Investigación espacial: Matemáticas e Inteligencia Artificial

Javier Yáñez

CU. Estadística e Investigación Operativa

Facultad de CC. Matemáticas (UCM)

Curso Exploración Espacial y su utilización

Madrid, 12 de noviembre de 2024

Esquema

1. **Planificación de las observaciones de un telescopio.**
2. **El modelo matemático.**
3. **Dificultades computacionales de la solución. Complejidad.**
4. **Metaheurísticas. Un enfoque basado en la Inteligencia Artificial.**
5. **Análisis de datos astronómicos.**
6. **Solución basada en redes neuronales. Aprendizaje automático**
7. **Memoria asociativa. Modelo de Hopfield.**

Planificación de las observaciones de un telescopio

1.- Entrada:

- **Conjunto de observaciones astronómicas solicitadas por los científicos durante un período de tiempo.**
- **Cada observación se caracteriza por:**
  - **Objeto a observar, coordenadas celestes.**
  - **Duración de la observación.**
  - **Tiempos posibles de la observación.**

Planificación de las observaciones de un telescopio

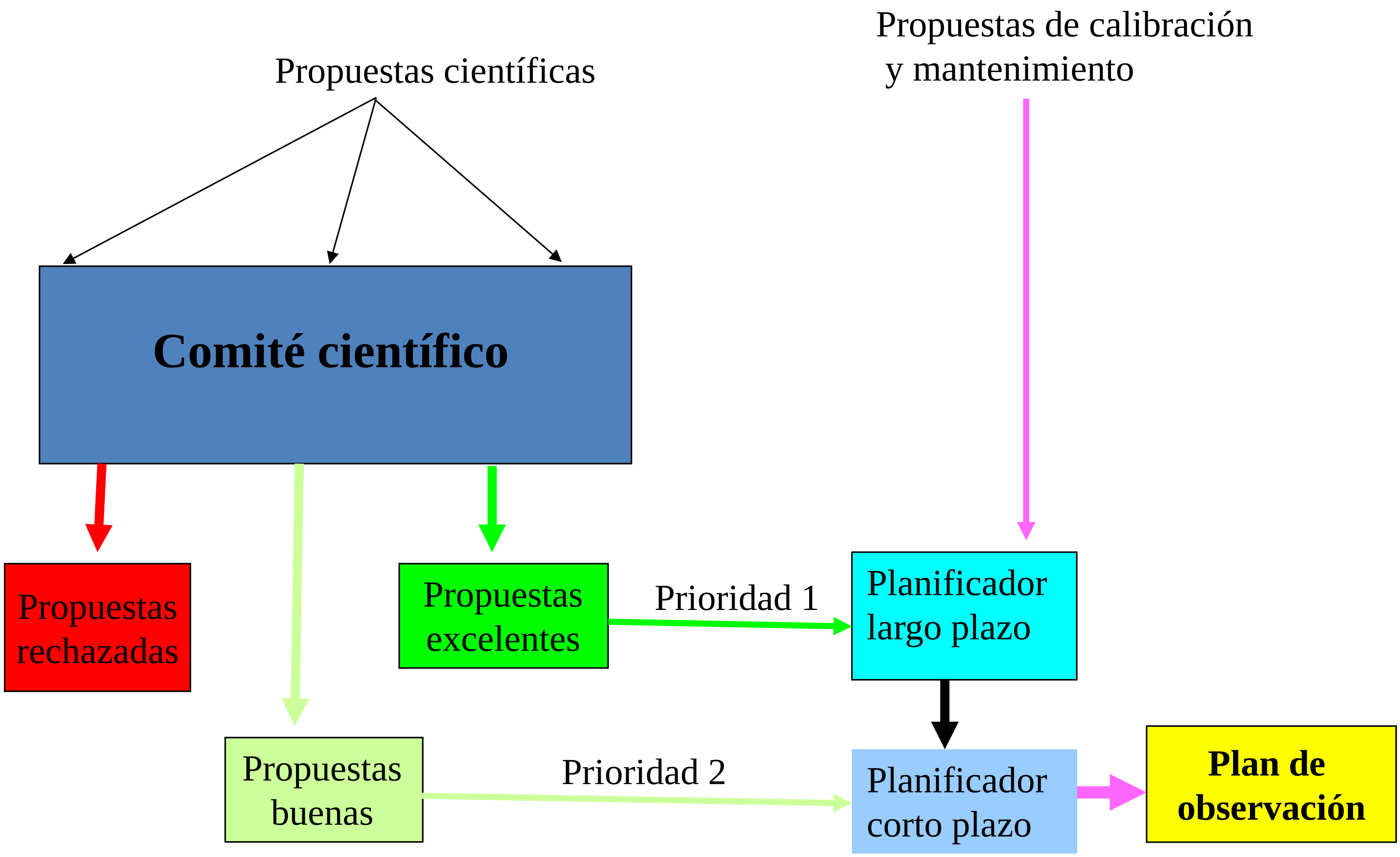
2.- Valoración de las solicitudes:

- **Comité científico.**
- **Cada observación es clasificada según su relevancia científica.**

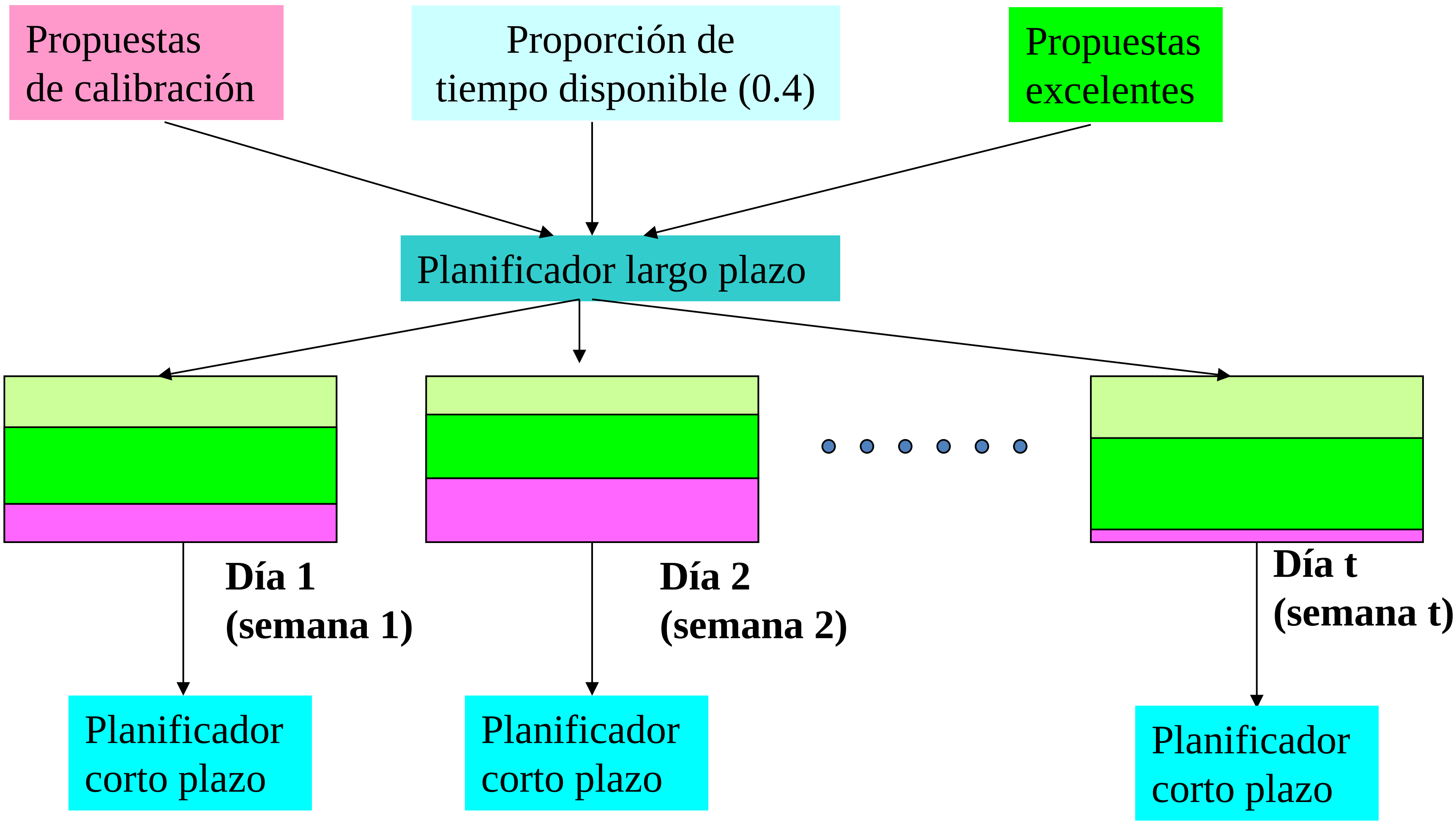
3.- Asignación de tiempos: el plan de observación:

- **Se distinguen dos niveles de planificación:**
  - **Planificación a largo plazo (un semestre, por ejemplo)**
  - **Planificación a corto plazo (una semana, un día, …)**
- **El objetivo es maximizar el retorno científico del uso del telescopio.**
- **Debe asignar tiempos para las operaciones de calibración y mantenimiento del propio telescopio.**
- **Esquemáticamente,**

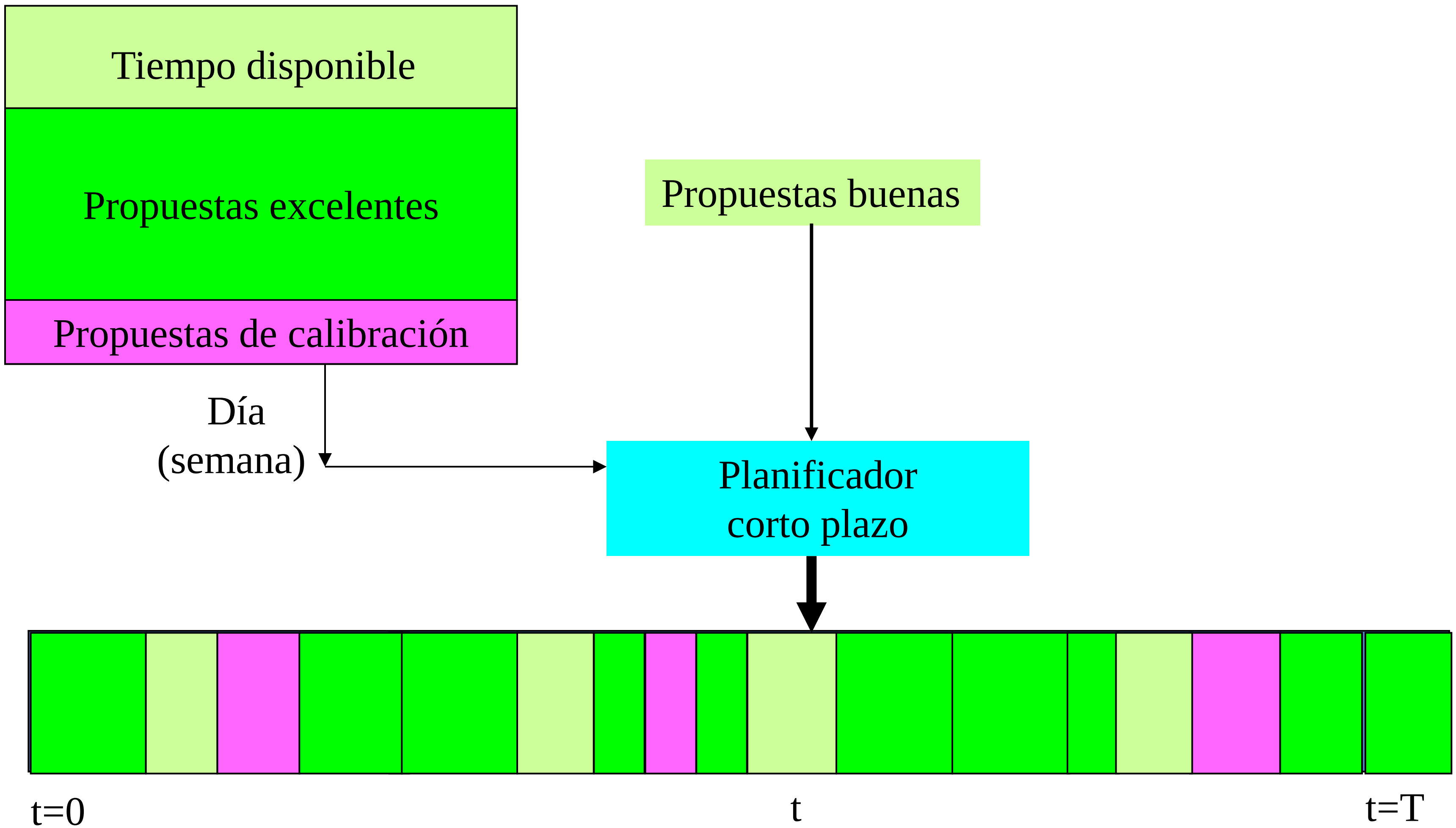
Planificación de las observaciones de un telescopio



