Modelo

1. Definir ecuaciones para el modelo y cargar los datos de la serie de tiempo

```
syms  Ms(t) Me(t) Mi(t) Hs(t) He(t) Hi(t) Hr(t) Hit(t)

syms lambda beta_m mu_m theta_m mu_h beta_h theta_h...
    gamma_h


H=Hs+He+Hi+Hr;
M=Ms+Me+Mi;
ode1 = diff(Ms) == lambda - beta_m*Hi*Ms/H - (mu_m)*Ms;
ode2 = diff(Me) == beta_m*Hi*Ms/H - (theta_m+mu_m)*Me;
ode3 = diff(Mi) == theta_m*Me - mu_m*Mi;
ode4 = diff(Hs) == -beta_h*Mi*Hs/M + (He+Hi+Hr)*mu_h;
ode5 = diff(He) == beta_h*Mi*Hs/M - (theta_h+mu_h)*He;
ode6 = diff(Hi) == theta_h*He - (gamma_h+mu_h)*Hi;
ode7 = diff(Hr) == gamma_h*Hi - mu_h*Hr;
ode8 = diff(Hit) == theta_h*He;
odes=[ode1; ode2 ;ode3; ode4; ode5 ;ode6; ode7; ode8];
vars=[Hit Hi Me Hr Hs He Ms Mi];
opts = odeset('NonNegative',1:8);
```

2. Preparar los datos:

Escogimos la ventana de tiempo (entre la semana 153 y la semana 233).

```
load Range7.mat
[T,~]=gsua_dpmat(odes,vars,[0 150],'7m','output',1,'opt',opts,'Range',Range);
```

```
Introduce ranges in the following order:
```

ans(t) = $\left(\mathrm{Hit}(t)\quad \mathrm{Hi}(t)\quad \mathrm{Me}(t)\quad \mathrm{Hr}(t)\quad \mathrm{Hs}(t)\quad \mathrm{He}(t)\quad \mathrm{Ms}(t)\quad \mathrm{Mi}(t)\quad \beta_h\quad \beta_m\quad \gamma_h\quad \lambda\quad \mu_h\quad \mu_m\quad \theta_h\quad \theta_m\right)$

```
load('DataBello_full.mat');
ydata=DataBello.cases(153:233)';
xdata=linspace(0,length(ydata)-1,length(ydata));
ydata2=ydata;
for i=1:length(ydata)
ydata2(i)=sum(ydata(1:i));
end
```

3. Analisis de sensibilidad

3.1. Preparamos los datos: usamos la tabla con los rangos y los valores nominales de los parametros y preparamos la matriz de diseño experimental.

```
[T,~]=gsua_dpmat(odes,vars,[0 80],'7m','output',1,'opt',opts,'Range',Range);
```

```
Introduce ranges in the following order:
```

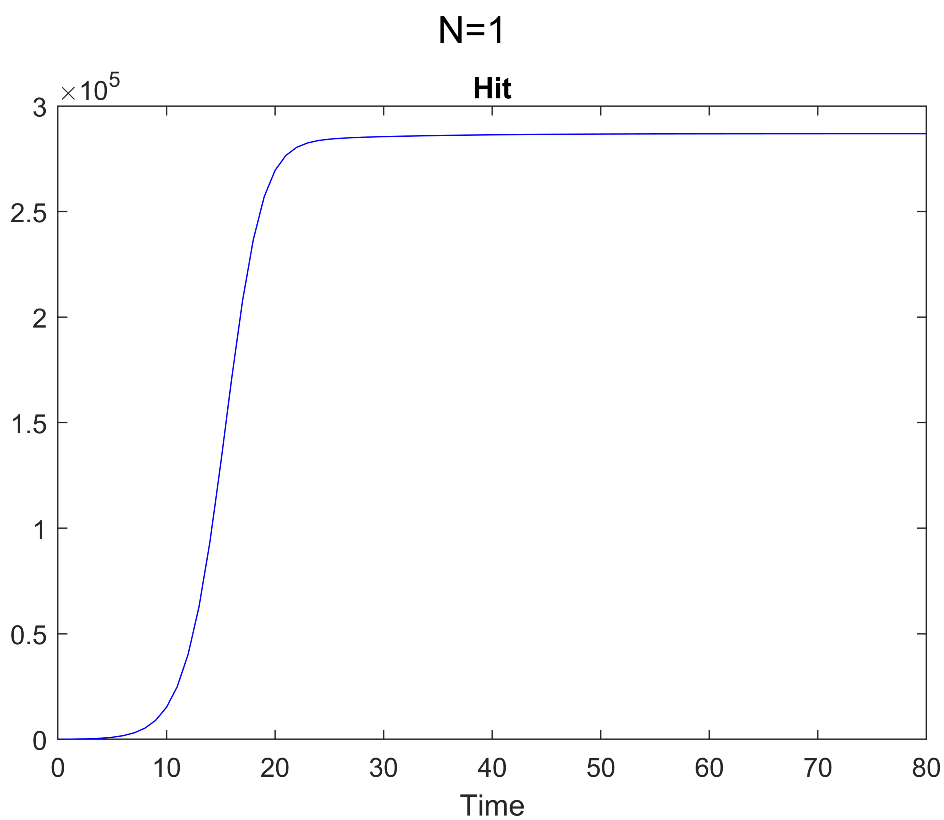
ans(t) = $\begin{pmatrix} \mathrm{Hit}(t) & \mathrm{Hi}(t) & \mathrm{Me}(t) & \mathrm{Hr}(t) & \mathrm{Hs}(t) & \mathrm{He}(t) & \mathrm{Ms}(t) & \mathrm{Mi}(t) & \beta_h & \beta_m & \gamma_h & \lambda & \mu_h & \mu_m & \theta_h & \theta_m \end{pmatrix}$

```
T.Properties.CustomProperties.output = 1;
M = gsua_dmatrix(T,100);
```

3.2. Análisis de sensibilidad

Usamos el toolbox para realizar el análisis de sensibilidad variando los parámetros según los rangos que nos dieron.

```
ynom = gsua_eval(T.Nominal,T);
```



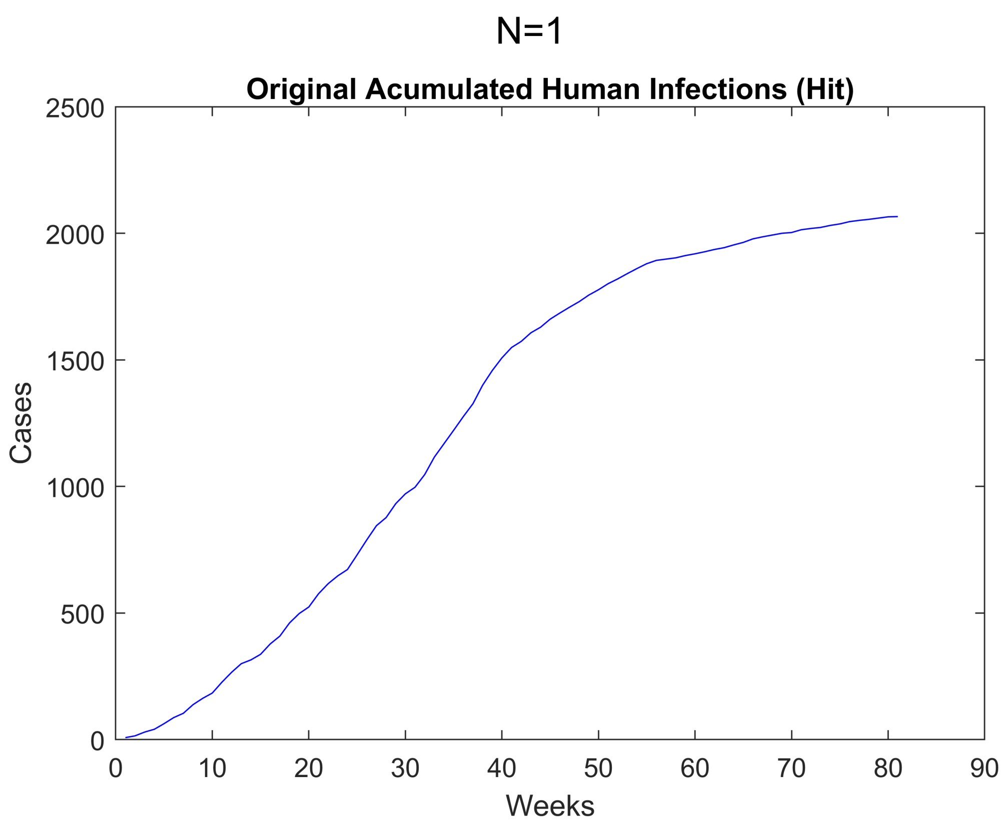

```
plot(cumsum(ydata),'b')
title('Original Acumulated Human Infections (Hit)')
xlabel('Weeks')
ylabel('Cases')
%savefig('iteration1/figures/NominalValues.fig')
```

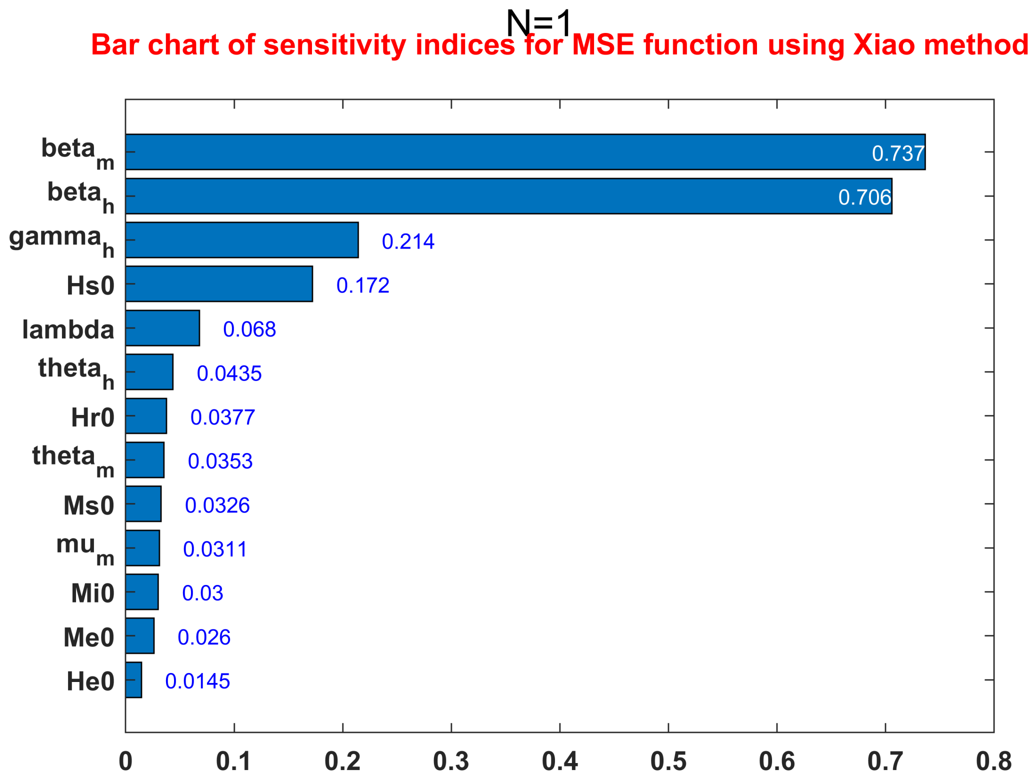
```
Tsa = gsua_sa(M,T,'parallel', false, 'SensMethod', 'Xiao', 'ynom', ynom);
```

```
Progress: 7%
Estimated processing time (h:m:s): 0:0:8
Remaining time (h:m:s): 0:0:8
Elapsed time (h:m:s): 0:0:0
Estimated stop time (h:m:s): 22:4:20
Number of simulations: 750
Progress: 14%
Estimated processing time (h:m:s): 0:0:8
Remaining time (h:m:s): 0:0:7
Elapsed time (h:m:s): 0:0:1
Estimated stop time (h:m:s): 22:4:20
Number of simulations: 750
Progress: 20%
Estimated processing time (h:m:s): 0:0:8
Remaining time (h:m:s): 0:0:6
Elapsed time (h:m:s): 0:0:1
Estimated stop time (h:m:s): 22:4:19
Number of simulations: 750
Progress: 27%
Estimated processing time (h:m:s): 0:0:8
Remaining time (h:m:s): 0:0:6
Elapsed time (h:m:s): 0:0:2
Estimated stop time (h:m:s): 22:4:19
Number of simulations: 750
Progress: 34%
Estimated processing time (h:m:s): 0:0:8
Remaining time (h:m:s): 0:0:5
Elapsed time (h:m:s): 0:0:2
Estimated stop time (h:m:s): 22:4:19
Number of simulations: 750
Progress: 40%
Estimated processing time (h:m:s): 0:0:7
Remaining time (h:m:s): 0:0:4
Elapsed time (h:m:s): 0:0:3
Estimated stop time (h:m:s): 22:4:19
Number of simulations: 750
Progress: 47%
Estimated processing time (h:m:s): 0:0:7
Remaining time (h:m:s): 0:0:4
Elapsed time (h:m:s): 0:0:3
Estimated stop time (h:m:s): 22:4:19
Number of simulations: 750
Progress: 54%
Estimated processing time (h:m:s): 0:0:7
Remaining time (h:m:s): 0:0:3
Elapsed time (h:m:s): 0:0:4
Estimated stop time (h:m:s): 22:4:19
Number of simulations: 750
Progress: 60%
Estimated processing time (h:m:s): 0:0:7
Remaining time (h:m:s): 0:0:3
Elapsed time (h:m:s): 0:0:4
Estimated stop time (h:m:s): 22:4:18
Number of simulations: 750
Progress: 67%
Estimated processing time (h:m:s): 0:0:7
Remaining time (h:m:s): 0:0:2
Elapsed time (h:m:s): 0:0:4
Estimated stop time (h:m:s): 22:4:18
Number of simulations: 750
Progress: 74%
```

```
Estimated processing time (h:m:s): 0:0:7
Remaining time (h:m:s): 0:0:1
Elapsed time (h:m:s): 0:0:5
Estimated stop time (h:m:s): 22:4:18
Number of simulations: 750
Progress: 81%
Estimated processing time (h:m:s): 0:0:7
Remaining time (h:m:s): 0:0:1
Elapsed time (h:m:s): 0:0:5
Estimated stop time (h:m:s): 22:4:18
Number of simulations: 750
Progress: 87%
Estimated processing time (h:m:s): 0:0:7
Remaining time (h:m:s): 0:0:0
Elapsed time (h:m:s): 0:0:6
Estimated stop time (h:m:s): 22:4:18
Number of simulations: 750
Progress: 94%
Estimated processing time (h:m:s): 0:0:7
Remaining time (h:m:s): 0:0:0
Elapsed time (h:m:s): 0:0:6
Estimated stop time (h:m:s): 22:4:18
Number of simulations: 750
Progress: 100%
Estimated processing time (h:m:s): 0:0:7
Remaining time (h:m:s): 0:0:0
Elapsed time (h:m:s): 0:0:7
Estimated stop time (h:m:s): 22:4:18
Number of simulations: 750
```

```
gsua_plot('Bar',Tsa,Tsa.STi)
```






```
%%savefig('iteration1/figures/SensibilityAnalisis.fig')
```

