

## Entrega 2 (Modelo 1)

---

### Programa

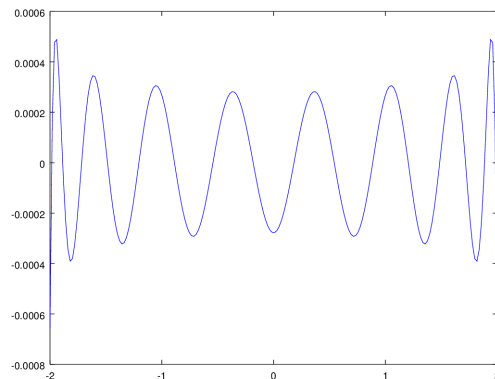
```

1 N = 200;
2 x = linspace(-2,2, N)';
3 y = sin(3*cos(x));
4 A = [ones(N,1), x.^2, x.^4];
5 P = inv(A'*A)*A';
6 c = P*y;
7 er = norm( A*c-y )
8
9 erinf = 1; % error L-infinito ficticio inicial
10 k = 2; % la última columna de A es x.^(2*k)
11 while erinf >= 1e-3
12     k = k+1; % incrementa k
13     A = [A, x.^(2*k)]; % añade la nueva columna
14     P = inv(A'*A)*A'; % calcula la pseudoinversa
15     erinf = norm( y-A*P*y, Inf); % calcula el error
16 end
17 disp(['El valor de k es ' num2str(k)]) % muestra k
18
19 figure(1) % dibuja la gráfica
20 plot(x, y-A*P*y )
21
22 % El espacio vectorial generado por [1,x^2,x^4,x^4(1-x^3)]
23 % es el mismo que el generado por [1,x^2,x^4,x^3]
24 % pero x^3 es ortogonal al vector 'y' porque la función f
25 % es par, por tanto desaparece en la proyección ortogonal.
26 % En el segundo caso se añade un x^6 que ya no es ortogonal.

```

### Salida

er = 2.8213  
El valor de k es 6



## Entrega 2 (Modelo 2)

---

### Programa

```

1 N = 200;
2 x = linspace(-1,1, N)';
3 y = cos(4*x.*cos(x));
4 A = [ones(N,1), x.^2, x.^4];
5 P = inv(A'*A)*A';
6 c = P*y;
7 er = norm( A*c-y )
8
9 erinf = 1; % error L-infinito ficticio inicial
10 k = 2; % la última columna de A es x.^(2*2)
11 while erinf >= 1e-3
12     k = k+1; % incrementa k
13     A = [A, x.^(2*k)]; % añade la nueva columna
14     P = inv(A'*A)*A'; % calcula la pseudoinversa
15     erinf = norm( y-A*P*y, Inf); % calcula el error
16 end
17 disp(['El valor de k es ' num2str(k)]) % muestra k
18
19 figure(1) % dibuja la gráfica
20 plot(x, y-A*P*y )
21
22 % El espacio vectorial generado por [1,x^2,x^4,x^4(1-x^3)]
23 % es el mismo que el generado por [1,x^2,x^4,x^3]
24 % pero x^3 es ortogonal al vector 'y' porque la función f
25 % es par, por tanto desaparece en la proyección ortogonal.
26 % En el segundo caso se añade un x^6 que ya no es ortogonal.

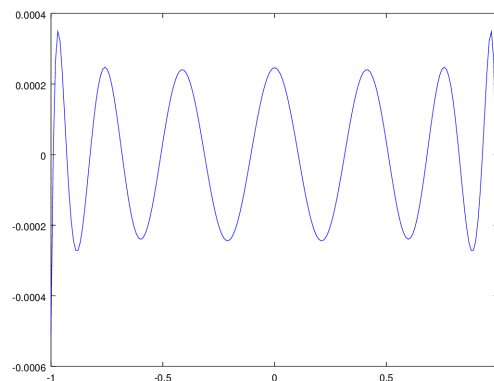
```