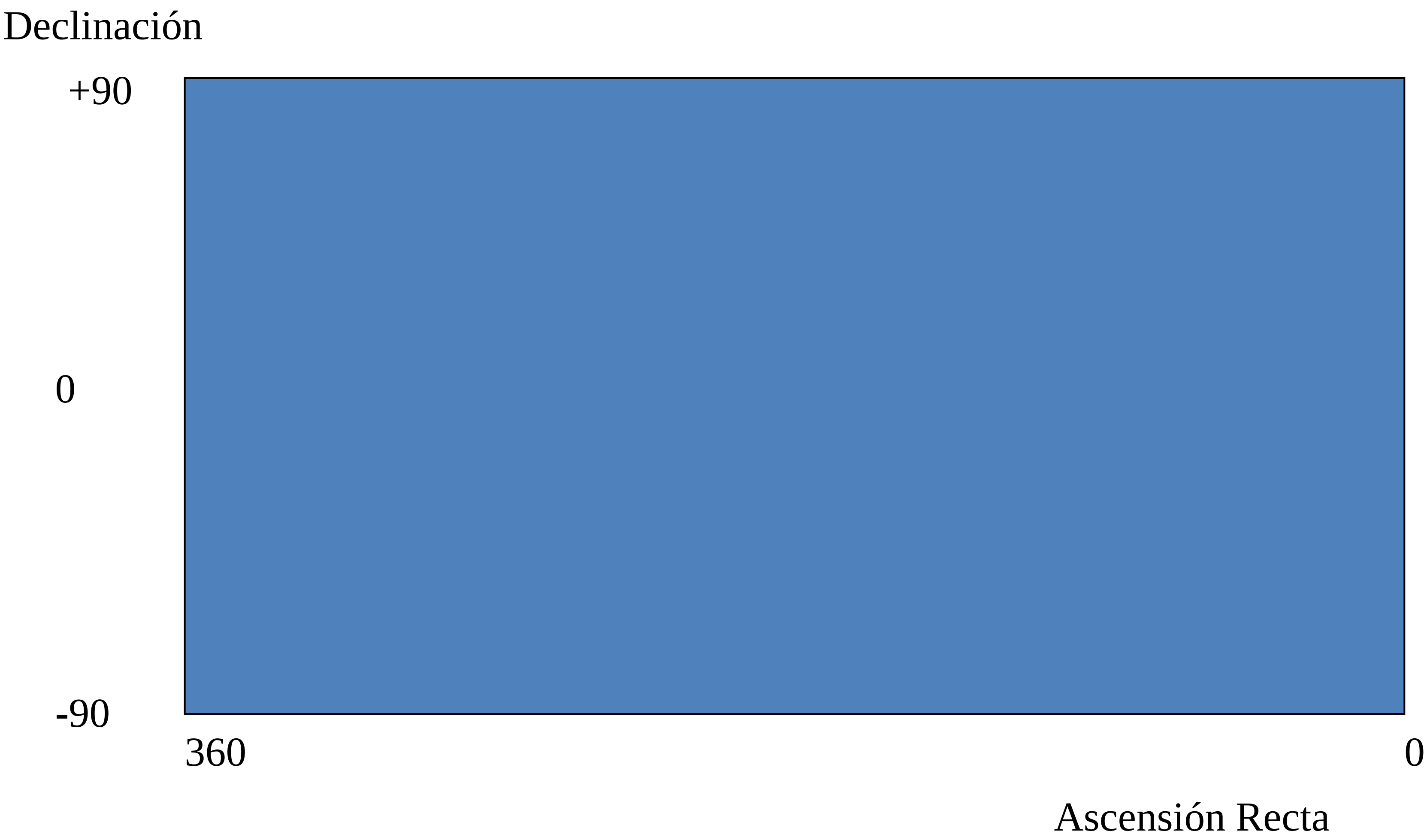
Planificación de las observaciones de un telescopio



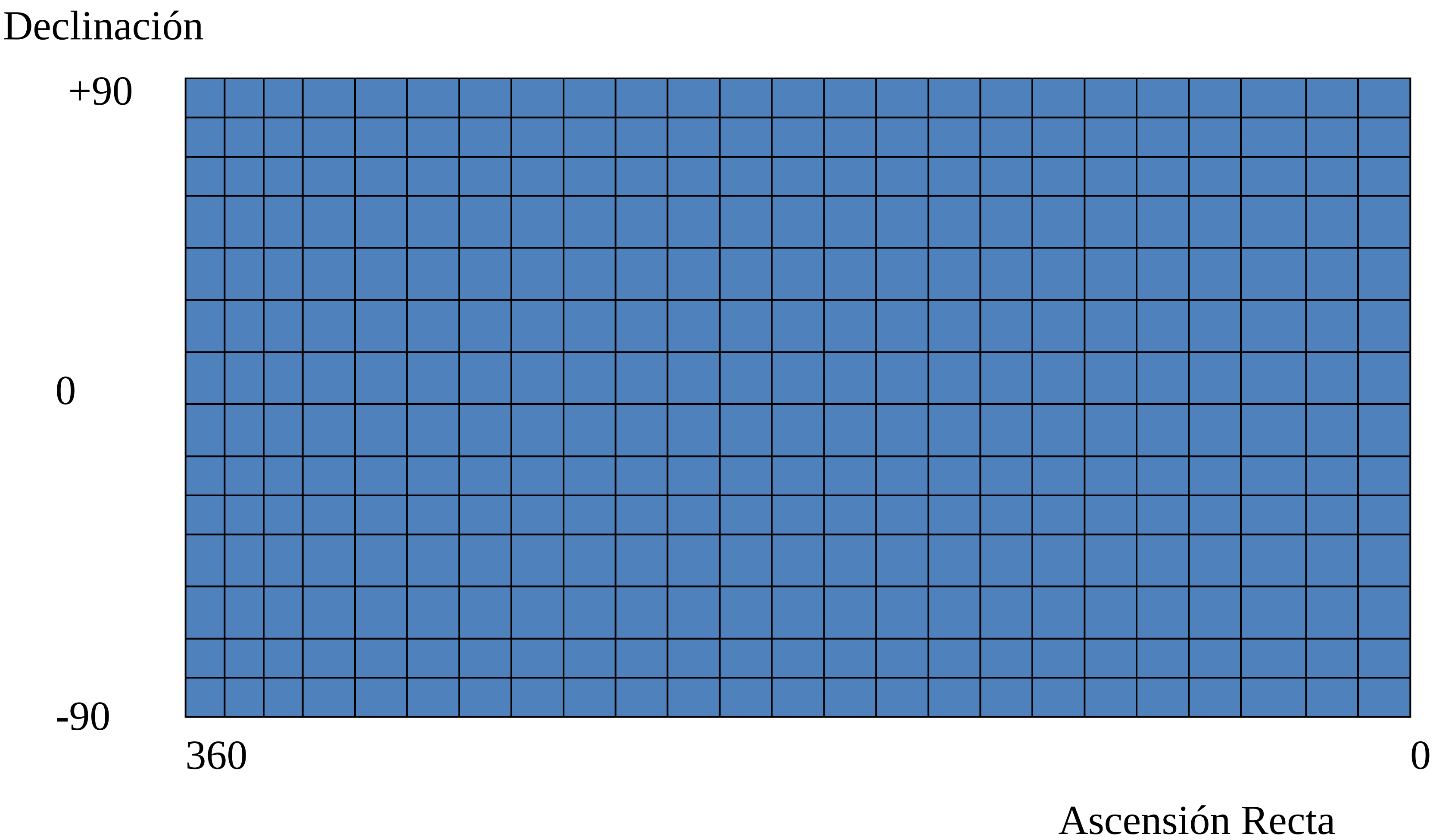
Planificación de las observaciones de un telescopio



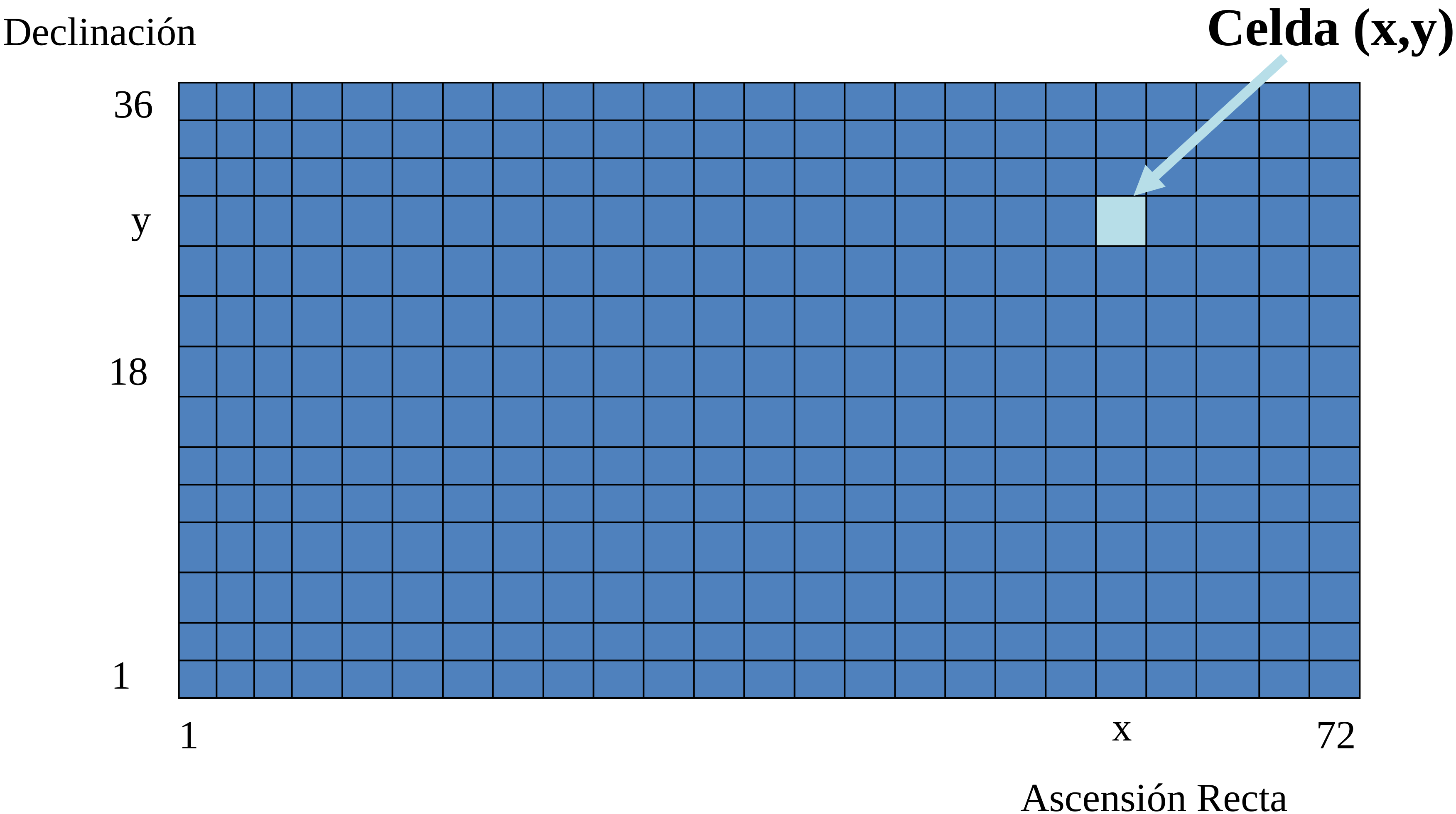
El “plano” (Asc.R.; Decl.)