3.3. Confiabilidad

Mientras mayor sea c, mayor confianza tenemos en los resultados del análisis.

```
c = sum(Tsa.Si)/sum(abs(Tsa.Si))
```

```
c = 0.9793
```

3.4 Fijar parámetros

Según el análisis previo,el ultimo parámetro es insensibles, entonces lo podemos fijar ya que no es identificable.

Este parámetros son He0

Para esto usamos los valores nominales de la tabla T.

```
vars=[Hit Hi He Me Hr Hs Ms Mi];

HeO = T.Nominal('He0');

RangeTemp = Range;
```

```
Range(3,:) = He0;
Range(4:6,:) = [RangeTemp(3,:); RangeTemp(4,:); RangeTemp(5,:)];

[T,~]=gsua_dpmat(odes,vars,[0 80],'7m','output',1,'opt',opts,'Range',Range);
```

```
Introduce ranges in the following order:
```

ans(t) = $\left(\mathrm{Hit}(t) \quad \mathrm{Hi}(t) \quad \mathrm{He}(t) \quad \mathrm{Me}(t) \quad \mathrm{Hr}(t) \quad \mathrm{Hs}(t) \quad \mathrm{Ms}(t) \quad \mathrm{Mi}(t) \quad \beta_h \quad \beta_m \quad \gamma_h \quad \lambda \quad \mu_h \quad \mu_m \quad \theta_h \quad \theta_m\right)$

```
T.Properties.CustomProperties.output = 1;
M = gsua_dmatrix(T,100);
```

4. Estimación de parámetros.

```

opt = optimoptions('lsqcurvefit','UseParallel',true,'Display','iter');
[T7,res] = gsua_pe(T,xdata,ydata2,'N',100,'opt',opt); %mover N ( 100 buena, 50 aceptable, 10 :(
%save('iteration1/values/Results7.mat','T7','res','xdata','ydata2')
```

5. Análisis de identificabilidad

Escogemos las curvas que estén más cercanas a la mejor.

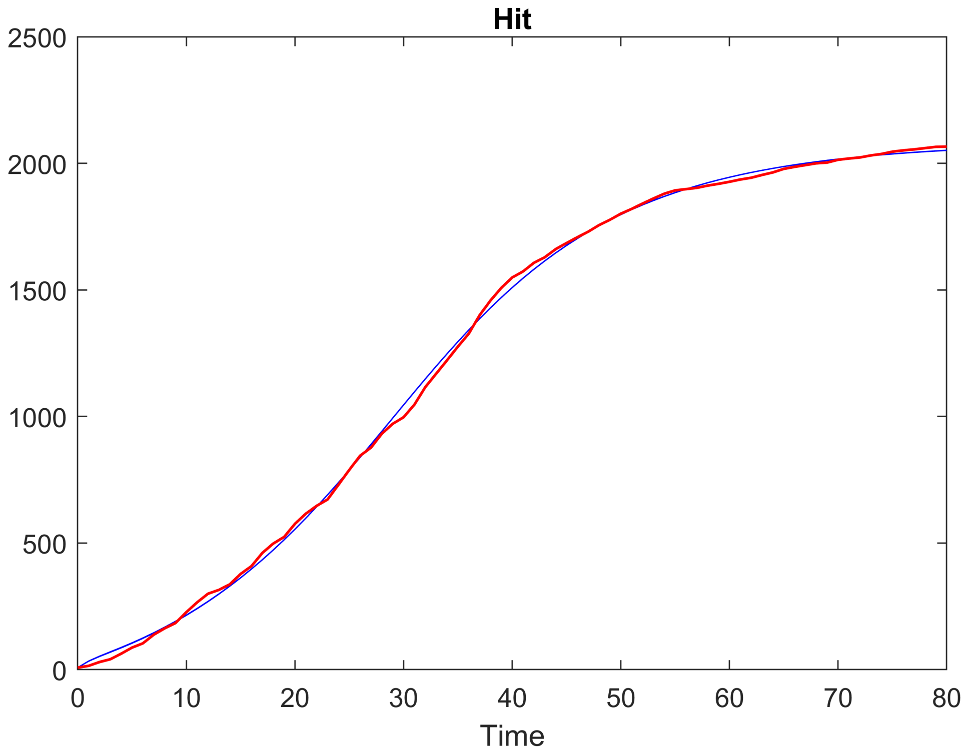
```
th = sum(res<res(1)*1.01)
```

```
th = 3
```

```
y7 = gsua_eval(T7.Estlsqc(:,1:th),T7,xdata,ydata2);
```

```
Sim3 Done
Sim2 Done
Sim1 Done
```

N=3

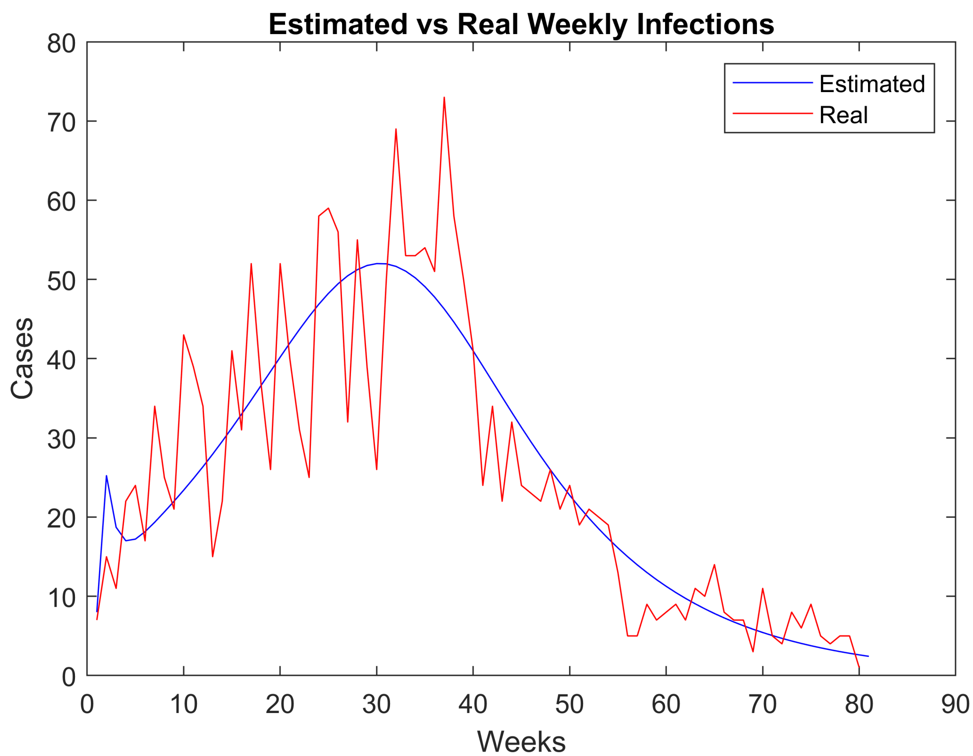



```
%savefig('iteration1/figures/curves.fig')
```

Comparamos la mejor curva con los valores reales para ver que tan bien ajusta.

```
bestest = y7(1,:);
trend = [bestest(1),bestest(2:end)-bestest(1:end-1)];
plot(trend,'b')
hold on
plot(diff(ydata2),'r')
title('Estimated vs Real Weekly Infections')
xlabel('Weeks')
ylabel('Cases')
legend({'Estimated','Real'})
```

N=3



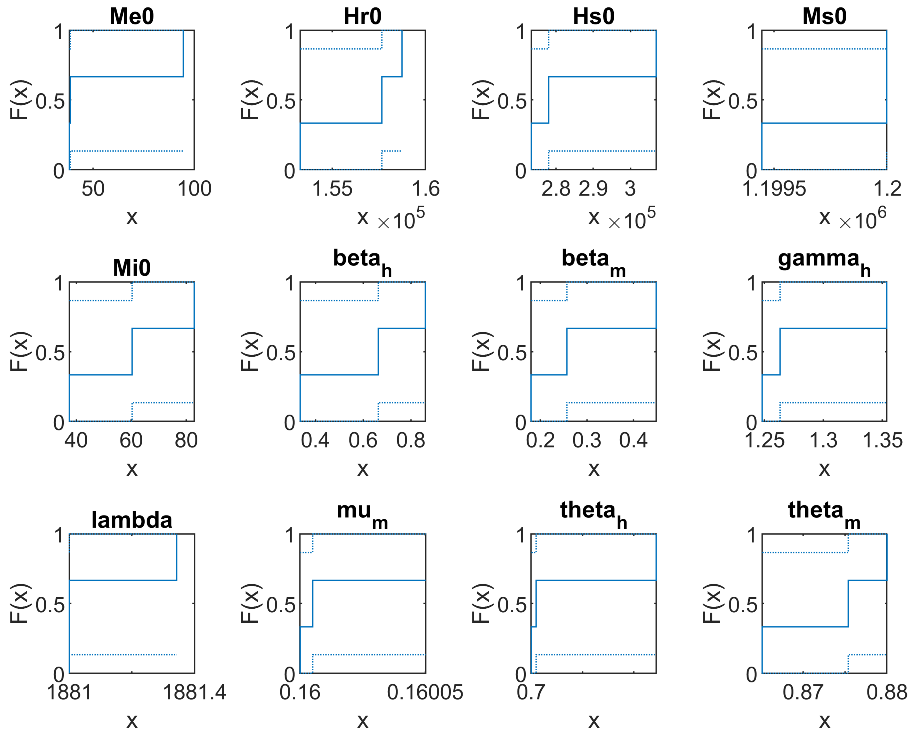

```
%savefig('iteration1/figures/EstimatedvsReal.fig')
```

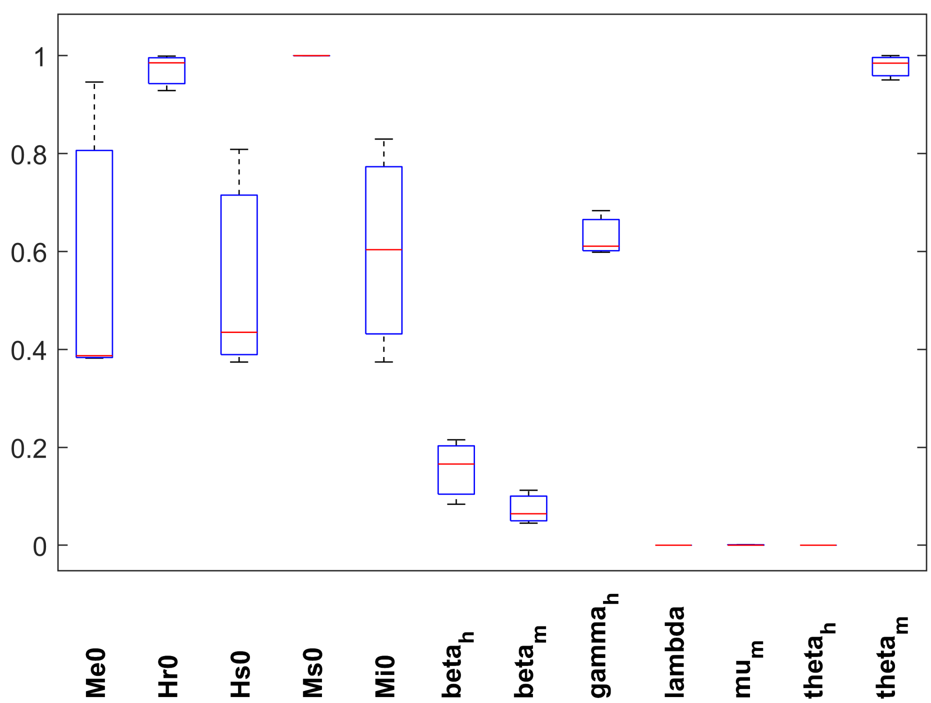
Ahora visualizamos el análisis de identificabilidad para asegurarnos de la fiabilidad del modelo ajustado.

