Моделирование динамики коронавируса

12-Feb-2020

Обновление: Власти Китая 12 февраля изменили методику подсчета числа заболевших. Отныне в число зараженных пациентов станут включать уже по данным компьютерной томографии, не дожидаясь результатов других исследований. Поэтому количество заболевших за 11 февраля (44653) и 12 февраля (59804) различается очень сильно и не согласуется с результатами моделирования. Тем не менее, описанный ниже подход будет применим к данным, собираемым по-новому.

Смоделируем динамику коронавируса 2019-nCoV в Mathematica основываясь на публичных данных.

Данные о распространении вируса

Актуальные данные можно найти на Википедии в секции Epidemiology. Перепишем динамику заболевших оттуда.

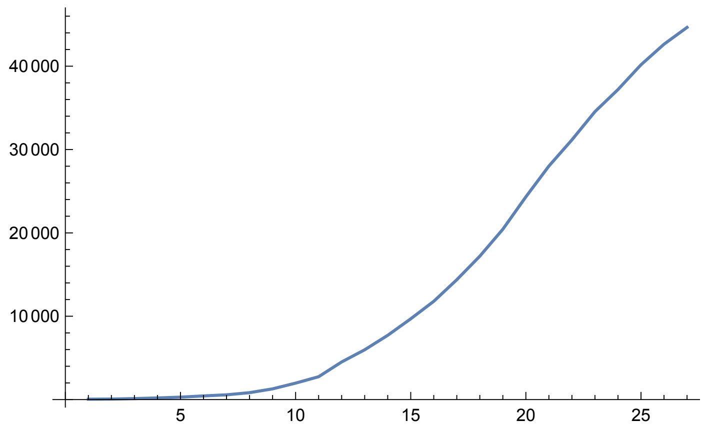
In[1]:=
```
data = {45, 62, 121, 198, 291, 440, 571, 830, 1287,
   1975, 2744, 4515, 5974, 7711, 9692, 11 791, 14 380, 17 205, 20 438,
   24 324, 28 018, 31 161, 34 546, 37 198, 40 171, 42 638, 44 653};
% //
 Length
```

Out[2]= 27

Построим график. Простейший способ сделать это — функция `ListLinePlot`.

In[3]:=
```
ListLinePlot[data]
```

Out[3]=

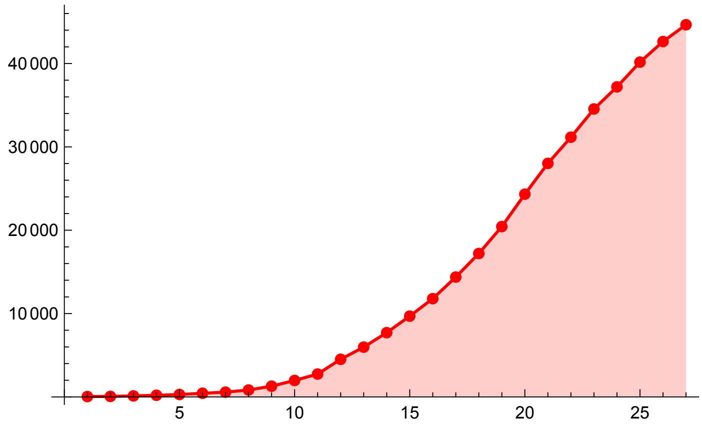


Моделирование динамики распространения вируса

Сделаем график более наглядным, добавив точки (опция `PlotMarkers→Automatic`) и заливку ниже линии (`Filling→Axis`). Также поменяем цвет на красный (`PlotStyle→Red`).

