COVID19 Data Analysis

Robert Gregg

University of Pittsburgh

March 18th 2020

Import the data

The data orgininates from the HGIS Lab at the University of Washinton

```
% Specify location and name of the file
filename = 'virus.csv';

% Format for each line of text
formatSpec = strcat('%{yyyy-MM-dd}D',repmat('%s',1,326));

% Read the data into memory
warning off %Warns about changing variable names for table
virusRaw = readtable(filename,'Format',formatSpec);
head(virusRaw,10)
```

```
ans = 10×327 table
```

...

	datetime	anhui	beijing	chongqing	fujian	gansu
1	2020-01-21	'0-3'	'10'	'5'	'0'	'0'
2	2020-01-22	'1-4'	'14'	'6'	'1'	'0'
3	2020-01-23	'9-4'	'22'	'9'	'5-2'	'2'
4	2020-01-24	'15-4'	'36-0-1'	'27-13'	'10-2'	'2'
5	2020-01-25	'39-4'	'41-0-2'	'57'	'18-20'	'4'
6	2020-01-26	'60-4'	'68-0-2'	'75'	'35-20'	'7'
7	2020-01-27	'70'	'72-0-2'	'110-0-0-1'	'56'	'14'
8	2020-01-28	'152-0-2-0'	'91-0-4-1'	'147-0-0-0'	'82-0-0-0'	'24-0-0-0'
9	2020-01-29	'152-0-2-0'	'102-0-4-1'	'147-0-1-0'	'84-0-0-0'	'24-0-0-0'
10	2020-01-30	'200-0-3-0'	'114-0-5-1'	'165-0-1-0'	'101-0-0-0'	'26-0-0-0'

Clean the data for analysis

```
%The list of numbers represent aggregated-active-recovered-death
%Convert the strings into vectors of numbers
virus = varfun(@(x) sscanf(x{1},'%f'),virusRaw,'GroupingVariables','datetime','OutputFormat','

%If an entry is empty, then the location has no reported cases
```

```
virus(cellfun(@isempty,virus)) = {[0,0,0,0]};

%Let's just look at the aggregated cases for now
cases = cellfun(@(x) x(1),virus);
```

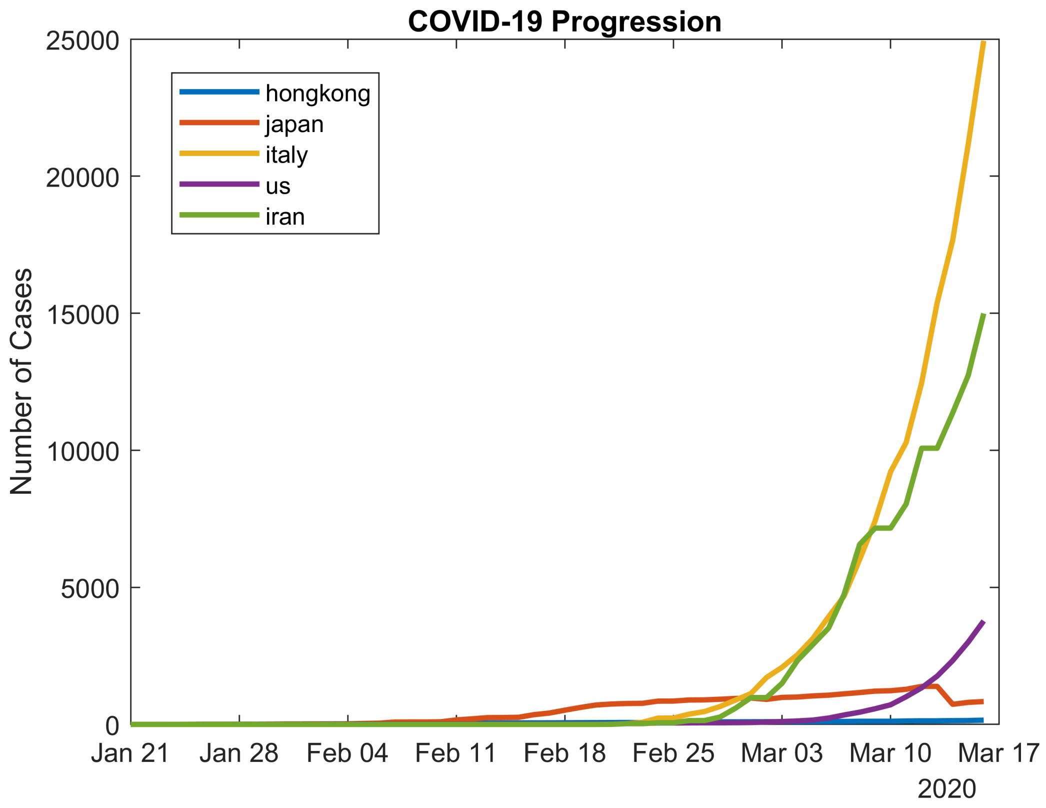
Plotting a few countries

```
%Picking some country names
countries = {'hongkong','southkorea','japan','us','italy','iran'};

%Find where they are located in the data
countryNames = virusRaw.Properties.VariableNames(2:end);
countryIdx = ismember(countryNames,countries);

%Generate a nice plot
fig1 = plot(virusRaw.datetime,cases(:,countryIdx),'linewidth',2);
ylabel('Number of Cases')
title('COVID-19 Progression')
legend(countryNames(countryIdx),'location','best')

%Prevent scientific notation on y axis
ax = ancestor(fig1, 'axes');
ax{1}.YAxis.Exponent = 0;
ytickformat('%.0f')
```