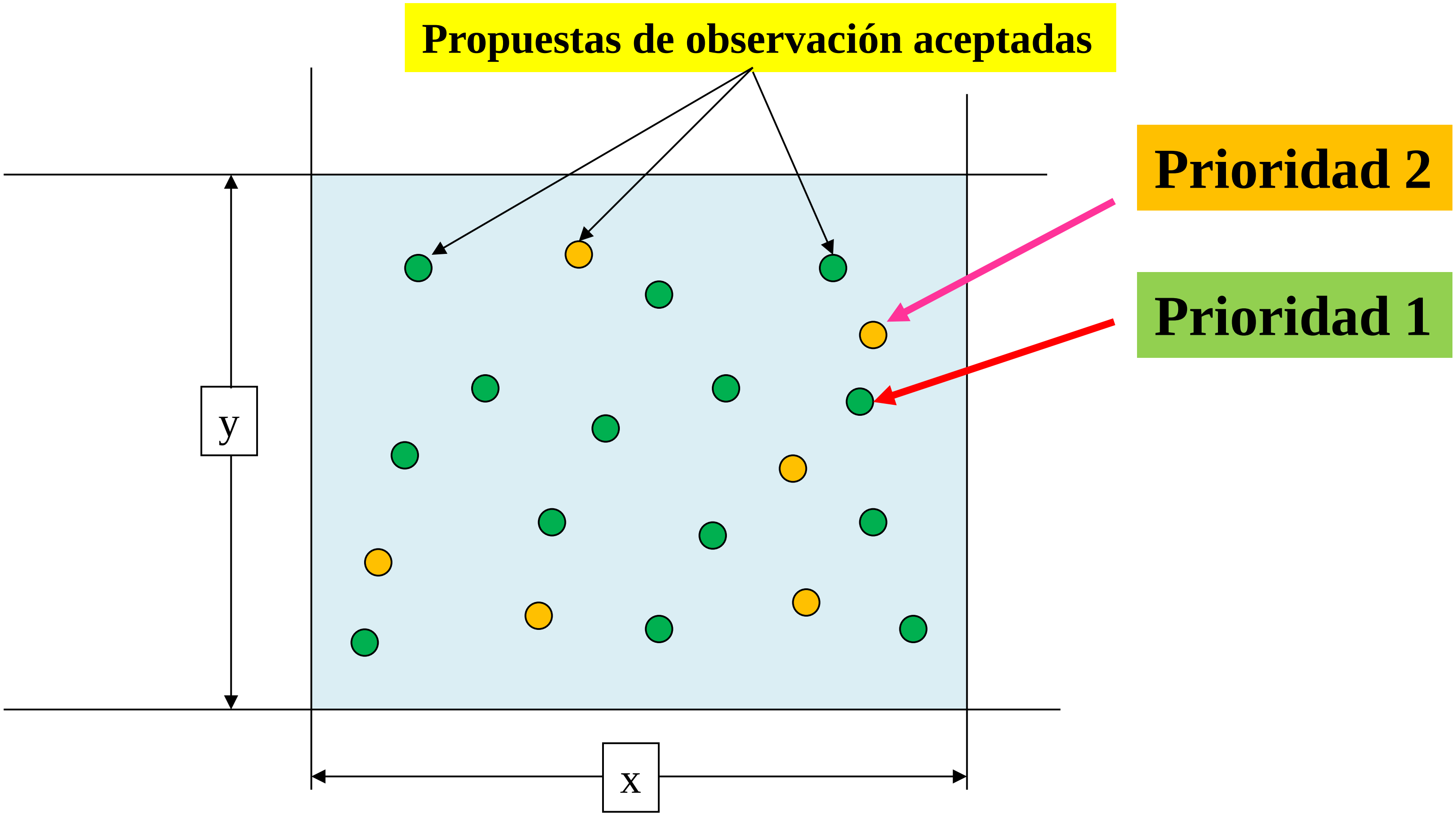
El “plano” (Asc.R.; Decl.)



El "plano" (Asc.R.; Decl.)

