

Example: Iterative fitting for multiple provinces in China (22-Jan-2020 -)

Table of Contents

Initialisation.....	1
Iterative application of fit_SEIQRDP.....	2

In this example, the generalized SEIR model is automatically fitted to multiple provinces in China. As in the previous examples, I am taking some data, collected into DATA.mat from John Hopkins university [1]. To keep the computation as short as possible, the option "iter" is removed using an optional argument.

[1] <https://github.com/CSSEGISandData/COVID-19>

Initialisation

The parameters are here taken as constant except the death rate and the cure rate.

```
clearvars;close all;clc;
clearvars;close all;clc;
% Download the data from ref [1] and read them with the function getDataCOVID
[tableConfirmed,tableDeaths,tableRecovered,time] = getDataCOVID();
% time = time(1:end-1);
fprintf(['Most recent update: ',datestr(time(end)),'\n'])
```

Most recent update: 13-Apr-2020

```
Location = 'China';
```

```
try
    indR = find(contains(tableRecovered.CountryRegion,Location)==1);
    indC = find(contains(tableConfirmed.CountryRegion,Location)==1);
    indD = find(contains(tableDeaths.CountryRegion,Location)==1);
catch exception
    searchLoc = strfind(tableRecovered.CountryRegion,Location);
    indR = find(~cellfun(@isempty,searchLoc)) ;

    searchLoc = strfind(tableConfirmed.CountryRegion,Location);
    indC = find(~cellfun(@isempty,searchLoc)) ;

    searchLoc = strfind(tableDeaths.CountryRegion,Location);
    indD = find(~cellfun(@isempty,searchLoc));
end

% disp(tableRecovered(indR,1:2))
disp(tableConfirmed(indC,1:2))
```

ProvinceState	CountryRegion
---------------	---------------

"Anhui"	"China"
"Beijing"	"China"
"Chongqing"	"China"
"Fujian"	"China"
"Gansu"	"China"
"Guangdong"	"China"
"Guangxi"	"China"
"Guizhou"	"China"
"Hainan"	"China"
"Hebei"	"China"
"Heilongjiang"	"China"
"Henan"	"China"
"Hong Kong"	"China"
"Hubei"	"China"
"Hunan"	"China"
"Inner Mongolia"	"China"
"Jiangsu"	"China"
"Jiangxi"	"China"
"Jilin"	"China"
"Liaoning"	"China"
"Macau"	"China"
"Ningxia"	"China"
"Qinghai"	"China"
"Shaanxi"	"China"
"Shandong"	"China"
"Shanghai"	"China"
"Shanxi"	"China"
"Sichuan"	"China"
"Tianjin"	"China"
"Tibet"	"China"
"Xinjiang"	"China"
"Yunnan"	"China"
"Zhejiang"	"China"

```
% disp(tableDeaths(indD,1:2))

% If the number of confirmed Confirmed cases is small, it is difficult to know whether
% the quarantine has been rigorously applied or not. In addition, this
% suggests that the number of infectious is much larger than the number of
% confirmed cases
```

Iterative application of fit_SEIQRDP

```
timeRef = time; % Used in the loop only

for ii = 1:min([numel(indR),numel(indC),numel(indD)])
    Recovered = table2array(tableRecovered(indR(ii),5:end));
    Deaths = table2array(tableDeaths(indD(ii),5:end));
    Confirmed = table2array(tableConfirmed(indC(ii),5:end));
    minNum= max(50,round(0.025*max(Confirmed)));
    % Warning: a dummy value of Npop is used here.
    Npop= 30e6; % population (It affects the values of the parameters)

    % Remove case where only few infectious are recorded (to avoid bad
```

```

% initial conditions)
Recovered(Confirmed<=minNum)=[];
Deaths(Confirmed<=minNum)=[];
time = timeRef; % trick to avoid reloading the variable "time" at each new loop
time(Confirmed<=minNum)= [];
Confirmed(Confirmed<=minNum)=[];

% The fitting is only applied if enough data is collected (that is why
% I use the case of China)
if numel(Confirmed)>30 % If more than 30 days of data, run the fit
    tic

    % Definition of the first estimates for the parameters
    alpha_guess = 0.06; % protection rate
    beta_guess = 0.8; % Infection rate
    LT_guess = 5; % latent time in days
    Q_guess = 0.5; % rate at which infectious people enter in quarantine
    lambda_guess = [0.1,0.05]; % recovery rate
    kappa_guess = [0.1,0.05]; % death rate

    guess = [alpha_guess,beta_guess,1/LT_guess, Q_guess,lambda_guess,kappa_guess];

    % Initial conditions
    E0 = Confirmed(1); % Initial number of exposed cases. Unknown but unlikely to b
    I0 = Confirmed(1); % Initial number of infectious cases. Unknown but unlikely t
    Q0 = Confirmed(1);
    R0 = Recovered(1);
    D0 = Deaths(1);

    Active = Confirmed-Recovered-Deaths;
    Active(Active<0) = 0; % No negative number possible

    [alpha1,beta1,gamma1,delta1,Lambda1,Kappa1] = ...
        fit_SEIQRDP(Active,Recovered,Deaths,Npop,E0,I0,time,guess,'Display','off');

    dt = 0.1; % time step
    timel = datetime(time(1)):dt:datetime(datestr(floor(datenum(now))+datenum(10)))
    N = numel(timel);
    t = [0:N-1].*dt;

    % Call of the function SEIQRDP.m with the fitted parameters
    [S,E,I,Q,R,D,P] = SEIQRDP(alpha1,beta1,gamma1,delta1,Lambda1,Kappa1,Npop,E0,I0,

    clf;close all;
    figure

    semilogy(timel,Q,'r',timel,R,'b',timel,D,'k');
    hold on
    semilogy(time,Active,'ro',time,Recovered,'bo',time,Deaths,'ko');
    % ylim([0,1.1*Npop])
    ylabel('Number of cases')
    xlabel('time (days)')

```

```

leg = {'Confirmed (fitted)',...
'Recovered (fitted)', 'Deceased (fitted)',...
'Confirmed (reported)', 'Recovered (reported)', 'Deceased (reported)'};
legend(leg{:}, 'location', 'southoutside');
set(gcf, 'color', 'w')

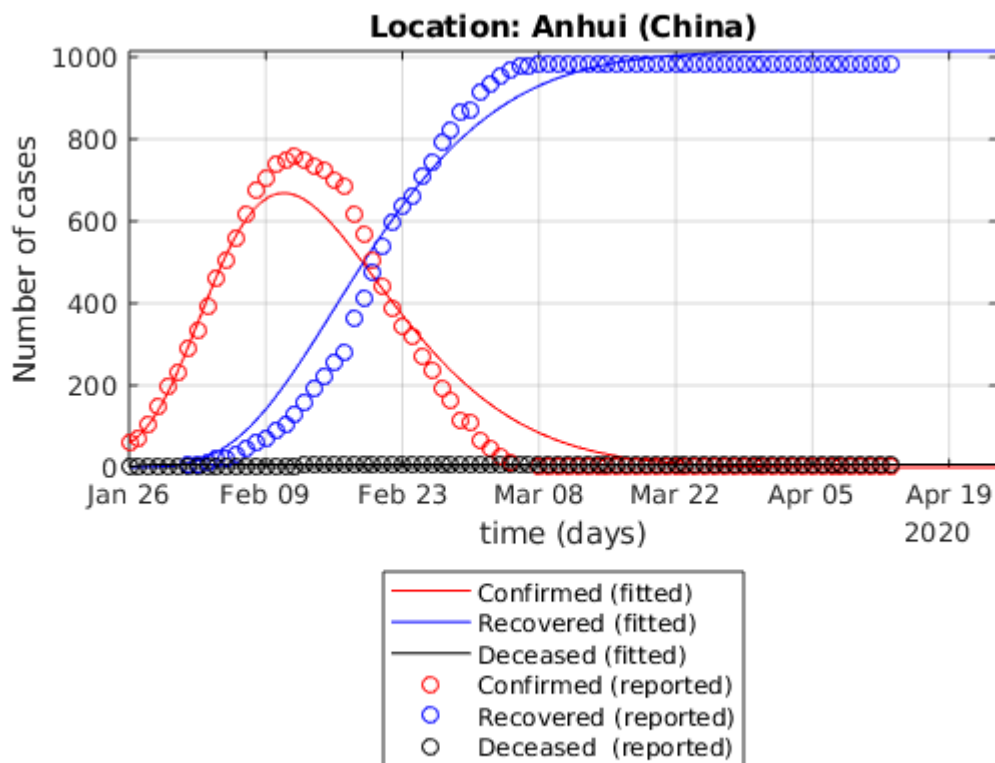
%%% title %%%
subLoc = char(table2array(tableRecovered(indR(ii),1)));
Loc = char(table2array(tableRecovered(indR(ii),2)));
title(['Location: ', subLoc, ' (', Loc, ')'])
%%%%%%%%%%%%%%

grid on
axis tight
set(gca, 'yscale', 'lin')
toc

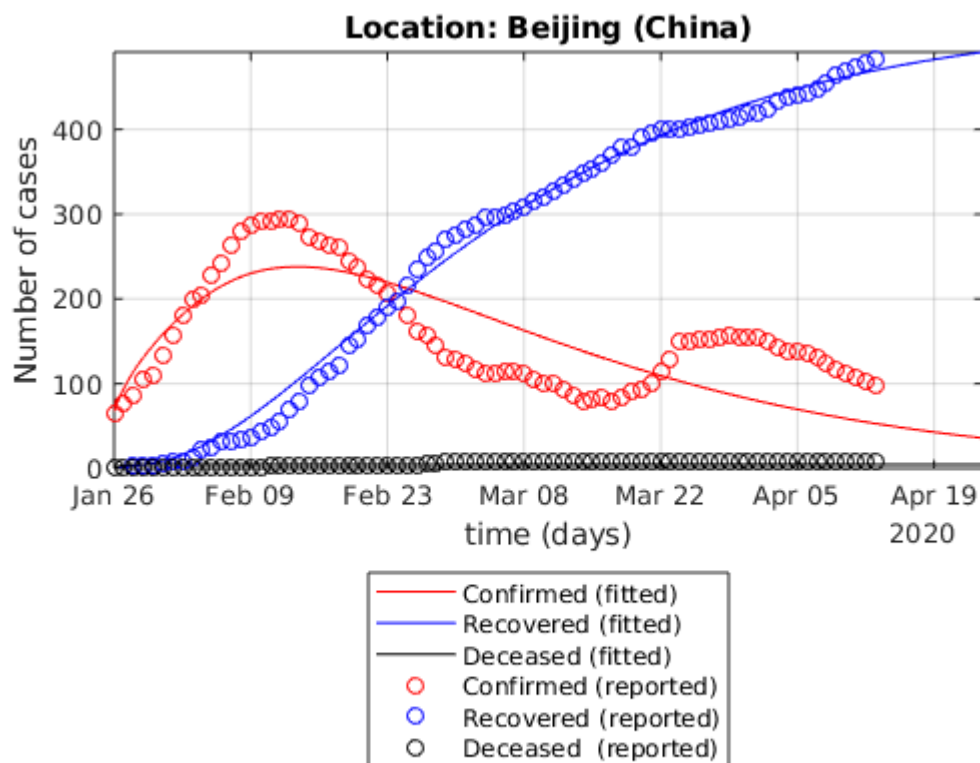
pause(1)

end
end

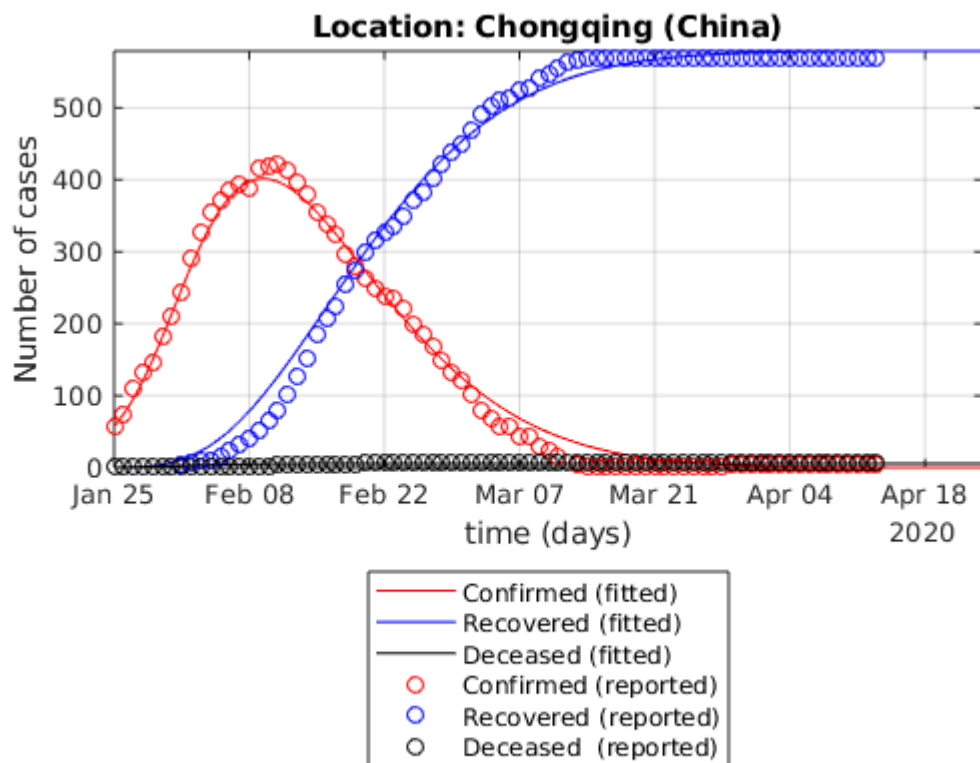
```



