

Modelo 1

$$S' = \pi - \beta \frac{M}{N} S - r_1 \frac{M}{N} S - \eta_1 S$$

$$M' = \beta \frac{M}{N} S - r_2 \frac{M}{N} S - \eta_2 M$$

$$R' = \eta_1 S + \frac{M}{N} S(r_1 + r_2) + \eta_2 M$$

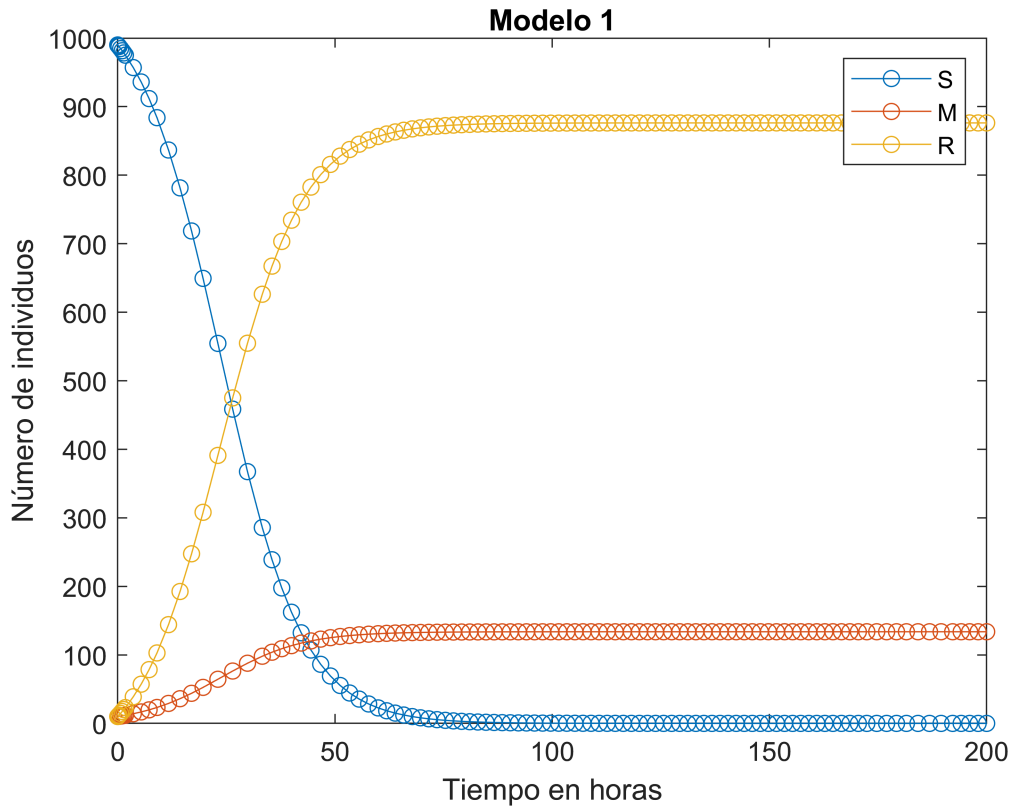
```
%Modelo 1
clear global;
%Establecer los parámetros
% r_1 + r_2 + beta < 1
%orden: pi, beta, r_1, eta_1, r_2, eta_2
set([0.00, 0.2, 0.6, 0.00, 0.1, 0.00]);

%Resolver el modelo 1
[t,y] = ode45(@modelo1,[0 200] ,[990 10 10]);

%Calcular e imprimir el R_0
disp("El R_0 del modelo 1 es " + r0Esquema1());
```

El R_0 del modelo 1 es 2

```
%Graficar el modelo 1
figure(1)
plot(t,y(1:end,1), '-o')
hold on
plot(t,y(1:end,2), '-o')
hold on
plot(t,y(1:end,3), '-o')
legend({'S','M','R'})
xlabel("Tiempo en horas")
ylabel("Número de individuos")
title('Modelo 1')
hold off
```



