Corona_Analysis

Steven Smith, PhD

3/18/2020

Contents

The 2019-2020 Coronavirus Pandemic Analysis

Contact: Smith Research

BACKGROUND & APPROACH

I wanted to track and trend the coronavirus outbreak on my own curiosity. There are some interesting questions that may fall out of this, as it is a very historic moment, including scientifically and analytically (we have a large amount of data being shared across the globe, analyzed in real-time). The world has come to a halt because of it.
This analysis attempts to answer the following questions (more to come):

1. What does the trend of the pandemic look like to date?

2. What are future case predictions based on historical model?
3. What interesting quirks or patterns emerge?

ASSUMPTIONS & LIMITATIONS: * This data is limited by the source. I realized early on that depending on source there were conflicting # of cases. Originally I was using JHU data... but this was always 'ahead' of the Our World In Data. I noticed that JHU's website was buggy- you clicked on the U.S. stats but it didn't reflect the U.S.. So I changed data sources to be more consistent with what is presented in the media (and Our World In Data has more extensive plots I can compare my own to). An interesting aside might be why the discrepancy? Was I missing something?
* Defintiions are important as is the idea that multiple varibales accumulate in things like total cases (more testing for example).

SOURCE RAW DATA: * https://ourworldindata.org/coronavirus
* https://github.com/CSSEGISandData/COVID-19/
*

INPUT DATA LOCATION: github (https://github.com/sbs87/coronavirus/tree/master/data)

OUTPUT DATA LOCATIOn: github (https://github.com/sbs87/coronavirus/tree/master/results)

TIMESTAMP

Start: ##—— Sat Apr 18 23:40:02 2020 ——##

PRE-ANALYSIS

The following sections are outside the scope of the 'analysis' but are still needed to prepare everything

UPSTREAM PROCESSING/ANALYSIS

1. Google Mobility Scraping, script available at get_google_mobility.py

```
# Mobility data has to be extracted from Google PDF reports using a web scraping script (python , writt

# See get_google_mobility.py for local script

python3 get_google_mobility.py
# writes csv file of mobility data as "mobility.csv"
```

SET UP ENVIORNMENT

Load libraries and set global variables

```
# timestamp start
timestamp()
## ##------ Sat Apr 18 23:40:03 2020 ------##

# clear previous enviornment
rm(list = ls())

##-----------------------------------------
## LIBRARIES
##-----------------------------------------
library(plyr)
library(tidyverse)
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
## v ggplot2 3.3.0     v purrr   0.3.3
## v tibble  3.0.0     v dplyr   0.8.5
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::arrange()   masks plyr::arrange()
## x purrr::compact()   masks plyr::compact()
## x dplyr::count()     masks plyr::count()
## x dplyr::failwith()  masks plyr::failwith()
## x dplyr::filter()    masks stats::filter()
## x dplyr::id()        masks plyr::id()
## x dplyr::lag()       masks stats::lag()
## x dplyr::mutate()    masks plyr::mutate()
```

```
## x dplyr::rename()    masks plyr::rename()
## x dplyr::summarise() masks plyr::summarise()
## x dplyr::summarize() masks plyr::summarize()
library(ggplot2)
library(reshape2)
##
## Attaching package: 'reshape2'
## The following object is masked from 'package:tidyr':
##
##     smiths
library(plot.utils)
library(utils)
library(knitr)

##----------------------------------------

##----------------------------------------
# GLOBAL VARIABLES
##----------------------------------------
user_name <- Sys.info()["user"]
working_dir <- paste0("/Users/", user_name, "/Projects/coronavirus/")  # don't forget trailing /
results_dir <- paste0(working_dir, "results/")  # assumes diretory exists
results_dir_custom <- paste0(results_dir, "custom/")  # assumes diretory exists


Corona_Cases.source_url <- "https://github.com/CSSEGISandData/COVID-19/raw/master/csse_covid_19_data/cs
Corona_Cases.US.source_url <- "https://github.com/CSSEGISandData/COVID-19/raw/master/csse_covid_19_data/
Corona_Deaths.US.source_url <- "https://github.com/CSSEGISandData/COVID-19/raw/master/csse_covid_19_dat
Corona_Deaths.source_url <- "https://github.com/CSSEGISandData/COVID-19/raw/master/csse_covid_19_data/cs

Corona_Cases.fn <- paste0(working_dir, "data/", basename(Corona_Cases.source_url))
Corona_Cases.US.fn <- paste0(working_dir, "data/", basename(Corona_Cases.US.source_url))
Corona_Deaths.fn <- paste0(working_dir, "data/", basename(Corona_Deaths.source_url))
Corona_Deaths.US.fn <- paste0(working_dir, "data/", basename(Corona_Deaths.US.source_url))
default_theme <- theme_bw() + theme(text = element_text(size = 14))  # fix this
##----------------------------------------
```