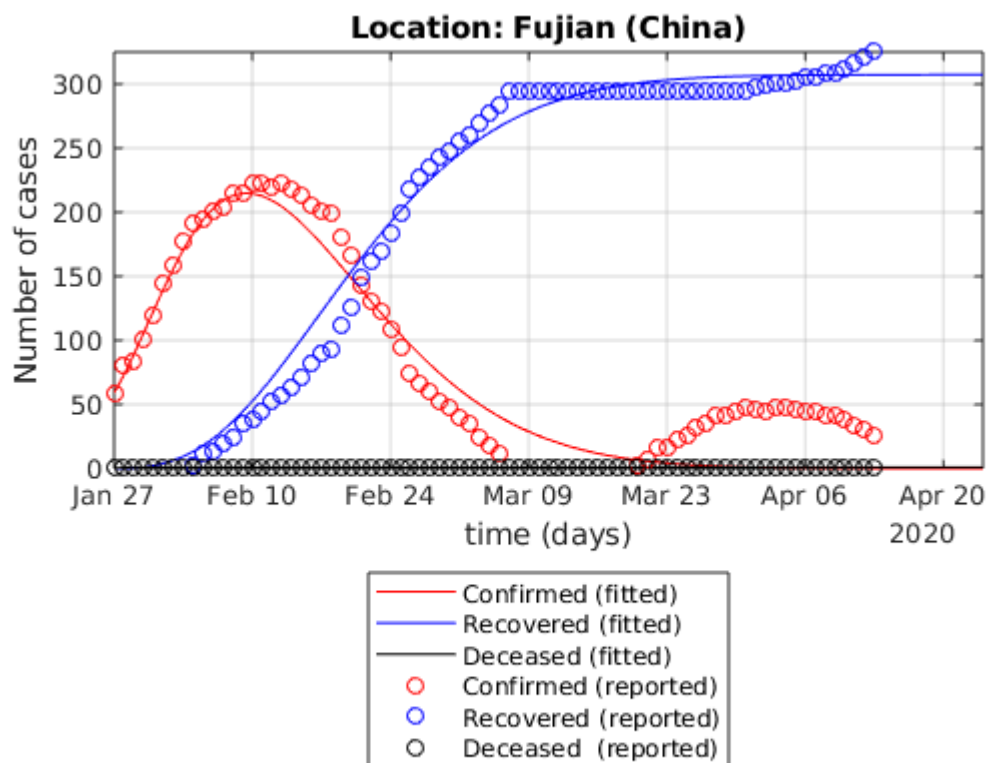
Elapsed time is 6.028812 seconds.
Elapsed time is 3.001728 seconds.



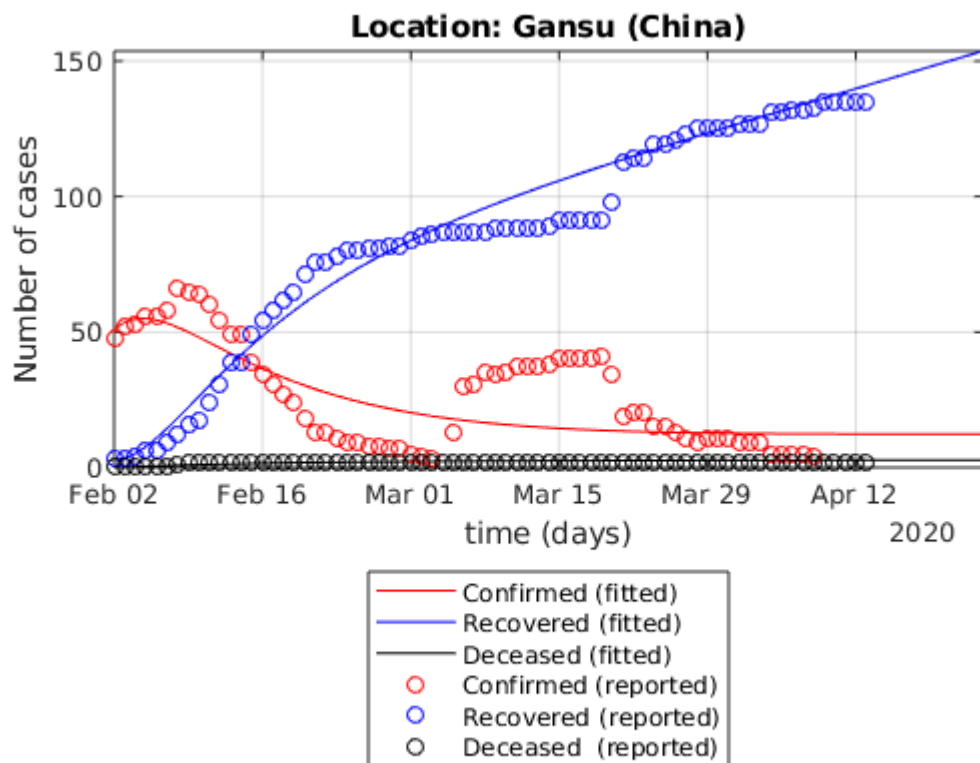
Elapsed time is 5.561491 seconds.



