

Zerfall und Pandemie - Gemeinsamkeiten

Bernd Laquai, 21.03.2020

Was hat eine Pandemie, wie die weltweite Ausbreitung der Covid-19 Erkrankung, mit dem radioaktiven Zerfall gemeinsam? Nun instinktiv würde man sagen, dass die Auswirkungen eines kerntechnischen Unfalls oder die Auswirkungen des Einsatzes von Atomwaffen ähnliche Anzahlen von Todesopfern und schlimme psychosoziale Folgen für die Gesellschaft haben können. Das leuchtet sicher ein. Aber die Ähnlichkeiten betreffen interessanterweise auch die mathematischen Grundlagen und es ist durchaus sinnvoll sich auch damit zu befassen, weil es dem Verständnis von Beidem dient und Möglichkeiten schafft Vorhersagen zu treffen.

Grundlage des radioaktiven Zerfalls ist das Zerfallsgesetz. Verbal ausgedrückt bedeutet es, dass die Zerfallsrate proportional zu den noch vorhandenen Atomen ist oder mathematisch ausgedrückt:

$$\frac{dN(t)}{dt} = -\lambda \cdot N(t)$$

Die Proportionalitätskonstante λ heißt Zerfallskonstante und $T_{1/2} = \ln(2)/\lambda$ ist die zugehörige Halbwertszeit. Diese Differentialgleichung hat die bekannte Lösung:

$$N(t) = N_0 \cdot e^{-\lambda \cdot t}$$

Die Zerfallskonstante λ ist in der Natur stets positiv, aber von der Mathematik her hätte man das Vorzeichen auch positiv machen können, dann wäre die Zerfallskonstante für den radioaktiven Zerfall negativ gewesen, so dass $N(t)$ auch wirklich stetig abnimmt. Dann aber hätte ein positiver Wert einen exponentiellen Anstieg bedeutet, eben genau wie wir es derzeit bei der Covid-19 Pandemie in Form des Anstiegs der Zahl der Infizierten sehen. Wir können also zunächst einmal vom gleichen Sachverhalt ausgehen. Am Anfang einer Pandemie steigt nämlich die Zahl der Infizierten proportional zur Zahl der bereits Infizierten. Also mit anderen Buchstaben ausgedrückt:

$$\frac{dI(t)}{dt} = b \cdot I(t)$$

Mit der Lösung:

$$I(t) = I_0 \cdot e^{b \cdot t}$$

Hierin ist $I(t)$ die Zahl der Infizierten und I_0 die Anzahl mit der man startet. Der Wert b ist die Wachstumskonstante und $T_2 = \ln(2)/b$ die Verdoppelungszahl, ganz analog zur Halbwertszeit. Sie definiert die Zeit, in der sich die Zahl der Infizierten verdoppelt. Und was sich ergibt ist eben ein exponentielles Anwachsen der Anzahl an Infizierten im Gegensatz zu exponentiellen Zerfall bei der Radioaktivität. Wenn die Übertragbarkeit der Viren hoch ist, dann ist T_2 kurz und umgekehrt.

Das Problem bei der Covid-19 Pandemie ist, dass T_2 in Deutschland in der Anfangsphase bei etwa 2.5 lag, was zu einem enorm schnellen exponentiellen Anstieg führte und auf Grund eines gewissen Anteils an schweren Erkrankungsverläufen die Krankenhäuser schnell an die Belastungsgrenze und in anderen Ländern, wie in Italien, das Gesundheitssystem zum Kollabieren brachte. Damit die Leute verstehen wo das Problem liegt, kann man nun extrapolieren und den exponentiellen Anstieg, der sich ohne Maßnahmen der Regierung einstellen würde versuchen vorherzusagen. Basierend auf den kumulativen Fallzahlen des Robert Koch-Instituts sieht das am 21.3.20 recht beeindruckend aus.