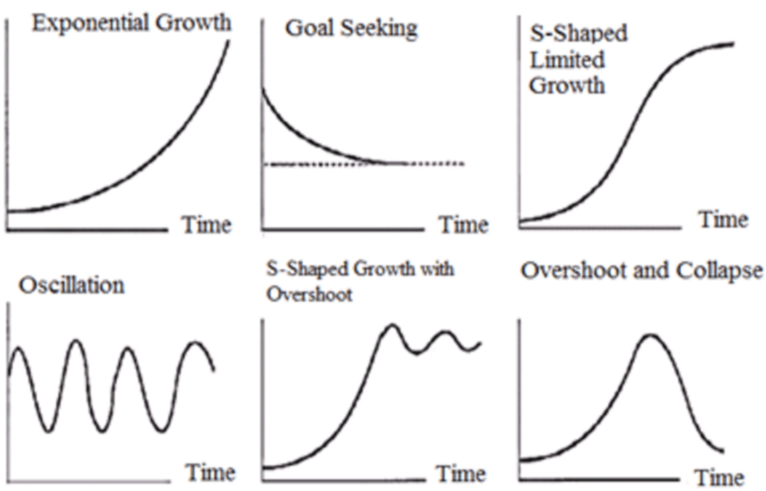
In[4]:=
```
plot1 =
 ListLinePlot[data, PlotMarkers → Automatic, PlotStyle → Red, Filling → Axis]
```

Out[4]=



Какой функцией можно смоделировать динамику этого графика? Вспомним (из курса по системному анализу), что у сложных систем существует несколько паттернов поведения.

In[5]:=



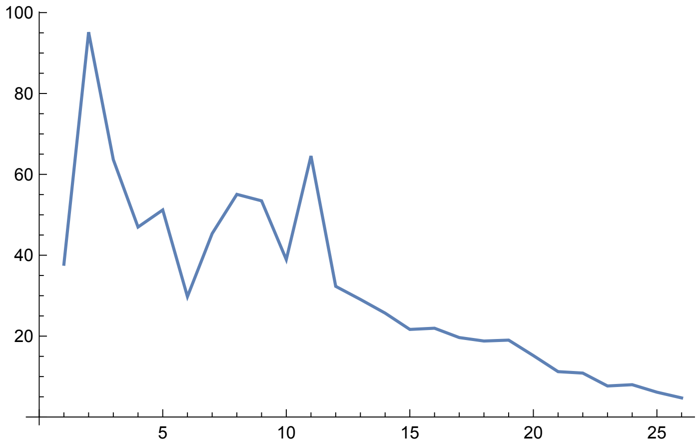
Даже по внешнему виду наш график напоминает S-кривую. Проверим это, вычислив процент прироста заболевших. Для этого разобьем список на пары (функция `Partition`), для каждой пары посчитаем прирост (функция `N[...,2]` нужна, чтобы значения были короче в записи), а затем нарисуем график.

In[6]:=
```
Partition[data, 2, 1]
N[100 (#[[2]] - #[[1]])/#[[1]], 2] & /@ %
ListLinePlot[%]
```

Out[6]=
```
{{45, 62}, {62, 121}, {121, 198}, {198, 291}, {291, 440}, {440, 571}, {571, 830},
 {830, 1287}, {1287, 1975}, {1975, 2744}, {2744, 4515}, {4515, 5974},
 {5974, 7711}, {7711, 9692}, {9692, 11 791}, {11 791, 14 380}, {14 380, 17 205},
 {17 205, 20 438}, {20 438, 24 324}, {24 324, 28 018}, {28 018, 31 161}, {31 161, 34 546},
 {34 546, 37 198}, {37 198, 40 171}, {40 171, 42 638}, {42 638, 44 653}}
```

Out[7]=
```
{38., 95., 64., 47., 51., 30., 45., 55., 53., 39., 65., 32.,
 29., 26., 22., 22., 20., 19., 19., 15., 11., 11., 7.7, 8.0, 6.1, 4.7}
```

Out[8]=



Прирост заболевших сначала рос, но потом стал постоянно падать. Это признаки S-кривой. Это обнадеживает, так как S-кривая возникает, если исходная популяция (в данном случае заболевшие) локализованна и не растет.

Уравнение S-кривой (логистической кривой) имеет вид

$$f(x) = \frac{L}{1+e^{-k(x-x_0)}}$$

где L — это максимум функции, k — коэффициент уклона кривой, а x_0 — срединная точка.

Посмотрим, сможем ли подобрать такие значения коэффициентов, чтобы получившаяся кривая соотвествовала нашим данным. Для этого воспользуемся функцией `FindFit`.