FUNCTIONS

List of functions

function_name	description
prediction_model	outputs case estumate for given log-linear moder parameters slope and intercept
make_long	converts input data to long format (specialized cases)

function_name	description
name_overlaps	outputs the column names intersection and set diffs of two data frame

```
##----------------------------------------
## FUNCTION: prediction_model
##----------------------------------------
## --- //// ----
# Takes days vs log10 (case) linear model parameters and a set of days since 100 cases and outputs a da
## --- //// ----
prediction_model<-function(m=1,b=0,days=1){
  total_cases.log<-m*days+b
  total_cases<-10^total_cases.log
  prediction<-data.frame(Days_since_100=days,Total_confirmed_cases=total_cases,Total_confirmed_cases.log
  return(prediction)
}
##----------------------------------------

##----------------------------------------
## FUNCTION: make_long
##----------------------------------------
## --- //// ----
# Takes wide-format case data and converts into long format, using date and total cases as variable/val
## --- //// ----
make_long<-function(data_in,variable.name = "Date",
                    value.name = "Total_confirmed_cases",
                    id.vars=c("case_type","Province.State","Country.Region","Lat","Long","City","Populati

long_data<-melt(data_in,
                id.vars = id.vars,
                variable.name=variable.name,
                value.name=value.name)
return(long_data)

}
##----------------------------------------

## THIS WILL BE IN UTILS AT SOME POINT
name_overlaps<-function(df1,df2){
i<-intersect(names(df1),
names(df2))
sd1<-setdiff(names(df1),
names(df2))
sd2<-setdiff(names(df2),names(df1))
cat("intersection:\n",paste(i,"\n"))
cat("in df1 but not df2:\n",paste(sd1,"\n"))
cat("in df2 but not df1:\n",paste(sd2,"\n"))
return(list("int"=i,"sd_1_2"=sd1,"sd_2_1"=sd2))
}
```

READ IN DATA

- total number of cases. current source: https://github.com/CSSEGISandData (precvious source https://ourworldindata.org/coronavirus)

```
# Q: do we want to archive previous versions? Maybe an auto git mv?

##----------------------------------------
## Download and read in latest data from github
##----------------------------------------
download.file(Corona_Cases.source_url, destfile = Corona_Cases.fn)
Corona_Totals.raw <- read.csv(Corona_Cases.fn, header = T, stringsAsFactors = F)

download.file(Corona_Cases.US.source_url, destfile = Corona_Cases.US.fn)
Corona_Totals.US.raw <- read.csv(Corona_Cases.US.fn, header = T, stringsAsFactors = F)

download.file(Corona_Deaths.source_url, destfile = Corona_Deaths.fn)
Corona_Deaths.raw <- read.csv(Corona_Deaths.fn, header = T, stringsAsFactors = F)

download.file(Corona_Deaths.US.source_url, destfile = Corona_Deaths.US.fn)
Corona_Deaths.US.raw <- read.csv(Corona_Deaths.US.fn, header = T, stringsAsFactors = F)

# latest date on all data:
paste("US deaths:", names(Corona_Deaths.US.raw)[ncol(Corona_Deaths.US.raw)])
```

```
## [1] "US deaths: X4.18.20"
```

```
paste("US total:", names(Corona_Totals.US.raw)[ncol(Corona_Totals.US.raw)])
```

```
## [1] "US total: X4.18.20"
```

```
paste("World deaths:", names(Corona_Deaths.raw)[ncol(Corona_Deaths.raw)])
```

```
## [1] "World deaths: X4.18.20"
```

```
paste("World total:", names(Corona_Totals.raw)[ncol(Corona_Totals.raw)])
```

```
## [1] "World total: X4.18.20"
```