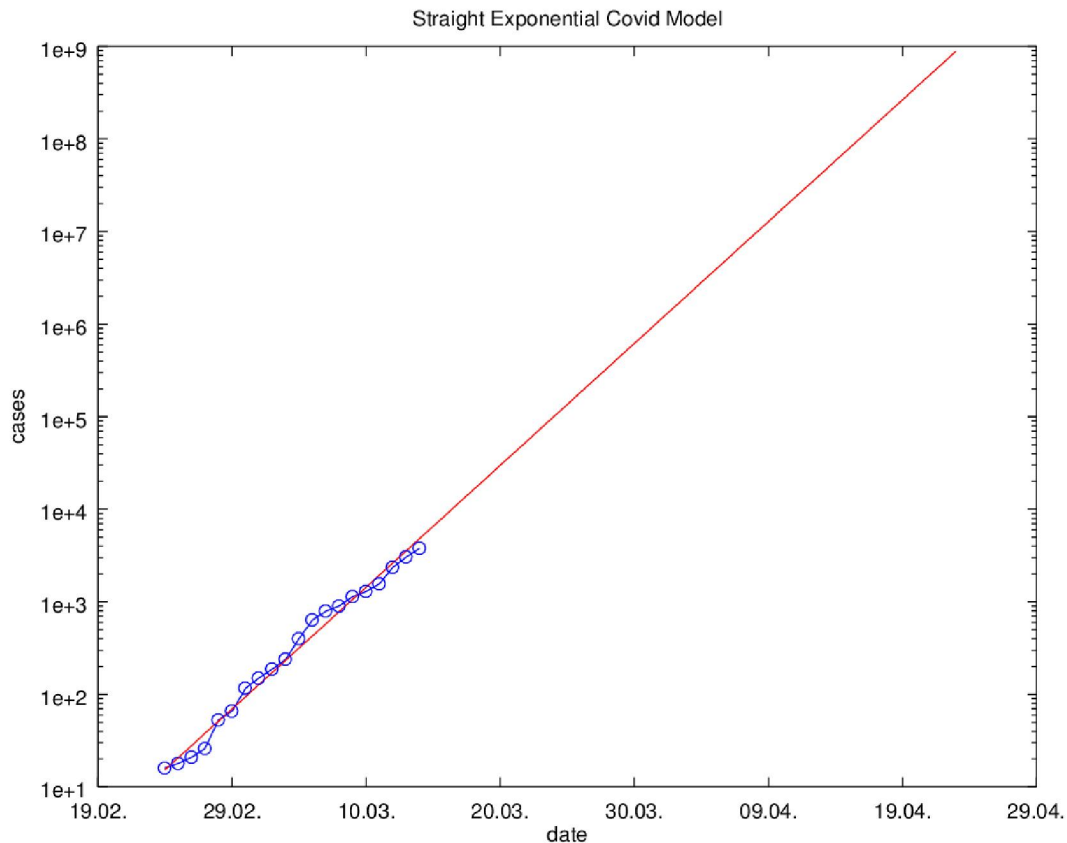


Abb. 1: Vorhersage der Fallzahlen der Covid-19 Pandemie in Deutschland am 21.3.20 auf Basis des exponentiellen Wachstumsgesetzes

Diese Projektion sagt aus, dass bereits Mitte April in Deutschland etwa 10^9 Menschen sich infiziert haben müssten. Allerdings beträgt die Zahl der Einwohner in Deutschland nur etwa $83 \cdot 10^6$ Menschen. Man erkennt also schnell, dass es ganz so dramatisch auch nicht laufen kann, obwohl es für manche uneinsichtigen Leute gut wäre, sie bekämen angesichts dieser Zahlen etwas mehr Angst.