Modelo 2

$$S' = \alpha_1 \pi - \beta \frac{M}{N} S - r_1 \frac{M}{N} S - \eta_1 S$$

$$I' = \alpha_2 \pi - r_1 \frac{M}{N} I - \eta_1 I$$

$$E' = \beta \frac{M}{N} S - \mu E - \eta_3 E - \eta_1 E$$

$$M' = \mu E - r_2 \frac{M}{N} (S + I) - \eta_2 M$$

$$R' = \eta_1 (I + S) + \eta_2 M + \eta_3 E + r_1 \frac{M}{N} (S + I) + r_2 \frac{M}{N} (S + I)$$

```
%Modelo 2
clear global;
%Establecer los parámetros
%eta_2 afecta el R_0
%orden: pi, beta, r_1, eta_1, r_2, eta_2, eta_3, mu
set2( [0, 0.35, 0.15, 0, 0.08, 0, 0.05, 0.1]);

%Calcular e imprimir el R_0
disp("El R_0 del modelo 2 es " + r0Esquema2());
```

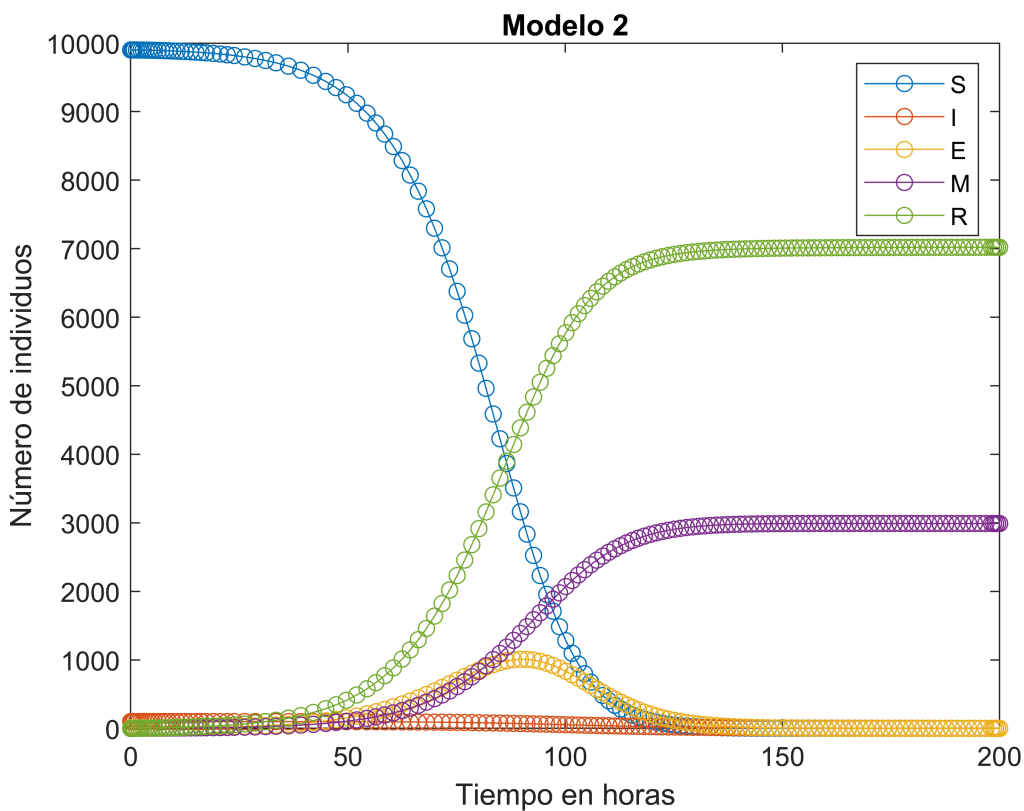
El R_0 del modelo 2 es 2.9167

```

%Resolver el modelo 2
[t,y] = ode45(@modelo2,[0 200] ,[9900 100 10 0 1]);

%Graficar el modelo 2
figure(2)
plot(t,y(1:end,1), '-o')
hold on
plot(t,y(1:end,2), '-o')
hold on
plot(t,y(1:end,3), '-o')
hold on
plot(t,y(1:end,4), '-o' )
hold on
plot(t,y(1:end,5), '-o' )
legend({'S','I', 'E', 'M', 'R'})
xlabel("Tiempo en horas")
ylabel("Número de individuos")
title('Modelo 2')
hold off