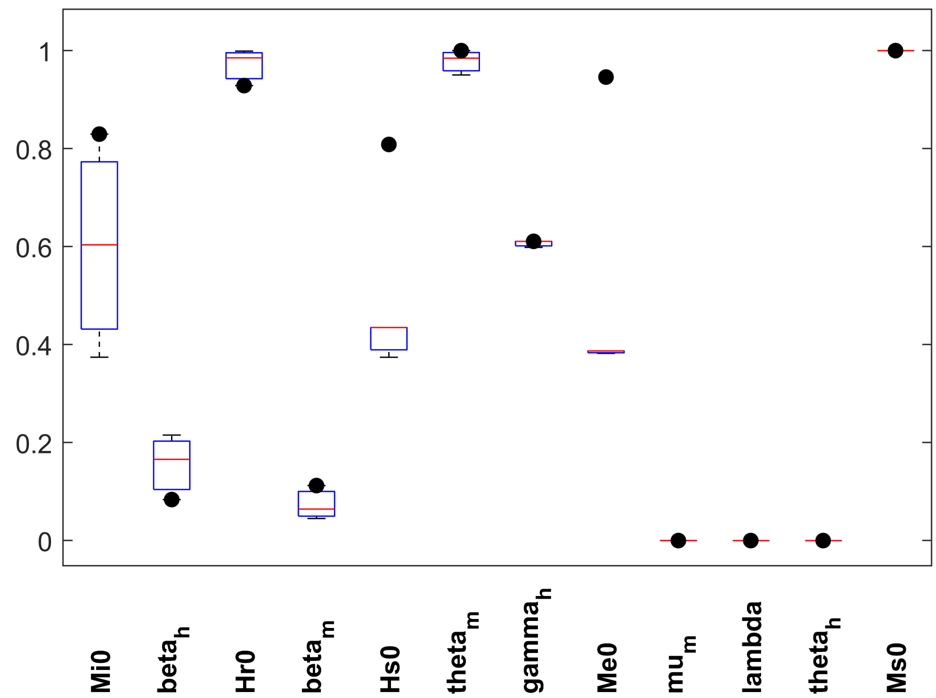
```
T7.Nominal = T7.Estlsqc(:,1);
%res : funciones de costo
th = sum(res<res(1)*1.01)
```

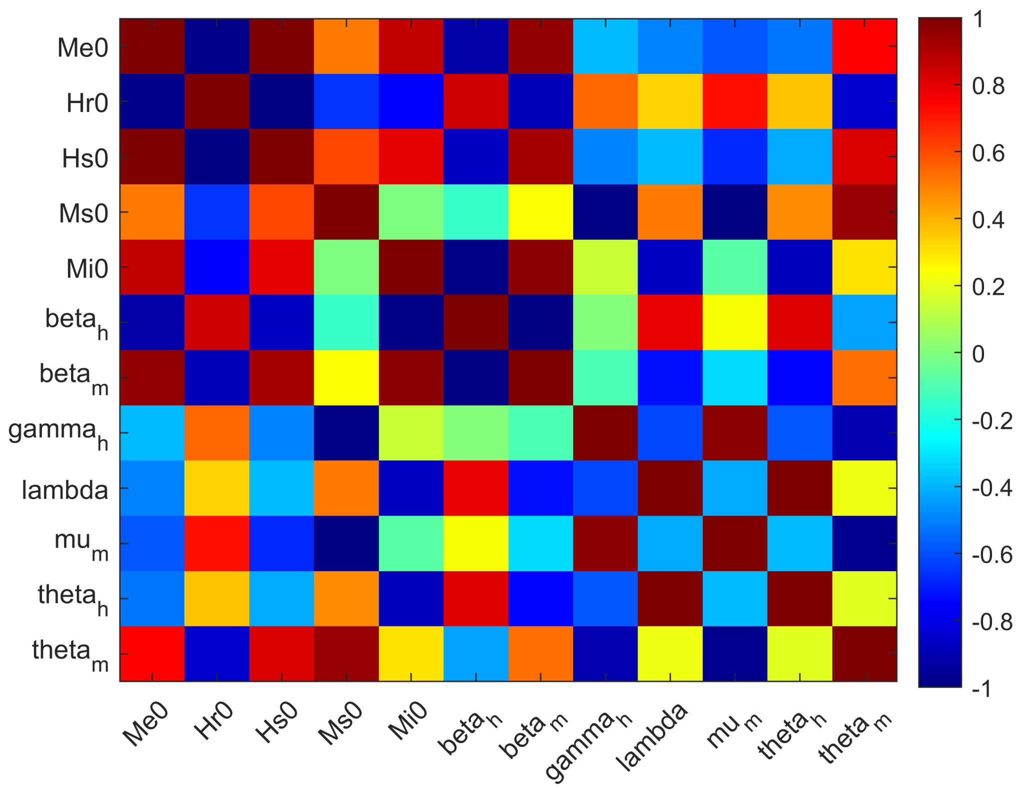
```
th = 3
```

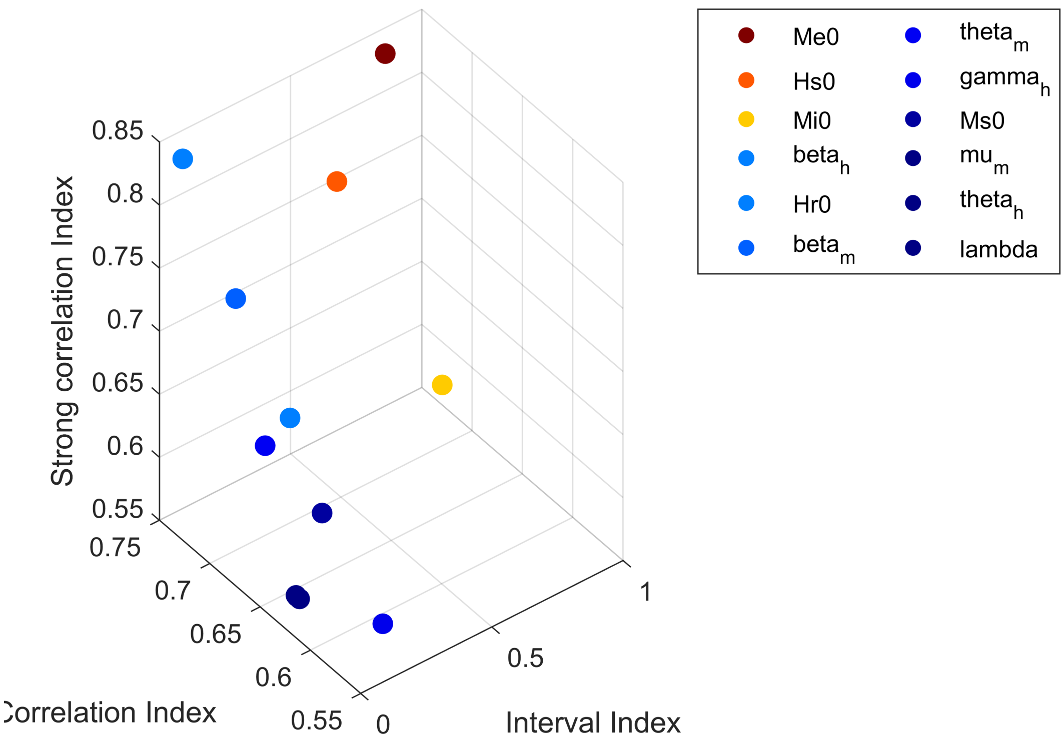
```
T7 = gsua_ia(T7,T7.Estlsqc(:,1:th), false, true);
```

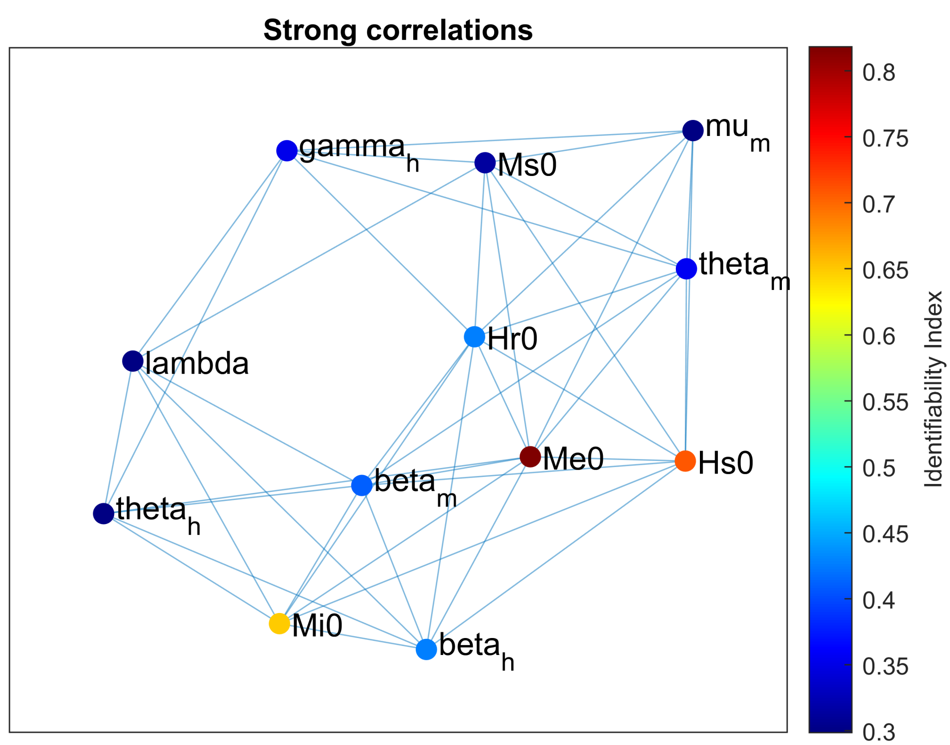

Me0
Hr0
Hs0
Ms0
Mi0
beta$_h$
beta$_m$
gamma$_h$
lambda
mu$_m$
theta$_h$
theta$_m$
F(x)
x
$\times 10^5$
$\times 10^6$


Me0
Hr0
Hs0
Ms0
Mi0
beta$_h$
beta$_m$
gamma$_h$
lambda
mu$_m$
theta$_h$
theta$_m$


1
0.8
0.6
0.4
0.2
0
Mi0
beta_h
Hr0
beta_m
Hs0
theta_m
gamma_h
Me0
mu_m
lambda
theta_h
Ms0


Me0
Hr0
Hs0
Ms0
Mi0
beta_h
beta_m
gamma_h
lambda
mu_m
theta_h
theta_m
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1





T = 11×2 table | Reduced from 12 rows

	Range		Nominal
1 Me0	81405	158809	120107

	Range		Nominal
2 Hs0	244402	321734	283068
3 Ms0	0	1200000	600000
4 Mi0	0	100	50
5 beta_h	0	4	2
6 beta_m	0	4	2
7 gamma_h	0.5000	1.7500	1.1250
8 lambda	1881	42694	2.2288e+04
9 mu_m	0.1600	0.2000	0.1800
10 theta_h	0.7000	1.7500	1.2250
11 theta_m	0.5800	0.8800	0.7300

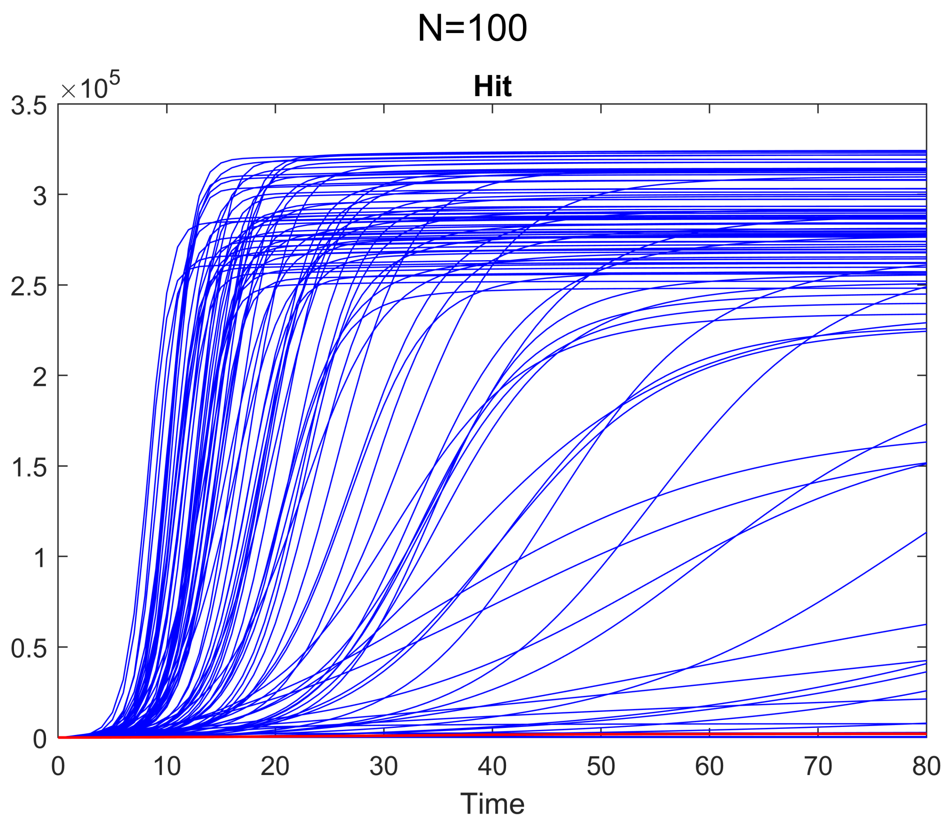
```
%savefig('iteration1/figures/CorrelationsDiag.fig')
```