Nun, die Annahme des exponentiellen Wachstums ist in der Anfangsphase einer Pandemie sicher richtig, aber sie setzt genau genommen voraus, dass auch unendlich viele potentielle Infektionsopfer vorhanden sind, so dass der Virus quasi „aus dem Vollen“ schöpfen kann. Betrachtet man aber nur Deutschland und geht davon aus, dass die Grenzen dicht sind, dann dauert es keinen Monat, dann wäre die Republik durchinfiziert und wenn dann jeder einigermaßen immun wäre bzw. wenn zusätzlich Menschen weggestorben sind, dann könnte die Zahl der Infizierten nicht mehr weiter anwachsen. Das exponentielle Wachstum wird daher automatisch gebremst. Man muss allerdings dazu sagen, dass die Mortalität bei Covid-19 etwa 0.2% beträgt, was bei 83 Mio. Infizierten etwa 166000 Tote bedeuten würde.

Schaut man sich aber die Zahlen der WHO aus China an, dann erkennt man, dass die Zahl der Infizierten im Februar zu stagnieren begann und derzeit bei einem Wert von etwas über 81000 konstant bleibt. Wenn man einmal davon ausgeht, dass die Zahlen aus China stimmen und wenn man bedenkt, dass China derzeit etwa 1.4 Milliarden Einwohner hat, dann haben die drastischen Maßnahmen der autoritären Regierung und andere Einflüsse dazu geführt, dass das Wachstum der Fallzahlen sich weit vor der Durchinfizierung alle Chinesen vom rein exponentiellen Wachstum wieder auf sehr geringe Werte zurückentwickelt hat.

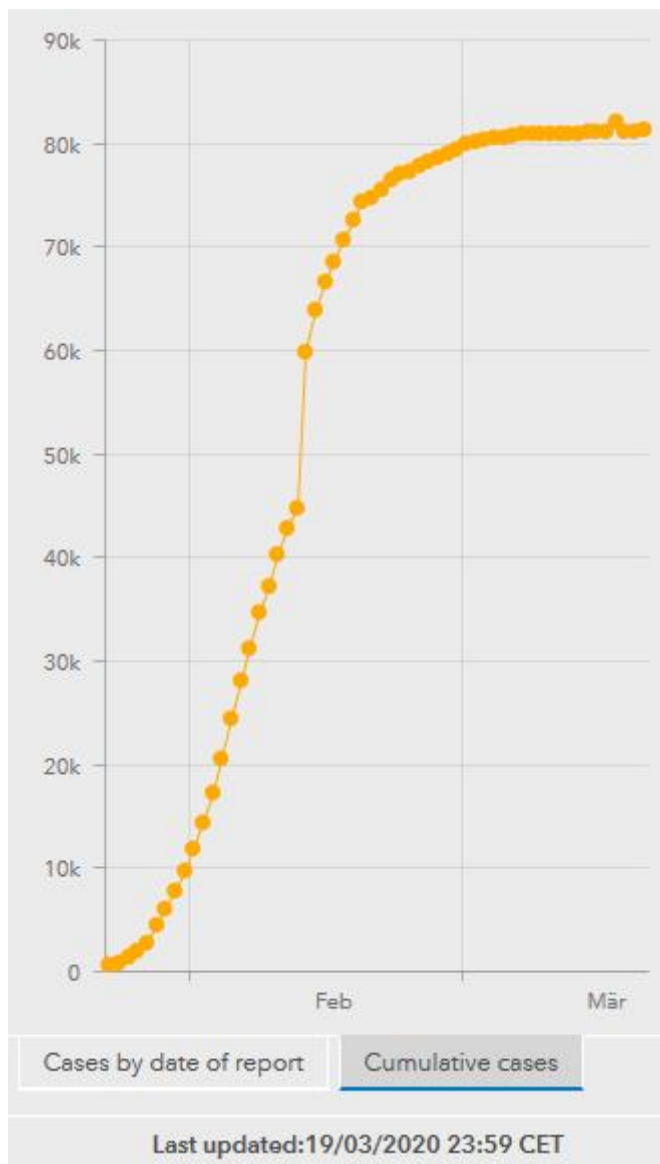


Abb.2: Entwicklung der kumulativen Covid-19 Fallzahlen in China bis Mitte März 2020 (Quelle WHO)

