Práctica 5: Sucesiones de Números Reales

Tratamos aquí diversos aspectos sobre sucesiones de números reales y sus aplicaciones mediante las utilidades de *Mathematica*: representación gráfica, cálculo de límites y resolución de ecuaciones en diferencias. Destacan, por su utilidad, las sucesiones recurrentes que permiten modelar problemas definidos mediante procesos inductivos.

1. Generalidades. Representación gráfica

En ocasiones, obtener algunos elementos de una sucesión $\{a_n\}$ es suficiente para intuir su comportamiento y disponer de una representación gráfica puede ayudarnos a descubrir la monotonía, acotación, convergencia, etc. Analizamos, a continuación, cómo realizar estas acciones con *Mathematica*.

Dado que una sucesión de números reales $\{a_n\}_{n=1}^{\infty}$ es una función real con dominio el conjunto de los números naturales $\mathbb{N}$, podemos definir una sucesión del mismo modo que una función, denotando su variable por n. Por ejemplo, la sucesión de término general $a_n = n^2$ puede verse como la función que asigna, a cada número natural n, el valor $a(n) = n^2$ (su cuadrado). Por tanto, podemos definirla con *Mathematica* mediante:

```
In[1]:= a[n_] := n^2
```

y calcular su valor, en $n = 10$, por ejemplo,

```
In[2]:= a[10]
Out[2]= 100
```

Los términos sucesivos $a_1, a_2, a_3, \ldots$ son, respectivamente, a[1], a[2], a[3], ... y podríamos obtenerlos de la misma manera. También es posible hallar una lista con unos cuantos términos de la sucesión utilizando el comando **Table**, cuya sintaxis es:

Table[expresión, {n, a, b, c}]

que devuelve una lista con los valores resultantes al sustituir la variable n en expresión, desde n = a hasta n = b, con un tamaño de paso igual a c. Si se omite el parámetro c, n varía de 1 en 1.

Ejemplo. Calcula los cinco primeros términos de la sucesión de término general $a_n = n^2$:

```
In[3]:= Table[a[n], {n, 1, 5}]
        tabla
Out[3]= {1, 4, 9, 16, 25}
```

Ejemplo. Halla los términos de la sucesión anterior que ocupan las posiciones pares entre a_6 y a_{11}:

```
In[4]:= Table[a[n], {n, 6, 11, 2}]
        tabla
Out[4]= {36, 64, 100}
```

Para generar un gráfico de la sucesión, necesitamos representar el conjunto de puntos de la forma $[n, a(n)]$. Para ello, podemos utilizar la instrucción

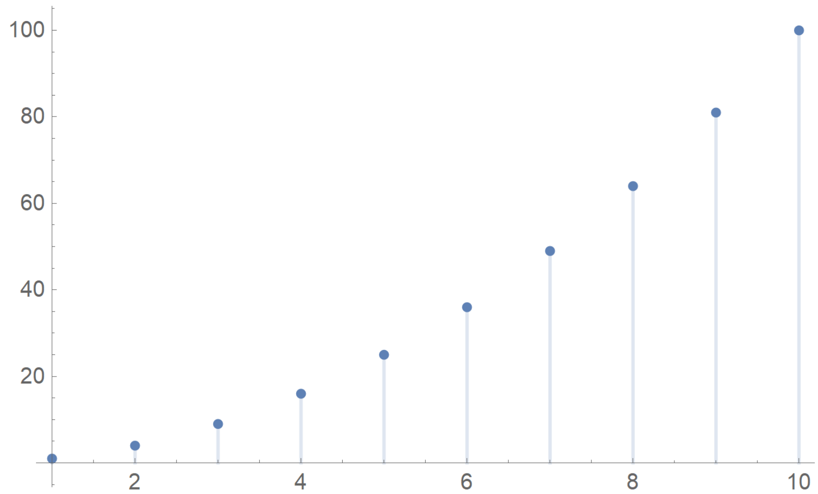
DiscretePlot[expresión, {n, a, b, c}]

cuyos parámetros son análogos a los descritos para la función **Table**. Con este comando se mostrará la gráfica de la sucesión cuyo término general se introduce en expresión, teniendo en cuenta ahora que, a diferencia del comando **Plot**, solo se representan las imágenes correspondientes a los valores naturales de la variable.