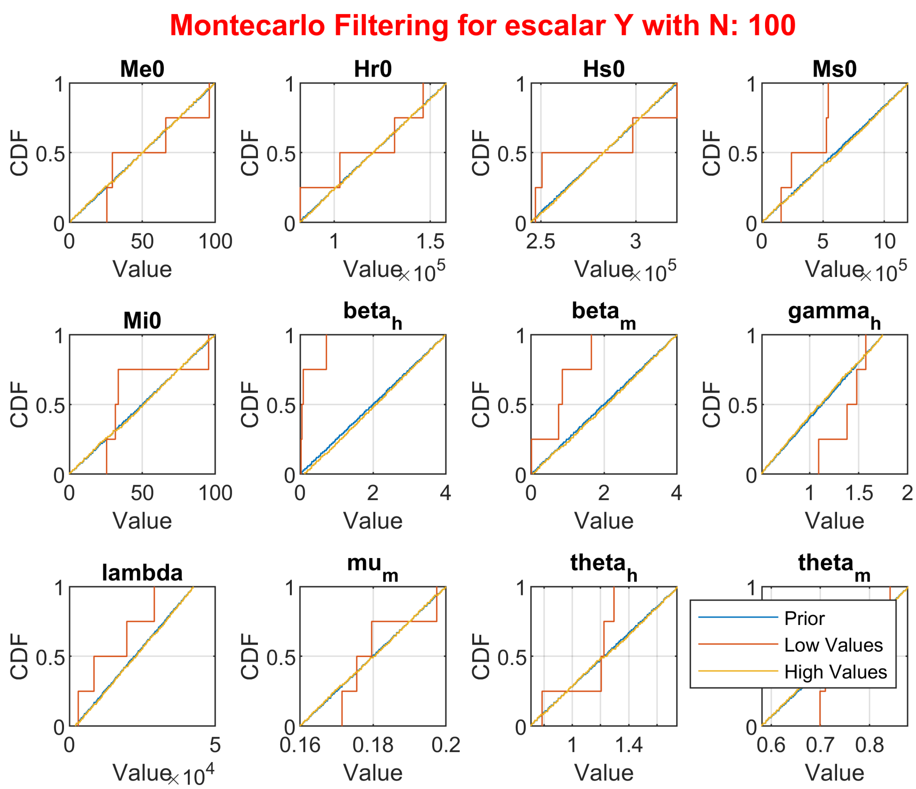
6. Análisis de incertidumbre

Veamos que las curvas hagan una banda al lado de los datos que estamos estimando. Si la banda es chevere, (visualmente) terminamos, si no, debemos ffijar algún parámetro.

```
Ua = gsua_ua(M, T7, 'parallel', false, 'ynom',ydata2);
```

```
Progress: 100%
Estimated processing time (h:m:s): 0:0:0
Remaining time (h:m:s): 0:0:0
Elapsed time (h:m:s): 0:0:0
Estimated stop time (h:m:s): 22:18:21
Number of simulations: 100
```






```
%savefig('iteration1/figures/Montecarlo.fig')
```

Segunda iteracion de los calculos

Debemos de repetir todo desde el paso 3 (el análisis de sensibilidad) pero habiendo aumentado los rangos de las

variables que se observan desde el boxplot de identificabilidad.

Aumentamos los rangos de theta_h, theta_m, lambda, mu_m, Ms0 .

```
T.Range('theta_h',1) = 0.5;
T.Range('lambda',1) = 1500;
T.Range('mu_m',1) = 0.12;
T.Range('Ms0',2) = 1.5e6;
T.Range('theta_m',2) = 1.2;
%save('iteration1/values/T.mat','T')
```

4. Estimación de parámetros

```
opt = optimoptions('lsqcurvefit','UseParallel',true,'Display','iter');
[T7_1,res_1] = gsua_pe(T,xdata,ydata2,'N',100,'opt',opt);
%save('iteration2/values/Results7.mat','T7_1','res_1','xdata','ydata2')
```

5. Análisis de identificabilidad

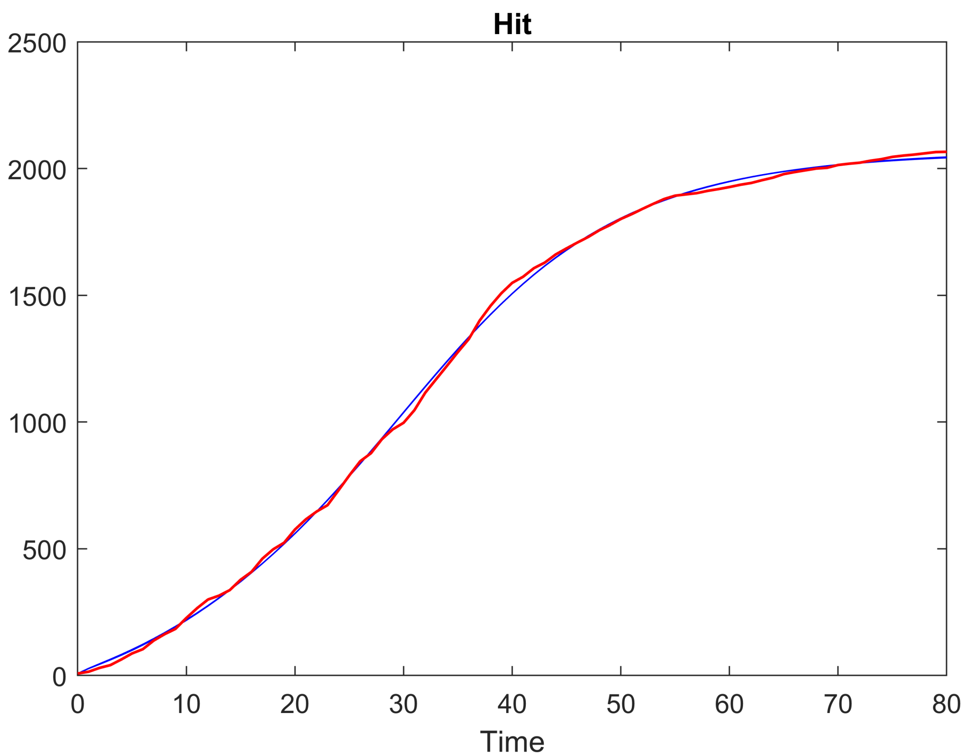
```
th_1 = sum(res_1<res_1(1)*1.01)
```

```
th_1 = 9
```

```
y7_1 = gsua_eval(T7_1.Estlsqc(:,1:th_1),T7_1,xdata,ydata2);
```

```
Sim9 Done
Sim8 Done
Sim7 Done
Sim6 Done
Sim5 Done
Sim4 Done
Sim3 Done
Sim2 Done
Sim1 Done
```

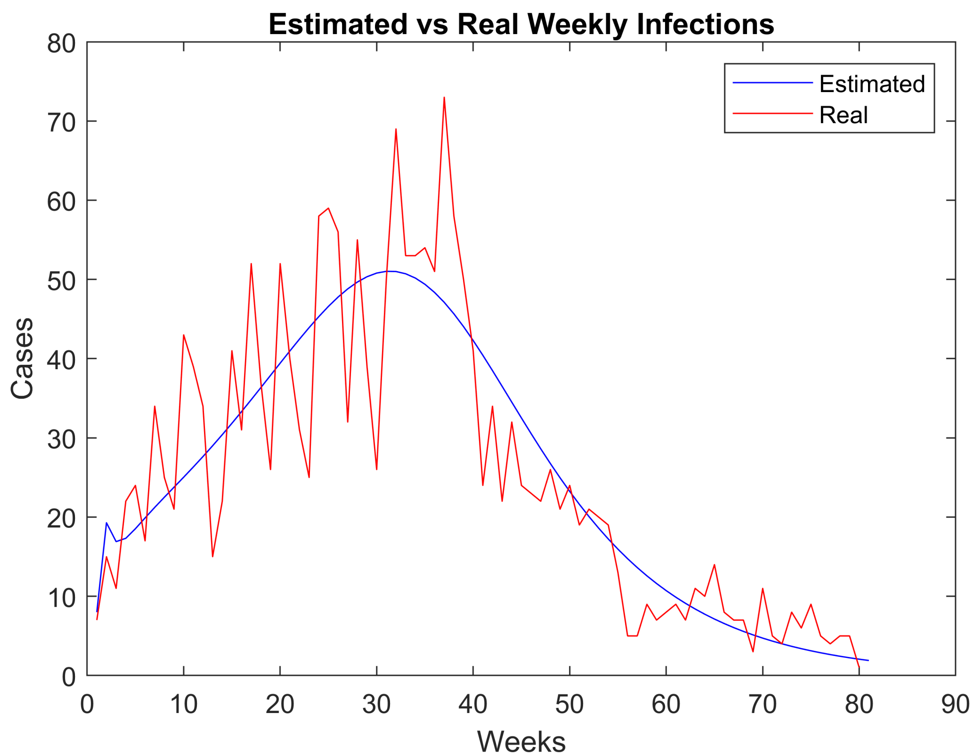
N=9




```
%savefig('iteration2/figures/curves.fig')
```

```
bestest_1 = y7_1(1,:);
trend = [bestest_1(1),bestest_1(2:end)-bestest_1(1:end-1)];
plot(trend,'b')
hold on
plot(diff(ydata2),'r')
title('Estimated vs Real Weekly Infections')
xlabel('Weeks')
ylabel('Cases')
legend({'Estimated','Real'})
```

N=9

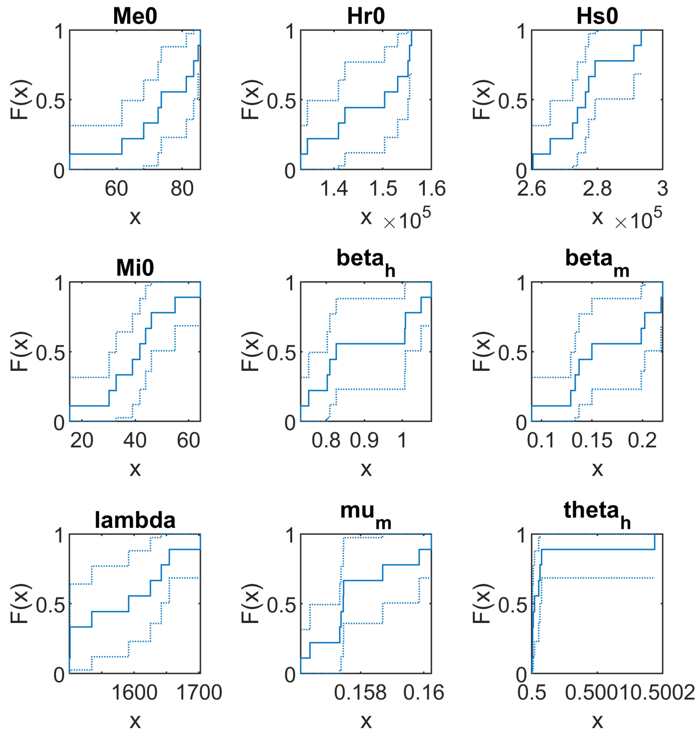



```
%savefig('iteration2/figures1/EstimatedvsReal.fig')
```

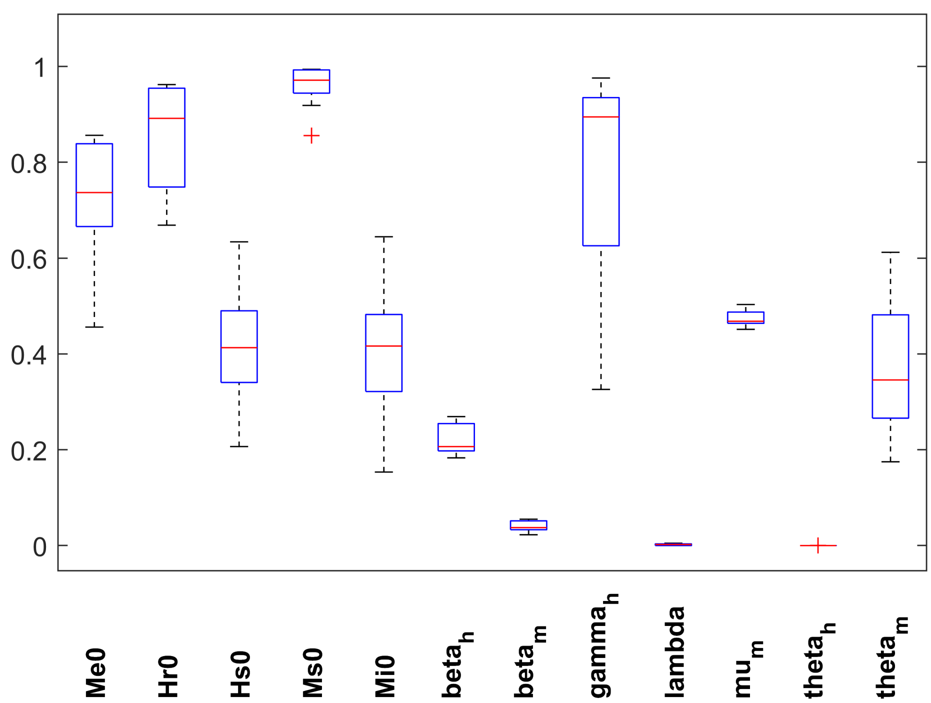
```
T7_1.Nominal = T7_1.Estlsqc(:,1);
%res : funciones de costo
th_1 = sum(res_1<res_1(1)*1.01) % threshold, cambiar el 1.01
```

```
th_1 = 9
```