Man kann sich nun überlegen, wie man zum Zweck der Vorhersage ein passenderes Modell als das des rein exponentiellen Wachstums erreichen kann. Dazu kann man sich klarmachen, dass man eine Gerade als Schaubild erhält, wenn man ein rein exponentielles Wachstum vorliegen hat und die Fallzahlen logarithmiert aufträgt. Man bekommt man den Zusammenhang für die Geraden-Gleichung im Exponenten wie folgt:

$$y = \ln(I(t)) = \ln(I_0) + b \cdot t = a + b \cdot t$$

Exponiert man den Zusammenhang wieder ergibt sich:

$$I(t) = e^{a+b \cdot t}$$

Wenn nun aber durch gewisse Einflüsse die logarithmierten Fallzahlen nicht konstant anwachsen, dann kann man unterstellen, dass die logarithmierten Fallzahlen keine Geraden ergeben, sondern eine gekrümmte Kurve. Dann liegt es nahe, dass man eine Kurve zweiter Ordnung für die logarithmierten Fallzahlen annimmt. Dafür erhält man einen Zusammenhang der Form:

$$I(t) = e^{a+b \cdot t+c \cdot t^2}$$

also einen quadratischen Term im Exponenten der e-Funktion.

Nun ist die Frage, wie kommt man zu den Koeffizienten a , b und c . Wenn man nur einen linearen Term $a+b*t$ im Exponenten hat, dann kann man a und b aus Achsenabschnitt und Steigung einer Regressionsgeraden für die logarithmierten Fallzahlen errechnen. Im Falle des quadratischen Terms im Exponenten braucht man eine Funktion, die statt einer linearen eine polynomische Anpassung im Sinne des kleinsten Fehlerquadrats an die logarithmierten Falldaten macht. In Microsoft Excel gibt es dazu beispielsweise die Auswahlmöglichkeit beim Einfügen einer Trendlinie. In Mathematikprogrammen wie Octave oder Matlab gibt es dafür spezielle Funktionen, die aus den logarithmierten Falldaten und der Zahl an Tagen seit Ausbruch die Werte von a , b und c ermitteln.