Ejemplo. Representa la gráfica de los 10 primeros términos de la sucesion $a_n = n^2$:

```
In[5]:= DiscretePlot[a[n], {n, 1, 10}]
        representación discreta
```

Out[5]=



Alternativamente, podemos usar la función **ListPlot** para generar la gráfica de los términos de la sucesión. Aunque admite varias sintaxis, la más útil, en este caso, es la que dibuja los elementos de la tabla, esto es,

ListPlot[**Table**[expresión, {n, a, b, c}]]

Ejemplo. Utiliza la gráfica de los 50 primeros términos de la sucesion

$$a_n = \frac{n+1}{n^2+1}$$

para analizar su comportamiento.

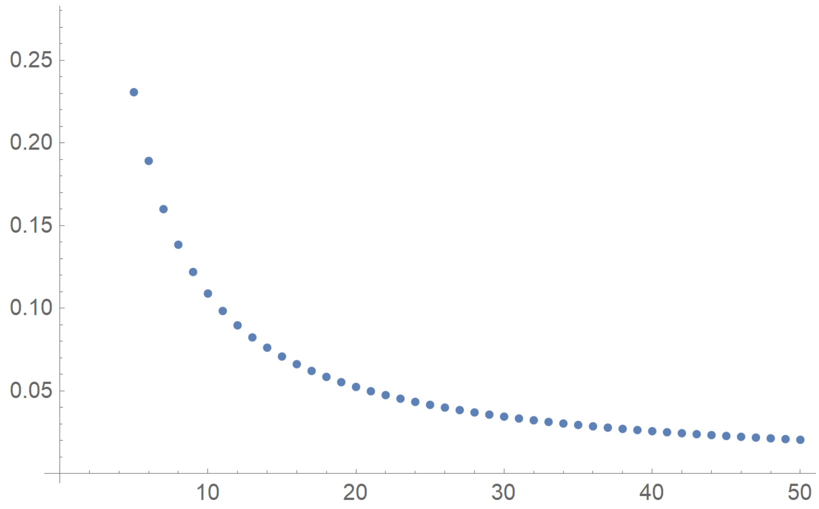
Definimos, en primer lugar, la sucesión con *Mathematica*:

```
In[6]:= a[n_] := (n + 1) / (n^2 + 1)
```

cuya representación gráfica será

```
In[7]:= ListPlot[Table[a[n], {n, 1, 50}]]
        representa·· tabla
```

Out[7]=



con lo que podemos intuir que se trata de una sucesión decreciente y acotada inferiormente (por 0). Aunque la gráfica, por sí misma, no prueba nada.

2. Criterios para el cálculo de límites

Como norma general, podemos usar el comando **Limit**, visto en la práctica segunda, para determinar el límite[1] de sucesiones definidas en forma explícita. En este caso, su sintaxis será

Limit[sucesion, n —> Infinity]

Mathematica aplica correctamente el proceso de cálculo de límites en muchas ocasiones. Así, sin más que usar el comando **Limit** a las sucesiones

$$\frac{6n^2+1}{3-2n^2} \quad , \quad \left(\frac{2n+1}{3n-\sqrt{n}}\right)^{n+2}, \quad \sqrt{n+2}\left(\sqrt{n+1}-\sqrt{n-1}\right)$$

[1]Cuando n tiende a $+\infty$, obviamente

obtendremos el valor correcto de los límites. Respectivamente -3, 0, 1, como puedes comprobar.

El programa también reconoce indeterminaciones del tipo 1^∞ y aplica la fórmula de Euler para resolverlas.

Ejemplo. Calcula el límite de la sucesión (del tipo 1^∞):

$$\left(\frac{n+1}{n}\right)^{\dfrac{\sqrt{n}}{\sqrt{n+1}-\sqrt{n}}}$$

En efecto, para obtener su límite (e^2) basta aplicar el comando

```
In[8]:= Limit[((n + 1) / n) ^ (Sqrt[n] / (Sqrt[n + 1] - Sqrt[n])), n → Infinity]
        límite                 raíz cuadr⋯  raíz cuadrada  raíz cuadrada   infinito
Out[8]= ⅇ^2
```

3. Sucesiones recurrentes

En Informática, con frecuencia, se presentan sucesiones definidas por un proceso inductivo. En estos casos, la expresión de su término general viene dada por una fórmula recurrente. Para definir con *Mathematica* una sucesión recurrente podemos usar la cláusula **If**, cuya sintaxis es:

If[condición, v, f]

donde condición es la cláusula a evaluar. Si esta es cierta, la función **If** devuelve v y, en caso de que sea falsa, el resultado será f.