```



Modelo 3

$$S' = \alpha_1 \pi - \beta \frac{M}{N} S - r_1 \frac{M}{N} S - \eta_1 S$$

$$I' = \alpha_2 \pi + iT - r_1 \frac{M}{N} I - \eta_1 I$$

$$E' = \beta \frac{M}{N} S - \mu E - \eta_3 E - \tau E - \eta_1 E$$

$$T' = \tau E - iT - \mu' T - \eta_4 T$$

$$M' = \mu E + \mu' T - r_2 \frac{M}{N} (I + S) - \eta_2 M$$

$$R' = \eta_1 (I + S + E) + \eta_3 E + \eta_4 T + \eta_2 M + r_1 \frac{M}{N} (I + S) + r_2 \frac{M}{N} (I + S)$$

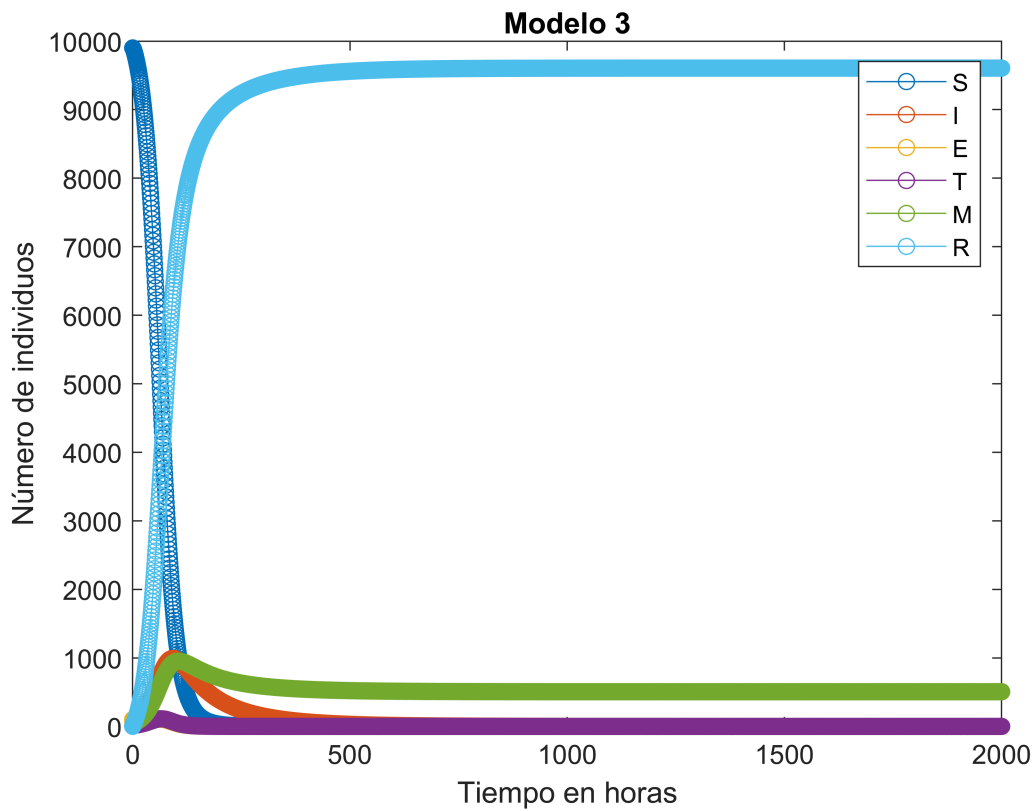
```
%Modelo 3
clear global;
% inm + eta_4 + mu_2 <= 1
%Establecer los parámetros
%orden: pi, beta, r_1, eta_1, r_2, eta_2, eta_3, mu, mu_2 , inm, eta_4, tau
set3( [0, 0.3, 0.15, 0 , 0.08, 0, 0.15, 0.3, 0.1, 0.2 , 0.08 , 0.4 ]);

%Calcular e imprimir el R_0
disp("El R_0 del modelo 3 es " + r0Esquema3());
```

El R_0 del modelo 3 es 1.7879

```
%Resolver el modelo 3
[t,y] = ode45(@modelo3,[0 2000] ,[9900 100 100 1 10 1]);

%Graficar el modelo 3
figure(3)
plot(t,y(1:end,1), '-o')
hold on
plot(t,y(1:end,2), '-o')
hold on
plot(t,y(1:end,3), '-o')
hold on
plot(t,y(1:end,4), '-o' )
hold on
plot(t,y(1:end,5), '-o' )
hold on
plot(t,y(1:end,6), '-o' )
legend({'S','I', 'E', 'T', 'M', 'R'})
xlabel("Tiempo en horas")
ylabel("Número de individuos")
title('Modelo 3')
hold off
```