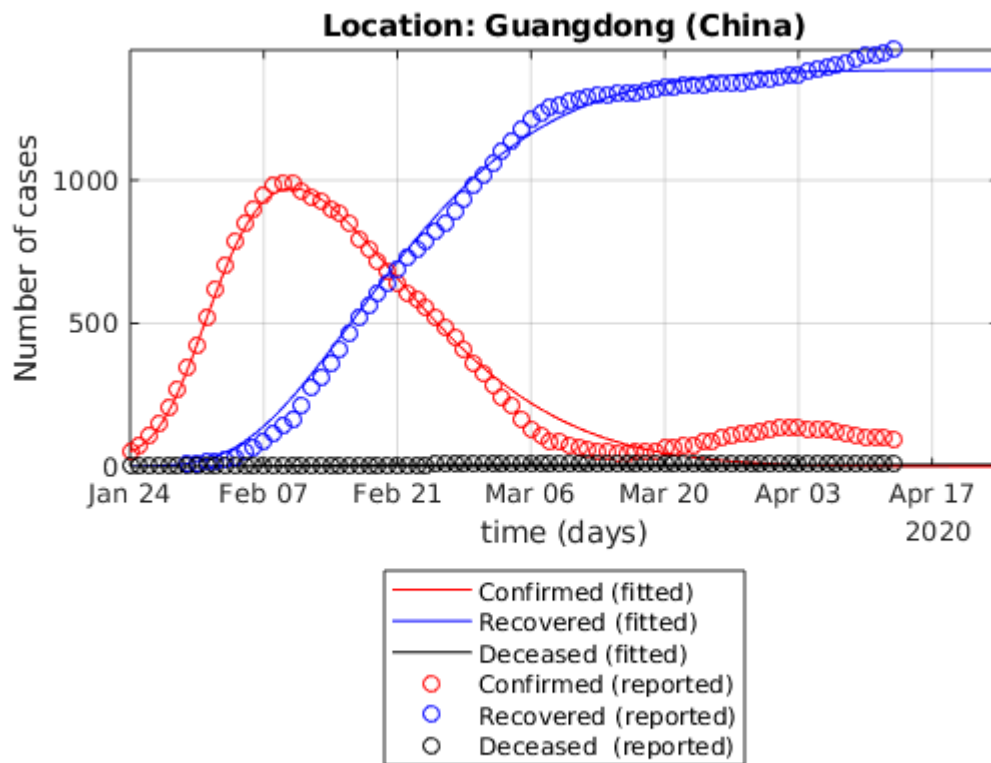
Elapsed time is 5.086324 seconds.



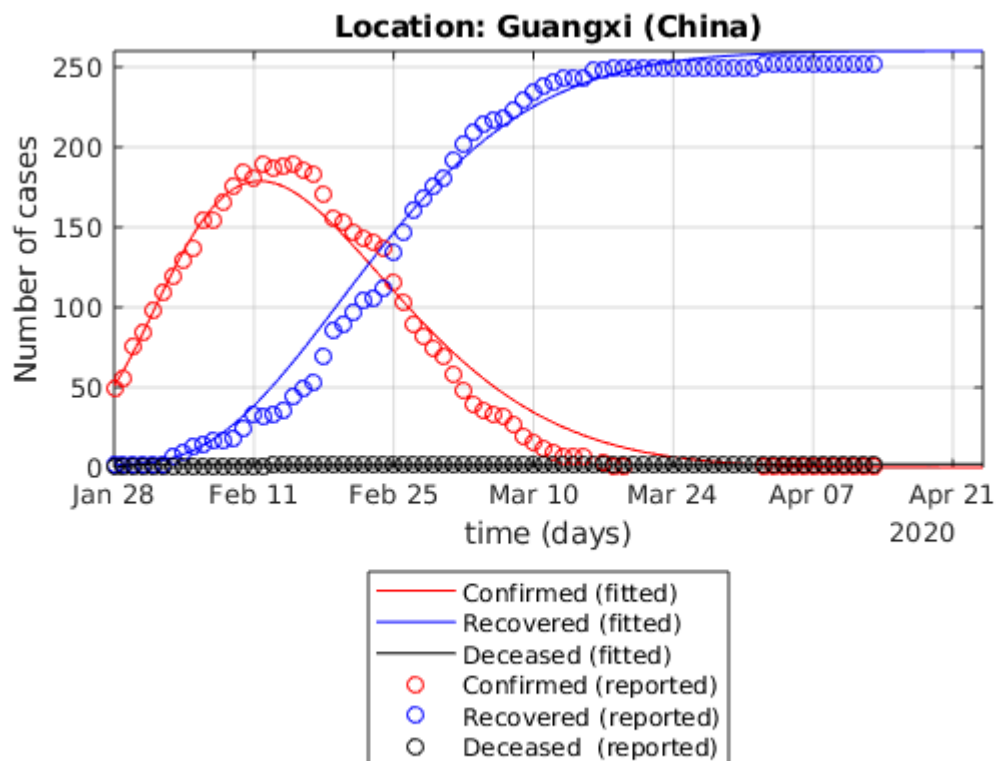
Elapsed time is 3.495556 seconds.



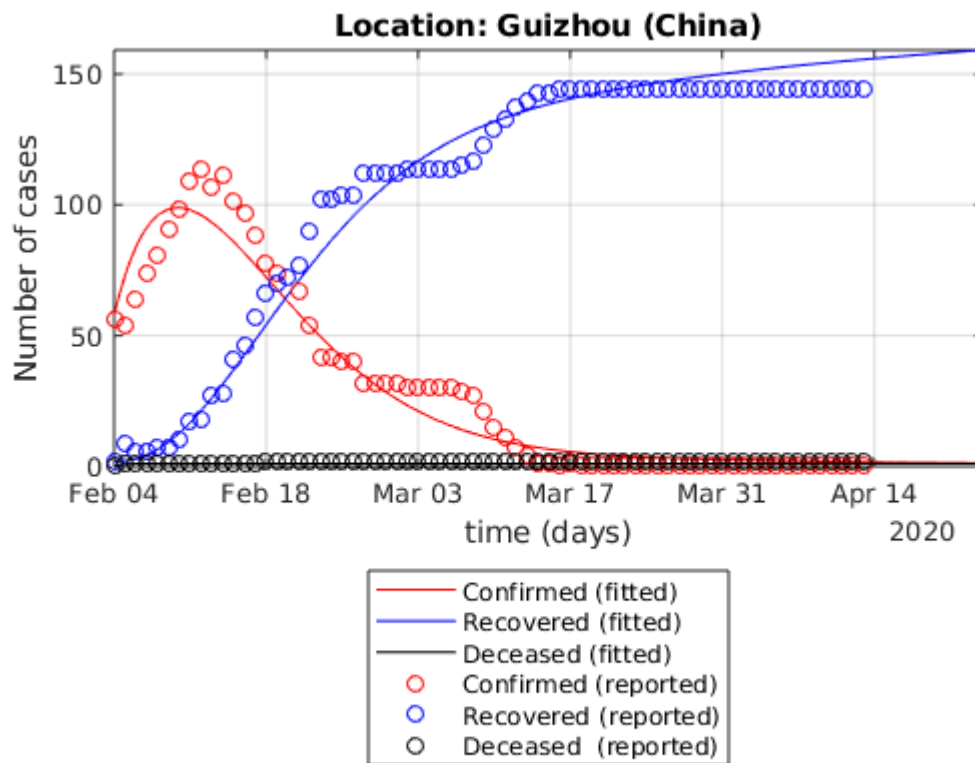
Elapsed time is 8.789640 seconds.



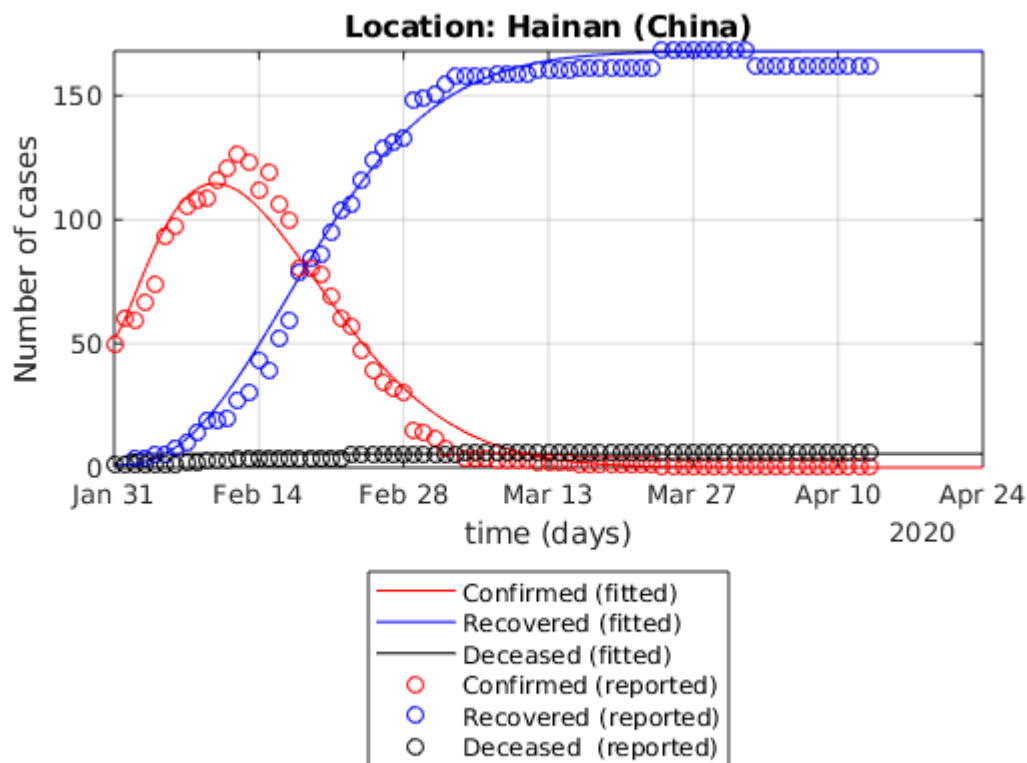
Elapsed time is 5.280773 seconds.