PROCESS DATA

- Convert to long format
- Fix date formatting/convert to numeric date
- Log10 transform total # cases

```
##----------------------------------------
## Combine death and total data frames
##----------------------------------------
Corona_Totals.raw$case_type<-"total"
Corona_Totals.US.raw$case_type<-"total"
Corona_Deaths.raw$case_type<-"death"
Corona_Deaths.US.raw$case_type<-"death"

# for some reason, Population listed in US death file but not for other data... Weird. When combining,
Corona_Totals.US.raw$Population<-"NA"
Corona_Totals.raw$Population<-"NA"
```

```
Corona_Deaths.raw$Population<-"NA"

Corona_Cases.raw<-rbind(Corona_Totals.raw,Corona_Deaths.raw)
Corona_Cases.US.raw<-rbind(Corona_Totals.US.raw,Corona_Deaths.US.raw)
#TODO: custom utils- setdiff, intersect names... option to output in merging too
##----------------------------------------
# prepare raw datasets for eventual combining
##----------------------------------------
Corona_Cases.raw$City<-"NA" # US-level data has Cities
Corona_Cases.US.raw$Country_Region<-"US_state" # To differentiate from World-level stats

Corona_Cases.US.raw<-plyr::rename(Corona_Cases.US.raw,c("Province_State"="Province.State",
                                                        "Country_Region"="Country.Region",
                                                        "Long_"="Long",
                                                        "Admin2"="City"))



##----------------------------------------
## Convert to long format
##----------------------------------------
#JHU has a gross file format. It's in wide format with each column is the date in MM/DD/YY. So read thi
# Furthermore, the World and US level data is formatted differently, containing different columns, etc.

Corona_Cases.long<-rbind(make_long(select(Corona_Cases.US.raw,-c(UID,iso2,iso3,code3,FIPS,Combined_Key))
make_long(Corona_Cases.raw))


##----------------------------------------
## Fix date formatting, convert to numeric date
##----------------------------------------
Corona_Cases.long$Date<-gsub(Corona_Cases.long$Date,pattern = "^X",replacement = "0") # leading 0 read
Corona_Cases.long$Date<-gsub(Corona_Cases.long$Date,pattern = "20$",replacement = "2020") # ends in .20
Corona_Cases.long$Date<-as.Date(Corona_Cases.long$Date,format = "%m.%d.%y")
Corona_Cases.long$Date.numeric<-as.numeric(Corona_Cases.long$Date)

kable(table(select(Corona_Cases.long,c("Country.Region","case_type"))),caption = "Number of death and t
```

Table 2: Number of death and total case longitudinal datapoints per geographical region

	death	total
Afghanistan	88	88
Albania	88	88
Algeria	88	88
Andorra	88	88
Angola	88	88
Antigua and Barbuda	88	88
Argentina	88	88
Armenia	88	88
Australia	704	704
Austria	88	88
Azerbaijan	88	88
Bahamas	88	88

	death	total
Bahrain	88	88
Bangladesh	88	88
Barbados	88	88
Belarus	88	88
Belgium	88	88
Belize	88	88
Benin	88	88
Bhutan	88	88
Bolivia	88	88
Bosnia and Herzegovina	88	88
Botswana	88	88
Brazil	88	88
Brunei	88	88
Bulgaria	88	88
Burkina Faso	88	88
Burma	88	88
Burundi	88	88
Cabo Verde	88	88
Cambodia	88	88
Cameroon	88	88
Canada	1320	1320
Central African Republic	88	88
Chad	88	88
Chile	88	88
China	2904	2904
Colombia	88	88
Congo (Brazzaville)	88	88
Congo (Kinshasa)	88	88
Costa Rica	88	88
Cote d'Ivoire	88	88
Croatia	88	88
Cuba	88	88
Cyprus	88	88
Czechia	88	88
Denmark	264	264
Diamond Princess	88	88
Djibouti	88	88
Dominica	88	88
Dominican Republic	88	88
Ecuador	88	88
Egypt	88	88
El Salvador	88	88
Equatorial Guinea	88	88
Eritrea	88	88
Estonia	88	88
Eswatini	88	88
Ethiopia	88	88
Fiji	88	88
Finland	88	88
France	968	968
Gabon	88	88
Gambia	88	88