Modelo 4

$$S' = \alpha_1 \pi - fS \left(\frac{M}{N} \right)^p \left(1 - \frac{M}{N} \right)^{q-1} - \beta \frac{M}{N} S - r_1 \frac{M}{N} S - \eta_1 S$$

$$I' = \alpha_2 \pi - fI \left(\frac{M}{N} \right)^p \left(1 - \frac{M}{N} \right)^{q-1} - r_1 \frac{M}{N} I - \eta_1 I$$

$$E' = \beta \frac{M}{N} S - \mu E - \eta_3 E - \eta_1 E$$

$$F' = (Sf + If) \left(\frac{M}{N} \right)^p \left(1 - \frac{M}{N} \right)^{q-1}$$

$$M' = \mu E - r_2 \frac{M}{N} (S + I) - \eta_2 M$$

```
%Modelo 4
clear global;
%Establecer los parámetros
%orden: pi, beta, r_1, eta_1, r_2, eta_2, eta_3, mu, p, q, f
set4( [0, 0.3, 0.2, 0, 0.08, 0, 0.15, 0.2, 1.1, 1.01, 0.1 ] );

%Calcular e imprimir el R_0
disp("El R_0 del modelo 4 es " + r0Esquema4());
```

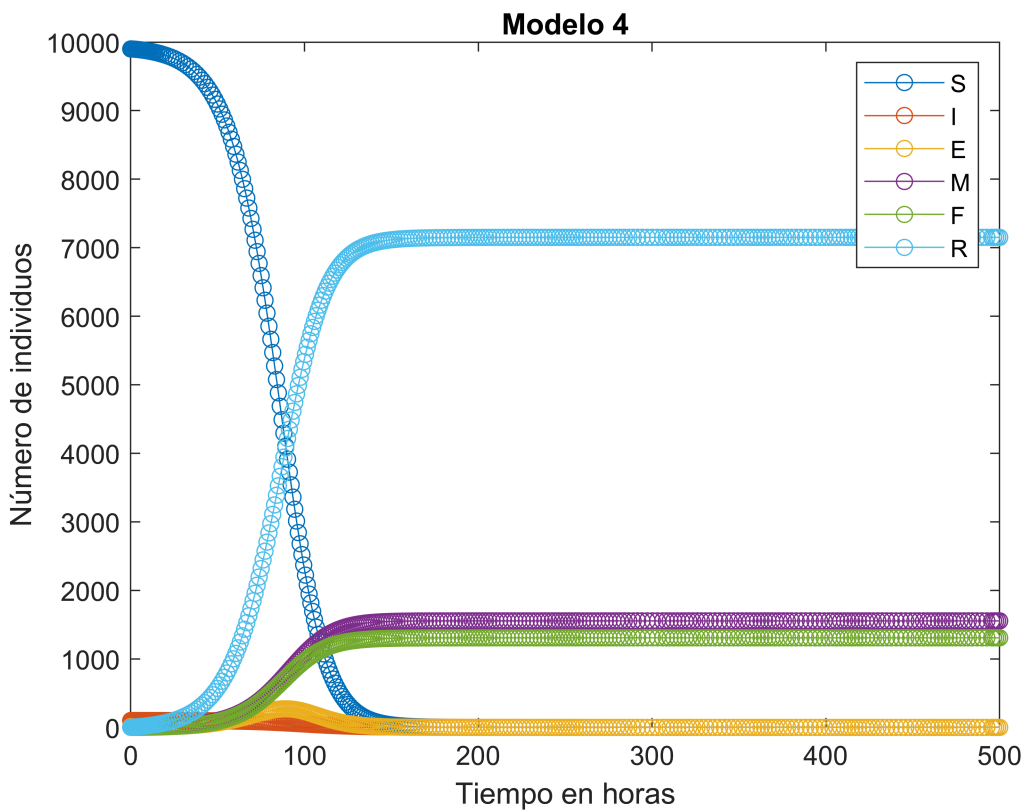
El R_0 del modelo 4 es 2.1429

```

%Resolver el modelo 4
[t,y] = ode45(@modelo4,[0 500] ,[9900 100 10 1 1 1]);

%Graficar el modelo 4
figure(4)
plot(t,y(1:end,1), '-o')
hold on
plot(t,y(1:end,2), '-o')
hold on
plot(t,y(1:end,3), '-o')
hold on
plot(t,y(1:end,4), '-o' )
hold on
plot(t,y(1:end,5), '-o' )
hold on
plot(t,y(1:end,6), '-o' )
hold on
legend({'S','I', 'E', 'M', 'F', 'R'})
xlabel("Tiempo en horas")
ylabel("Número de individuos")
title('Modelo 4')
hold off

