_B & x_D + 3 \leq x_E & x_F + 6 \leq x_G & x_H + 1 \leq x_J & x_J + 2 \leq x_L & x_L + 1 \leq 18 \\
x_B + 2 \leq x_C & x_E + 2 \leq x_F & x_G + 1 \leq x_H & x_I + 2 \leq x_J & x_K + 1 \leq x_L & x_A \geq 0 \\
x_B + 2 \leq x_E & x_C + 2 \leq x_G & x_G + 1 \leq x_I & x_I + 2 \leq x_K & & x_D \geq 0
\end{array}$$

Códigos de R

```
c2<-c(rep(1,12))
b2<-c(rep(-2,3),-3, rep(-2,2),-6, rep(-1,3), rep(-2,3),-1,17)
restmat2=matrix(c(
1,-1, rep(0,10),
0,1,-1, rep(0,9),
0,1,0,0,-1, rep(0,7),
0,0,0,1,-1, rep(0,7),
0,0,0,0,1,-1, rep(0,6),
0,0,1,0,0,0,-1, rep(0,5),
rep(0,5),1,-1, rep(0,5),
rep(0,6),1,-1, rep(0,4),
rep(0,6),1,0,-1, rep(0,3),
rep(0,7),1,0,-1, rep(0,2),
rep(0,8),1,-1, rep(0,2),
rep(0,8),1,0,-1, rep(0,1),
rep(0,9),1,0,-1,
rep(0,10),1,-1,
rep(0,11),1
),nrow=15,byrow=TRUE)
dirv<-c(rep("<=",15))
res2<-solveLP(c2,b2,restmat2,maximum=FALSE,const.dir=dirv,lpSolve=TRUE)
res2
```

Códigos de R

```
c3<-c(rep(1,12))
b3<-c(rep(-2,3),-3, rep(-2,2),-6, rep(-1,3), rep(-2,3),-1,17)
restmat3=matrix(c(
1,-1, rep(0,10),
0,1,-1, rep(0,9),
0,1,0,0,-1, rep(0,7),
0,0,0,1,-1, rep(0,7),
0,0,0,0,1,-1, rep(0,6),
0,0,1,0,0,0,-1, rep(0,5),
rep(0,5),1,-1, rep(0,5),
rep(0,6),1,-1, rep(0,4),
rep(0,6),1,0,-1, rep(0,3),
rep(0,7),1,0,-1, rep(0,2),
rep(0,8),1,-1, rep(0,2),
rep(0,8),1,0,-1, rep(0,1),
rep(0,9),1,0,-1,
rep(0,10),1,-1,
rep(0,11),1
),nrow=15,byrow=TRUE)
dirv<-c(rep("<=",15))
res3<-solveLP(c3,b3,restmat3,maximum=TRUE,const.dir=dirv,lpSolve=TRUE)
res3
```

Resultado

	x_k
A	0
B	2
C	4
D	0
E	4
F	6
G	12
H	13
I	13
J	15
K	15
L	17

	x_k
A	0
B	2
C	10
D	1
E	4
F	6
G	12
H	14
I	13
J	15
K	16
L	17

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
A	A																
		B	B														
				E	E												
						F	F	F	F	F	F						
												G					
													I	I			
															J	J	
																	L
D	D	D	D														
				C	C	C	C	C	C	C	C						
													H	H			
															K	K	

Ventajas de usar la programación lineal

- No es necesario construir el grafo para realizar la planificación del proyecto.
- Se puede usar cualquier solver de programación lineal para resolver el problema.
- La programación lineal permite modelizar otras restricciones que pueden afectar al proyecto.

Aplicaciones de la programación lineal en planificación de proyectos

Ignacio García-Jurado

Departamento de Matemáticas
Universidade da Coruña

Xornada de formación para elaborar proyectos de estatística: aplicación á incubadora