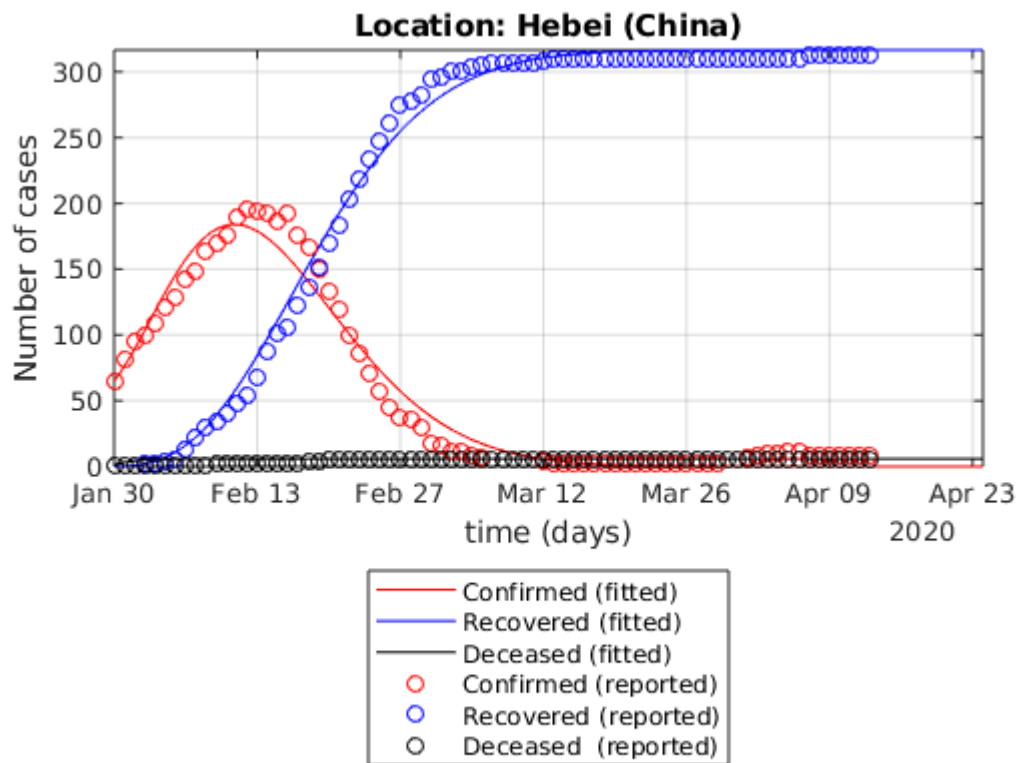
Elapsed time is 2.244720 seconds.



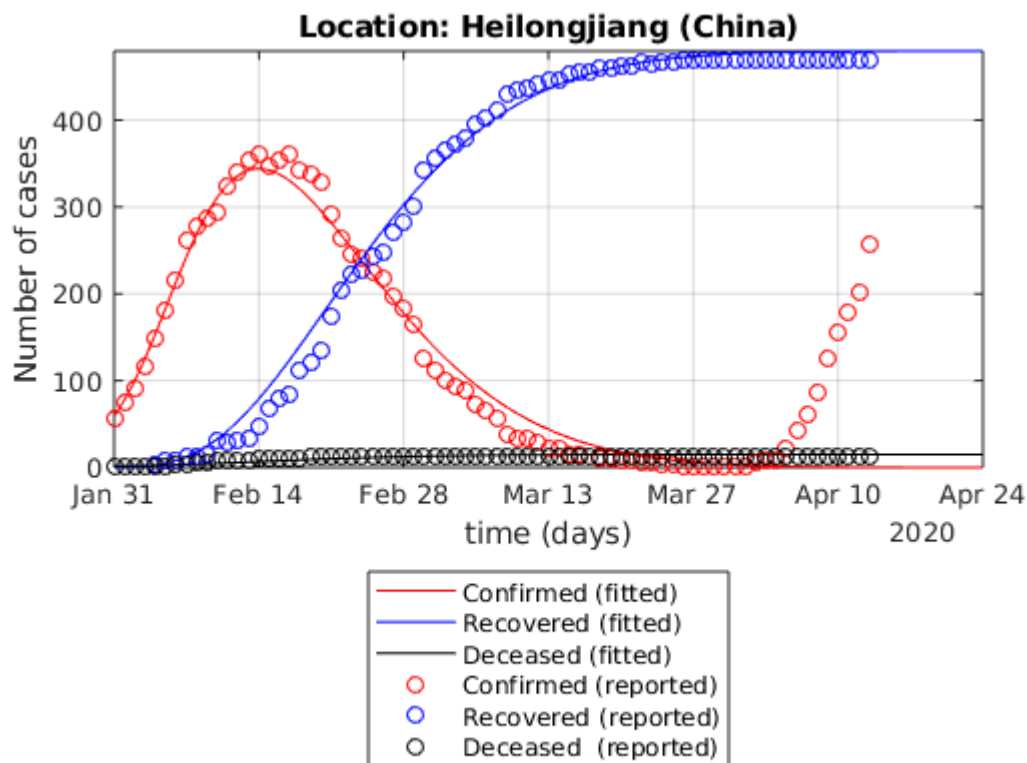
Elapsed time is 7.369017 seconds.



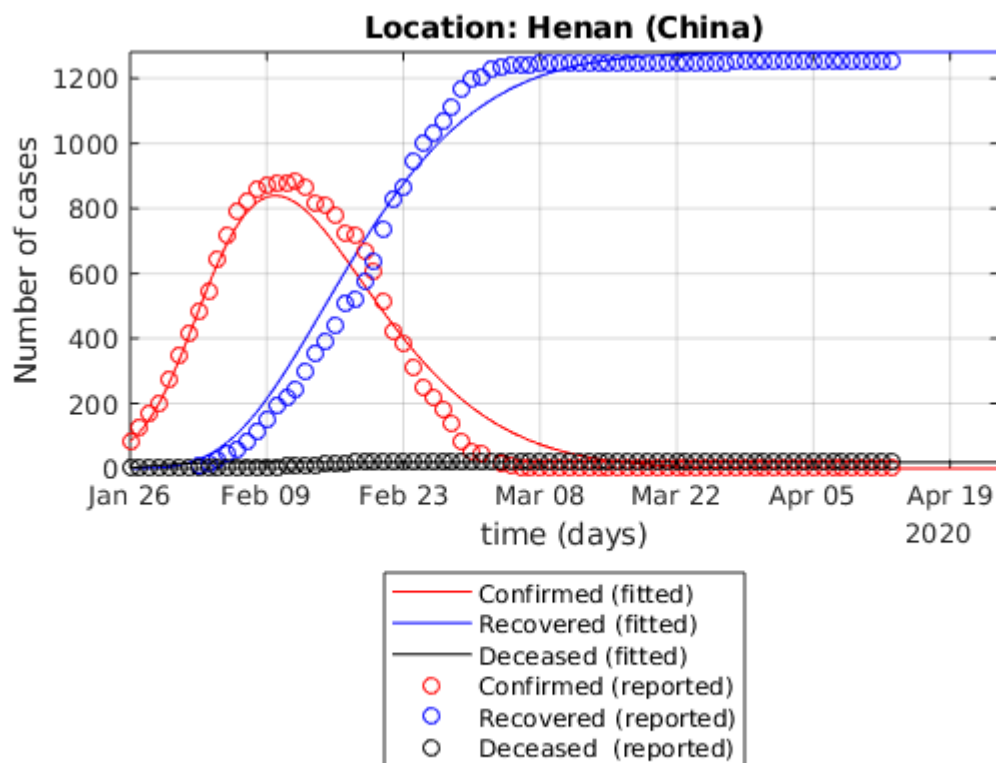
Elapsed time is 6.084266 seconds.



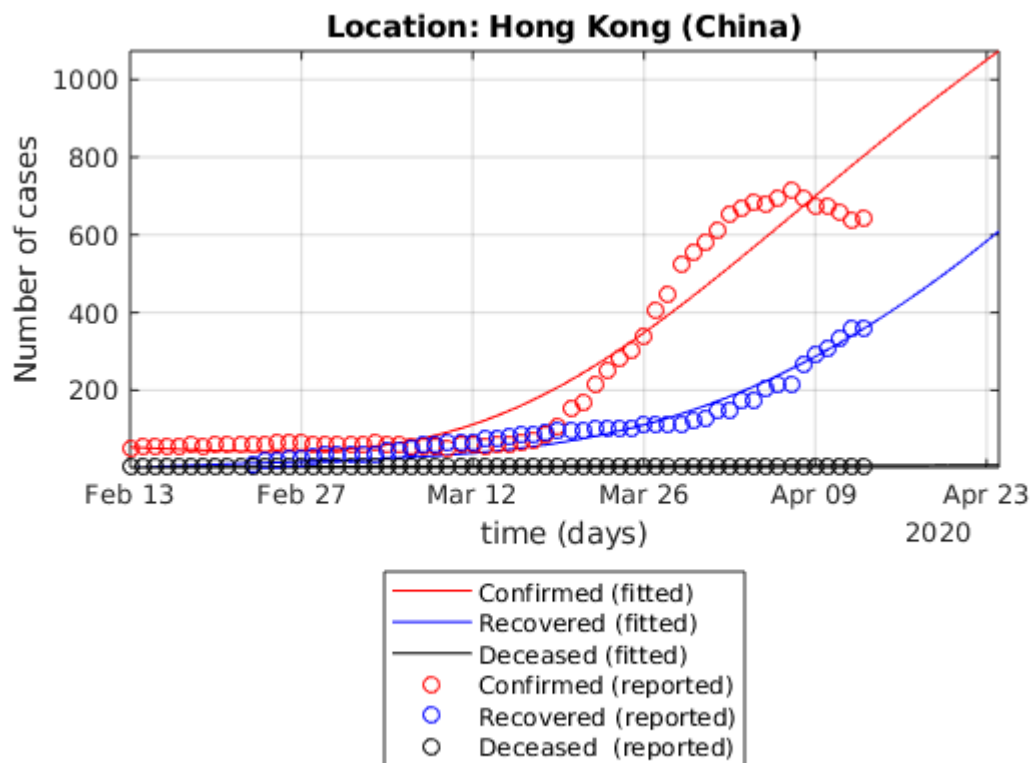
Elapsed time is 7.106727 seconds.