### Salida

```

er = 1.1911
El valor de k es 6

```



## Entrega 2 (Modelo 3)

---

### Programa

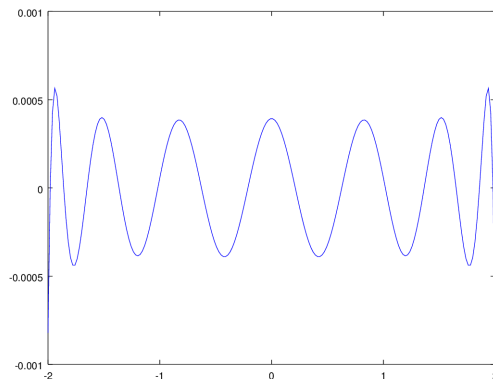
```

1 N = 200;
2 x = linspace(-2,2, N)';
3 y = cos(1.5*x+sin(x));
4 A = [ones(N,1), x.^2, x.^4];
5 P = inv(A'*A)*A';
6 c = P*y;
7 er = norm( A*c-y )
8
9 erinf = 1; % error L-infinito ficticio inicial
10 k = 2; % la última columna de A es x.^(2*2)
11 while erinf >= 1e-3
12     k = k+1; % incrementa k
13     A = [A, x.^(2*k)]; % añade la nueva columna
14     P = inv(A'*A)*A'; % calcula la pseudoinversa
15     erinf = norm( y-A*P*y, Inf); % calcula el error
16 end
17 disp(['El valor de k es ' num2str(k)]) % muestra k
18
19 figure(1) % dibuja la gráfica
20 plot(x, y-A*P*y )
21
22 % El espacio vectorial generado por [1,x^2,x^4,x^4(1-x^3)]
23 % es el mismo que el generado por [1,x^2,x^4,x^3]
24 % pero x^3 es ortogonal al vector 'y' porque la función f
25 % es par, por tanto desaparece en la proyección ortogonal.
26 % En el segundo caso se añade un x^6 que ya no es ortogonal.

```

### Salida

er = 2.2649  
El valor de k es 6



## Entrega 2 (Modelo 4)

---

### Programa

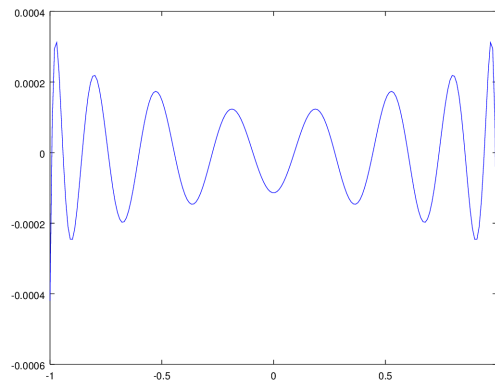
```

1 N = 200;
2 x = linspace(-1,1, N)';
3 y = sin(x.^2.*cos(4*x));
4 A = [ones(N,1), x.^2, x.^4];
5 P = inv(A'*A)*A';
6 c = P*y;
7 er = norm( A*c-y )
8
9 erinf = 1; % error L-infinito ficticio inicial
10 k = 2; % la última columna de A es x.^(2*2)
11 while erinf >= 1e-3
12     k = k+1; % incrementa k
13     A = [A, x.^(2*k)]; % añade la nueva columna
14     P = inv(A'*A)*A'; % calcula la pseudoinversa
15     erinf = norm( y-A*P*y, Inf); % calcula el error
16 end
17 disp(['El valor de k es ' num2str(k)]) % muestra k
18
19 figure(1) % dibuja la gráfica
20 plot(x, y-A*P*y )
21
22 % El espacio vectorial generado por [1,x^2,x^4,x^4(1-x^3)]
23 % es el mismo que el generado por [1,x^2,x^4,x^3]
24 % pero x^3 es ortogonal al vector 'y' porque la función f
25 % es par, por tanto desaparece en la proyección ortogonal.
26 % En el segundo caso se añade un x^6 que ya no es ortogonal.

```

### Salida

er = 0.87167  
El valor de k es 6



## Entrega 2 (Modelo 5)

---

### Programa

```

1 N = 200;
2 x = linspace(-1,1, N)';
3 y = sin(3*x.^2.*cos(x));
4 A = [ones(N,1), x.^2, x.^4];
5 P = inv(A'*A)*A';
6 c = P*y;
7 er = norm( A*c-y )
8
9 erinf = 1; % error L-infinito ficticio inicial
10 k = 2; % la última columna de A es x.^(2*2)
11 while erinf >= 1e-3
12     k = k+1; % incrementa k
13     A = [A, x.^(2*k)]; % añade la nueva columna
14     P = inv(A'*A)*A'; % calcula la pseudoinversa
15     erinf = norm( y-A*P*y, Inf); % calcula el error
16 end
17 disp(['El valor de k es ' num2str(k)]) % muestra k
18
19 figure(1) % dibuja la gráfica
20 plot(x, y-A*P*y )
21
22 % El espacio vectorial generado por [1,x^2,x^4,x^4(1-x^3)]
23 % es el mismo que el generado por [1,x^2,x^4,x^3]
24 % pero x^3 es ortogonal al vector 'y' porque la función f
25 % es par, por tanto desaparece en la proyección ortogonal.
26 % En el segundo caso se añade un x^6 que ya no es ortogonal.

```

### Salida

```

er = 0.34313
El valor de k es 6

```