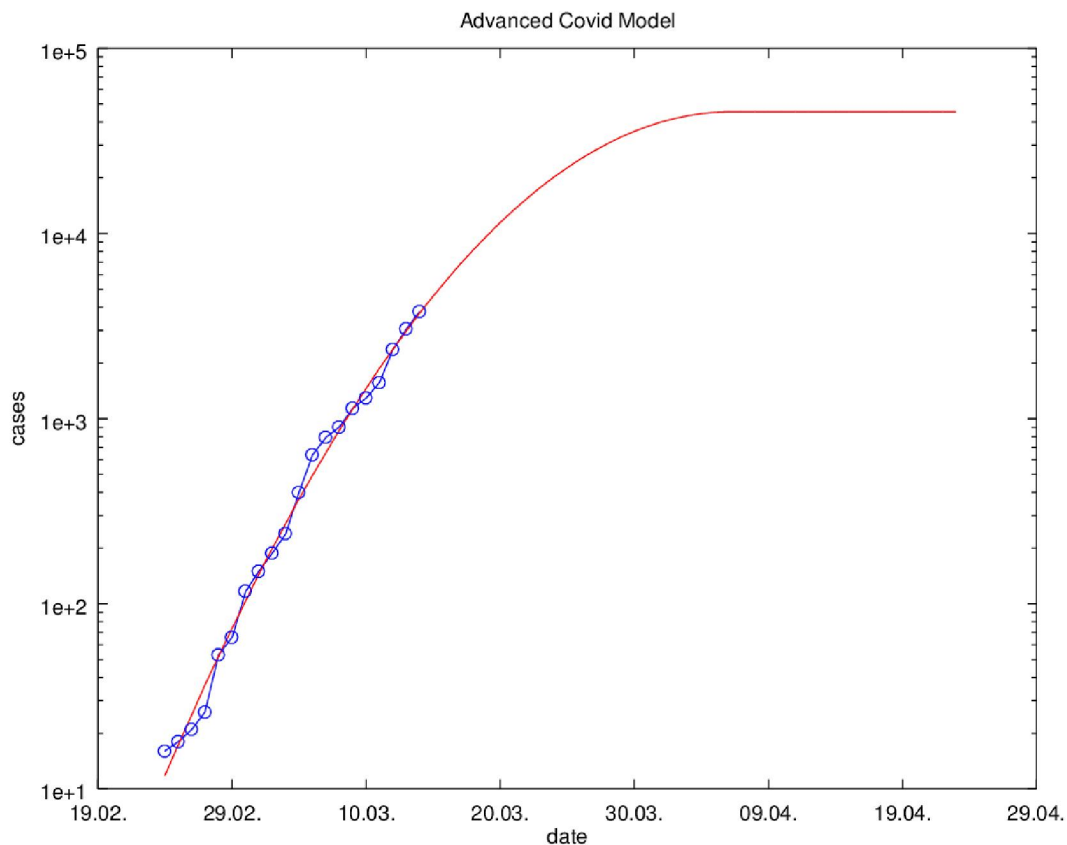
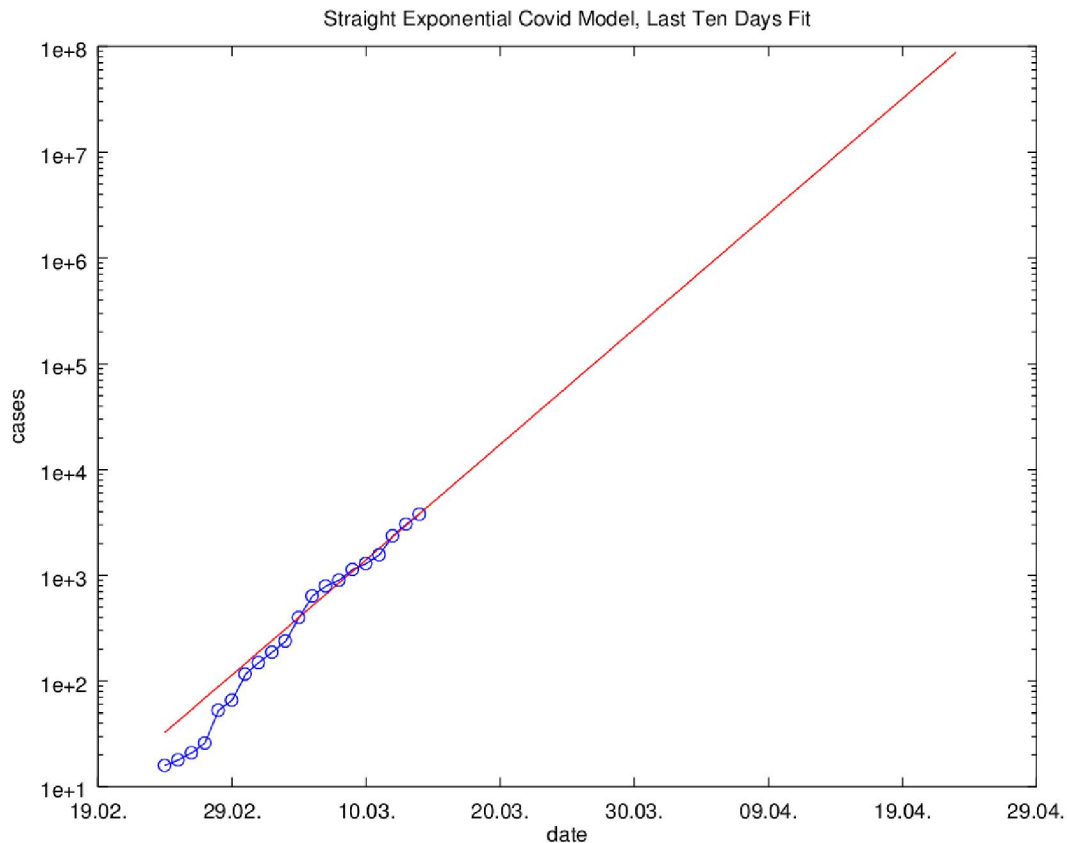