Ejemplo. Halla el término a_{10} de la sucesión recurrente definida por:

$$\begin{cases} a_1 & = & \sqrt{2} \\ a_{n+1} & = & \sqrt{2a_n}\ , \quad n \geq 1 \end{cases}$$

Observa que en la expresión anterior el primer término vale $\sqrt{2}$ y el resto se calcula a partir de la fórmula recurrente. Sus términos son:

$$a_1 = \sqrt{2} \quad , \quad a_2 = \sqrt{2a_1} = \sqrt{2\sqrt{2}} \quad , \quad a_3 = \sqrt{2a_2} = \sqrt{2\sqrt{2\sqrt{2}}} \quad , \ldots$$

Así, utilizando la cláusula **If**, deberemos escribir

```
In[9]:= a[n_] := If[n == 1, Sqrt[2], Sqrt[2 a[n - 1]]]
                 si          raíz cuad⋯ raíz cuadrada
```

En adelante, podemos usar esta expresión para calcular uno o más términos. Así, hallamos el término a_{10} escribiendo (y aproximando, después):

In[10]:=
```
a[10]
```

Out[10]=

$2^{1023/1024}$

In[11]:=
```
N[%]
```
valor numérico

Out[11]=

1.99865

o bien, generamos una tabla de valores, mediante

In[12]:=
```
Table[a[n], {n, 1, 5}]
```
tabla

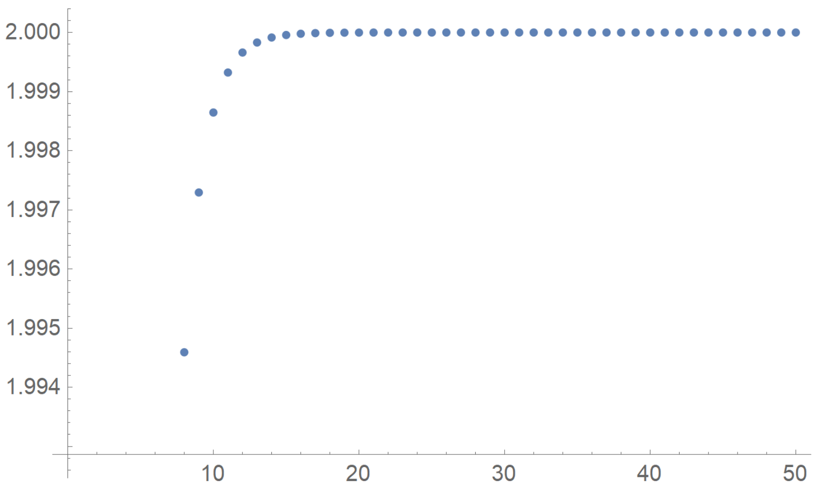
Out[12]=

$\{\sqrt{2}, 2^{3/4}, 2^{7/8}, 2^{15/16}, 2^{31/32}\}$

Para representarla gráficamente procederemos como en la forma explícita:

In[13]:=
```
ListPlot[Table[a[n], {n, 1, 50}]]
```
representa·· tabla

Out[13]=



Si intentas hallar el límite de la sucesión recurrente mediante el comando **Limit**, no obtendrás una respuesta válida. Esto sucede porque el programa no es capaz de calcular límites de sucesiones definidas mediante el comando **If**.

Para obtener el valor del límite, supuesto que $\alpha = \lim_n a_n$ existe, aplicamos límites en la fórmula de recurrencia. Tendremos así que α debe de satisfacer la ecuación

$$\alpha = \sqrt{2\alpha}$$

y podemos resolver ésta usando el comando **Solve**. *Mathematica* te devolverá

$$\alpha = 0 \ , \ \alpha = 2$$

de donde podemos concluir que el valor del límite es 2; puesto que no puede ser 0.