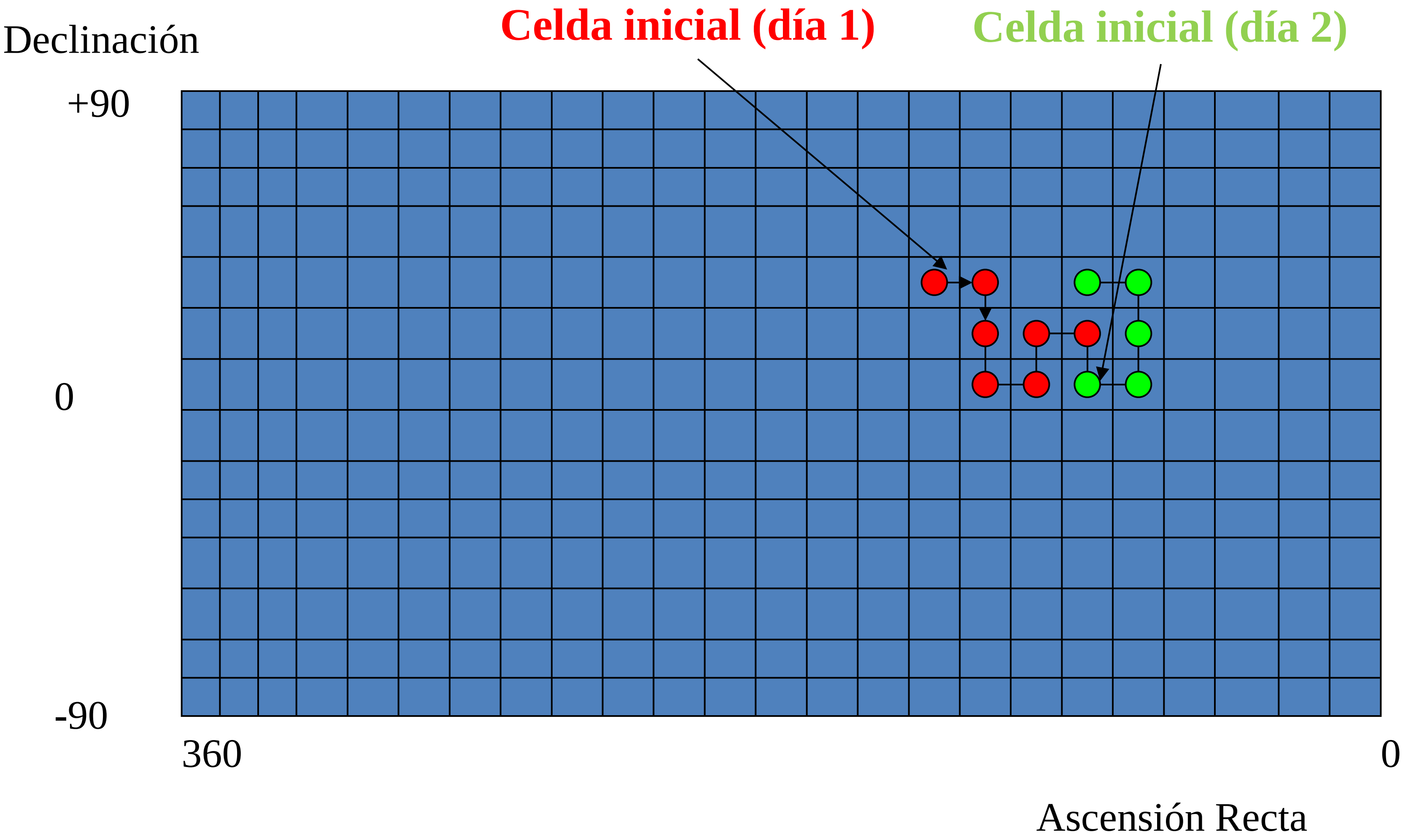


Las celdas

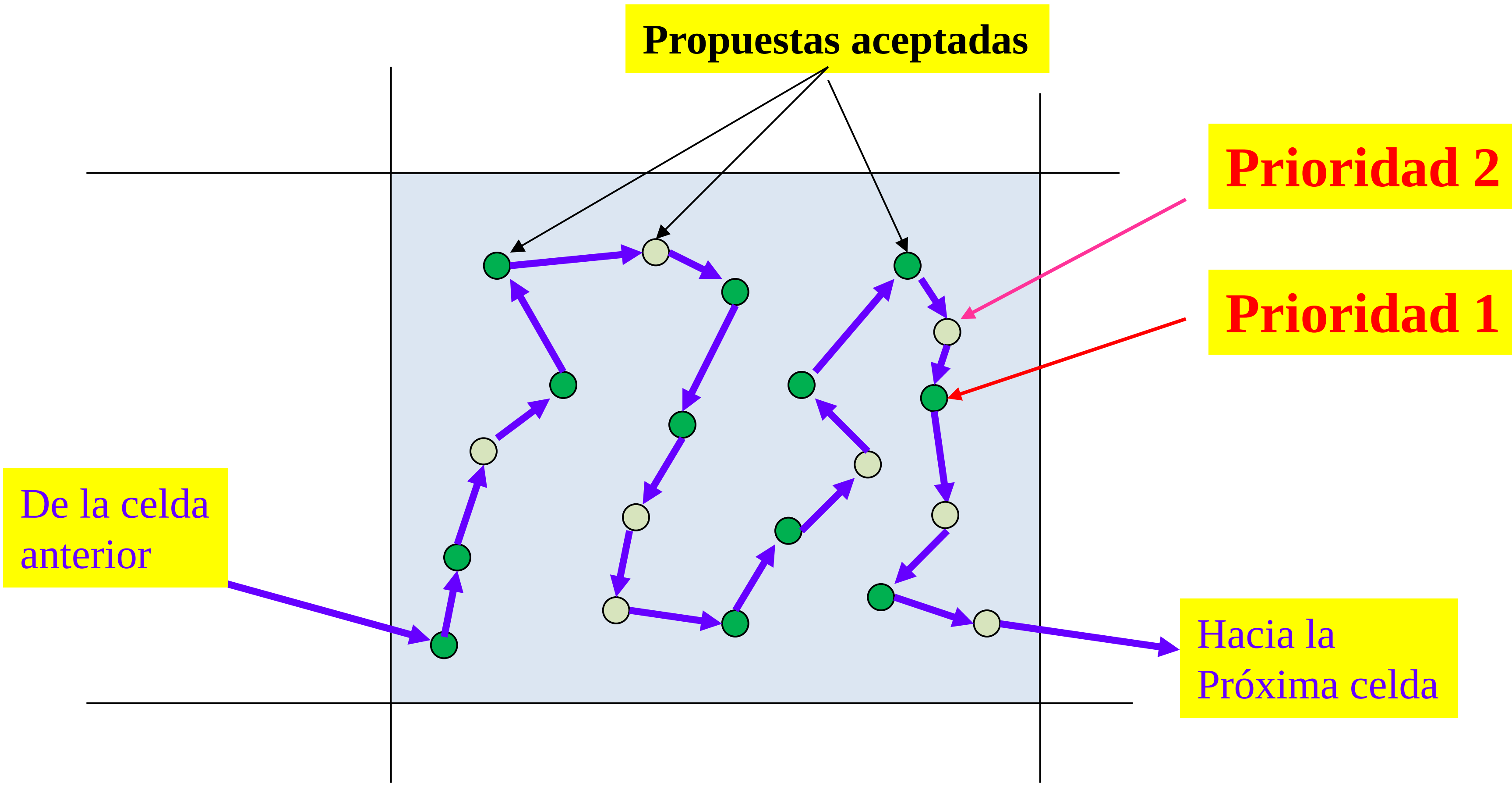


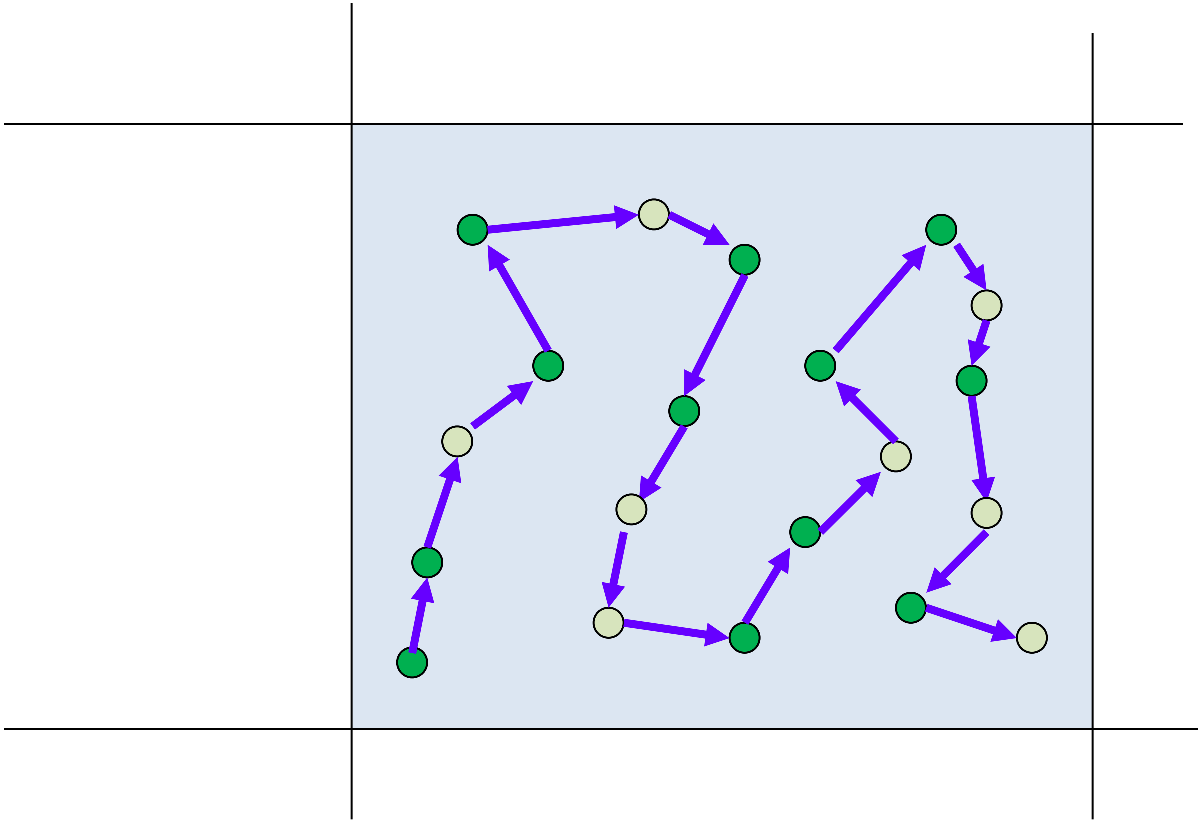
val(x,y)=Nro. Prop. excelentes / Duración prop.exc.

Solución de la planificación a largo plazo

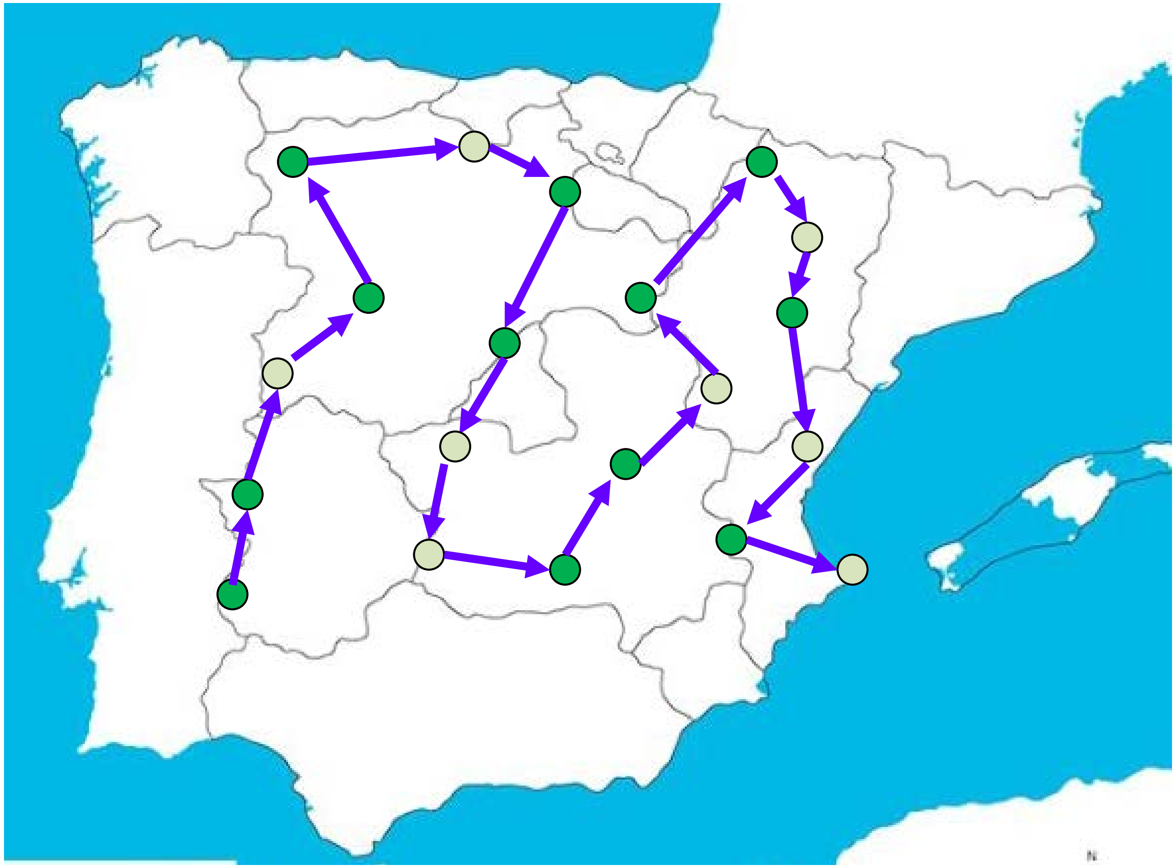


Solución de la planificación a corto plazo

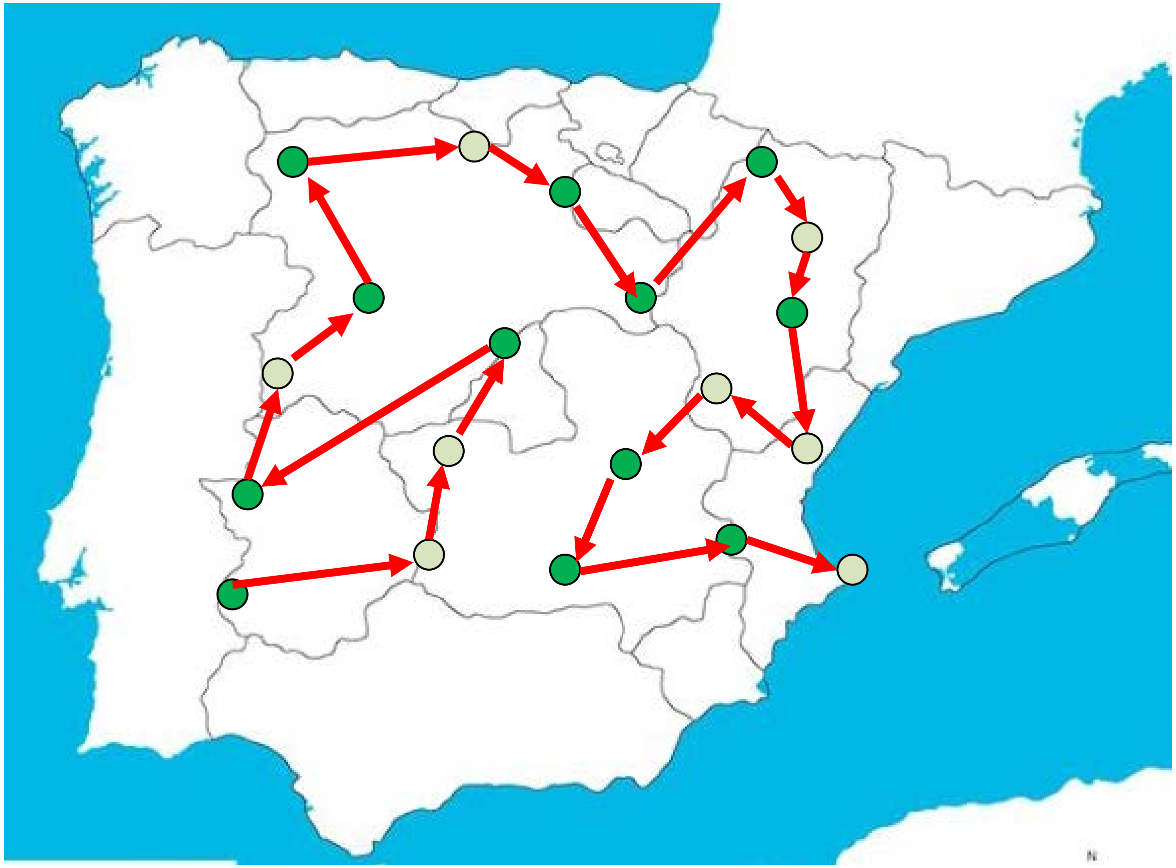




¿Un problema de ruta óptima?



¿Un problema de ruta óptima?



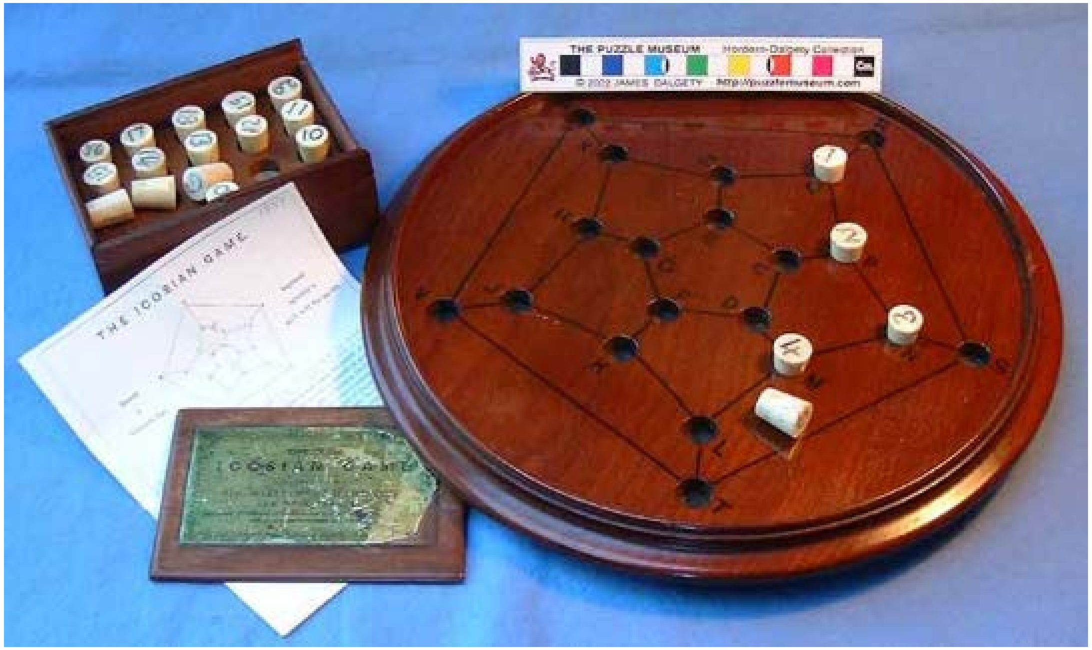
¿Un problema de ruta óptima?

