

## Introducción

En este número os vamos a hablar de polinomios: polinomios de una variable con coeficientes reales. Sea el polinomio  $p(x)$  de grado  $n$  dado como sigue en la ecuación (0.1),

$$(0.1) \quad P(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0,$$

donde  $n \in \mathbb{N}$ , y  $a_i \in \mathbb{R}$  con  $i \in \{0, 1, 2, \dots, n\}$ . El término  $a_n$  debe ser distinto de cero y se denomina coeficiente director de  $P(x)$  y  $n$  el grado del polinomio.

En Octave los polinomios se representan matricialmente como un vector fila. De este modo el polinomio dado en la ecuación (0.1) se representa en Octave con el vector fila siguiente:  $[a_n \ a_{n-1} \ \dots \ a_1 \ a_0]$ . Por ejemplo el polinomio  $3x^3 + 2x + 1$  es representado en Octave como el vector fila  $[3 \ 0 \ 2 \ 1]$ . Veamos el gráfico de algunos polinomios en la Figura 1.

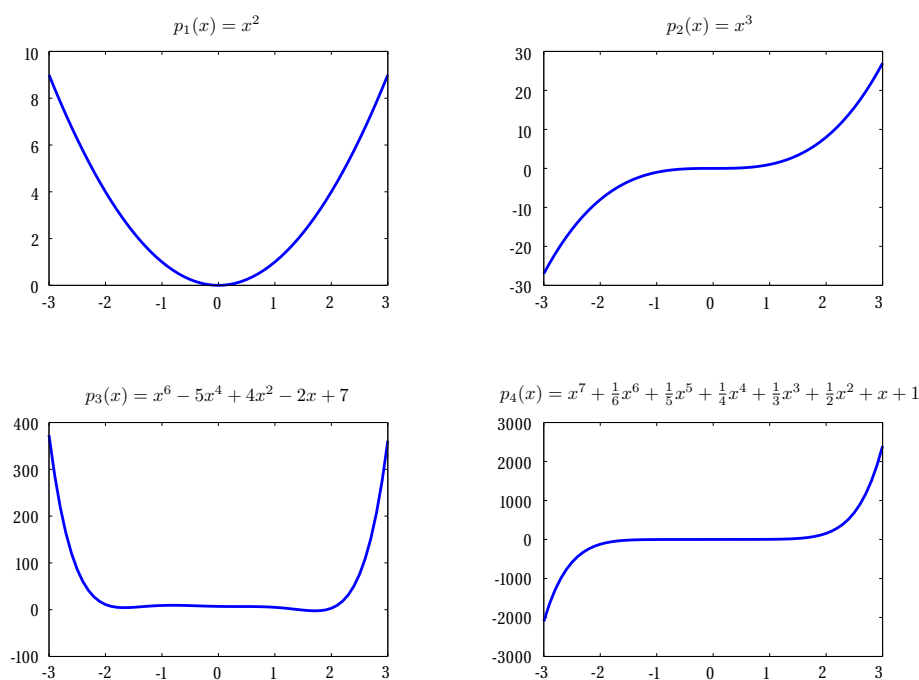


Figura 1: Ejemplos de polinomios de una variable y coeficientes reales

La importancia de los polinomios es altísima. ¿Te has preguntado alguna vez cómo calcula el  $\sin(15)$  una calculadora o tu ordenador?, ¿cómo se detecta la huella de tu mano en una impresora? ¿Cómo se realizan predicciones de mercado? Pues en esto y muchísimo más están implicados los polinomios. Los polinomios tienen la bondad de que operarlos no es hacer otras cosas que sumas y multiplicaciones y tienen tan buenas propiedades como continuidad,

diferenciabilidad,  $C^\infty$  que podemos aproximar funciones extrañas localmente por polinomios. Por ello vamos a dedicarles este número y el siguiente para que aprendan como usarlos en Octave y transversalmente aprendan algunas de sus aplicaciones. Diviértanse con ello.

## 1 Funciones de Octave que involucran polinomios

Vamos a ver operaciones entre polinomios, evaluación de un polinomio, construcción de un polinomio a partir de sus raíces y el cálculo de las raíces de un polinomio.

### 1.1 Operaciones entre polinomios

Las operaciones suma y resta se realizarían con los comandos `+` y `-`. Por ejemplo probemos que queremos sumar el polinomio  $p(x) = x^3 + x^2 + x + 1$  con  $q(x) = 2x^2 + 2x$ . ¿Cómo lo haríamos en Octave? Veamos el siguiente código:

```
p=[1 1 1 1];  
q=[2 2 0];  
s=p+q
```

Si lo has ejecutado te habrás encontrado con el siguiente error:

**operator +: nonconformant arguments (op1 is 1x4, op2 is 1x3).**

Si recuerdas bien en el número anterior vimos operaciones entre matrices y estamos representando un polinomio con una matriz, en particular un vector fila. Vimos que para sumar dos matrices estas deberían ser del mismo tamaño. Esa es la razón por la que no se ha realizado la operación correctamente. Lo corregimos, pues, en el siguiente código donde ya realizamos la operación resta también, ¿te ha coincidido con tus cuentas?

```
p=[1 1 1 1];  
q=[0 2 2 0];  
s=p+q  
r=p-q
```

Como sabemos dos polinomios de cualquier grado pueden ser multiplicados y además el resultado es un polinomio de grado la suma de los grados de los polinomios que se desean multiplicar. Para la multiplicación existe un comando que la realiza que es el comando `conv()`. Multipliquemos los polinomios  $p(x)$  y  $q(x)$  de antes.

```
p=[1 1 1 1];  
q=[2 2 0];  
conv(p,q)
```

Dos polinomios también pueden ser divididos si el grado del dividendo es mayor o igual que el grado del divisor. El resultado de la operación nos da un cociente y un resto cuyo grado es menor que el del divisor. En Octave podemos realizar esta operación usando el comando `deconv()` y nos da dos salidas, el cociente y el resto. Damos un ejemplo en el siguiente código tomando como dividendo (D) el polinomio  $p(x)$  y el divisor (d) el polinomio  $q(x)$ .

```
D=[1 1 1 1];  
d=[2 2 0];  
[Q R]=deconv(p,q)
```

obteniendo como cociente el polinomio  $\frac{1}{2}x$  y como resto el polinomio  $x + 1$ .