	death	total
Georgia	88	88
Germany	88	88
Ghana	88	88
Greece	88	88
Grenada	88	88
Guatemala	88	88
Guinea	88	88
Guinea-Bissau	88	88
Guyana	88	88
Haiti	88	88
Holy See	88	88
Honduras	88	88
Hungary	88	88
Iceland	88	88
India	88	88
Indonesia	88	88
Iran	88	88
Iraq	88	88
Ireland	88	88
Israel	88	88
Italy	88	88
Jamaica	88	88
Japan	88	88
Jordan	88	88
Kazakhstan	88	88
Kenya	88	88
Korea, South	88	88
Kosovo	88	88
Kuwait	88	88
Kyrgyzstan	88	88
Laos	88	88
Latvia	88	88
Lebanon	88	88
Liberia	88	88
Libya	88	88
Liechtenstein	88	88
Lithuania	88	88
Luxembourg	88	88
Madagascar	88	88
Malawi	88	88
Malaysia	88	88
Maldives	88	88
Mali	88	88
Malta	88	88
Mauritania	88	88
Mauritius	88	88
Mexico	88	88
Moldova	88	88
Monaco	88	88
Mongolia	88	88
Montenegro	88	88
Morocco	88	88

	death	total
Mozambique	88	88
MS Zaandam	88	88
Namibia	88	88
Nepal	88	88
Netherlands	440	440
New Zealand	88	88
Nicaragua	88	88
Niger	88	88
Nigeria	88	88
North Macedonia	88	88
Norway	88	88
Oman	88	88
Pakistan	88	88
Panama	88	88
Papua New Guinea	88	88
Paraguay	88	88
Peru	88	88
Philippines	88	88
Poland	88	88
Portugal	88	88
Qatar	88	88
Romania	88	88
Russia	88	88
Rwanda	88	88
Saint Kitts and Nevis	88	88
Saint Lucia	88	88
Saint Vincent and the Grenadines	88	88
San Marino	88	88
Sao Tome and Principe	88	88
Saudi Arabia	88	88
Senegal	88	88
Serbia	88	88
Seychelles	88	88
Sierra Leone	88	88
Singapore	88	88
Slovakia	88	88
Slovenia	88	88
Somalia	88	88
South Africa	88	88
South Sudan	88	88
Spain	88	88
Sri Lanka	88	88
Sudan	88	88
Suriname	88	88
Sweden	88	88
Switzerland	88	88
Syria	88	88
Taiwan*	88	88
Tanzania	88	88
Thailand	88	88
Timor-Leste	88	88
Togo	88	88

	death	total
Trinidad and Tobago	88	88
Tunisia	88	88
Turkey	88	88
Uganda	88	88
Ukraine	88	88
United Arab Emirates	88	88
United Kingdom	968	968
Uruguay	88	88
US	88	88
US_state	286440	286440
Uzbekistan	88	88
Venezuela	88	88
Vietnam	88	88
West Bank and Gaza	88	88
Western Sahara	88	88
Yemen	88	88
Zambia	88	88
Zimbabwe	88	88

```
# Decouple population and lat/long data, refactor to make it more tidy
metadata_columns<-c("Lat","Long","Population")
metadata<-unique(select(filter(Corona_Cases.long,case_type=="death"),c("Country.Region","Province.State
Corona_Cases.long<-select(Corona_Cases.long,-all_of(metadata_columns))

# Some counties are not summarized on the country level. collapse all but US
Corona_Cases.long<-rbind.fill(ddply(filter(Corona_Cases.long,!Country.Region=="US_state"),c("case_type"

# Put total case and deaths side-by-side (wide)
Corona_Cases<-spread(Corona_Cases.long,key = case_type,value = Total_confirmed_cases)

#Compute moratlity rate
Corona_Cases$mortality_rate<-Corona_Cases$death/Corona_Cases$total

#TMP
Corona_Cases<-plyr::rename(Corona_Cases,c("total"="Total_confirmed_cases","death"="Total_confirmed_deat

##----------------------------------------
## log10 transform total # cases
##----------------------------------------
Corona_Cases$Total_confirmed_cases.log<-log(Corona_Cases$Total_confirmed_cases,10)
Corona_Cases$Total_confirmed_deaths.log<-log(Corona_Cases$Total_confirmed_deaths,10)
##----------------------------------------

##----------------------------------------
## Compute # of days since 100th for US data
##----------------------------------------

# Find day that 100th case was found for Country/Province. NOTE: Non US countries may have weird provin
# TODO: consider city-level summary as well. This data may be sparse

Corona_Cases<-merge(Corona_Cases,ddply(filter(Corona_Cases,Total_confirmed_cases>100),c("Country.Region
```

```
Corona_Cases$Days_since_100<-Corona_Cases$Date.numeric-Corona_Cases$case100_date

##-----------------------------------------
## Add population and lat/long data (CURRENTLY US ONLY)
##-----------------------------------------

kable(filter(metadata,(is.na(Country.Region) | is.na(Population) )) %>% select(c("Country.Region","Provi
```