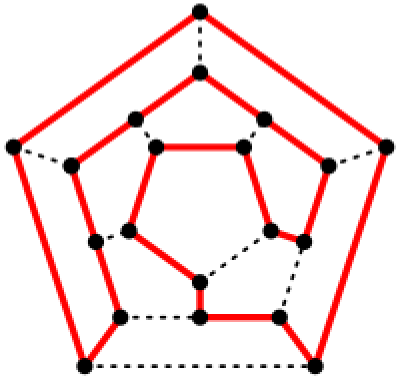
Problema del Viajante (TSP, Traveling Salesman Problem):
Dados una serie de elementos (observaciones, ciudades, tiendas, etc.) y conocidas las distancias (coste, tiempo, etc.) entre cada uno de ellos, determinar el recorrido óptimo (mínimo) que recorre todos los elementos y vuelve al elemento inicial.

Características del Problema del Viajante:
- Problema muy antiguo (1832, manual alemán para viajantes, sXIX Hamilton “Icosian Game”)
- Admite diversas formulaciones matemáticas (Grafo, Programación matemática, …)
- Computacionalmente muy complejo.

TSP, the Icosian Game

Introducido por Hamilton en 1857.





TSP, modelo de programación matemática

TSP, modelo de optimización combinatoria

Problema del Viajante:
Dados una serie de elementos

$$\{1, 2, \ldots, n\}$$

y conocidas las distancias

$$\left\{c_{i,j}\ , i, j \in \{1,2, \ldots, n\}\right\}$$

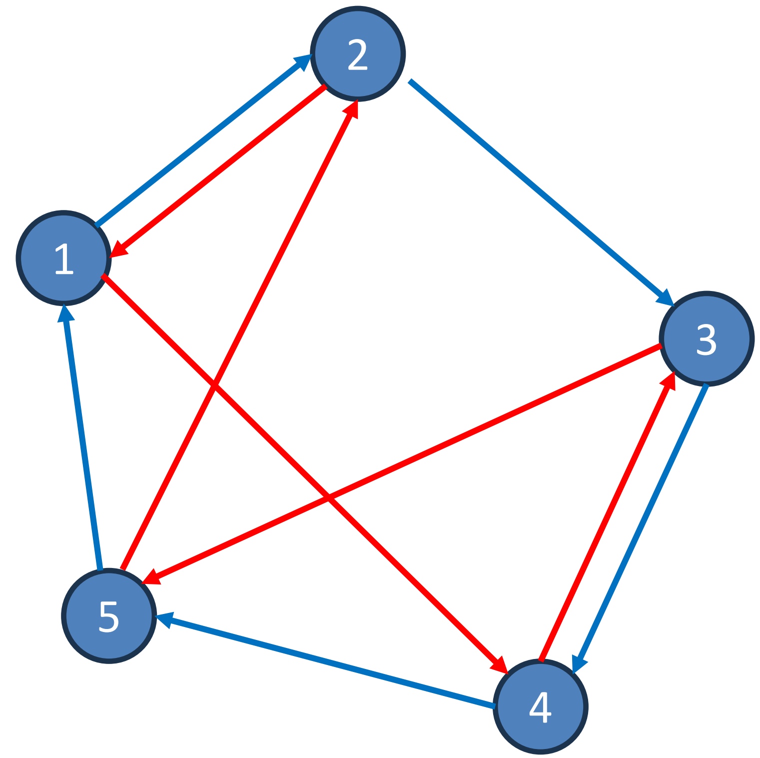
Determinar una permutación de los n elementos

$$\left(\sigma(1), \sigma(2), \ldots, \sigma(n)\right)$$

De forma que la distancia total del recorrido asociado a dicha permutación, que comienza en $\sigma(1)$, va a $\sigma(2)$, y termina en $\sigma(n)$ para volver a $\sigma(1)$:

$$C(\sigma) = \sum_{i=1}^{n-1} c_{\sigma(i),\sigma(i+1)} + c_{\sigma(n),\sigma(1)}$$

TSP, modelo de optimización combinatoria



Solución azul:

$\big(\sigma(1), \sigma(2), \sigma(3), \sigma(4), \sigma(5)\big)$ = (1,2,3,4,5)

Solución roja:

$\big(\sigma(1), \sigma(2), \sigma(3), \sigma(4), \sigma(5)\big)$ = (1,4,3,5,2)

TSP, complejidad del problema

¿Importa el tamaño?

La respuesta es afirmativa, independientemente del modelo matemático utilizado, es una característica del problema en sí.

El problema de programación matemática tiene n^2 variables binarias.
El problema de optimización combinatoria admite $(n-1)!$ Soluciones distintas:

n	6	11	13	14
$(n-1)!$	120	3628800	**479001600**	$\mathbf{6.227 \cdot 10^9}$

TSP, complejidad del problema

En el estado actual de la ciencia, no es posible encontrar un algoritmo eficiente, que resuelva en tiempo aceptable, que no crezca exponencialmente con el tamaño del problema, y de forma exacta un problema del TSP de tamaño grande o muy grande.

Si alguien encontrara o propusiera tal algoritmo, habría resuelto uno de los 7 problemas del milenio (fundación Clay Mathematics Institute de Cambridge), pues se demostraría que $P = NP$.

Complejidad algorítmica

Para el TSP ¿Cómo resolverlo si el problema es suficientemente grande?

Respuesta: Utilizar heurísticas.

En el lenguaje coloquial, una **heurística** se refiere a una técnica, método o procedimiento inteligente de realizar una tarea que no es producto de un riguroso análisis formal, sino de conocimiento experto sobre la tarea.

En Matemáticas, una heurística es un procedimiento o algoritmo que aporta muy buenas soluciones en muy poco tiempo, aunque no garantiza la optimalidad de la solución.

Metaheurística: Algoritmos genéticos

La sistematización de la construcción de heurísticas lleva al concepto de **Metaheurística**: Estrategia general que permite el diseño e implementación de algoritmos aproximados, identificando así una familia de heurísticas, caracterizada por un conjunto de parámetros.

Los **algoritmos genéticos** identifican una de estas metaheurísticas, y representa una opción válida para resolver los problemas complejos computacionalmente, como es el caso del TSP:

Inteligencia Artificial

Los **algoritmos genéticos** son procedimientos evolutivos que establecen estrategias que mejoran conjuntos de soluciones o poblaciones. El aspecto fundamental de las heurísticas evolutivas consiste en la interacción entre los miembros de la población frente a las búsquedas que se guían por la información de soluciones individuales.

Estas heurísticas toman como referencia la teoría de la evolución de Darwin y combinan conceptos de genética, selección natural y reproducción.

En este sentido, los algoritmos genéticos se pueden considerar una técnica de **Inteligencia Artificial**.

Inteligencia Artificial

La **Inteligencia Artificial** se refiere a los sistemas de computación capaces de realizar tareas complejas que históricamente sólo los seres vivos, y en particular los humanos, son capaces de realizar.

La Inteligencia Artificial es la “inteligencia de las máquinas o el software” en oposición a la inteligencia de los humanos o los animales.

Actualmente, el término AI recoge un conjunto muy amplio de procedimientos automáticos que mejoran nuestra vida: desde aplicaciones que recomiendan libros o películas, direccionamiento de mensajes comerciales particularizados, etc.

TSP. Algoritmos genéticos

Enfocando el TSP para que sea resuelto por AG:

- Caracterizar cada solución del TSP:

$$\big(\sigma(1), \sigma(2), \dots, \sigma(n)\big)$$

- Determinar la población de tamaño ***m***:

$$\Big\{\Big(\sigma^i(1), \sigma^i(2), \dots, \sigma^i(n)\Big), i = 1, \dots, m\Big\}$$

- Cada elemento de la población tiene una valoración que debe ser maximizada:

$$V^i = V\big(\sigma^i\big), i = 1, \dots, m$$

- Ejemplo, *n=7* y *m=4*:

$$\begin{array}{ll} \sigma^1 = (1,2,3,4,5,6,7) & V^1 = 20 \\ \sigma^2 = (3,2,7,5,1,6,4) & V^2 = 10 \\ \sigma^3 = (7,6,5,4,3,2,1) & V^3 = 3 \\ \sigma^4 = (2,4,7,1,5,6,3) & V^4 = 40 \end{array}$$

TSP. Algoritmos genéticos

Los algoritmos genéticos se basan en tres operadores para obtener una población diferente a la anterior, el número de generaciones se puede fijar ***G***:

1. **Selección**: se eligen aleatoriamente *m* individuos de la población con probabilidad proporcional al valor.
2. **Cruce**: con una determinada probabilidad ***prc***, cada pareja de soluciones es seleccionada para obtener otra pareja diferente que la sustituirá.
3. **Mutación**: con una determinada probabilidad ***prm***, alguna de las soluciones es alterada levemente.

TSP. Algoritmos genéticos, selección

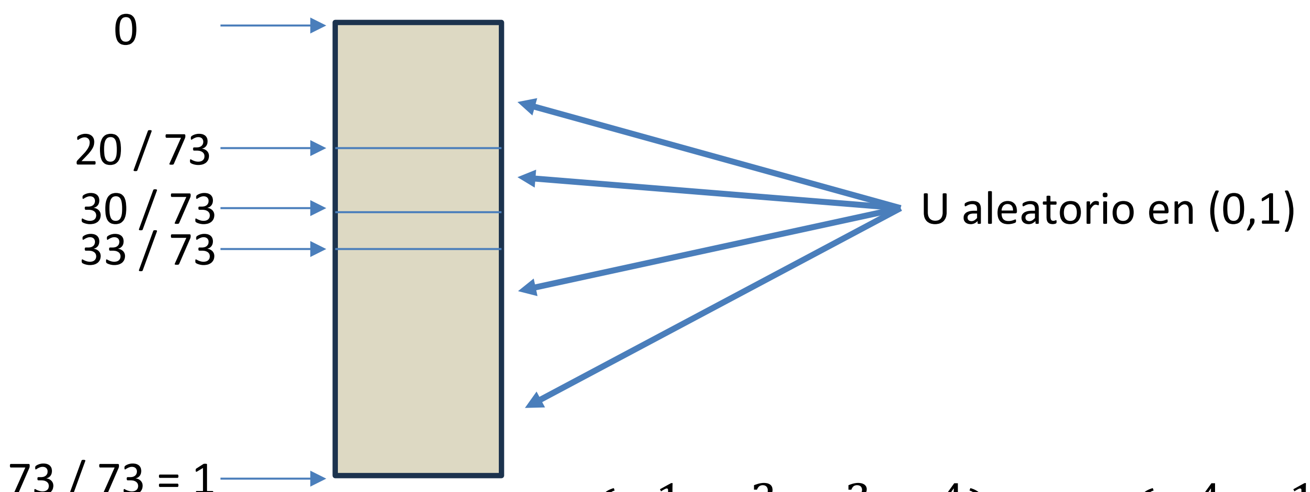
Ejemplo, *n=7* y *m=4*:

$\sigma^1 = (1,2,3,4,5,6,7) \qquad V^1 = 20$

$\sigma^2 = (3,2,7,5,1,6,4) \qquad V^2 = 10$

$\sigma^3 = (7,6,5,4,3,2,1) \qquad V^3 = 3$

$\sigma^4 = (2,4,7,1,5,6,3) \qquad V^4 = 40$



$$(\sigma^1, \sigma^2, \sigma^3, \sigma^4) \longrightarrow (\sigma^4, \sigma^1, \sigma^2, \sigma^4)$$

TSP. Algoritmos genéticos, cruce

prc = 0.5

$$\sigma^1 = (2,4,7,1,5,6,3) \qquad V^1 = 40$$
$$\sigma^2 = (1,2,3,4,5,6,7) \qquad V^2 = 20$$
$$\sigma^3 = (3,2,7,5,1,6,4) \qquad V^3 = 10$$
$$\sigma^4 = (2,4,7,1,5,6,3) \qquad V^4 = 40$$