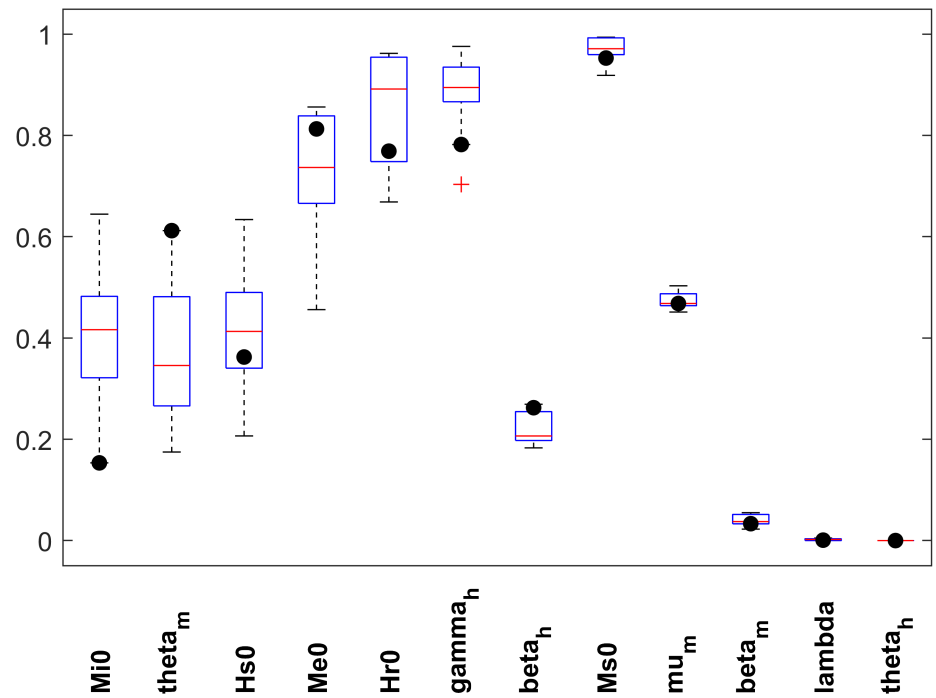
```
%th = 476 %el parametro que habia dado el profe, a nosotros nos da 1????

T7_1 = gsua_ia(T7_1,T7_1.Estlsqc(:,1:th_1), false, true);
```



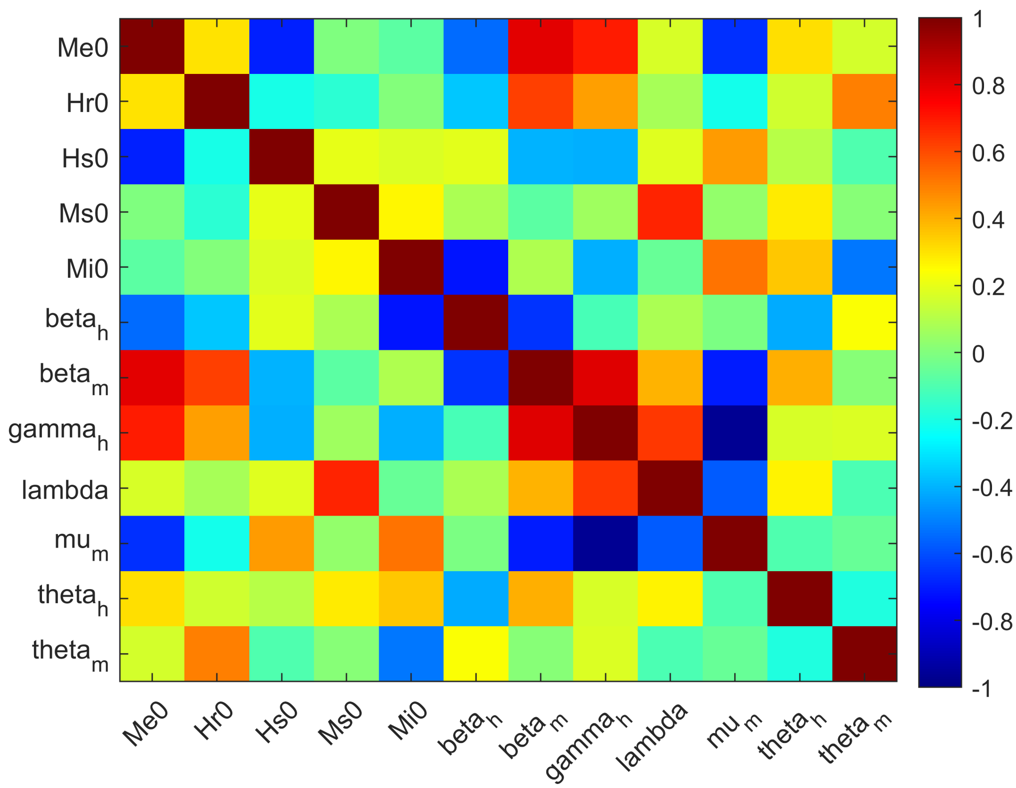
Me0
Hr0
Hs0
Ms0
Mi0
beta_h
beta_m
gamma_h
lambda
mu_m
theta_h
theta_m
F(x)
x



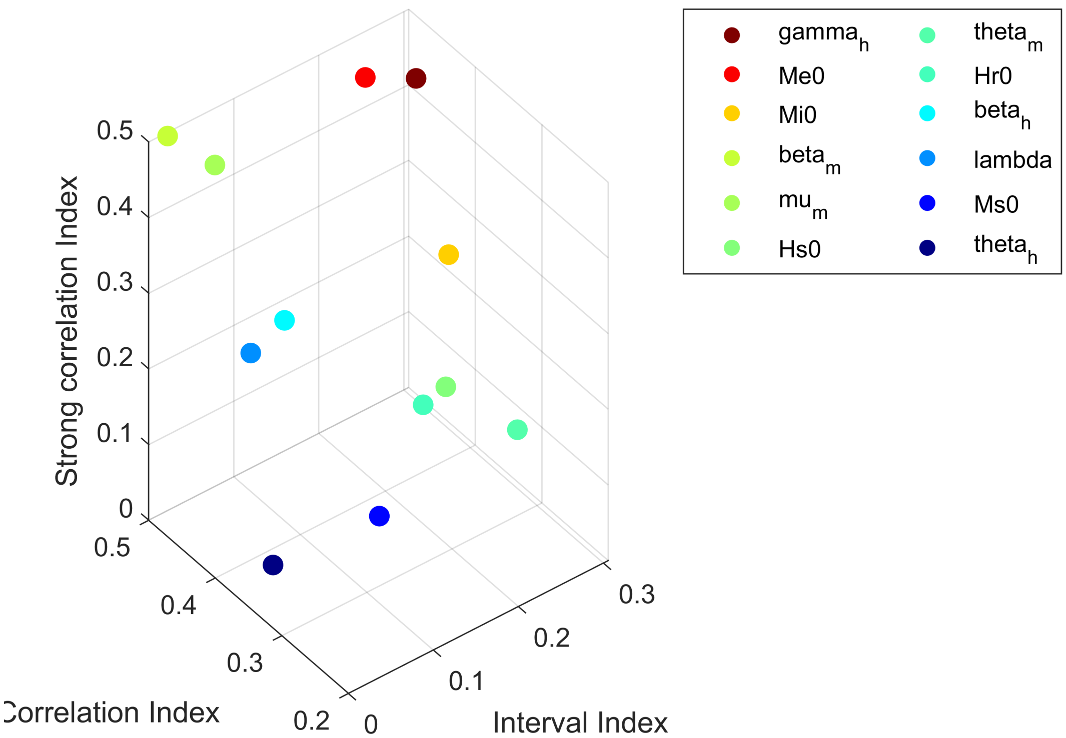
Me0
Hr0
Hs0
Ms0
Mi0
beta_h
beta_m
gamma_h
lambda
mu_m
theta_h
theta_m

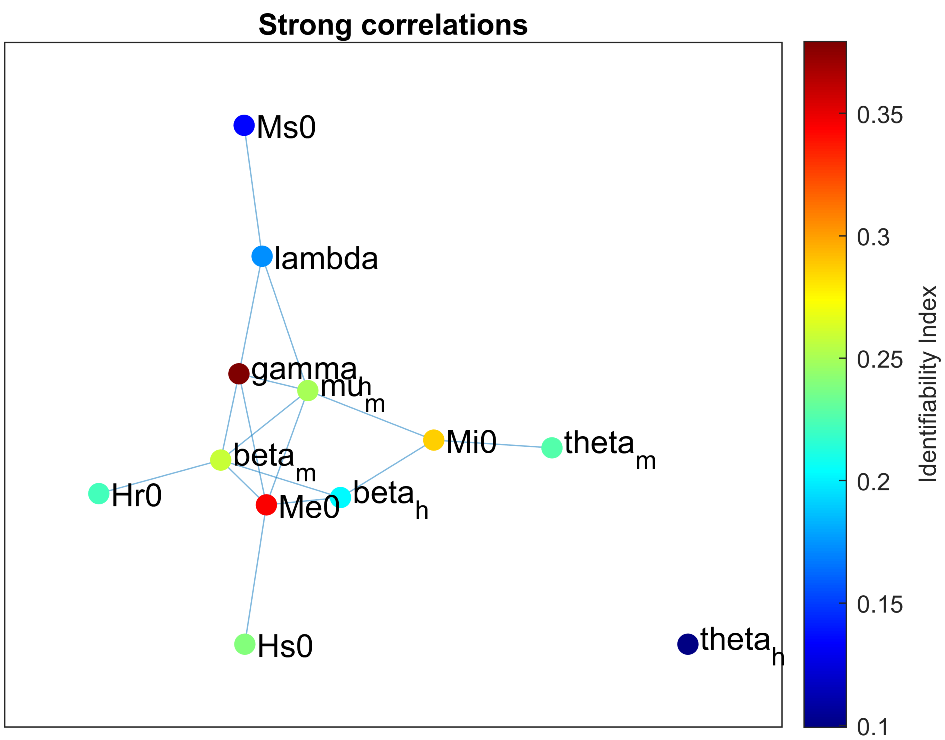


1
0.8
0.6
0.4
0.2
0
Mi0
theta_m
Hs0
Me0
Hr0
gamma_h
beta_h
Ms0
mu_m
beta_m
lambda
theta_h



Me0
Hr0
Hs0
Ms0
Mi0
beta_h
beta_m
gamma_h
lambda
mu_m
theta_h
theta_m
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1





T = 11×2 table | Reduced from 12 rows

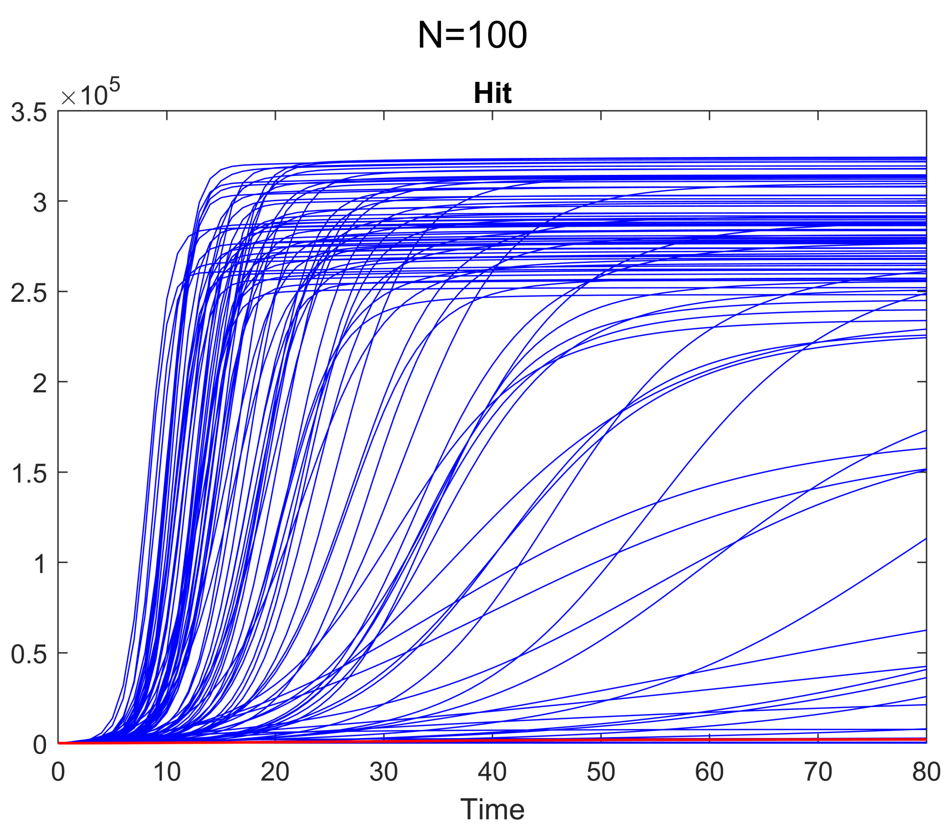
	Range		Nominal
1 Me0	81405	158809	120107