Fitting a Model to Disease Progression

```
%Choose a country to model
myCountry = 'us';
myCountryIdx = ismember(countryNames,myCountry);

%Choose a model, here I chose a logistic growth model
logisticfit = @(L,k,t0,x) L./(1+exp(-k.*(x-t0)));

% Extract the data for your country
timePast = datenum(virusRaw.datetime) - min(datenum(virusRaw.datetime));
caseNumbers = cases(:,myCountryIdx);

% Define a start points for parameters
xGuess = [5000 .5 50];

%Fit the model with the fit() function
fitfun = fittype( logisticfit );
[fittedCurve,goodnessOfFit] = fit(timePast,caseNumbers,fitfun,'StartPoint',xGuess)
```

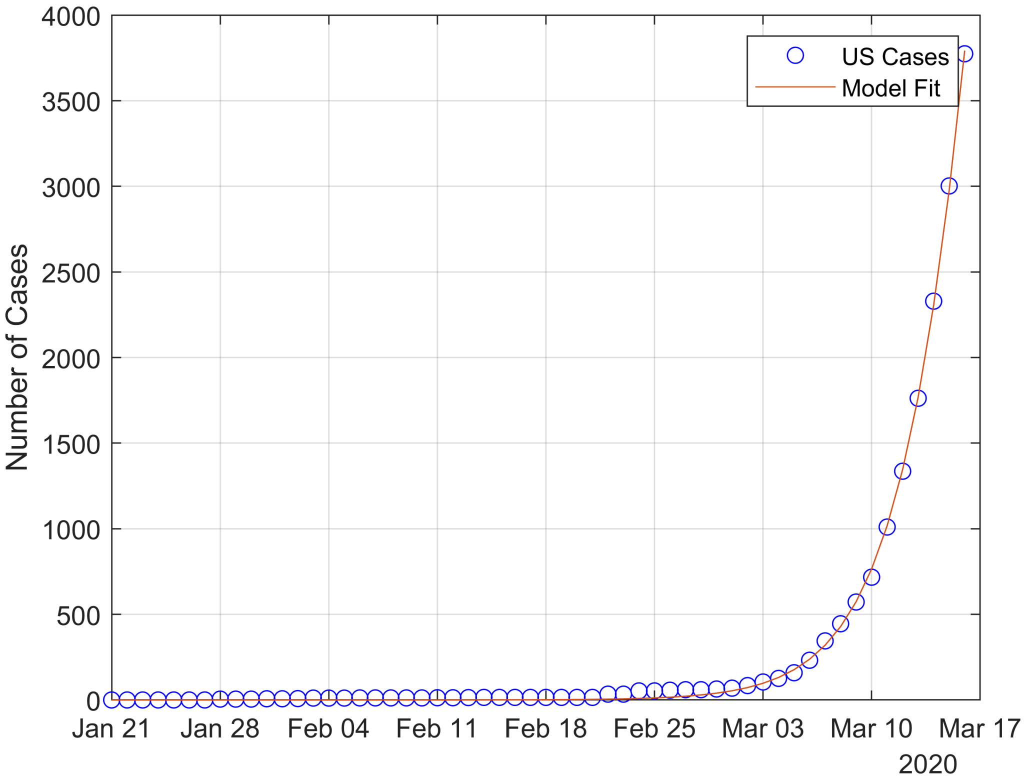
```
fittedCurve = 

     General model:
     fittedCurve(x) = L./(1+exp(-k.*(x-t0)))
     Coefficients (with 95% confidence bounds):
       L =   1.759e+04  (1.26e+04, 2.258e+04)
       k =      0.2998  (0.2884, 0.3111)
       t0 =       59.31  (57.97, 60.65)
goodnessOfFit = struct with fields:

           sse: 1.8893e+04
       rsquare: 0.9994
           dfe: 53
    adjrsquare: 0.9994
          rmse: 18.8805
```