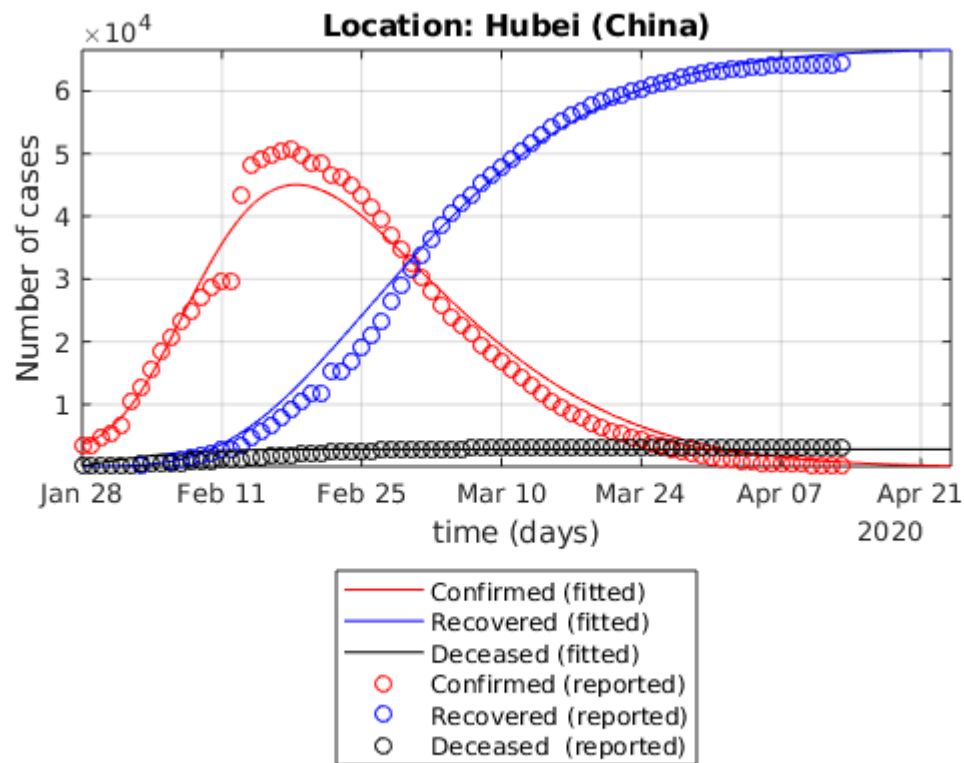
Elapsed time is 5.492636 seconds.



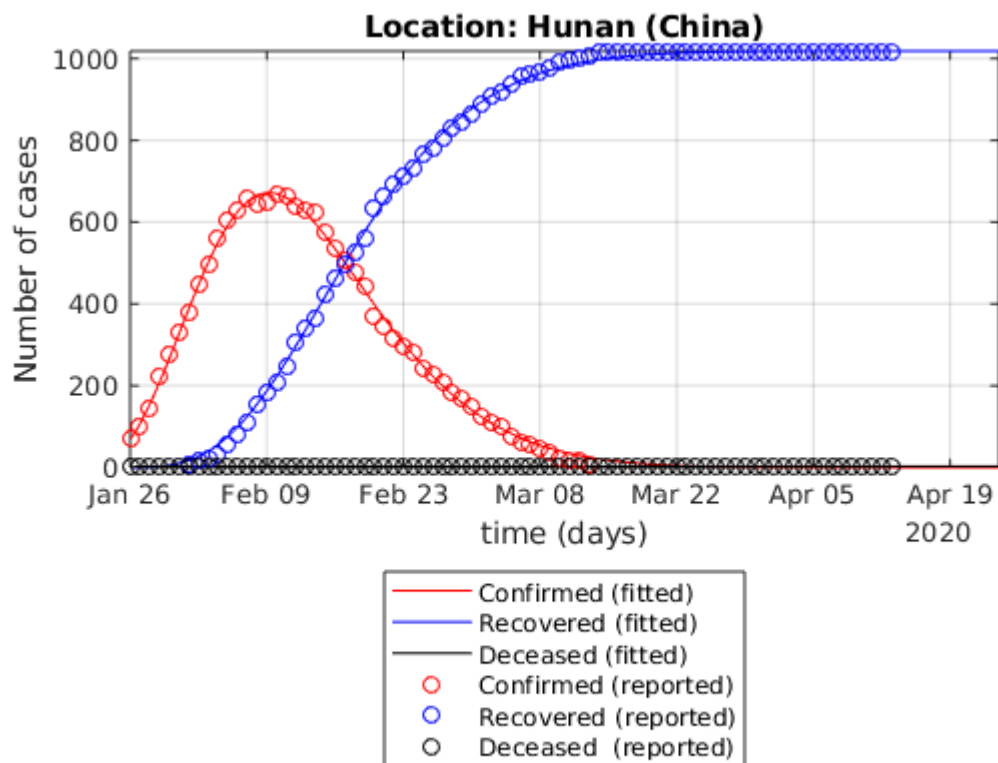
Elapsed time is 7.390638 seconds.



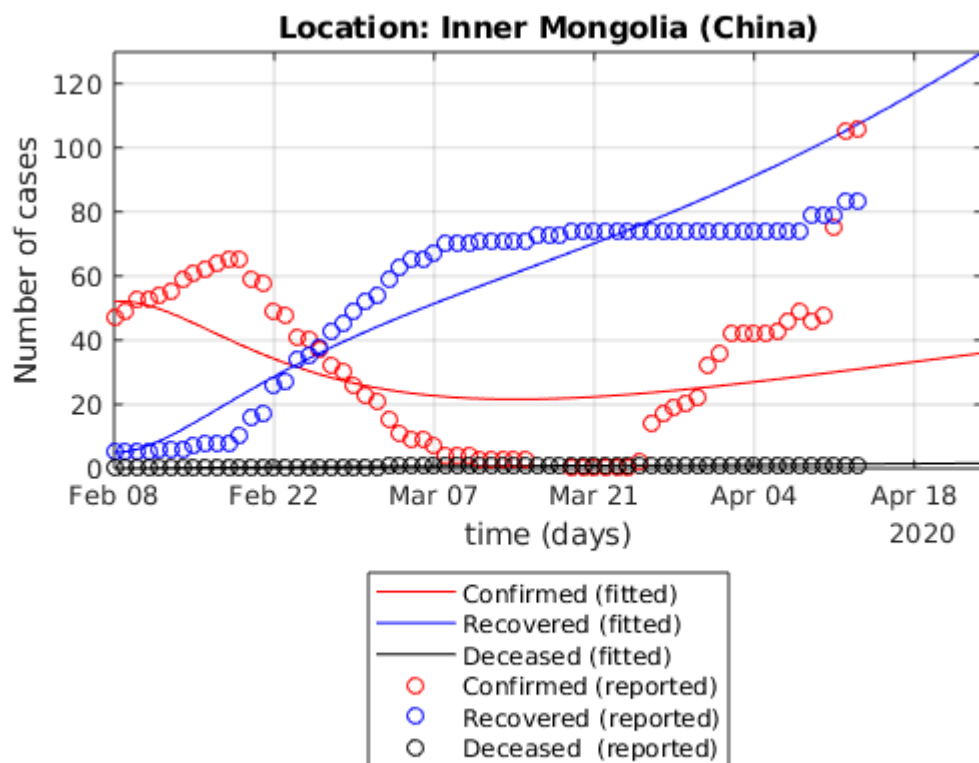
Elapsed time is 6.688305 seconds.



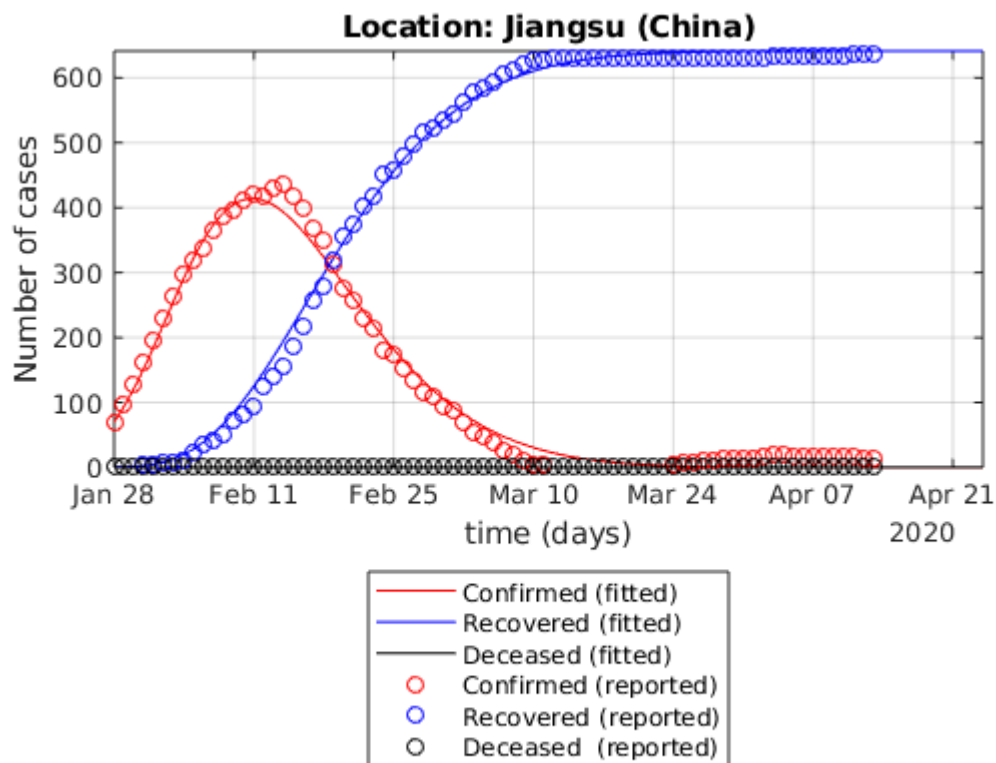
Elapsed time is 8.010915 seconds.