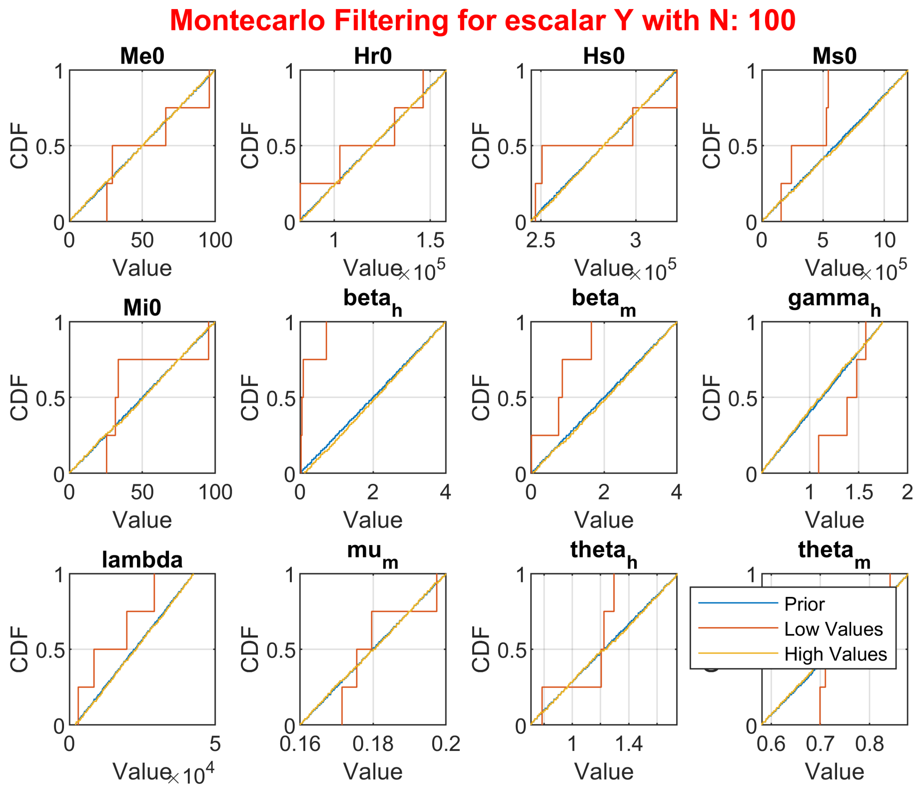
	Range		Nominal
2 Hs0	244402	321734	283068
3 Ms0	0	1200000	600000
4 Mi0	0	100	50
5 beta_h	0	4	2
6 beta_m	0	4	2
7 gamma_h	0.5000	1.7500	1.1250
8 lambda	1881	42694	2.2288e+04
9 mu_m	0.1600	0.2000	0.1800
10 theta_h	0.7000	1.7500	1.2250
11 theta_m	0.5800	0.8800	0.7300

```
%savefig('iteration2/figures/Correlations.fig')
```

```
gsua_ua(M, T7_1, 'parallel', false, 'ynom',ydata2);
```

```
Progress: 100%
Estimated processing time (h:m:s): 0:0:0
Remaining time (h:m:s): 0:0:0
Elapsed time (h:m:s): 0:0:0
Estimated stop time (h:m:s): 22:32:17
Number of simulations: 100
```





Tercera Iteracion

Vemos que todavia debemos aumentar los rangos de lambda, theta_h, Ms0 y theta_m

```
T.Range('lambda',1) = 1100;
T.Range('theta_h',1) = 0.2;
T.Range('Ms0',2) = 1.6e6;
T.Range('theta_m',2) = 1.2;
```

4. Estimamos los parámetros

```
opt = optimoptions('lsqcurvefit','UseParallel',false,'Display','iter');
[T7_2,res_2] = gsua_pe(T,xdata,ydata2,'N',100,'opt',opt);
%save('iteration3/values/Results7.mat','T7_2','res_2','xdata','ydata2')
```

5. Análisis de identificabilidad

```
th_2 = sum(res_2<res_2(1)*1.03)
```

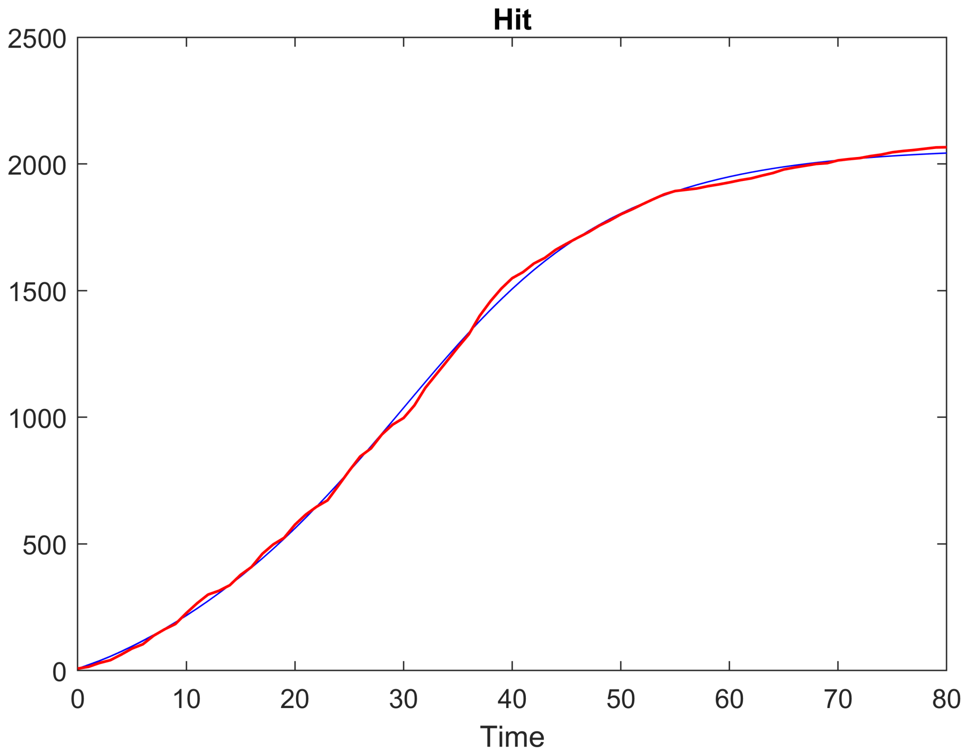
```
th_2 = 10
```

```
y7_2 = gsua_eval(T7_2.Estlsqc(:,1:th_2),T7_2,xdata,ydata2);
```

```
Sim10 Done
Sim9 Done
Sim8 Done
```

```
Sim7 Done
Sim6 Done
Sim5 Done
Sim4 Done
Sim3 Done
Sim2 Done
Sim1 Done
```

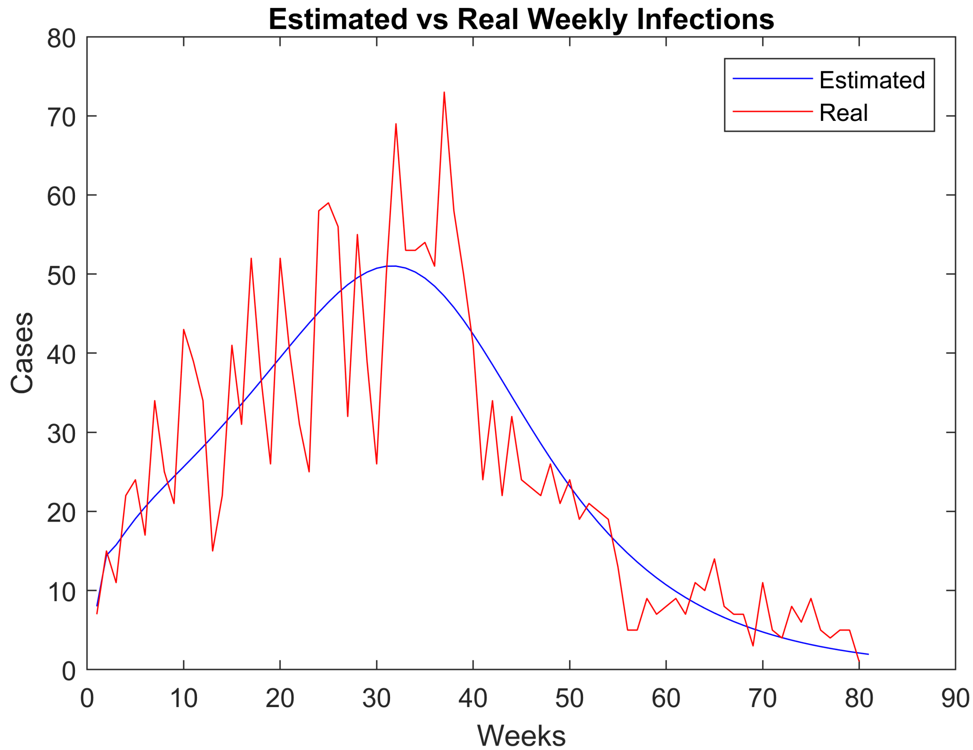
N=10




```
%savefig('iteration3/figures/curves.fig')
```

```
bestest_2 = y7_2(1,:);
trend = [bestest_2(1),bestest_2(2:end)-bestest_2(1:end-1)];
plot(trend,'b')
hold on
plot(diff(ydata2),'r')
title('Estimated vs Real Weekly Infections')
xlabel('Weeks')
ylabel('Cases')
legend({'Estimated','Real'})
```

N=10

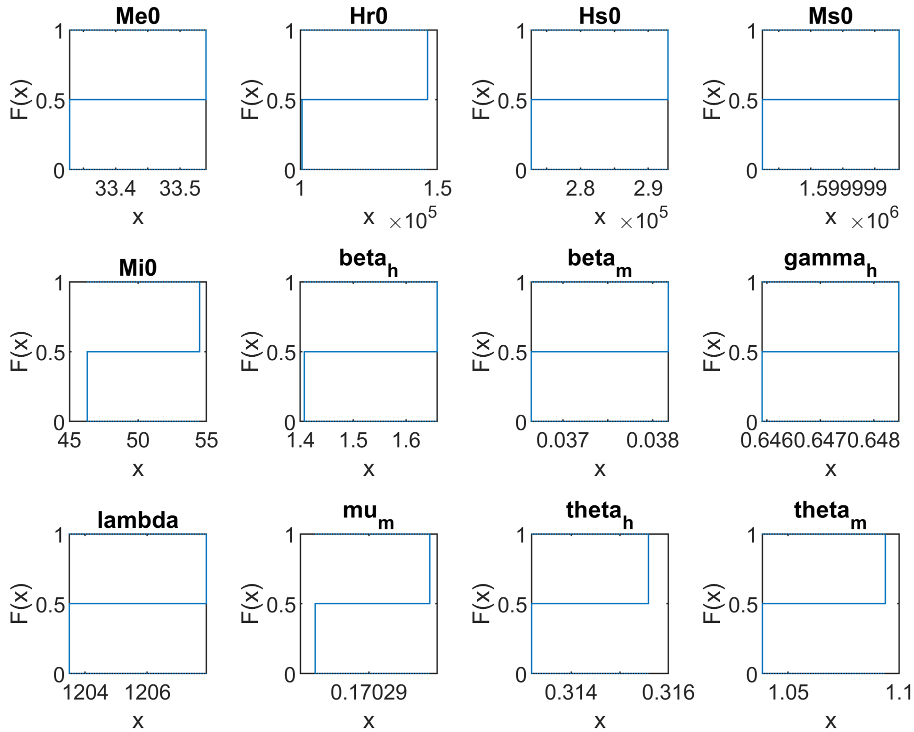


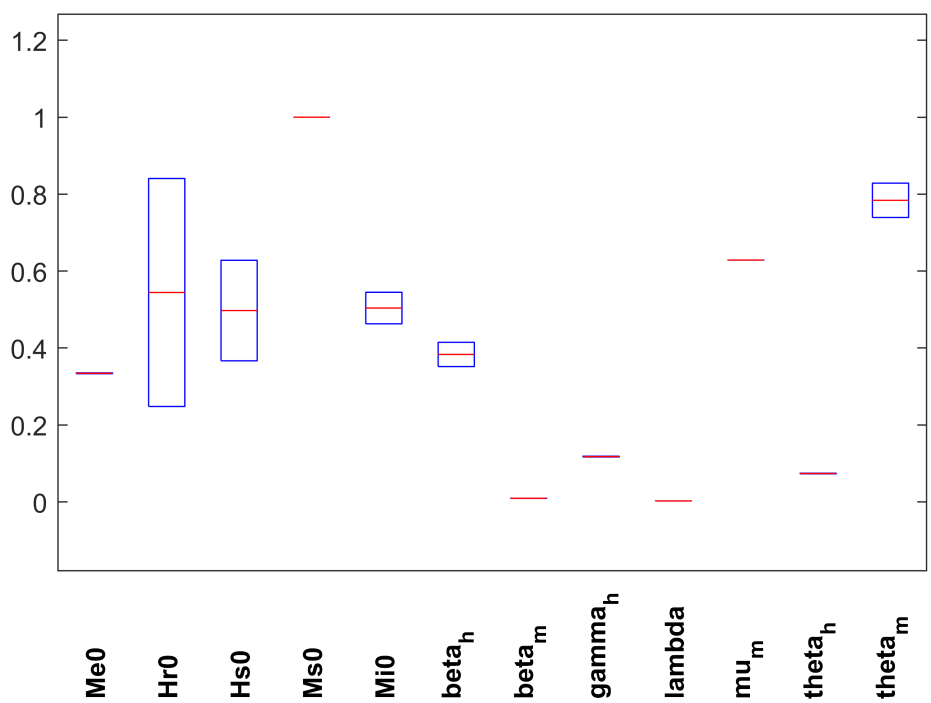

```
%savefig('iteration3/figures1/EstimatedvsReal.fig')
```