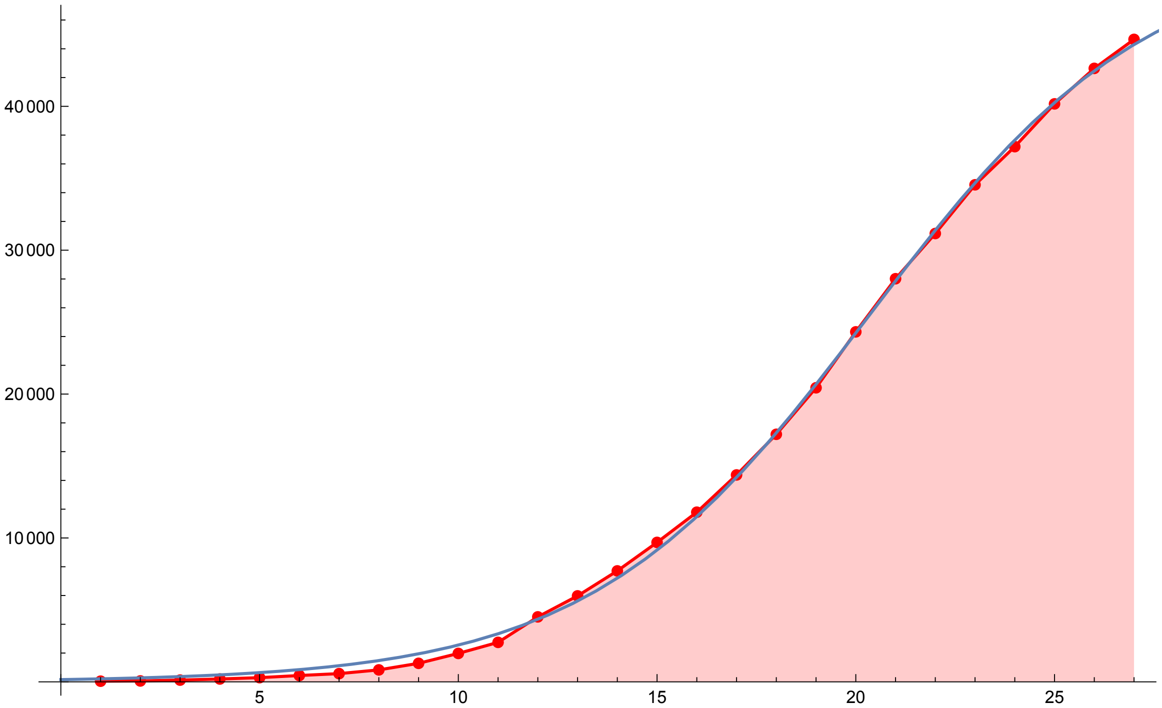
In[9]:=
```
coeffs = FindFit[data, L/(1 + e^(-k (x-x0))), {L, k, x0}, x]
```

Out[9]=
```
{L → 50 956.4, k → 0.284508, x0 → 20.3404}
```

Значения есть! Мы даже видим максимум функции, а именно 50956 заболевших. Проверим, насколько получившиеся значения соответствуют нашим данным. Для этого нарисуем график логистической кривой и совместим его с графиком исходных данных.

In[10]:=
```
plot2 = Plot[L/(1 + e^(-k (x-x0))) /. coeffs, {x, 0, 30}];
Show[plot1, plot2, ImageSize → 600]
```

Out[11]=



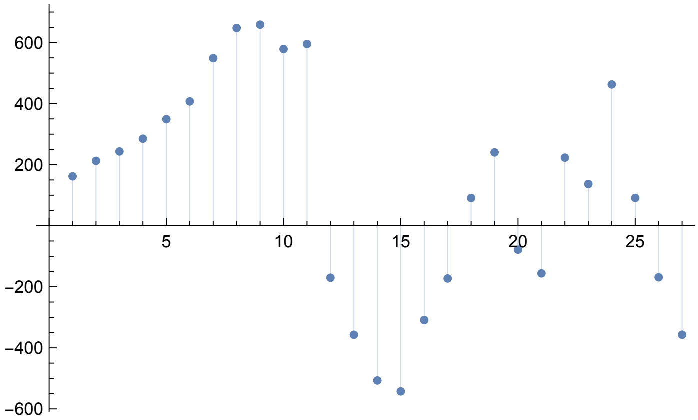
Как мы видим, данные практически совпадают. Построим график разностей предсказаний и фактических данных (эти разности назовем `residuals`). Дополнительно, найдем минимальное, среднее и максимальное значения разностей.