Ejemplo. Usa la cláusula **If** para definir la sucesión de Fibonacci:

$$\begin{cases} a_1 & = & 1 \\ a_2 & = & 1 \\ a_n & = & a_{n-2} + a_{n-1} \ , \quad n \geq 3 \end{cases}$$

En este caso usaremos una cláusula **If** anidada. Así, podemos definirla en la forma

```
In[14]:= a[n_] := If[n == 1, 1, If[n == 2, 1, a[n - 1] + a[n - 2]]]
                  si              si
```

Con la definición anterior, calcular términos, una tabla o representar su gráfica es inmediato. Así, podemos intuir que la sucesión de Fibonacci tiende a $+\infty$; sin embargo, no podemos determinar su orden de magnitud, un concepto muy útil para estudiar la eficiencia de los algoritmos. Para ello, resulta conveniente escribir la recurrencia de forma explícita. En general, *Mathematica* dispone de un comando que permite resolver recurrencias de determinados tipos, es decir, hallar la expresión explícita de la sucesión que define la recurrencia. Se trata del comando **RSolve**, cuya sintaxis es

RSolve[{ecuación, condiciones iniciales}, a[n], n]

donde ecuación representa la fórmula de la recurrencia, condiciones iniciales son los valores de los r primeros términos de la sucesión (separados por comas), siendo r el orden de la recurrencia, a[n] es el nombre de la sucesión que define la recurrencia y cuya expresión explícita devuelve este comando, y n es la variable de la sucesión. En caso de querer hallar la solución explícita de una recurrencia sin especificar condiciones iniciales, podemos escribir

RSolve[ecuación, a[n], n]

y *Mathematica* devolverá una expresión explícita para a_n con tantas constantes arbitrarias como el orden de la recurrencia.

Para evitar errores y, dado que solemos denotar por a[n] a las sucesiones con las que trabajamos, antes de aplicar el comando **RSolve**, conviene borrar asignaciones anteriores para a[n]. Para ello, usaremos la instrucción **Clear**[a].

Ejemplo. Resuelve la sucesión recurrente definida por:

$$\begin{cases} a_1 & = & 2 \\ a_{n+1} & = & a_n + \frac{2n+3}{3} \ , \quad n \geq 1 \end{cases}$$

Observa que se trata de una ecuación en diferencias lineal de *primer orden*, cuya expresión general es de la forma

$$a_{n+1} = s_n \cdot a_n + t_n$$

con s_n y t_n sucesiones conocidas. Como se conoce una condición inicial ($a_1 = 2$) existe una solución única que, además, podría resolverse para a_n de forma directa.

Para hallar la solución de la ecuación en diferencias anterior, basta escribir:

```
In[15]:= Clear[a]
         borra

In[16]:= RSolve[{a[n + 1] == a[n] + (2 n + 3) / 3, a[1] == 2}, a[n], n]
         resuelve recurrencia

Out[16]=
```

$$\left\{\left\{a[n] \to \frac{1}{3}\left(3 + 2n + n^2\right)\right\}\right\}$$

El término general de la sucesión es, por lo tanto, $a_n = \frac{n^2+2n+3}{3}$ y, por tanto, $a_n \approx n^2$.

De manera análoga, mediante el comando **RSolve** con los argumentos apropiados, se puede hallar la solución de una ecuación en diferencias lineal de *segundo orden con coeficientes constantes* que, en su forma general, se expresa como:

$$a_{n+2} + p \cdot a_{n+1} + q \cdot a_n = t_n$$

sujeta a dos condiciones iniciales conocidas. Si estas se omiten, la solución incluirá dos constantes arbitrarias, y si solo se da una condición inicial, la solución dependerá de una constante arbitraria.