Abb. 3: Fortgeschrittenes Modell für die Covid-19 Fallzahlen Prognose

Der quadratische Term im Exponent führt schließlich zu einem parabelartigen Verlauf, der aber nach Überschreiten des Höhepunkts auch wieder abnimmt. Bei kumulativen Fallzahlen ist dies nicht möglich. Man muss daher davon ausgehen, dass dies auch nur eine Approximation ist, die davon ausgeht, dass sich der lineare Anstieg abflacht und im Grenzfall konstant bleibt. Man kann daher davon ausgehen, dass die Fallzahlen ab Erreichen des Scheitelpunkts konstant bleiben, was auch der Fall sein dürfte, wenn die Maßnahmen und Einflüsse, die zur Stagnation geführt haben, erhalten bleiben.

Wie gut das Modell ist, wird sich aber erst beweisen müssen. Ob das Modell gerechtfertigt ist, wird sich an der Krümmung des Wachstums zeigen, die auch dazu führen muss, dass die differentiell berechnete Verdoppelungszahl deutlich zunimmt. Diese differentielle Verdoppelungszahl kann man anstatt aus allen Fallzahlen beispielsweise aus den Fallzahlen der letzten 10 Tage durch einen begrenzten linearen Fit bestimmen.



Bereits jetzt, Mitte März kann man damit erkennen, dass diese differentiell bestimmte Verdoppelungszahl von anfänglichen 2.5Tagen auf jetzt 3 Tage zunimmt. Allerdings muss dieser Wert noch erheblich mehr zunehmen, damit das Gesundheitssystem nicht doch noch in echte Schwierigkeiten kommt.

Das hier zugrunde gelegte Modell ist durch ein Matlab Simulationsskript im Anhang beschrieben und ist auch unter der (fast) kompatiblen Freeware „Octave“ lauffähig. Am heutigen Tag generiert es die obigen Grafiken und folgendes textuelles Ergebnis:

>>>

Covid-19 statistical projection

Advanced model stagnation date: 11.04.2020, max cases: 96679, max deaths: 267, mortality: 0.28

doubling time (from last 10 days): 3.09, quadratic term of adv. model exponent, -0.0040

Status as of: 21.03.2020, Author: Bernd Laquai

Data Source: Robert Koch-Institute, Berlin, Germany

<<<

Die zugrunde gelegte Eingabe-Datendatei „FallzahlenRki.txt“, die aus den Angaben des Robert Koch-Instituts (www.rki.de) erzeugt wurde, ist auf dieser Webseite verfügbar und wird bestmöglich aktualisiert (ohne jede Gewähr).