```



Funciones

```
function[] = set4(par)
% Declara los parámetros globales modelo 4
%Entradas: par --- vector que contiene los parámetros en el siguiente
%orden: pi, beta, r_1, eta_1, r_2, eta_2, eta_3, mu, p , q, f
global par1
par1 = par(1);

global par2
par2 = par(2);

global par3
par3 = par(3);

global par4
par4 = par(4);

global par5
par5 = par(5);

global par6
par6 = par(6);

global par7
par7 = par(7);

global par8
par8 = par(8);

global par9
par9 = par(9);

global par10
par10 = par(10);

global par11
par11 = par(11);

end

function dydt = modelo4(~,y)
% Esta función modela el esquema 4

N = sum(y);

global par1
pi = par1;

global par2
beta = par2;
global par3
r_1 = par3;
```

```

global par4
eta_1= par4;
global par5
r_2 = par5;
global par6
eta_2 = par6;
global par7
eta_3 = par7;
global par8
mu = par8;
global par9
p = par9;
global par10
q = par10;
global par11
f = par11;

dydt = zeros(6,1);
fac = y(4)/N * y(1);
fac_2 = y(4)/N * y(2);
ent = (y(4)/N)^p*(1 - y(4)/N)^(q-1);

%S
dydt(1) = 0.99*pi - beta*fac - r_1*fac - eta_1*y(1) - y(1)*ent*f;
%I
dydt(2) = 0.01*pi - r_1*fac_2 - eta_1*y(2) - y(2)*ent*f;
%E
dydt(3) = beta*fac - y(3)*(mu + eta_3 + eta_1);
%M
dydt(4) = y(3)*mu - r_2*(fac + fac_2) - eta_2*y(2);
%F
dydt(5) = y(1)*ent*f + y(2)*ent*f;
%R
dydt(6) = eta_1*(y(1) + y(2) + y(3)) + (fac + fac_2)*(r_1 + r_2) + eta_3*y(3) + eta_2*y(4);
end

function[R0] = r0Esquema4()
% Calcula el R0 del modelo 4
global par2
beta = par2;
global par5
r_2 = par5;
global par8
mu = par8;
global par7
eta_3 = par7;

R0 = (mu*beta)/(r_2*(mu + eta_3));

end

```



```

function[] = set3(par)
% Declara los parámetros globales modelo 3
%Entradas: par --- vector que contiene los parámetros en el siguiente
%orden: pi, beta, r_1, eta_1, r_2, eta_2, eta_3, mu, mu_2 , inm, eta_4, tau
global par1
par1 = par(1);

global par2
par2 = par(2);

global par3
par3 = par(3);

global par4
par4 = par(4);

global par5
par5 = par(5);

global par6
par6 = par(6);

global par7
par7 = par(7);

global par8
par8 = par(8);
global par9
par9 = par(9);
global par10
par10 = par(10);
global par11
par11 = par(11);
global par12
par12 = par(12);
end

function dydt = modelo3(~,y)
% Función que representa el modelo 3

N = sum(y);

global par1

```

```

pi = par1;

global par2
beta = par2;
global par3
r_1 = par3;

global par4
eta_1= par4;
global par5
r_2 = par5;
global par6
eta_2 = par6;
global par7
eta_3 = par7;
global par8
mu = par8;

global par9
mu_2 = par9;
global par10
inm = par10;
global par11
eta_4 = par11;
global par12
tau = par12;

dydt = zeros(6,1);
fac = (y(5)/N) * y(1);
fac_2 = (y(5)/N) * y(2);

%S
dydt(1) = 0.99*pi - beta*fac - r_1*fac -eta_1*y(1);
%I
dydt(2) = 0.01*pi - r_1*fac_2 - eta_1*y(2) + y(4)*inm;
%E
dydt(3) = beta*fac - y(3)*(mu + eta_3 + eta_1 + tau);
%T
dydt(4) = tau*y(3) - y(4)*(inm + mu_2 + eta_4);
%M
dydt(5) = y(3)*mu - r_2*(fac + fac_2) - eta_2*y(2) + y(4)*mu_2;
%R
dydt(6) = eta_1*(y(1) + y(2) + y(3)) + (fac + fac_2)*(r_1 + r_2) + eta_3*y(3) + eta_2*y(5) + et

end

function[R0] = r0Esquema3()
% Calcula el R0 del modelo 3
global par2
beta = par2;
global par5
r_2 = par5;
global par8
mu = par8;