Plot the Model Against the Data

```
% Plot results
scatter(virusRaw.datetime, caseNumbers, 'bo','DisplayName','US Cases')
hold on
plot(virusRaw.datetime,fittedCurve(timePast),'DisplayName','Model Fit')
box on
legend
ylabel('Number of Cases')
grid on
hold off
```



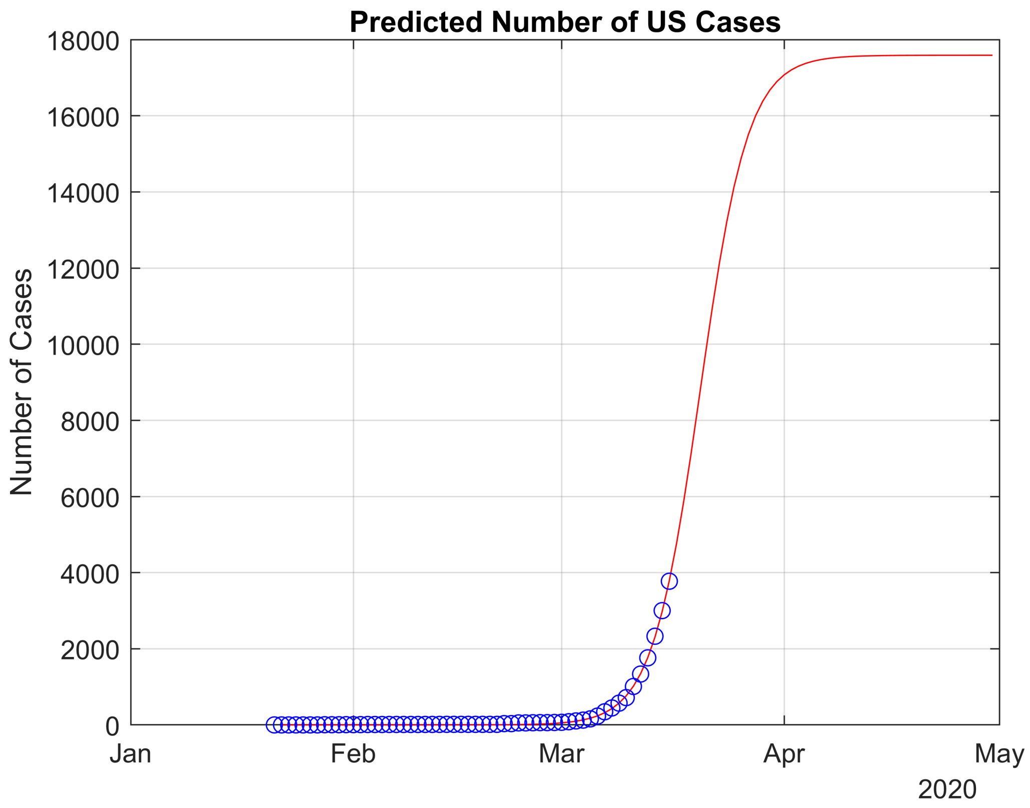
Extrapolate the Model

```
%Plot 100 days past Jan-21 (first recorded datapoint)
daysPast = 1:100;

%Calculate the number of cases
predExtrap = fittedCurve(daysPast);

%Convert dats past to dates
extrapDates = datetime(datevec( min(datenum(virusRaw.datetime))+daysPast ));

plot(extrapDates,predExtrap,'r')
hold on
scatter(virusRaw.datetime, caseNumbers, 'bo','DisplayName','US Cases')
ylabel('Number of Cases')
title('Predicted Number of US Cases')
grid on
```


Predicted Number of US Cases
Number of Cases
18000
16000
14000
12000
10000
8000
6000
4000
2000
0
Jan
Feb
Mar
Apr
May
2020