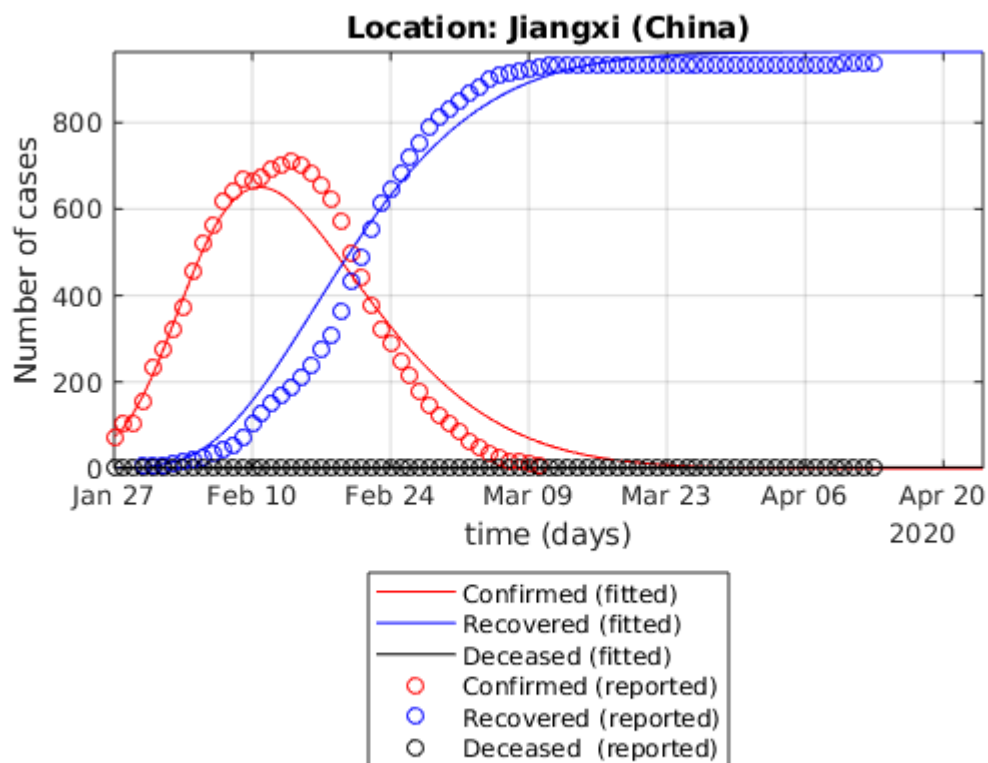
Elapsed time is 3.468896 seconds.



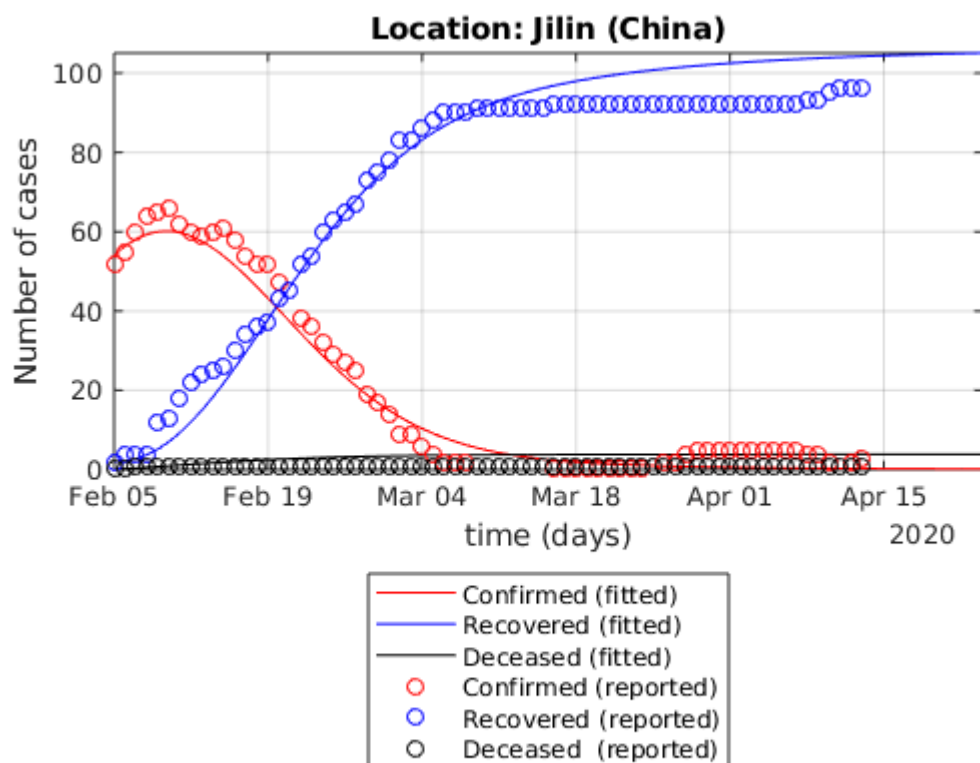
Elapsed time is 6.870132 seconds.



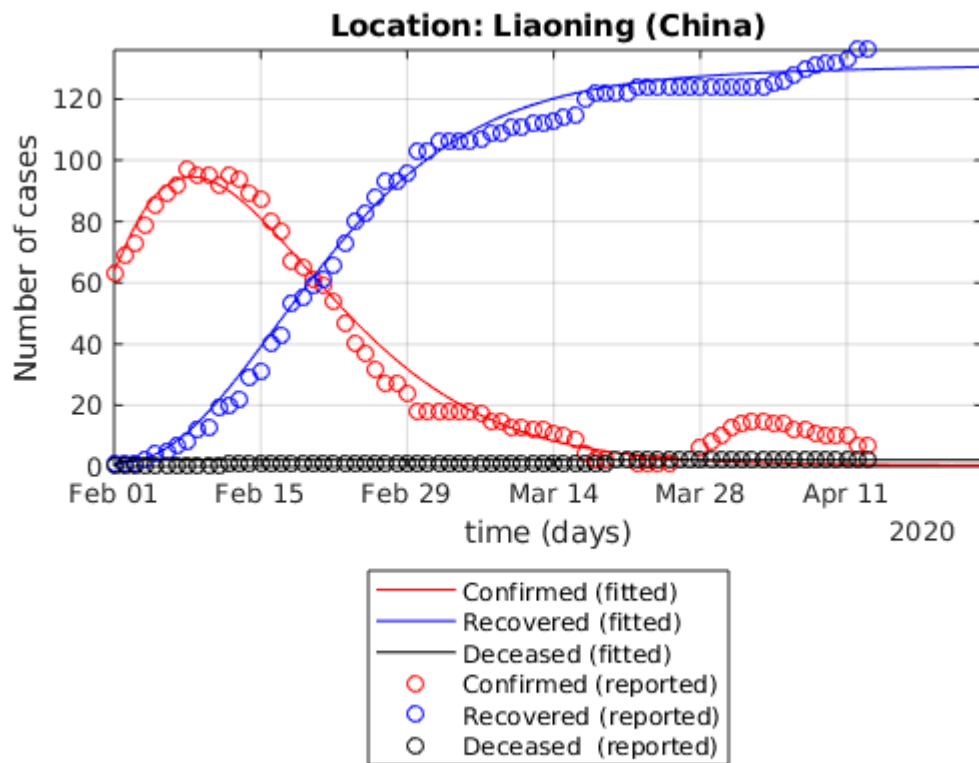
Elapsed time is 4.834306 seconds.



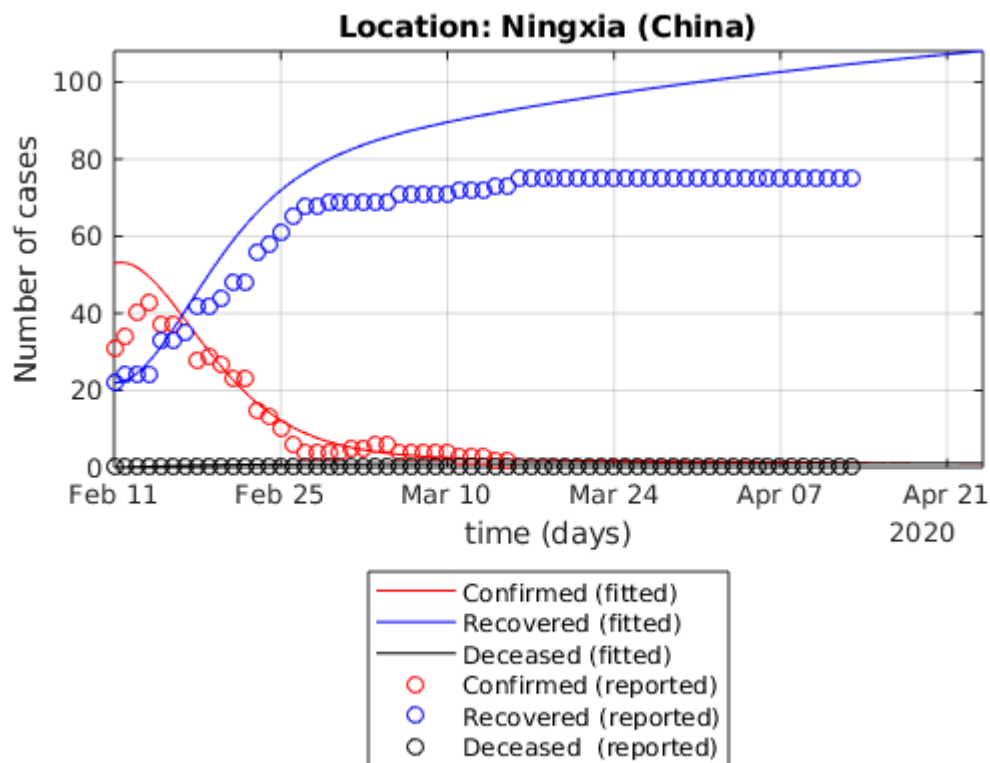
Elapsed time is 5.803504 seconds.