```

```

global par7
eta_3 = par7;

global par9
mu_2 = par9;
global par10
inm = par10;
global par11
eta_4 = par11;
global par12
tau = par12;
%disp(inm + mu_2 + eta_4);

R0 = (beta*(tau*mu_2 + mu*(mu_2 + inm + eta_4)))/(r_2*(mu + eta_3 + tau)*(inm + mu_2 + eta_4)

end

function[] = set2(par)
% Declara los parámetros globales modelo 2
%Entradas: par --- vector que contiene los parámetros en el siguiente
%orden: pi, beta, r_1, eta_1, r_2, eta_2, eta_3, mu
global par1
par1 = par(1);

global par2
par2 = par(2);

global par3
par3 = par(3);

global par4
par4 = par(4);

global par5
par5 = par(5);

global par6
par6 = par(6);

global par7
par7 = par(7);

global par8
par8 = par(8);
end

function dydt = modelo2(~,y)
% Esta es el sistema que corresponde al modelo 2
N = sum(y);

global par1
pi = par1;

```

```

global par2
beta = par2;
global par3
r_1 = par3;

global par4
eta_1= par4;
global par5
r_2 = par5;
global par6
eta_2 = par6;
global par7
eta_3 = par7;
global par8
mu = par8;

dydt = zeros(5,1);
fac = y(4)/N * y(1);
fac_2 = y(4)/N * y(2);

%S
dydt(1) = 0.99*pi - beta*fac - r_1*fac -eta_1*y(1);
%I
dydt(2) = 0.01*pi - r_1*fac_2 - eta_1*y(2);
%E
dydt(3) = beta*fac - y(3)*(mu + eta_3 + eta_1);
%M
dydt(4) = y(3)*mu - r_2*(fac + fac_2) - eta_2*y(2);
%R
dydt(5) = eta_1*(y(1) + y(2) + y(3)) + (fac + fac_2)*(r_1 + r_2) + eta_3*y(3) + eta_2*y(4);

end

function[R0] = r0Esquema2()
% Calcula el R0 del modelo 2
global par2
beta = par2;
global par5
r_2 = par5;
global par8
mu = par8;
global par7
eta_3 = par7;

R0 = (mu*beta)/(r_2*(mu + eta_3));

end

function[] = set(par)
% Declara los parámetros globales modelo 1
%Entradas: par --- vector que contiene los parámetros en el siguiente
%orden: pi, beta, r_1, eta_1, r_2, eta_2

```

```

global par1
par1 = par(1);

global par2
par2 = par(2);

global par3
par3 = par(3);

global par4
par4 = par(4);

global par5
par5 = par(5);

global par6
par6 = par(6);
end

function dydt = modelo1(~,y)
% Esta es el sistema que corresponde al modelo 1
N = sum(y);
global par1
pi = par1;
global par2
beta = par2;
global par3
r_1 = par3;

global par4
eta_1= par4;
global par5
r_2 = par5;
global par6
eta_2 = par6;

dydt = zeros(3,1);
fac = y(2)/N * y(1);

%S
dydt(1) = pi - beta*fac - r_1*fac -eta_1*y(1);
%M
dydt(2) = beta*fac - r_2*fac - eta_2*y(2);
%R
dydt(3) = eta_1*y(1) + fac*(r_1 + r_2) + eta_2*y(2);

end

function[R0] = r0Esquema1()
% Calcula el R0 del modelo 1 usando los parámetros globales
global par6
eta_2 = par6;
global par2

```

```
beta = par2;  
global par5  
r_2 = par5;  
  
R0 = beta/(r_2 + eta_2);  
  
end
```