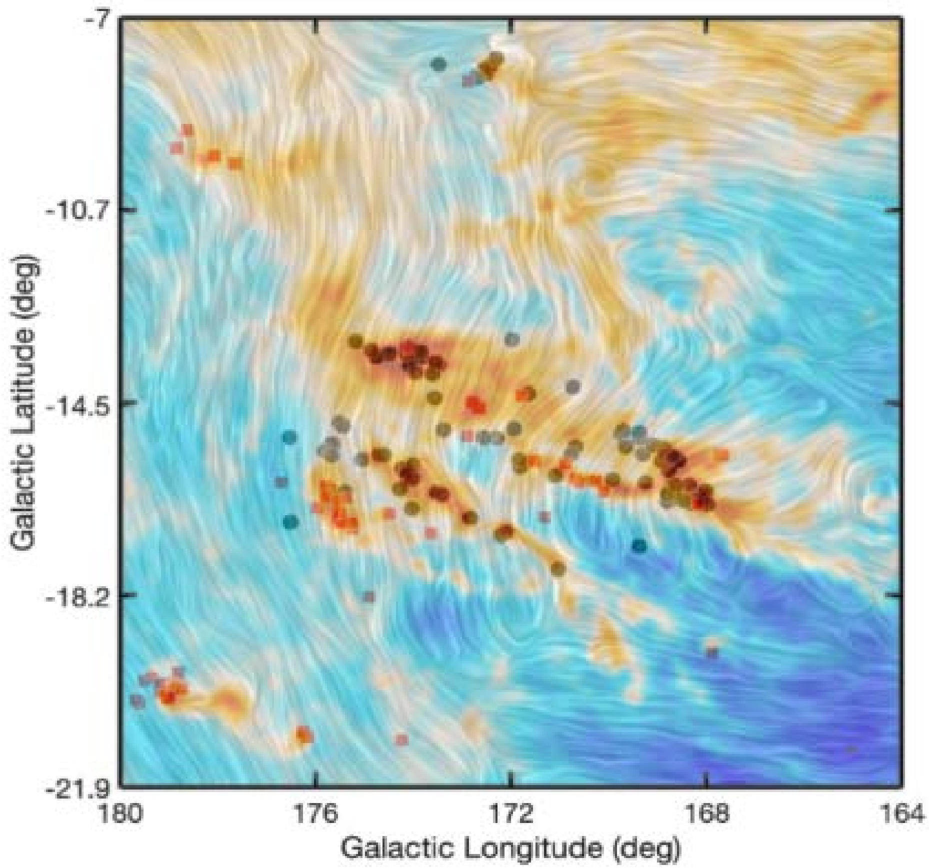
$$\sigma^1 = (2,4,7,1,5,6,3) \qquad \longrightarrow \qquad \sigma^1 = (2,4,7,5,6,3,1)$$
$$\sigma^2 = (1,2,3,4,5,6,7) \qquad \qquad \qquad \sigma^2 = (1,2,3,5,6,7,4)$$

TSP. Algoritmos genéticos, mutación

prm = 0.05

$$\sigma^1 = (2, \mathbf{4}, 7,5,6, \mathbf{3}, 1) \qquad \sigma^1 = (2, \mathbf{3}, 7,5,6, \mathbf{4}, 1)$$
$$\sigma^2 = (1,2,3,5,6,7,4) \qquad \sigma^2 = (1,2,3,5,6,7,4)$$
$$\sigma^3 = (3,2,7, \mathbf{5}, \mathbf{1}, 6,4) \quad \longrightarrow \quad \sigma^3 = (3,2,7, \mathbf{1}, \mathbf{5}, 6,4)$$
$$\sigma^4 = (2,4,7,1,5,6,3) \qquad \sigma^4 = (2,4,7,1,5,6,3)$$

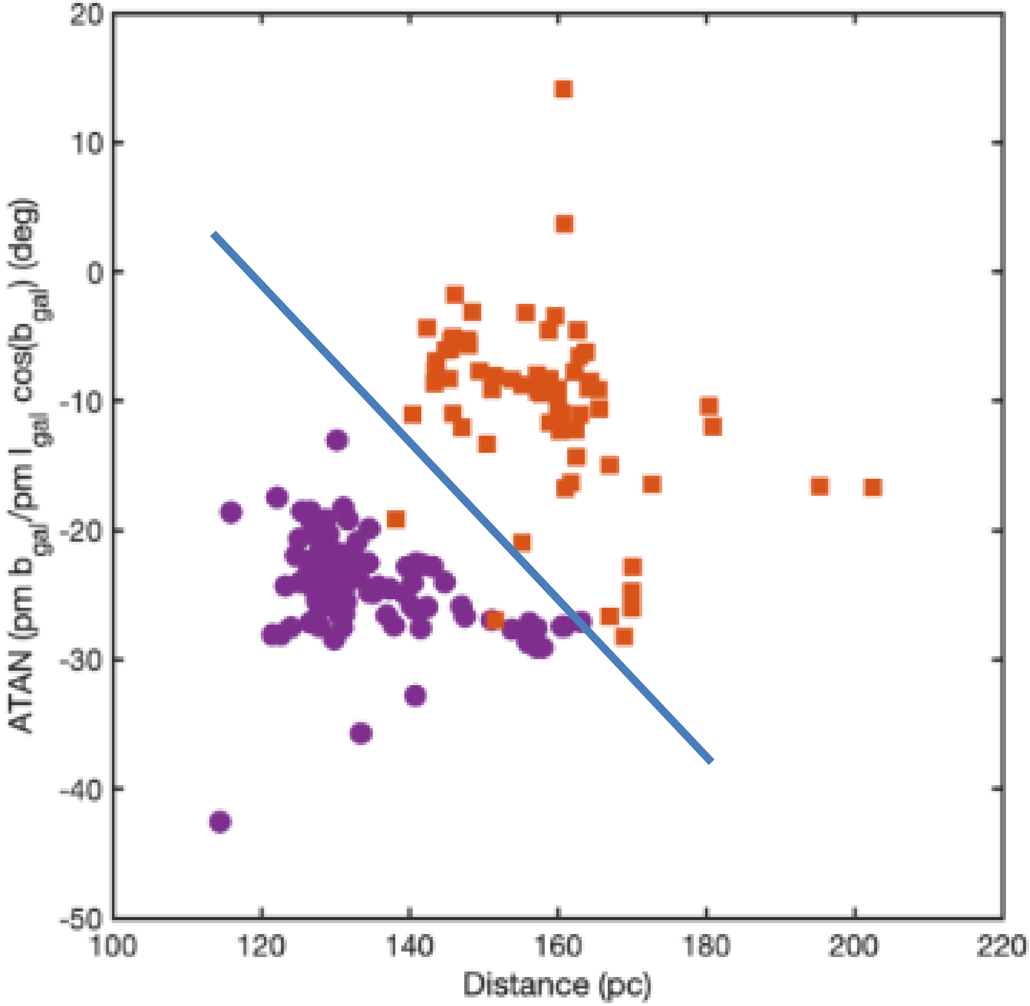
Análisis de datos astronómicos



(Declinación)

(Ascensión recta)

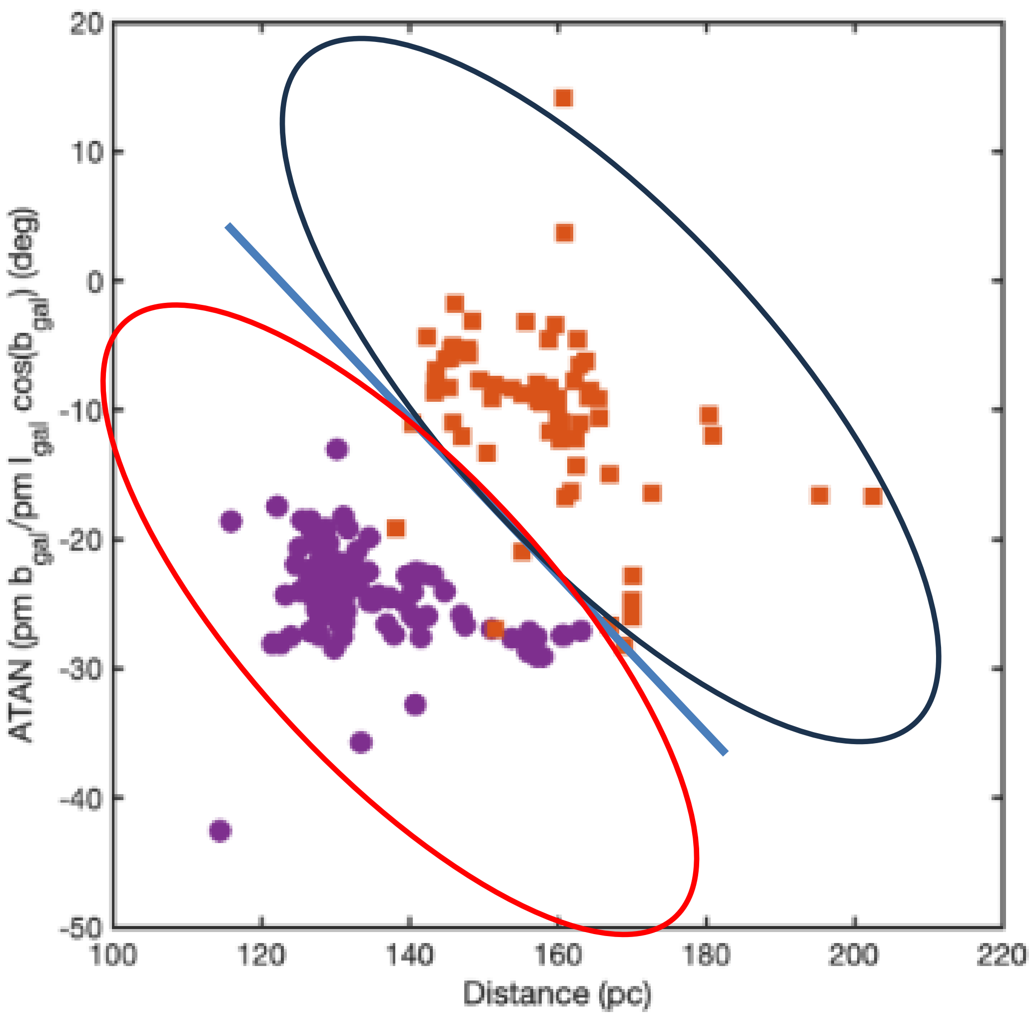
Análisis de datos astronómicos



Análisis de datos astronómicos

- ¿Se pueden distinguir los dos grupos de estrellas a partir del conocimiento de las correspondientes variables *x* e *y*? Es decir, dado un objeto astronómico del que son conocidas dichas variables, asignarle una de las dos categorías.
- Es un problema de **clasificación supervisada**. En la vida real hay muchos ejemplos: identificación de un potencial cliente como *fiable* o *no fiable*, diagnosticar un tumor como *cancerígeno* o *no cancerígeno*, etc.
- Tradicionalmente, la Estadística proponía determinados modelos (matemáticos) de comportamiento suponiendo una determinada distribución de probabilidad y resolviendo el problema como un problema de estimación estadística, por ejemplo, el Análisis discriminante.

Análisis de datos astronómicos



Análisis de datos astronómicos

- Este enfoque (clásico) ha sido sustituido por un nuevo paradigma: **el Aprendizaje automático**, (en inglés **Machine Learning**).
- **Aprender** en este contexto quiere decir **identificar patrones** complejos en los datos. La máquina (ordenador) que realmente aprende es un algoritmo que revisa los datos y es capaz de predecir comportamientos futuros.
- Se incluye el Aprendizaje automático dentro de la **Inteligencia Artificial**.
- Dentro de estos procedimientos se pueden encajar las redes neuronales artificiales, árboles de decisión, máquinas de vector soporte, redes bayesianas, etc.

Redes neuronales. El perceptrón.