In[12]:=
```
L/(1 + ⅇ^(-k (x-x0))) /. coeffs /. x → # & /@ Range[Length[data]]
residuals = % - data
ListPlot[residuals, Filling → Axis]
{Min[Abs[residuals]], Mean[Abs[residuals]], Max[Abs[residuals]]}
```

Out[12]=
```
{206.885, 274.606, 364.335, 483.104, 640.1, 847.26, 1119.97, 1477.87, 1945.68,
 2553.93, 3339.37, 4344.68, 5616.93, 7204.16, 9149.4, 11482., 14207.3, 17296.,
 20678.3, 24245.6, 27862., 31384.1, 34682.7, 37660.9, 40262.1, 42469.1, 44296.}
```

Out[13]=
```
{161.885, 212.606, 243.335, 285.104, 349.1, 407.26,
 548.972, 647.874, 658.683, 578.93, 595.367, -170.324, -357.069,
 -506.839, -542.601, -308.96, -172.712, 90.9564, 240.269, -78.4165,
 -155.994, 223.052, 136.693, 462.911, 91.1493, -168.85, -356.971}
```

Out[14]=



Out[15]=
```
{78.4165, 324.181, 658.683}
```

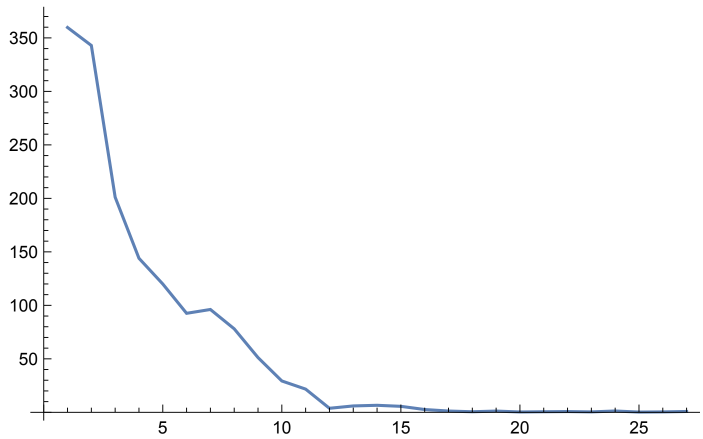
Как мы видим, среднее отклонение модели (`Mean[Abs[residuals]]`) составляет 324.181 или менее 1% от последних фактических данных. Но как менялась относительная погрешность по мере роста функции?

In[16]:=
```
L/(1 + e^(-k (x-x0))) /. coeffs /. x → # & /@ Range[Length[data]]
MapThread[#1/#2 100 &, {Abs[residuals], data}]
ListLinePlot[%]
```

Out[16]= {206.885, 274.606, 364.335, 483.104, 640.1, 847.26, 1119.97, 1477.87, 1945.68, 2553.93, 3339.37, 4344.68, 5616.93, 7204.16, 9149.4, 11482., 14207.3, 17296., 20678.3, 24245.6, 27862., 31384.1, 34682.7, 37660.9, 40262.1, 42469.1, 44296.}

Out[17]= {359.745, 342.913, 201.103, 143.992, 119.966, 92.5591, 96.1421, 78.0571, 51.1797, 29.3129, 21.6971, 3.77239, 5.97704, 6.57294, 5.59844, 2.6203, 1.20105, 0.528662, 1.1756, 0.322383, 0.556764, 0.715806, 0.395685, 1.24445, 0.226903, 0.396008, 0.799433}

Out[18]=



Как мы видим, погрешность модели составляет ~1% для последних 11 показаний. Это указывает либо на точность нашей модели, либо на то, что публикуемые данные также смоделированы. Хочется верить, что верно первое утверждение -- ведь тогда модель явно указывает на то, что вспышка вируса локализованна, находится под контролем, а общее количество заболевших не будет расти экспоненциально и очень скоро достигнет своего максимума (ориентировочно, 51-52 тыс человек), после чего пойдет на спад.