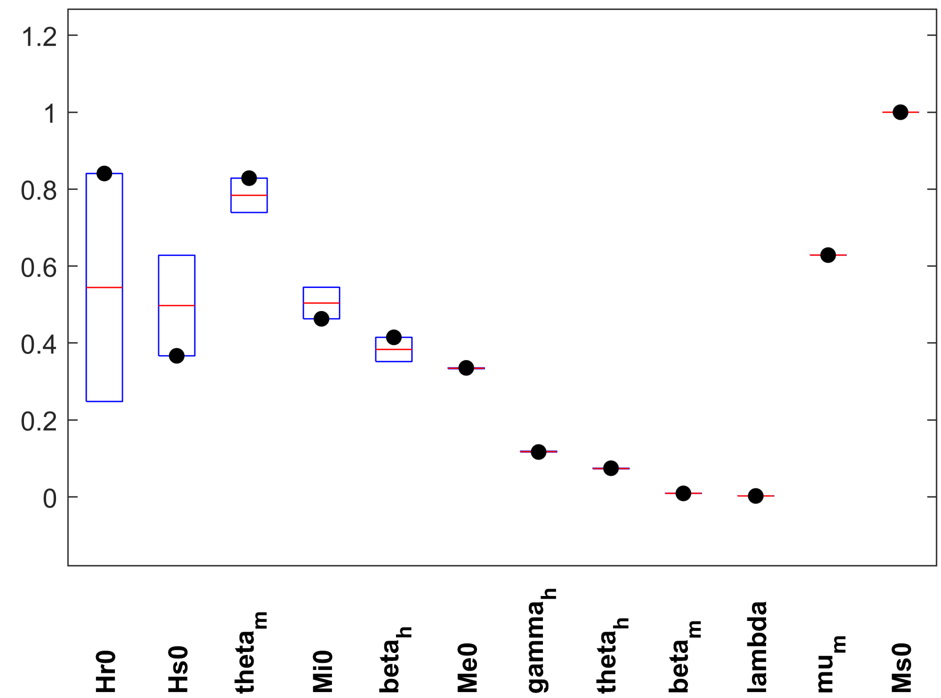
```
T7_2.Nominal = T7_2.Estlsqc(:,1);
th_2 = sum(res_2<res_2(1)*1.01)
```

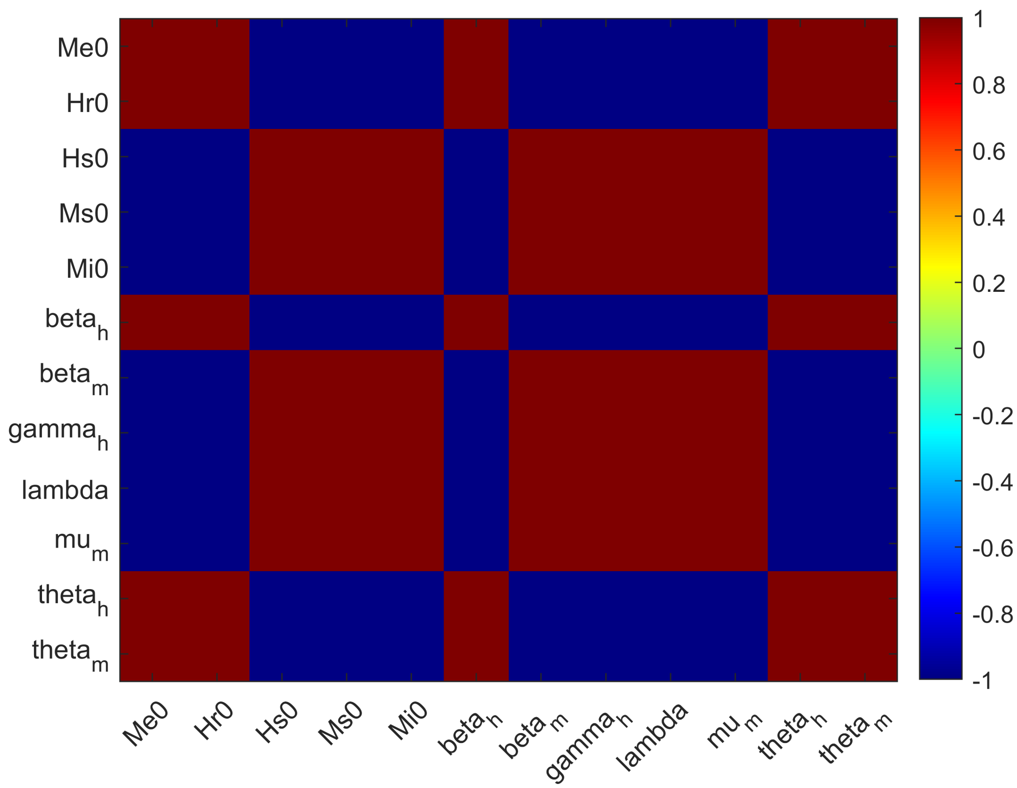
```
th_2 = 2
```

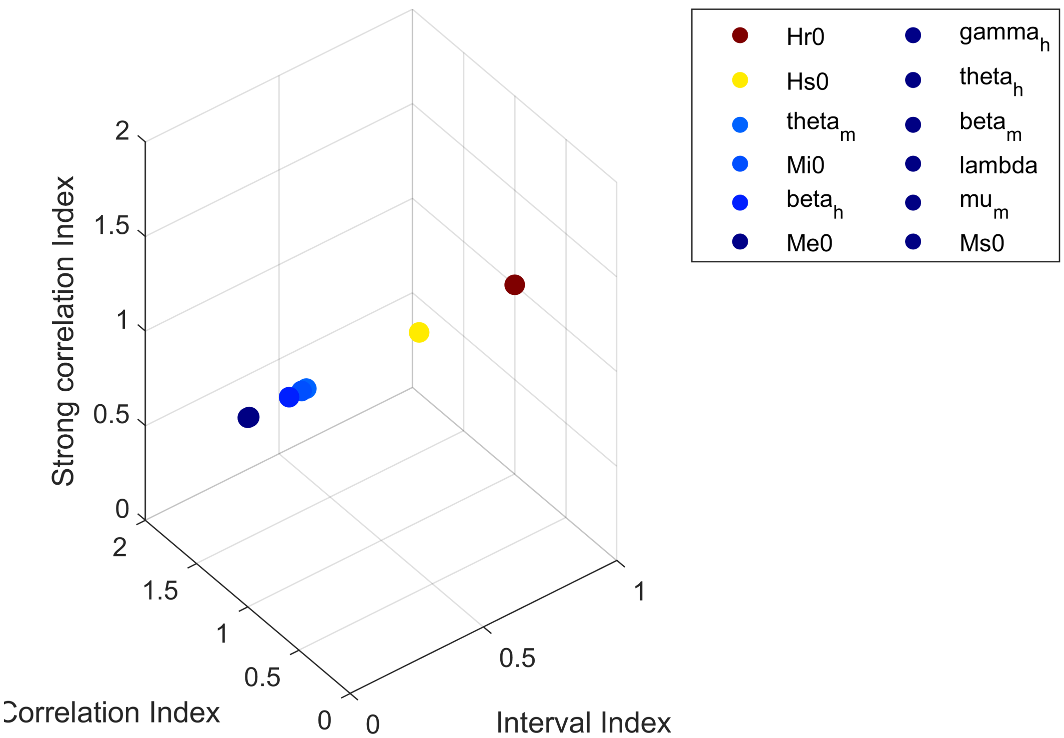
```
T7_2 = gsua_ia(T7_2,T7_2.Estlsqc(:,1:th_2), false, true);
```

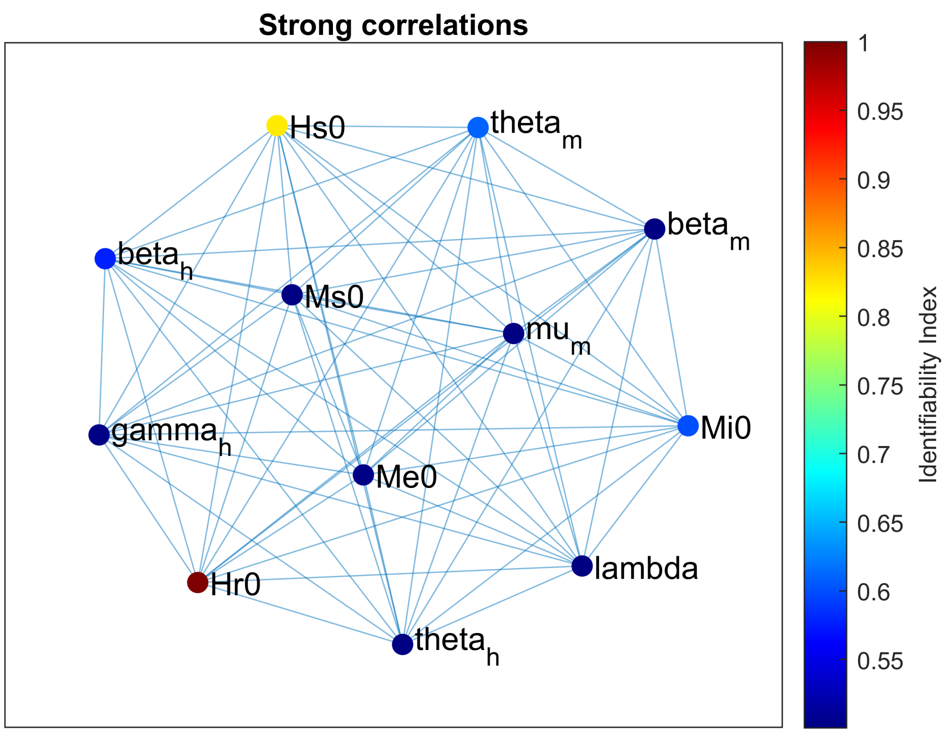

Me0
Hr0
Hs0
Ms0
Mi0
beta$_h$
beta$_m$
gamma$_h$
lambda
mu$_m$
theta$_h$
theta$_m$
F(x)
x
33.4
33.5
$\times10^5$
$\times10^6$
1.599999
0.17029


Me0
Hr0
Hs0
Ms0
Mi0
beta$_h$
beta$_m$
gamma$_h$
lambda
mu$_m$
theta$_h$
theta$_m$


1.2
1
0.8
0.6
0.4
0.2
0
Hr0
Hs0
theta_m
Mi0
beta_h
Me0
gamma_h
theta_h
beta_m
lambda
mu_m
Ms0


Me0
Hr0
Hs0
Ms0
Mi0
beta_h
beta_m
gamma_h
lambda
mu_m
theta_h
theta_m
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1





T = 11×2 table | Reduced from 12 rows

	Range		Nominal
1 Me0	81405	158809	120107

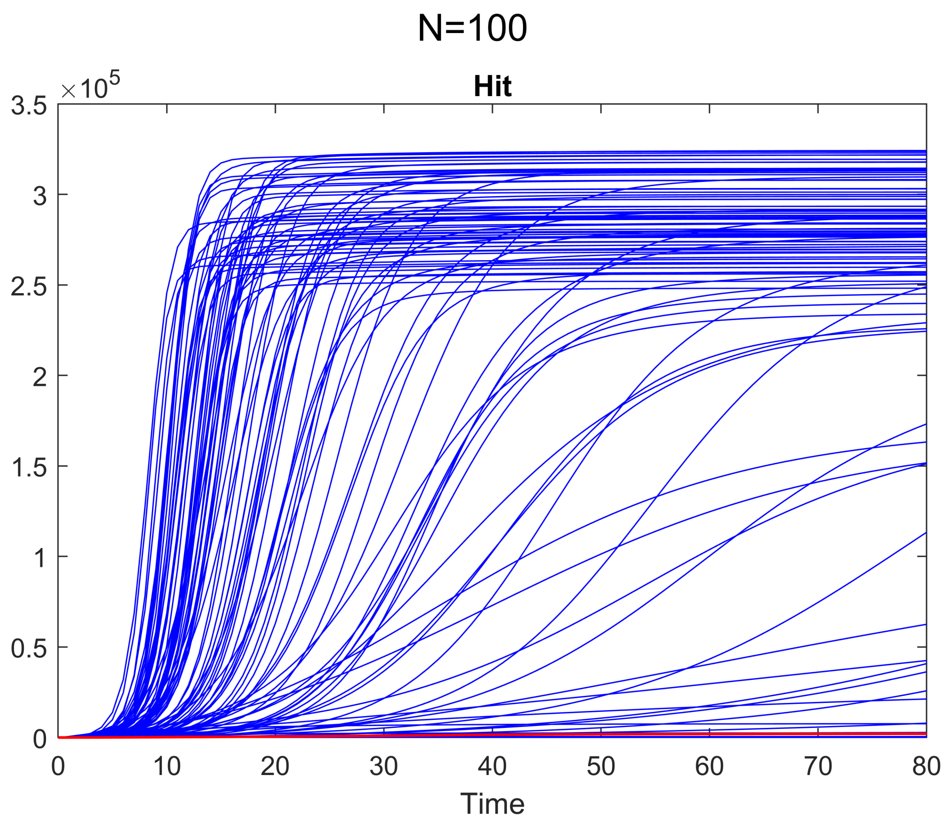
	Range		Nominal
2 Hs0	244402	321734	283068
3 Ms0	0	1200000	600000
4 Mi0	0	100	50
5 beta_h	0	4	2
6 beta_m	0	4	2
7 gamma_h	0.5000	1.7500	1.1250
8 lambda	1881	42694	2.2288e+04
9 mu_m	0.1600	0.2000	0.1800
10 theta_h	0.7000	1.7500	1.2250
11 theta_m	0.5800	0.8800	0.7300