Table 3: Regions for which either population or Country is NA

Country.Region	Province.State	City

```
# Drop missing data
metadata<-filter(metadata,!(is.na(Country.Region) | is.na(Population) ))
# Convert remaining pop to numeric
metadata$Population<-as.numeric(metadata$Population)
```

```
## Warning: NAs introduced by coercion
```

```
# Add metadata to cases
Corona_Cases<-merge(Corona_Cases,metadata,all.x = T)

##-----------------------------------------
## Compute total and death cases relative to population
##-----------------------------------------

Corona_Cases$Total_confirmed_cases.per100<-100*Corona_Cases$Total_confirmed_cases/Corona_Cases$Populati
Corona_Cases$Total_confirmed_deaths.per100<-100*Corona_Cases$Total_confirmed_deaths/Corona_Cases$Popula


##-----------------------------------------
## Filter df for US state-wide stats
##-----------------------------------------

Corona_Cases.US_state<-filter(Corona_Cases,Country.Region=="US_state" & Total_confirmed_cases>0 )
kable(table(select(Corona_Cases.US_state,c("Province.State"))),caption = "Number of longitudinal datapoi
```

Table 4: Number of longitudinal datapoints (total/death) per state

Var1	Freq
Alabama	1720
Alaska	253
Arizona	506
Arkansas	1645
California	2034
Colorado	1616
Connecticut	295
Delaware	111
Diamond Princess	33
District of Columbia	34
Florida	1985
Georgia	4085
Grand Princess	34

Var1	Freq
Guam	34
Hawaii	182
Idaho	776
Illinois	2037
Indiana	2421
Iowa	1872
Kansas	1363
Kentucky	2150
Louisiana	1810
Maine	449
Maryland	761
Massachusetts	553
Michigan	2023
Minnesota	1711
Mississippi	2227
Missouri	2059
Montana	680
Nebraska	899
Nevada	311
New Hampshire	335
New Jersey	797
New Mexico	629
New York	1887
North Carolina	2515
North Dakota	667
Northern Mariana Islands	19
Ohio	2230
Oklahoma	1497
Oregon	913
Pennsylvania	1854
Puerto Rico	34
Rhode Island	196
South Carolina	1351
South Dakota	891
Tennessee	2379
Texas	4352
Utah	499
Vermont	435
Virgin Islands	34
Virginia	2939
Washington	1318
West Virginia	912
Wisconsin	1614
Wyoming	493

```
Corona_Cases.US_state<-merge(Corona_Cases.US_state,ddply(filter(Corona_Cases.US_state,Total_confirmed_ca
Corona_Cases.US_state$Days_since_100_state<-Corona_Cases.US_state$Date.numeric-Corona_Cases.US_state$cas
```

ANALYSIS

Q1: What is the trend in cases, mortality across geopgraphical regions?