Ejemplo. Encuentra la solución al problema de valores iniciales

$$\begin{cases} a_{n+2} - 5 \cdot a_{n+1} + 6 \cdot a_n = n^2 \\ a_1 = 3, \;\; a_2 = 5 \end{cases}$$

Para hallar la solución, introducimos las siguientes expresiones:

```
In[17]:= Clear[a]
         borra

In[18]:= RSolve[{a[n + 2] - 5 a[n + 1] + 6 a[n] == n^2, a[1] == 3,  a[2] == 5}, a[n], n]
         resuelve recurrencia

Out[18]=
```

$$\left\{\left\{a[n] \to \frac{1}{6}\left(15 - 3 \times 2^{1+n} + 3^n + 9n + 3n^2\right)\right\}\right\}$$

Observa, simplificando y puesto que conoces la forma de las soluciones, que la solución puede expresarse como suma de una combinación lineal de dos soluciones de la ecuación homogénea y una solución particular de la completa

$$a_n^1 = 3^n \quad , \quad a_n^2 = 2^n \quad ; \quad u_n = \frac{n^2 + 3n + 5}{2}.$$

Puedes visualizar mejor este resultado si hallas la solución general de la recurrencia, sin tener en cuenta las condiciones iniciales. En efecto, sin condiciones iniciales la solución será:

```
In[19]:= RSolve[a[n + 2] - 5 a[n + 1] + 6 a[n] == n^2, a[n], n]
         resuelve recurrencia
```

Out[19]=

$$\left\{\left\{a[n] \to \frac{1}{2}\left(5 + 3\,n + n^2\right) + 2^n\,c_1 + 3^n\,c_2\right\}\right\}$$

donde las constantes toman valores concretos al añadir las condiciones iniciales. En el ejemplo, $c_1 = -1$ y $c_2 = 1/6$.

Aunque no las hemos visto en teoría, hay que señalar que *Mathematica* es capaz de resolver, en algunos casos, recurrencias lineales de segundo orden de coeficientes variables y algunas recurrencias no lineales, como puedes observar con los siguientes ejemplos.

Ejemplo. Resuelve la recurrencia lineal de segundo orden, con coeficientes variables:

$$(n+4)a_{n+2} - a_{n+1} - (n+1)a_n = 0$$

En este caso se tendrá:

```
In[20]:= RSolve[(n + 4) a[n + 2] - a[n + 1] - (n + 1) a[n] == 0, a[n], n]
         resuelve recurrencia
```

Out[20]=

$$\left\{\left\{a[n] \to \frac{(3 + 2\,n)\,c_1}{3\,(1 + n)\,(2 + n)} - \frac{\left(-3 + (-1)^{1+n} - 2\,n\right)\,c_2}{4\,(1 + n)\,(2 + n)}\right\}\right\}$$

Ejemplo. Resuelve la recurrencia no lineal:

$$a_n = a_{n+1} \cdot a_{n-1}$$

Para ello, escribiremos

```
In[21]:= RSolve[a[n] == a[n + 1] × a[n - 1], a[n], n]
         resuelve recurrencia
```

Out[21]=

$$\left\{\left\{a[n] \to e^{\left(\frac{1}{2}+\frac{i\sqrt{3}}{2}\right)^n c_1+\left(\frac{1}{2}-\frac{i\sqrt{3}}{2}\right)^n c_2}\right\}\right\}$$

Finalmente, introducimos una instrucción que genera una lista de valores de una recurrencia, sin necesidad de hallar su expresión explícita previamente. Su sintaxis es:

RecurrenceTable[{ecuación, condiciones iniciales}, a[n], {n, a, b, c}]

La orden **RecurrenceTable** funciona igual que la instrucción **Table** para sucesiones definidas de manera explícita, es decir, devuelve una lista con los términos de la recurrencia introducida en ecuación, sujeta a las condiciones iniciales, desde el término que ocupa la posición n = a hasta n = b, con un tamaño de paso igual a c. Si se omite el parámetro c, n varía de 1 en 1.

Ejemplo. Halla los diez primeros términos de la sucesión recurrente dada por:

$$\begin{cases} a_1 & = & 7 \\ a_{n+1} & = & 3a_n \end{cases}$$

Para ello, basta escribir:

```
In[22]:= RecurrenceTable[{a[n + 1] == 3 a[n], a[1] == 7}, a[n], {n, 1, 10}]
         tabla de recurrencia

Out[22]=
         {7, 21, 63, 189, 567, 1701, 5103, 15309, 45927, 137781}
```

Tal y como hemos hecho con las tablas de sucesiones explícitas, podemos usar el comando **ListPlot** para representar gráficamente los términos de la sucesión.