- Uno de los hitos de la Inteligencia artificial fue la introducción del perceptrón en 1959 por Rosenblatt.
- Ilustra el proceso por el que el modelo “aprende” de los datos.
- Es un modelo muy simple de red neuronal artificial que propone un modelo de clasificación supervisada a partir de un conjunto de entrenamiento.
- Se basa en el condicionamiento clásico o pauloviano (Paulov 1849-1936) que explica cómo respuestas automáticas se asocian a estímulos que inicialmente no provocan ninguna respuesta o respuestas diferentes.

2.- Clasificando en dos clases. El perceptrón.

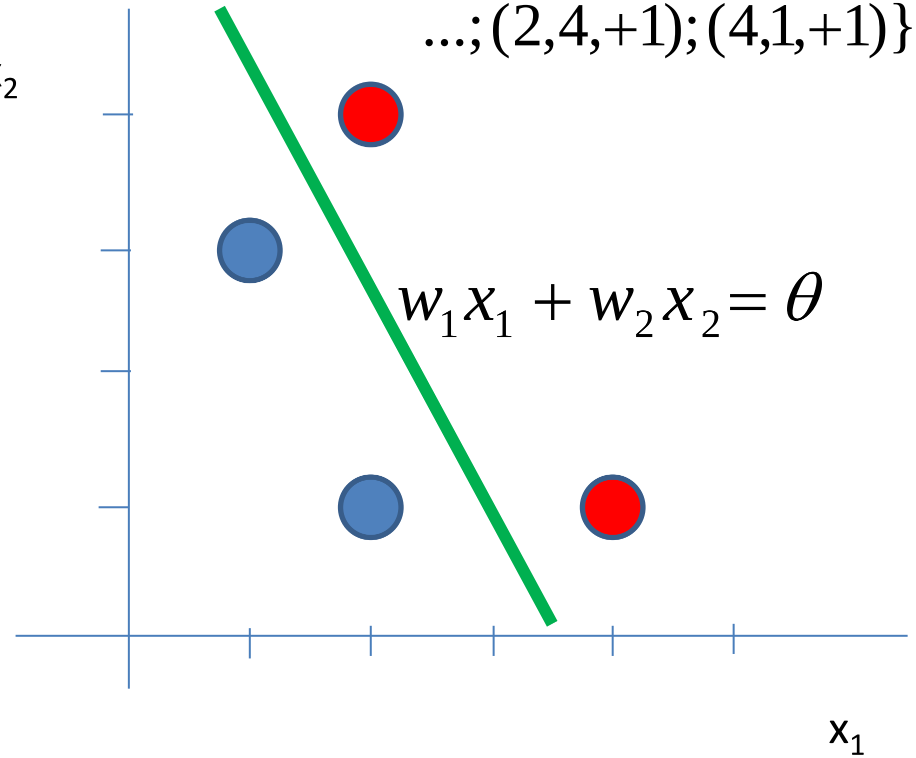
Conjunto de datos
de entrenamiento:

$$C = \{(x^i, y^i); \quad i = 1, m\}$$

$$x^i \in R^d; y^i \in \{-1, +1\}$$

$$m = 4, d = 2$$

$$C = \{(1,3,-1); (2,1,-1); ..$$
$$...; (2,4,+1); (4,1,+1)\}$$



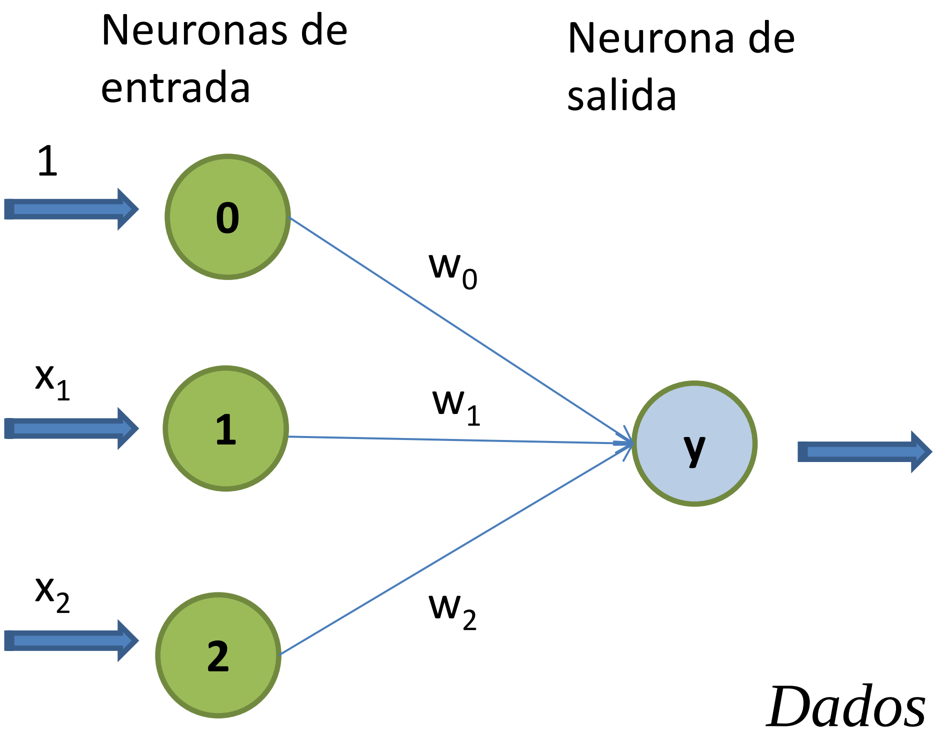
$$Objetivo:$$

$$¿w \in R^d, \theta \in R /$$

$$w^t x > \theta \qquad y = +1$$

$$w^t x < \theta \qquad y = -1?$$

2.- Clasificando en dos clases. El perceptrón.



$$y = \begin{cases} +1 & 1w_0 + x_1 w_1 + x_2 w_2 \geq 0 \\ -1 & 1w_0 + x_1 w_1 + x_2 w_2 < 0 \end{cases}$$

$$Dados \qquad X = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 2 & 4 \\ 3 & 1 & 4 & 1 \end{pmatrix}$$

$$e \qquad y = \begin{pmatrix} -1 & -1 & 1 & 1 \end{pmatrix}$$

¿Cuáles deben ser los valores de (w_0 , w_1, w_2)?

2.- Clasificando en dos clases. El perceptrón.

- A partir de un valor arbitrario w^0:
- $\alpha \in (0,1)$ es la tasa de aprendizaje

$$
\begin{aligned}
& t = 1,2,\ldots,m \\
& \quad if \quad w^{t-1} x^t \geq 0 \\
& \quad\quad \widehat{y}^t = +1 \\
& \quad else \\
& \quad\quad \widehat{y}^t = -1 \\
& \quad end \\
& \quad \delta^t = y^t - \widehat{y}^t \\
& \quad w^t = w^{t-1} + \alpha \cdot \delta^t \cdot x^t
\end{aligned}
$$

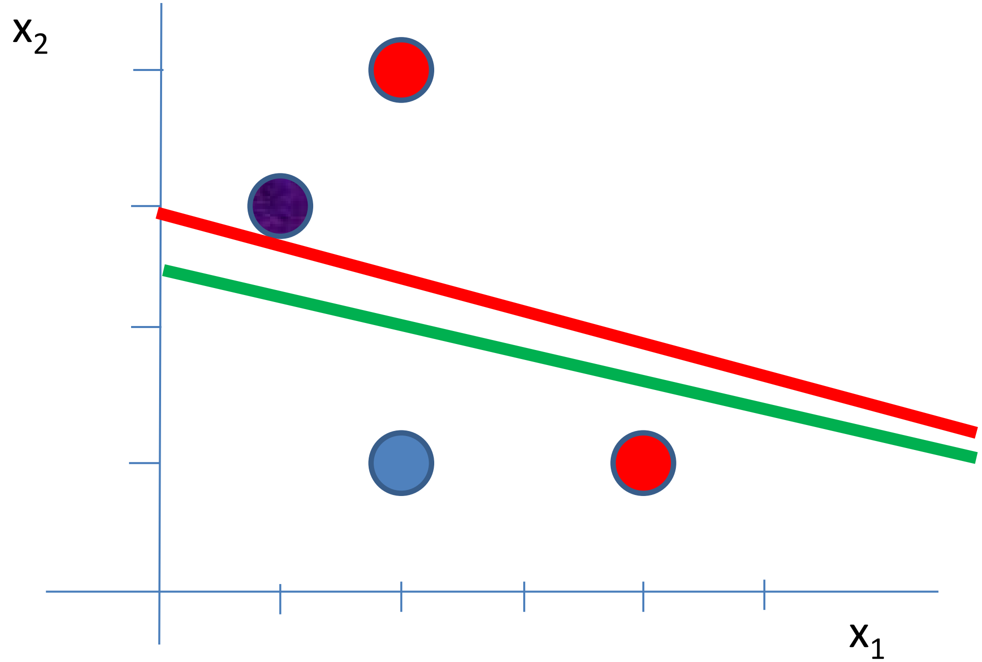
- Repetir el proceso hasta que no haya errores:

$\delta^t = 0$ para todo $t = 1,2,..,m$

2.- Clasificando en dos clases. El perceptrón.

$$w^0 = \begin{pmatrix} -10 \\ 1 \\ 4 \end{pmatrix} \qquad w^0 \cdot x^1 = (-10,1,4)\begin{pmatrix} 1 \\ 1 \\ 3 \end{pmatrix} = +3 \geq 0 \qquad \delta^1 = -1-(+1) = -2$$

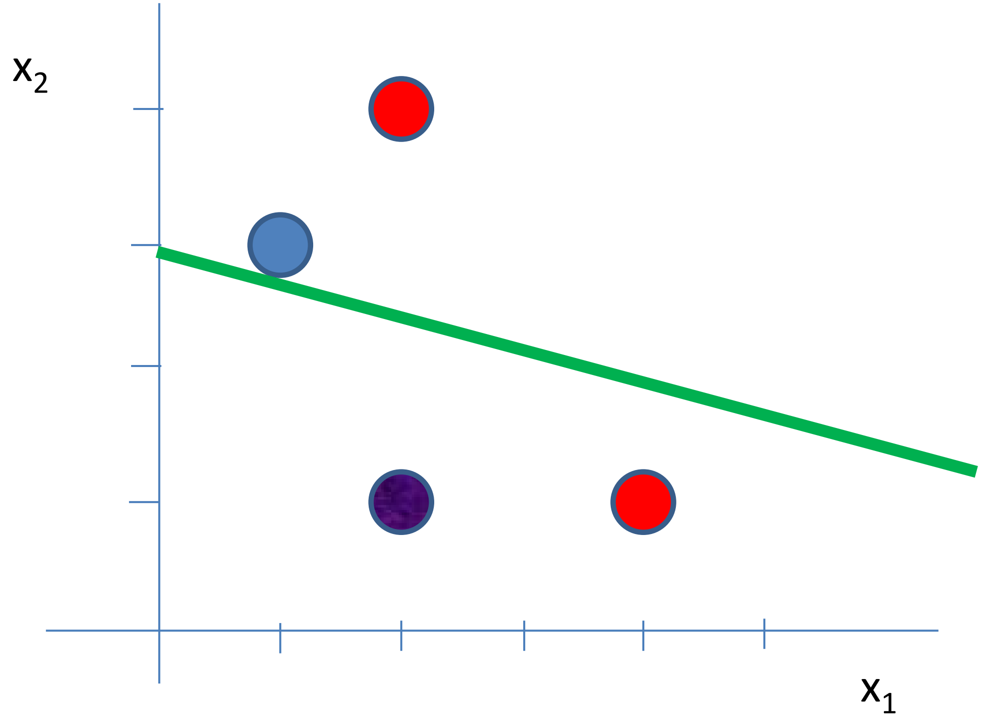
$$w^1 = w^0 + \alpha \cdot \delta^1 \cdot x^1 = \begin{pmatrix} -10 \\ 1 \\ 4 \end{pmatrix} + 0.1 \cdot (-2) \cdot \begin{pmatrix} 1 \\ 1 \\ 3 \end{pmatrix} = \begin{pmatrix} -10.2 \\ 0.8 \\ 3.4 \end{pmatrix}$$