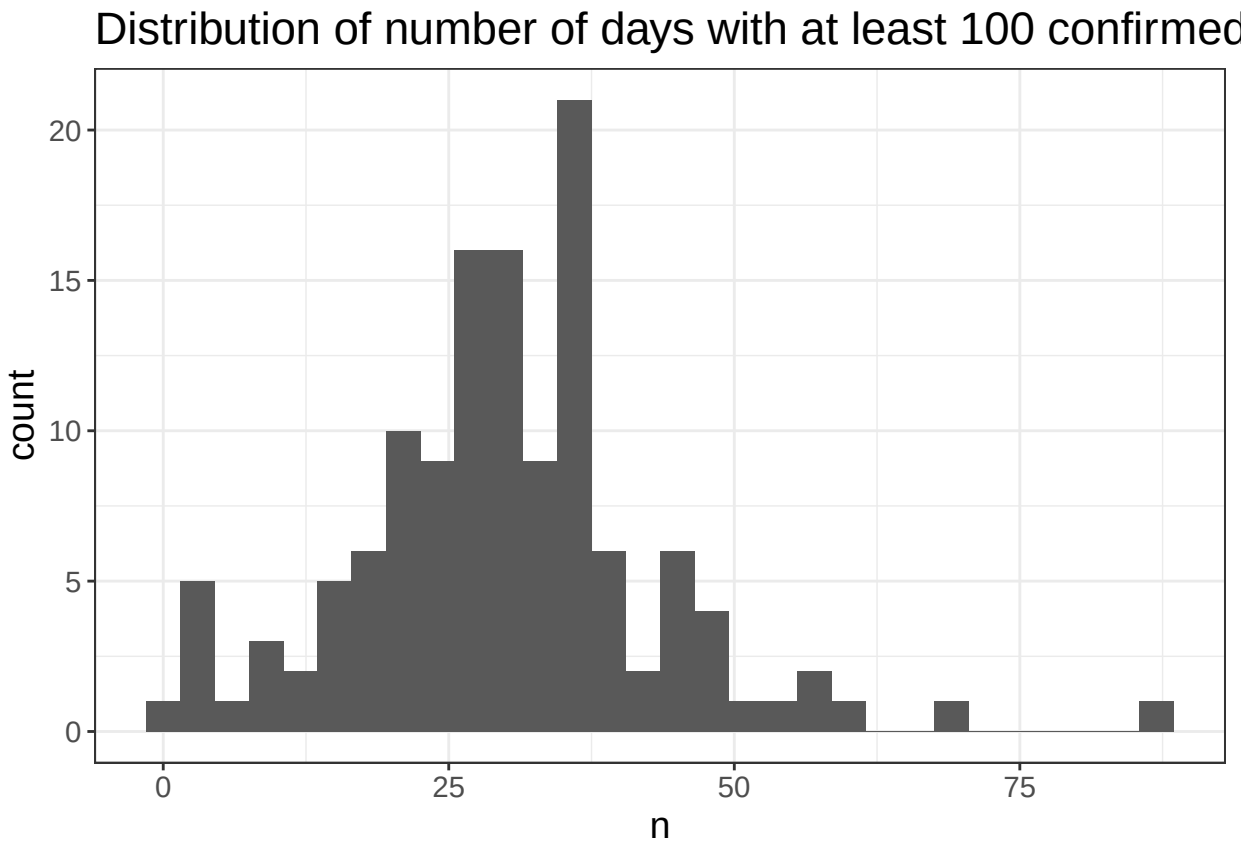
Plot # of cases vs time
* For each geographical set:
* comparative longitudinal case trend (absolute & log scale)
* comparative longitudinal mortality trend
* death vs total correlation