## 1.2 Evaluación de un polinomio y sus raíces

Para evaluar un polinomio en un punto o en un conjunto de puntos usaremos el comando `polyval(p,x)` donde `p` será el polinomio a evaluar en notación Octave y `x` un valor o una matriz de valores donde ser evaluado el polinomio. Ejecuta el siguiente código y observa los resultados.

```
p=[1 2 3 4];  
polyval(p,0)  
polyval(p,1)  
polyval(p,[1 2 3;4 5 6])
```

El cálculo de las raíces de un polinomio  $p(x)$  (esto son los puntos del eje  $x$  por donde el corta polinomio o más formalmente los valores  $x$  tales que  $p(x) = 0$ ) no es un cálculo trivial. En la escuela nos enseñan como calcular las raíces de un polinomio de grado dos, pero a partir de grado tres el cómputo es mucho más complejo y además nos ponían ejemplos cuyas raíces eran enteras o como mucho racionales. Pero, esto parece un poco manipulado, ¿verdad? y bastante lejano a la realidad. El cálculo de raíces de un polinomio cualesquiera conlleva en la práctica un conjunto de conocimientos más sofisticados que involucran cálculo numérico. En un futuro podremos hablar un poco de ello. El comando de matlab que calcula, o más convenientemente dicho, que aproxima las raíces de un polinomio cualesquiera es el comando `roots(·)`, y los algoritmos que usa para el cálculo están actualizados constantemente según las mejoras de las investigaciones que se realizan para depurar todos esos algoritmos existentes.

Además tenemos el comando `poly(v)` que calcula un polinomio de grado mínimo y mónico (esto es, que el coeficiente director vale 1) cuyas raíces son los elementos del vector  $v$ . Para ver como funcionan estos comandos ejecuta el siguiente código.

```
p=[1 2 3 4];  
% Calculemos las raíces del polinomio p(x)  
roots(p)  
% Construyamos un polinomio cuyas raíces sean los elementos del vector fila v  
v=[0 1 2 3 4]  
t=poly(v)  
% Calculemos sus raíces y comparemos con v, ¿son iguales?  
roots(poly(v))
```

## 2 Un reto: cálculo de un polinomio de grado tres

En esta sección te proponemos un reto, escribir una función en Octave que realice lo siguiente: calcular los coeficientes del polinomio de grado tres que pase por cuatro puntos dados y que dibuje el polinomio resultante junto con los

puntos dados. El output de la función serán los coeficientes y la figura. Mándanos tu programa y si eres el primero en tenerlo bien lo compartiremos con el resto en el próximo número y *recibirás un obsequio*.

Lo que esperamos es algo parecido a lo que se muestra en la Figura 2.

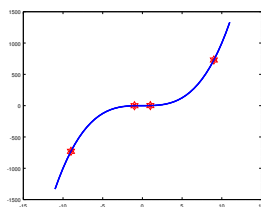


Figura 2: Polinomio calculado en azul, puntos dados en rojo

## 3 Programando: expresiones lógicas

Según vayamos avanzando en la complejidad de los programas que vayamos desarrollando vamos a tener que hacer uso de expresiones lógicas, por ejemplo para incluirlas en expresiones condicionales. En la siguiente tabla mostramos la terminología a usar en las expresiones lógicas.

<	menor que	&&	Conjunción
<=	menor o igual que		Disyunción
>	mayor que	~	Negación
>=	mayor o igual que	xor	Disyunción exclusiva
==	igual a	~=	distinto de

Las expresiones lógicas dan como resultado booleanos: verdadero o falso. En Octave verdadero es representado con el dígito 1 y falso con el dígito 0. Prueba el siguiente código.

```
x=2; y=3; z=4; A=[-3 0 3];
% Expresiones verdaderas

x>0
x+y>=z
x==y-1
x>0 && y<z
x<0 || x>0
A>=-3

% Expresiones falsas

xor(x>0,y>0)
xor(x<0,y<0)
```

$\sim(x>0)$

Las expresiones lógicas pueden aparecer en expresiones condicionales (con el comando **if**) o expresiones de flujo de eventos (con los comandos **while** o **for**). Todas ellas las explicaremos con detalle en el siguiente número. Intenta escribir un programa que dado un vector cuyo primer término es distinto de cero y por tanto representa un polinomio nos devuelva el grado de este. Te dejamos el código en el formato electrónico de la revista. No olvides chequearlo. Se llama grado-pol.m.



107 108 109 110

3D ■ ADVERTISING ■ ANIMATION ■ SPECIAL EFFECTS  
DIGITAL CONTENTS ■ RECORDING ■ DESIGN  
SIMULATION ■ POST-PRODUCTION ■ COMPUTER GRAPHICS

**MAD**mouse  
studio

[www.madmousestudio.com](http://www.madmousestudio.com)