2.- Clasificando en dos clases. El perceptrón.

$$w^1 = \begin{pmatrix} -10.2 \\ 0.8 \\ 3.4 \end{pmatrix} \qquad w^1 \cdot x^2 = (-10.2, 0.8, 3.4)\begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix} = -5.2 < 0 \qquad \delta^2 = 0$$

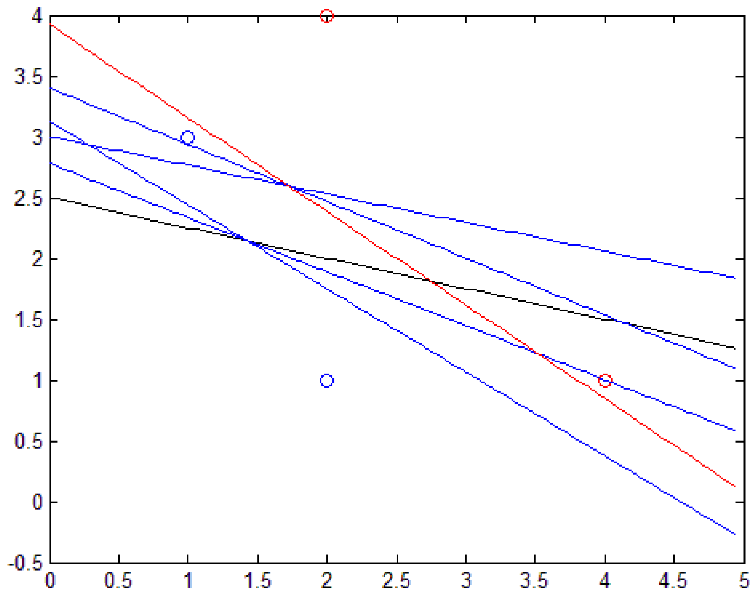
$$w^2 = w^1$$



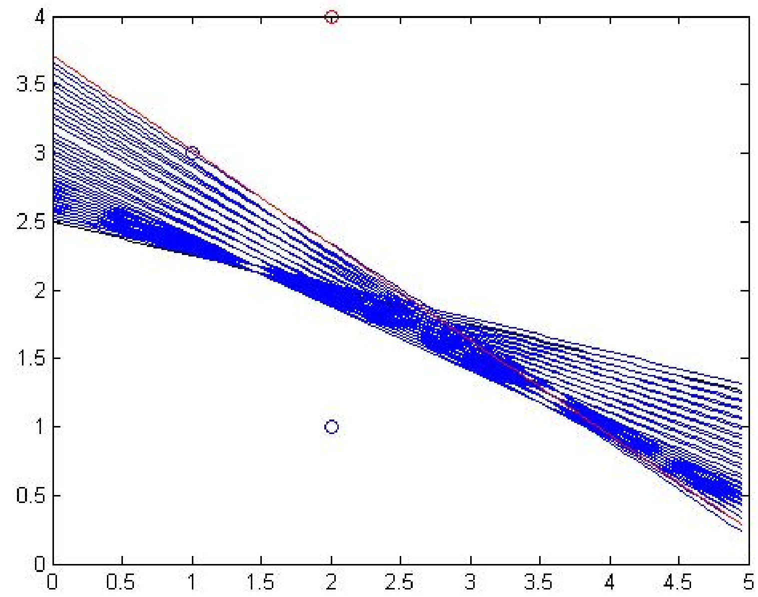
2.- Clasificando en dos clases. El perceptrón.

- Evolución del hiperplano:

$\alpha = 0.1$



$\alpha = 0.01$

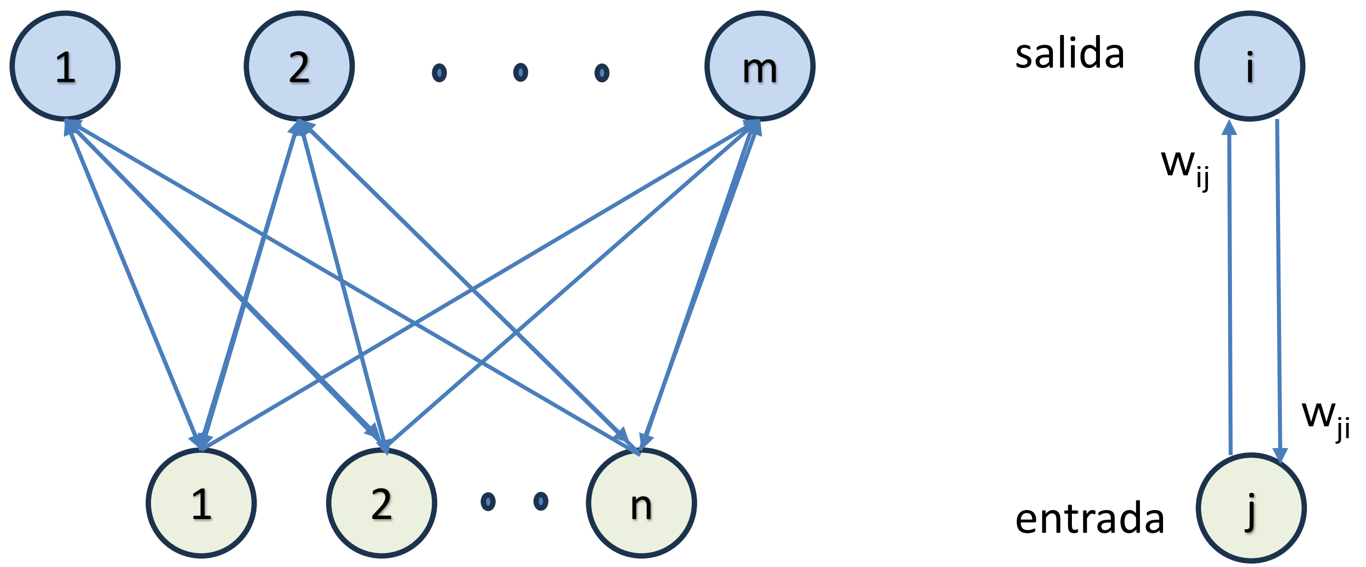


7.- Redes neuronales. Modelo de Hopfield.

- El premio Nobel de Física 2024 ha ido a parar a John Hopfield (91) y Geoffrey Hinton (76).
- “… *han sentado las bases para que las máquinas aprendan*”,
- “…*descubrimientos e invenciones que han sido fundamentales para el aprendizaje automático con redes neuronales artificiales…*"

Modelo de Hopfield. Memoria asociativa bidireccional

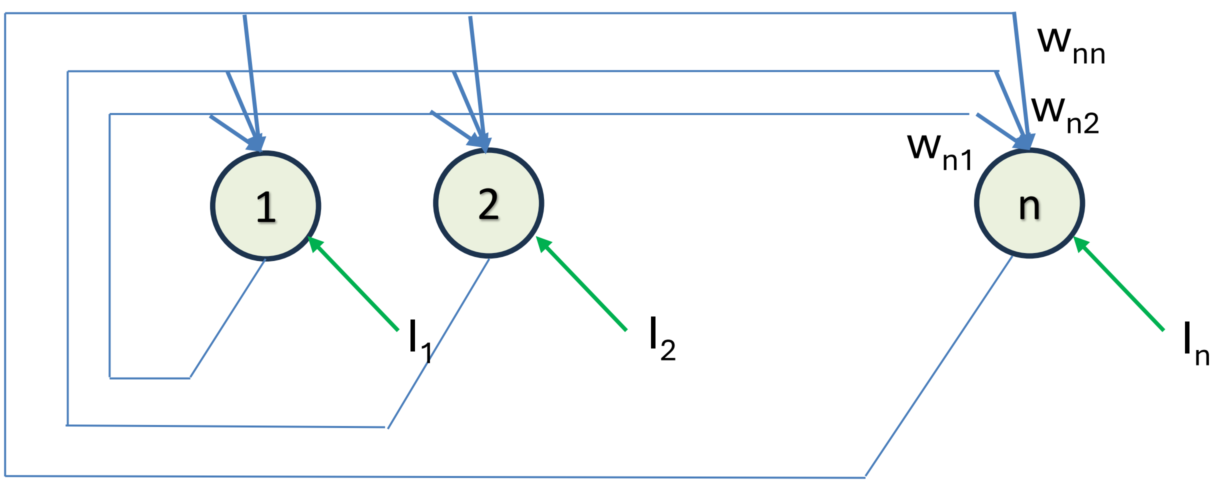
- BAM. Reproduce la capacidad del cerebro de asociar y relacionar datos.



Modelo de Hopfield. Memoria autoasociativa

- Una capa única, inicialización:

$$x(0) = \left(x_1(0), x_2(0), \dots, x_n(0)\right) \qquad x_i(0) \in \{0,1\} \quad i=1,\dots,n$$



- Evolución:

$$Neta_i = \sum_j w_{ij} x_j(t) + I_i \quad i=1,\dots,n$$

$$x_i(t+1) = (1, x_i(t), 0) \quad si \quad Neta_i\ (>0, =0, <0) \quad t=0,1,\dots$$

Modelo de Hopfield 1.

- En cada instante t, el estado del sistema viene determinado por:

$$x(t) = \left(x_1(t), x_2(t), \dots, x_n(t)\right) \qquad\qquad x_i(0) \in \{0,1\}$$

- El modelo de Hopfield exige:

$$w_{ij} = w_{ji} \quad \forall\, i, j = 1, \dots, n \quad w_{ii} = 0 \;\forall\, i = 1, \dots, n$$

- La entrada I_i de cada neurona i se puede interpretar como un umbral de activación $T_i = -I_i$.
- El aprendizaje de este modelo difiere radicalmente del perceptrón:
  - La matriz de pesos W es fija
  - Con dicha matriz el estado del sistema (x(t)) evoluciona hasta conseguir un estado estable. Esta propiedad se interpreta a través de la existencia de una función de energía, que es una función de Lyapunov:

$$H(x) = -\frac{1}{2}\sum_i \sum_j w_{ij}\, x_i\, x_j + \sum_i T_i\, x_i$$

Modelo de Hopfield 2.

- No es complicado demostrar (a partir de las propiedades de w_{ij}) que esta función de energía H(x) siempre decrece, y se llegará a un mínimo local.
- Si se consigue que el mínimo local identifique la solución del problema, el modelo de Hopfield podrá ser útil. Es fundamental definir adecuadamente la matriz de pesos de las conexiones neuronales.
- El problema será el de recuperar información parcialmente desconocida o dañada en un conjunto de datos, por ejemplo, una base de datos con nombres y teléfonos asociados:

  (nombre_1, teléfono_1; …; nombre_p,teléfono_p)

El objetivo es asociar un teléfono a un nombre (o un nombre a un teléfono) suponiendo que la información de partida puede ser parcial o estar dañada.

Modelo de Hopfield 3.

- El procedimiento comienza con codificar los *P* patrones de entrada (código binario para nombres y teléfonos):

$$d^p = \left(d_1^p, d_2^p, \dots, d_n^p\right) \; p \in \{1, \dots, P\} \;\; d_i^p \in \{0,1\}$$

- Asignación de pesos de conexiones:

$$w_{ij} = \frac{1}{n}\sum_{p=1}^{P} d_i^p d_j^p \qquad i,j \in \{1,\dots,n\}, i \neq j$$

$$w_{ii} = 0 \qquad i \in \{1,\dots,n\}$$

- Umbral de las neuronas:

$$T_i = 0 \qquad i \in \{1,\dots,n\}$$

- Inicialización:

$$x(0) = \left(x_1(0), x_2(0), \dots, x_n(0)\right)$$

- Evolución:

$$Neta_i(0) = \sum_j w_{ij} x_j(0) \quad i=1,\dots,n$$

$$x_i(t+1) = (1, x_i(t), 0) \quad si \quad Neta_i\ (>0, =0, <0) \quad t=0,1,\dots$$

- Finalización:

$$x^* = x(t^*) \;\; si \;\; x(t^*) = x(t^* - 1)$$

¡Muchas gracias!