question	dataset	x	y	color	facet	pch	dimentions
comparative_longitudinal_case_trend	long	time	cases	geography	none (case type?)	case_type	[15, 50, 4] geography x (2 scale?) case type
comparative longitudinal case trend	long	time	cases	geography	case_type	?	[15, 50, 4] geography x (2+ scale) case type
comparative longitudinal mortality trend	wide	time	mortality rate	geography	none	none	[15, 50, 4] geography
death vs total correlation	wide	cases	deaths	geography	none	none	[15, 50, 4] geography

```
# total cases vs time
# death cases vs time
# mortality rate vs time
# death vs mortality


  # death vs mortality
  # total & death case vs time (same plot)

#<question> <x> <y> <colored> <facet> <dataset>
## trend in case/deaths over time, comapred across regions <time> <log cases> <geography*> <none> <.wid
## trend in case/deaths over time, comapred across regions <time> <cases> <geography*> <case_type> <.lo
## trend in mortality rate over time, comapred across regions <time> <mortality rate> <geography*> <non
## how are death/mortality related/correlated? <time> <log cases> <geography*> <none>
## how are death and case load correlated? <cases> <deaths>

# lm for each?? - > apply lm from each region starting from 100th case. m, b associated with each.
    # input: geographical regsion, logcase vs day (100th case)
    # output: m, b for each geographical region ID



#total/death on same plot-  diffeer by 2 logs, so when plotting log, use pch. when plotting absolute, n
#when plotting death and case on same, melt.

#CoronaCases - > filter sets (3)
```

```
  #world - choose countries with sufficent data

N<-ddply(filter(Corona_Cases,Total_confirmed_cases>100),c("Country.Region"),summarise,n=length(Country.
ggplot(filter(N,n<100),aes(x=n))+
  geom_histogram()+
  default_theme+
  ggtitle("Distribution of number of days with at least 100 confirmed cases for each region")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
kable(arrange(N,-n),caption="Sorted number of days with at least 100 confirmed cases")
```

Table 6: Sorted number of days with at least 100 confirmed cases

Country.Region	n
US_state	8246
China	88
Diamond Princess	69
Korea, South	59
Japan	58
Italy	56
Iran	53
Singapore	50
France	49
Germany	49
Spain	48
US	47

Country.Region	n
Switzerland	45
United Kingdom	45
Belgium	44
Netherlands	44
Norway	44
Sweden	44
Austria	42
Malaysia	41
Australia	40
Bahrain	40
Denmark	40
Canada	39
Qatar	39
Iceland	38
Brazil	37
Czechia	37
Finland	37
Greece	37
Iraq	37
Israel	37
Portugal	37
Slovenia	37
Egypt	36
Estonia	36
India	36
Ireland	36
Kuwait	36
Philippines	36
Poland	36
Romania	36
Saudi Arabia	36
Indonesia	35
Lebanon	35
San Marino	35
Thailand	35
Chile	34
Pakistan	34
Luxembourg	33
Peru	33
Russia	33
Ecuador	32
Slovakia	32
South Africa	32
United Arab Emirates	32
Armenia	31
Colombia	31
Croatia	31
Mexico	31
Panama	31
Serbia	31
Taiwan*	31
Turkey	31

Country.Region	n
Argentina	30
Bulgaria	30
Latvia	30
Algeria	29
Costa Rica	29
Dominican Republic	29
Hungary	29
Uruguay	29
Andorra	28
Bosnia and Herzegovina	28
Jordan	28
Lithuania	28
Morocco	28
New Zealand	28
North Macedonia	28
Vietnam	28
Albania	27
Cyprus	27
Malta	27
Moldova	27
Brunei	26
Burkina Faso	26
Sri Lanka	26
Tunisia	26
Ukraine	25
Azerbaijan	24
Ghana	24
Kazakhstan	24
Oman	24
Senegal	24
Venezuela	24
Afghanistan	23
Cote d'Ivoire	23
Cuba	22
Mauritius	22
Uzbekistan	22
Cambodia	21
Cameroon	21
Honduras	21
Nigeria	21
West Bank and Gaza	21
Belarus	20
Georgia	20
Bolivia	19
Kosovo	19
Kyrgyzstan	19
Montenegro	19
Congo (Kinshasa)	18
Kenya	17
Niger	16
Guinea	15
Rwanda	15

Country.Region	n
Trinidad and Tobago	15
Paraguay	14
Bangladesh	13
Djibouti	11
El Salvador	10
Guatemala	9
Madagascar	8
Mali	7
Congo (Brazzaville)	4
Jamaica	4
Gabon	2
Somalia	2
Tanzania	2
Ethiopia	1

```
# Pick top 15 countries with data
max_colors<-12
# find way to fix this- China has diff provences. Plot doesnt look right...
sufficient_data<-arrange(filter(N,!Country.Region %in% c("US_state", "Diamond Princess")),-n)[1:max_colo
kable(sufficient_data,caption = paste0("Top ",max_colors," countries with sufficient data"))
```