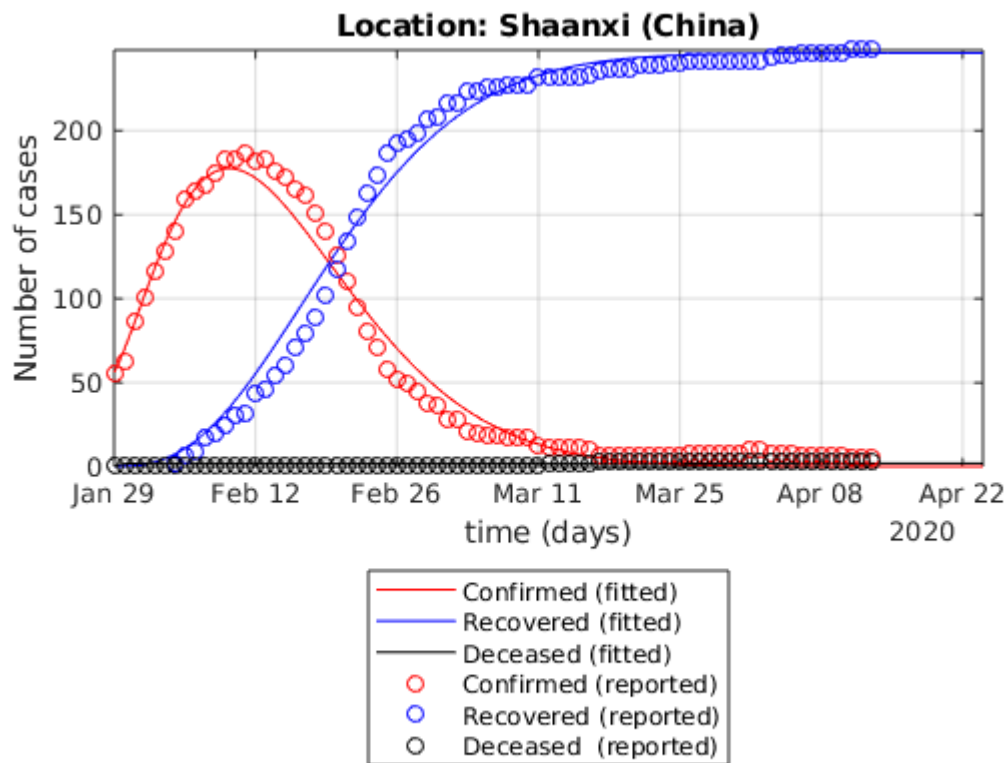
Elapsed time is 2.808234 seconds.



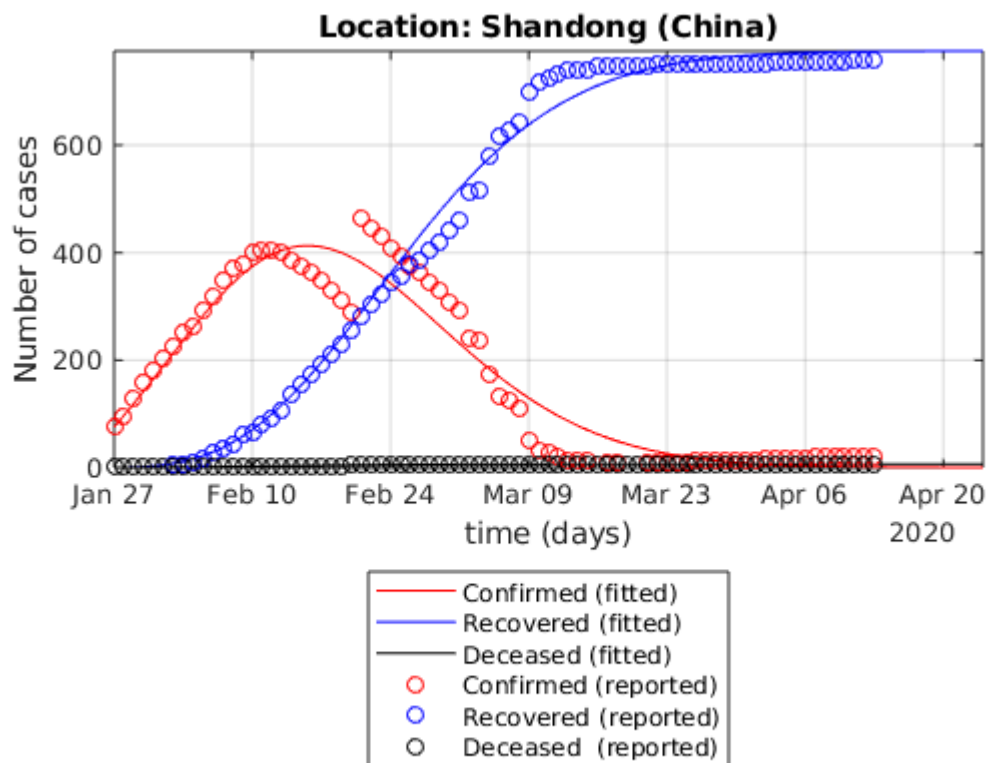
Elapsed time is 2.586329 seconds.



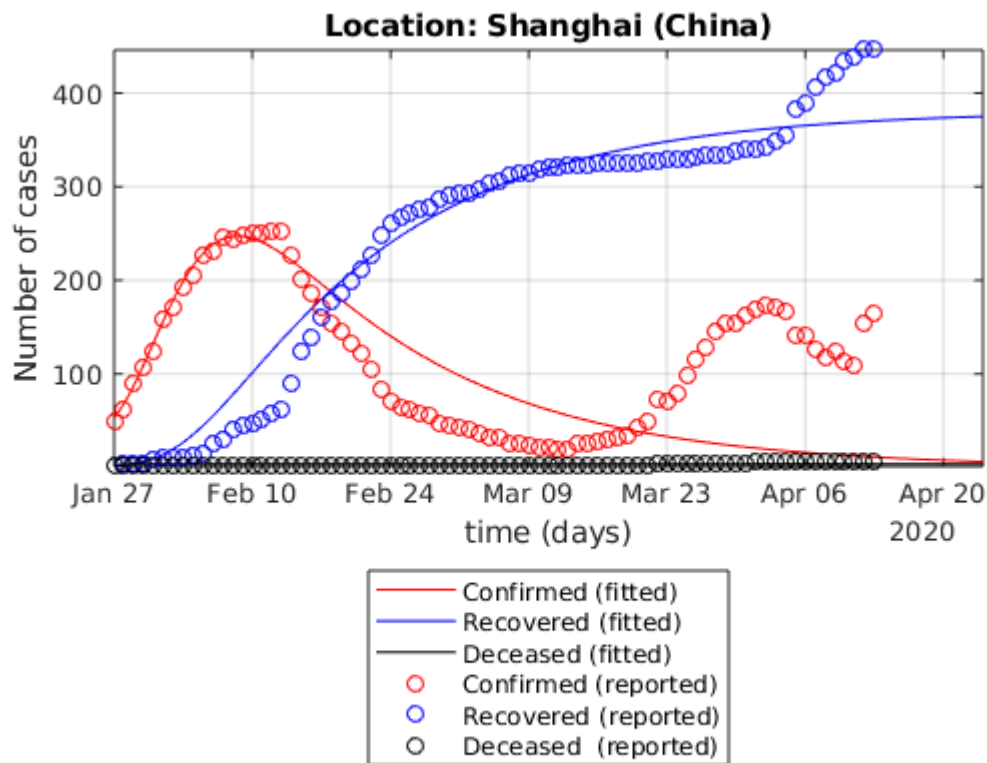
Elapsed time is 6.355090 seconds.



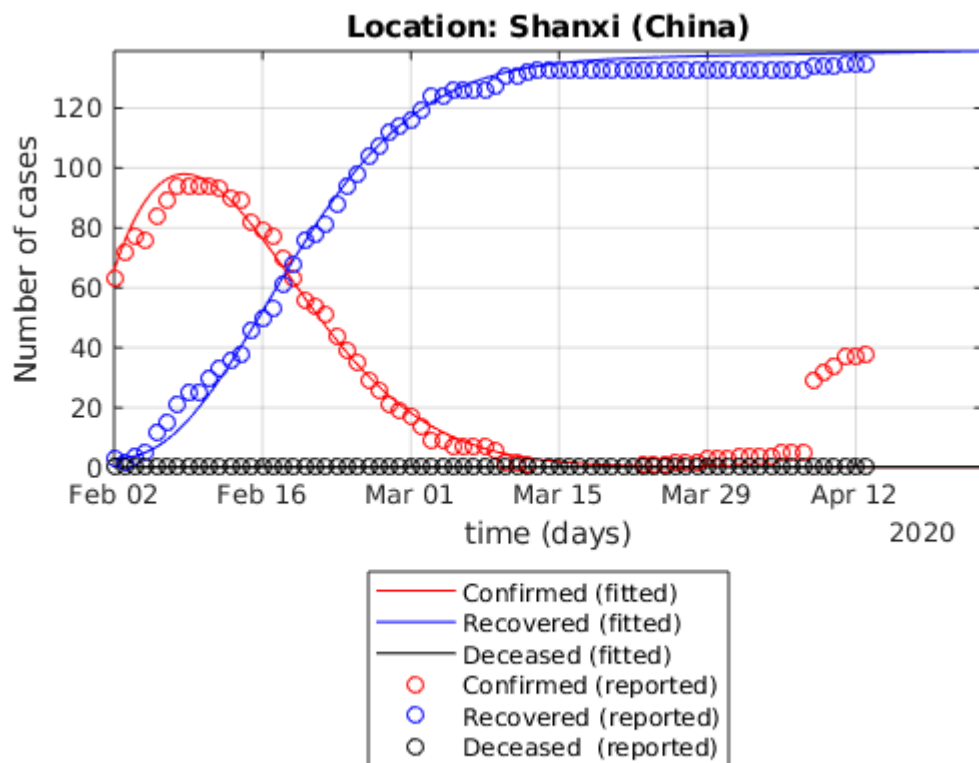
Elapsed time is 6.267219 seconds.