Anhang: Matlab Skript zur Covid-19 Prognose

```
%Covid-19 Simulation, Bernd Laquai 21.03.20
clear;
close('all');
if (isempty(regexpi(version, 'R')) matlabSys=0;
else matlabSys=1;
end
infid = fopen('FallzahlenRKI.txt');
rkiLines = textscan(infid, '%s', 'delimiter', '\n');
fclose(infid);
nrDays = length(rkiLines{1});
datesRki = zeros(nrDays, 1);
casesRki = zeros(nrDays, 1);
deathsRki = zeros(nrDays, 1);
k=0;
for i = 1:nrDays
    if (~isempty(char(rkiLines{1}(i))))
        if matlabSys
            lineData = textscan(char(rkiLines{1}(i)), '%s', 'delimiter',
'\t');
        else
            lineData = textscan(char(rkiLines{1}(i)), '%s', 'delimiter', '
'); %octave
        end
        datesRki(i)=datenum(lineData{1}(1), 'dd.mm.yyyy');
        casesRki(i)=str2double(char(lineData{1}(2)));
        deathsRki(i)=str2double(char(lineData{1}(3)));
        k=k+1;
    end
end
nrDays=k;
datesRki=datesRki(1:nrDays);
casesRki=casesRki(1:nrDays);
deathsRki=deathsRki(1:nrDays);
days = datesRki-datesRki(1,1)+1;
mortality = deathsRki./casesRki;
logCases=log(casesRki);

%
if(0)
figure;plot(datesRki, casesRki, '-o');
set(gca, 'XTick', datesRki);
datetick('x', 'dd.mm.', 'keepticks');
datetick('x', 'dd.mm. ');
title('Covid-19 Cases in Germany');

figure;plot(datesRki, deathsRki, '-o');
set(gca, 'XTick', datesRki);
datetick('x', 'dd.mm. ');
title('Covid-19 Deaths in Germany');
end

%1st order fit
[p1,S] = polyfit(days, logCases, 1);
daysProj = (1:60);
datesProj = datesRki(1)+daysProj-1;
fitCases = exp(p1(1)*daysProj+p1(2));
maxCases = round(fitCases(end));
maxDeaths = round(maxCases*mortality(end));
figure;semilogy(datesRki, casesRki, '-o'); hold on;
```

```

semilogy(datesProj, fitCases,'r'); hold off;
set(gca,'XTick',datesRki);
datetick('x','dd.mm.');
```

title('Straight Exponential Covid Model');

```

ylabel('cases');
xlabel('date');
```

%

%1st order fit only last 10 data points

```

[p1d,S] = polyfit(days(end-10:end),logCases(end-10:end),1);
daysProj = (1:60);
datesProj = datesRki(1)+daysProj-1;
T2 = log(2)/p1d(1); %doubling time
fitCases = exp(p1d(1)*daysProj+p1d(2));
maxCases = round(fitCases(end));
maxDeaths = round(maxCases*mortality(end));
figure;semilogy(datesRki, casesRki,'-o'); hold on;
semilogy(datesProj, fitCases,'r'); hold off;
set(gca,'XTick',datesRki);
datetick('x','dd.mm.');
```

title('Straight Exponential Covid Model, Last Ten Days Fit');

```

ylabel('cases');
xlabel('date');
```

%

%2nd order fit

```

[p2,S] = polyfit(days,logCases,2);
daysProj = (1:60);
datesProj = datesRki(1)+daysProj-1;
fitCases = exp(p2(1)*daysProj.^2+p2(2)*daysProj+p2(3));
dFitCases = diff(fitCases);
stagIdx = find(dFitCases<0,1,'first'); %stagnation idx
maxCases = round(fitCases(stagIdx));
maxDeaths = round(maxCases*mortality(end));
fitCases(stagIdx+1:end) = maxCases;
figure;semilogy(datesRki, casesRki,'-o'); hold on;
semilogy(datesProj, fitCases,'r'); hold off;
set(gca,'XTick',datesRki);
datetick('x','dd.mm.');
```

title('Advanced Covid Model');

```

ylabel('cases');
xlabel('date');
```

%

%Mortality

```

figure;plot(datesRki, mortality*100,'-o');
set(gca,'XTick',datesRki);
datetick('x','dd.mm.');
```

title('Covid-19 Mortality in Germany');

```

ylabel('%');
xlabel('date');
```

fprintf(1, 'Covid-19 statistical projection\n\n');

```

fprintf(1, 'Advanced model stagnation date: %s, max cases: %d, max deaths:
%d, mortality: %.2f\n', datestr(datesProj(stagIdx),'dd.mm.yyyy'), maxCases,
maxDeaths, mortality(end)*100);
fprintf(1, 'doubling time (from last 10 days): %.2f, quadratic term of adv.
model exponent, %.4f\n', T2, p2(1));
fprintf(1, 'Status as of: %s, Author: Bernd Laquai\n',
datestr(now,'dd.mm.yyyy'));
fprintf(1, 'Data Source: Robert Koch-Institute, Berlin, Germany\n');
```

fclose('all');