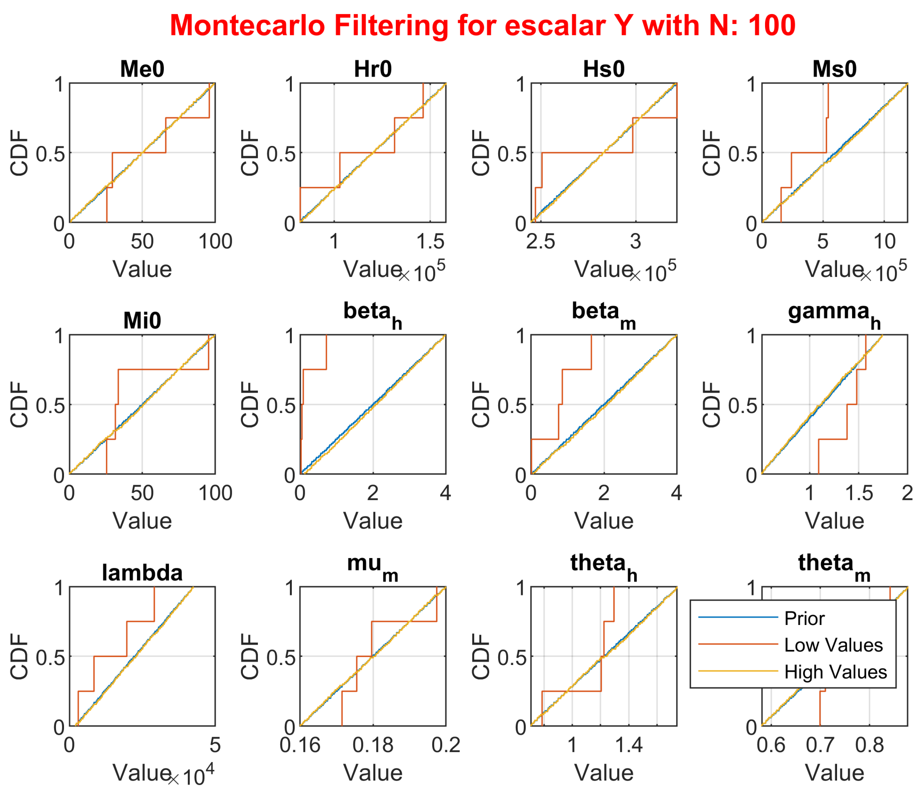
```
%savefig('iteration3/figures/Correlations.fig')
```

6. Análisis de incertidumbre

```
gsua_ua(M, T7_2, 'parallel', false, 'ynom',ydata2);
```

```
Progress: 100%
Estimated processing time (h:m:s): 0:0:0
Remaining time (h:m:s): 0:0:0
Elapsed time (h:m:s): 0:0:0
Estimated stop time (h:m:s): 22:51:15
Number of simulations: 100
```






```
%savefig('iteration3/figures/Montecarlo.fig
```

Cuarta Iteracion

Fijamos Hr) dado que este esta muy correlacionado a las otras variables

```
vars=[Hit Hi He Hr Me Hs Ms Mi];

HrO = T7_2.Estlsqc('Hr0',1);

RangeTemp = Range;
Range(4,:) = HeO;

[T,~]=gsua_dpmat(odes,vars,[0 80],'7m','output',1,'opt',opts,'Range',Range);
```

```
Introduce ranges in the following order:
```

ans(t) = $\left(\mathrm{Hit}(t) \quad \mathrm{Hi}(t) \quad \mathrm{He}(t) \quad \mathrm{Hr}(t) \quad \mathrm{Me}(t) \quad \mathrm{Hs}(t) \quad \mathrm{Ms}(t) \quad \mathrm{Mi}(t) \quad \beta_h \quad \beta_m \quad \gamma_h \quad \lambda \quad \mu_h \quad \mu_m \quad \theta_h \quad \theta_m\right)$

```
T.Properties.CustomProperties.output = 1;
M = gsua_dmatrix(T,100);
```

4. Estimamos los parámetros

```
opt = optimoptions('lsqcurvefit','UseParallel',true,'Display','iter');
[T7_3,res_3] = gsua_pe(T,xdata,ydata2,'N',100,'opt',opt);
%save('iteration3/values/Results7.mat','T7_2','res_2','xdata','ydata2')
```

5. Análisis de identificabilidad

```
th_3 = sum(res_3<res_3(1)*1.01)
```

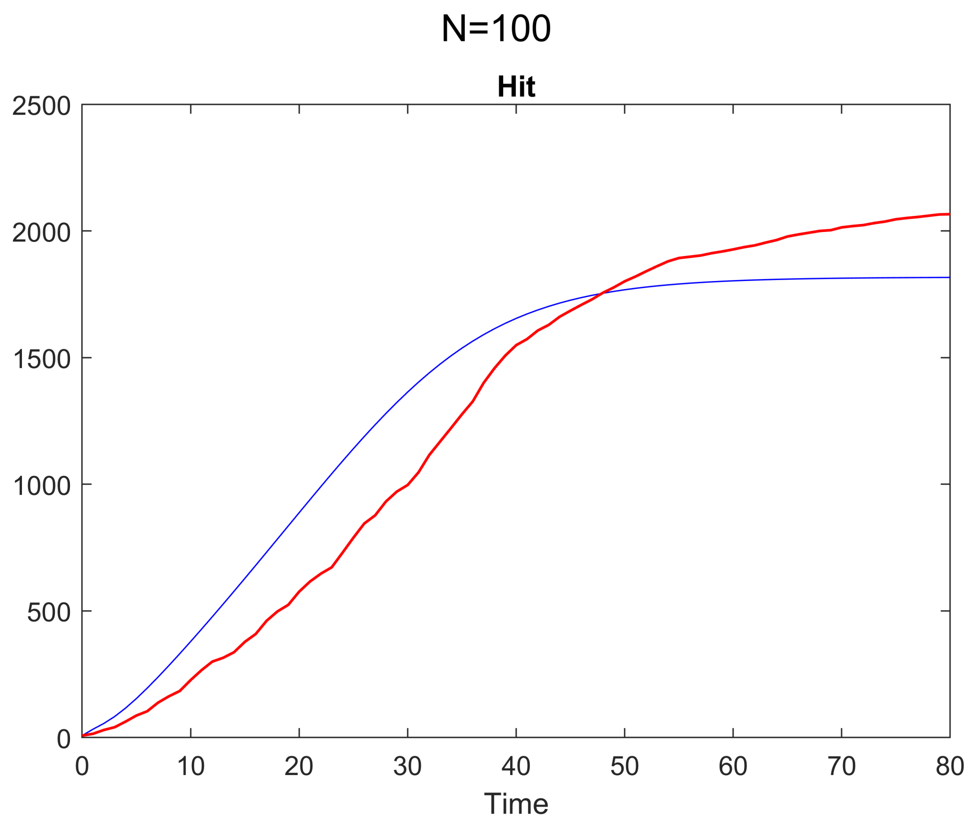
```
th_3 = 100
```

```
y7_3 = gsua_eval(T7_3.Estlsqc(:,1:th_3),T7_3,xdata,ydata2);
```

```
Sim100 Done
Sim99 Done
Sim98 Done
Sim97 Done
Sim96 Done
Sim95 Done
Sim94 Done
Sim93 Done
Sim92 Done
Sim91 Done
Sim90 Done
Sim89 Done
Sim88 Done
Sim87 Done
Sim86 Done
Sim85 Done
Sim84 Done
Sim83 Done
Sim82 Done
Sim81 Done
Sim80 Done
Sim79 Done
```

```
Sim78 Done
Sim77 Done
Sim76 Done
Sim75 Done
Sim74 Done
Sim73 Done
Sim72 Done
Sim71 Done
Sim70 Done
Sim69 Done
Sim68 Done
Sim67 Done
Sim66 Done
Sim65 Done
Sim64 Done
Sim63 Done
Sim62 Done
Sim61 Done
Sim60 Done
Sim59 Done
Sim58 Done
Sim57 Done
Sim56 Done
Sim55 Done
Sim54 Done
Sim53 Done
Sim52 Done
Sim51 Done
Sim50 Done
Sim49 Done
Sim48 Done
Sim47 Done
Sim46 Done
Sim45 Done
Sim44 Done
Sim43 Done
Sim42 Done
Sim41 Done
Sim40 Done
Sim39 Done
Sim38 Done
Sim37 Done
Sim36 Done
Sim35 Done
Sim34 Done
Sim33 Done
Sim32 Done
Sim31 Done
Sim30 Done
Sim29 Done
Sim28 Done
Sim27 Done
Sim26 Done
Sim25 Done
Sim24 Done
Sim23 Done
Sim22 Done
Sim21 Done
Sim20 Done
Sim19 Done
Sim18 Done
Sim17 Done
Sim16 Done
Sim15 Done
```

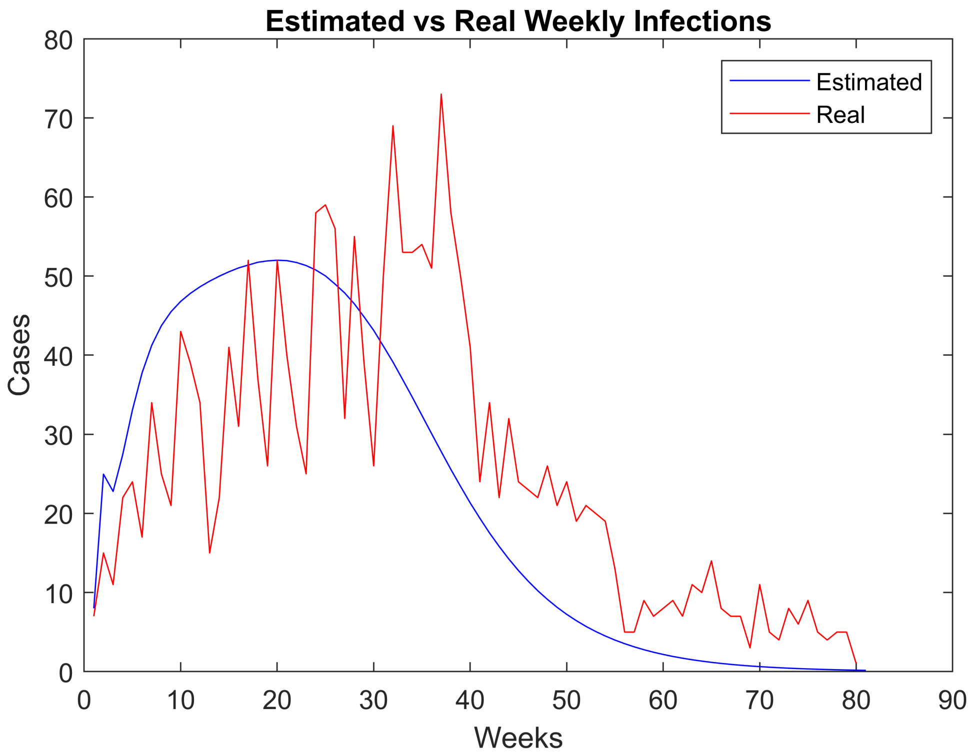
```
Sim14 Done
Sim13 Done
Sim12 Done
Sim11 Done
Sim10 Done
Sim9 Done
Sim8 Done
Sim7 Done
Sim6 Done
Sim5 Done
Sim4 Done
Sim3 Done
Sim2 Done
Sim1 Done
```




```
%savefig('iteration3/figures/curves.fig')
```

```
bestest_3 = y7_3(1,:);
trend = [bestest_3(1),bestest_3(2:end)-bestest_3(1:end-1)];
plot(trend,'b')
hold on
plot(diff(ydata2),'r')
title('Estimated vs Real Weekly Infections')
xlabel('Weeks')
ylabel('Cases')
legend({'Estimated','Real'})
```

N=100




```
%savefig('iteration3/figures1/EstimatedvsReal.fig')
```

Al fijar Hr las estimaciones dejan de funcionar y el modelo deja de ajustar correctamente, esto tiene multiples distintas razones por las que puede ser, no s eampliaron los rangos lo suficiente en las iteraciones anteriores, el parametro que se fijo, tenia mucha incertidumbre sobre su valor, esto se observa en el boxplot de la iteracion anterior, donde Hr, es el parametro con mayor caja, por lo que aunque afectaba mucho a los otros parametros, habia muy poca certeza sobre su valor real. O que posiblemente el threshold que se tomo fue muy restrictivo, lo cual afecto mucho los analisis de identificabilidad, por tanto se termino eligiendo el parametro incorrecto, con un valor posiblemente inadecuado. Tambien se observa como al realizar el analisis de identificabilidad, el parametro mu quedo con un rango basicamente nulo, por lo que esto tambien pudo haber afectado el analisis y la estimacion de parametros en la siguiente iteracion.