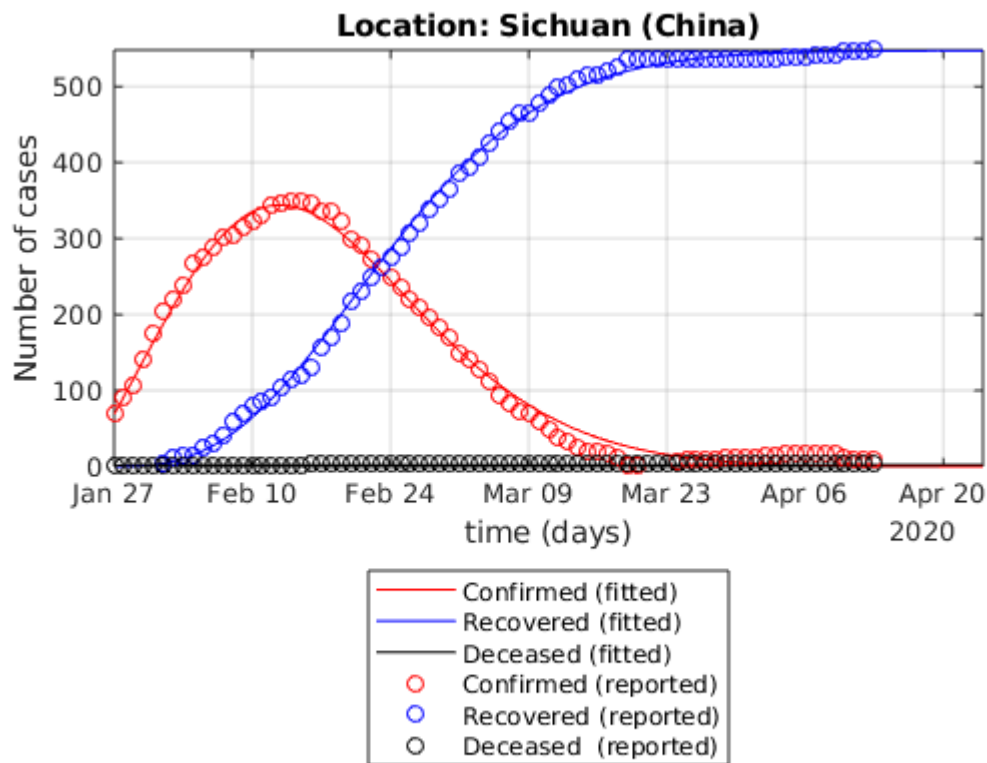
Elapsed time is 2.936280 seconds.



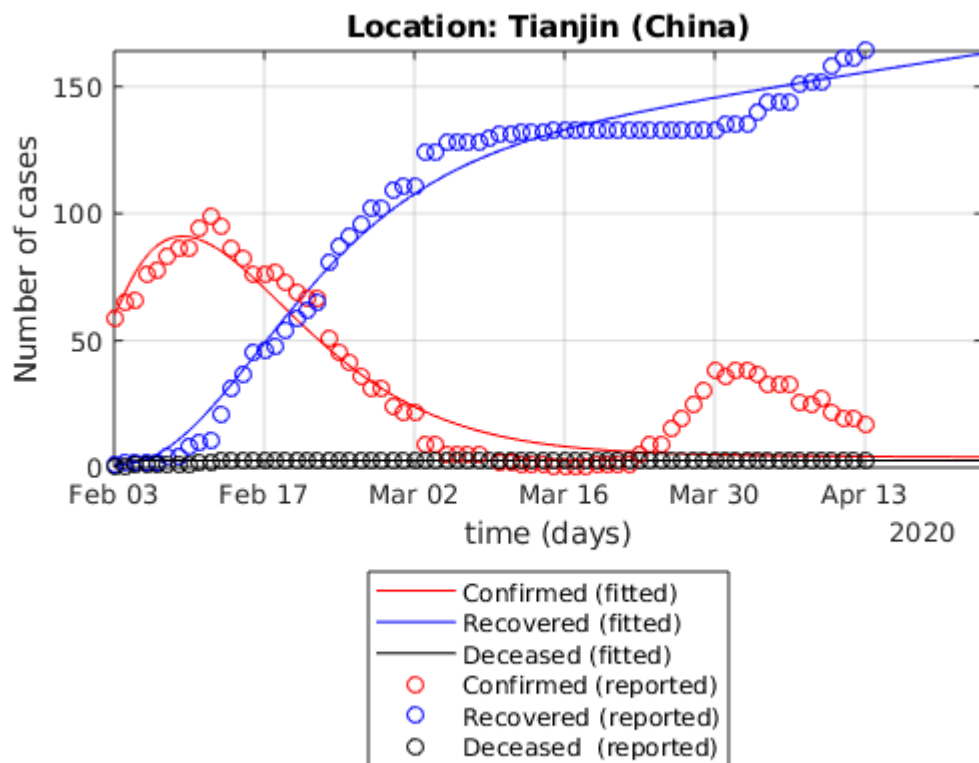
Elapsed time is 5.103272 seconds.



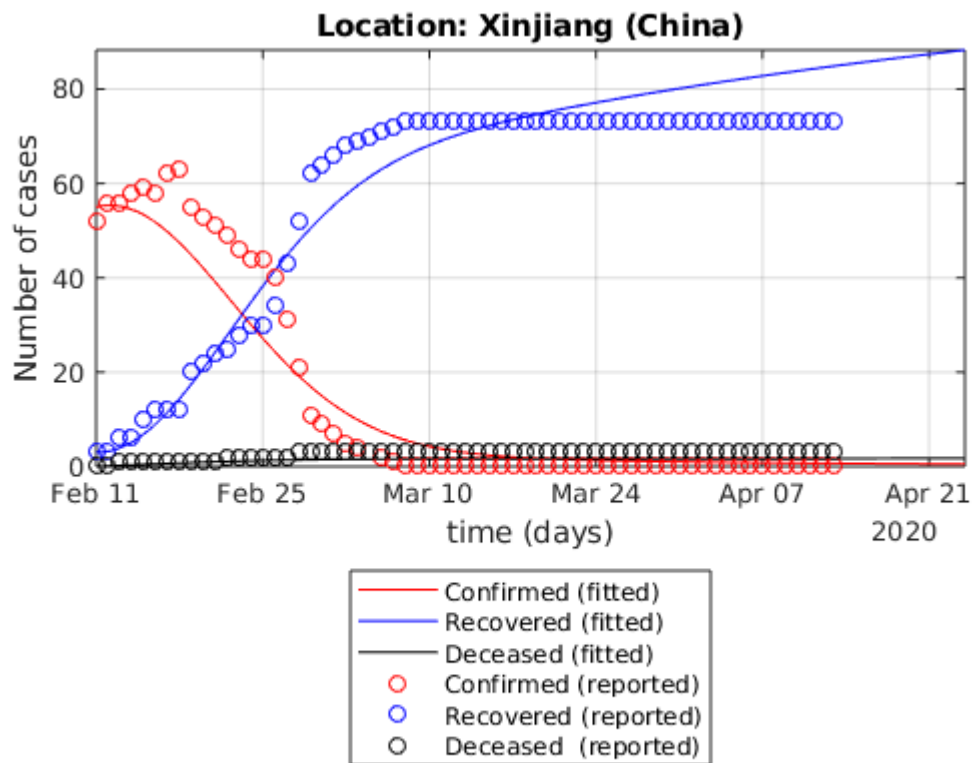
Elapsed time is 6.751732 seconds.



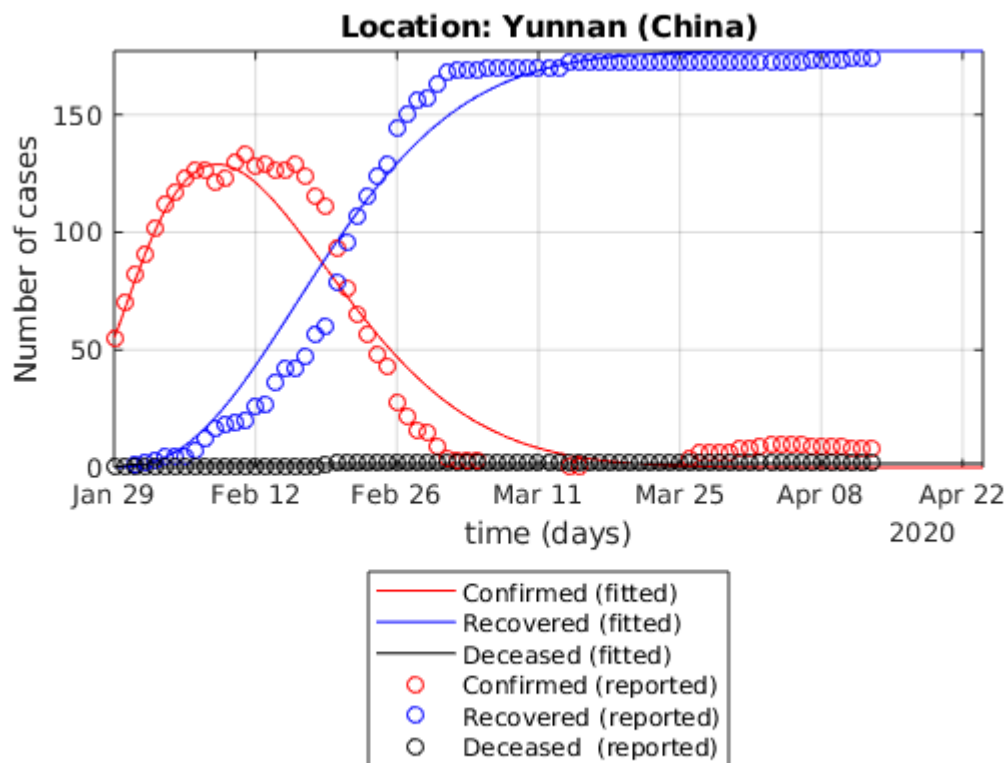
Elapsed time is 3.344490 seconds.



Elapsed time is 4.616368 seconds.



Elapsed time is 5.105739 seconds.



Elapsed time is 7.922835 seconds.