Table 7: Top 12 countries with sufficient data

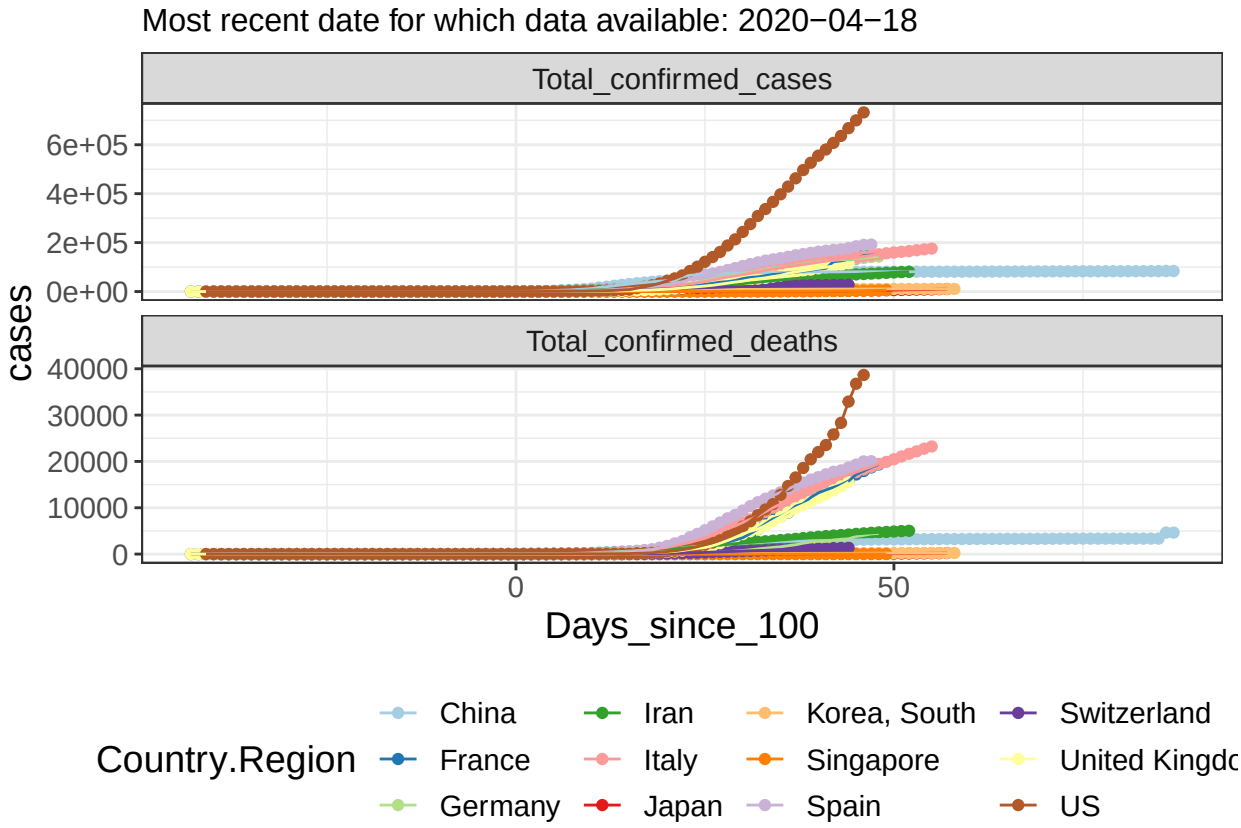
Country.Region	n
China	88
Korea, South	59
Japan	58
Italy	56
Iran	53
Singapore	50
France	49
Germany	49
Spain	48
US	47
Switzerland	45
United Kingdom	45

```
Corona_Cases.world<-filter(Corona_Cases,Country.Region %in% c(sufficient_data$Country.Region))


  #us
  #    - by state
Corona_Cases.US<-filter(Corona_Cases,Country.Region=="US" & Total_confirmed_cases>0)
# summarize
#!City %in% c("Unassigned")
  #    - specific cities
#mortality_rate!=Inf & mortality_rate<=1
Corona_Cases.UScity<-filter(Corona_Cases,Province.State %in% c("Pennsylvania","Maryland","New York","New

measure_vars_long<-c("Total_confirmed_cases.log","Total_confirmed_cases","Total_confirmed_deaths","Total
```

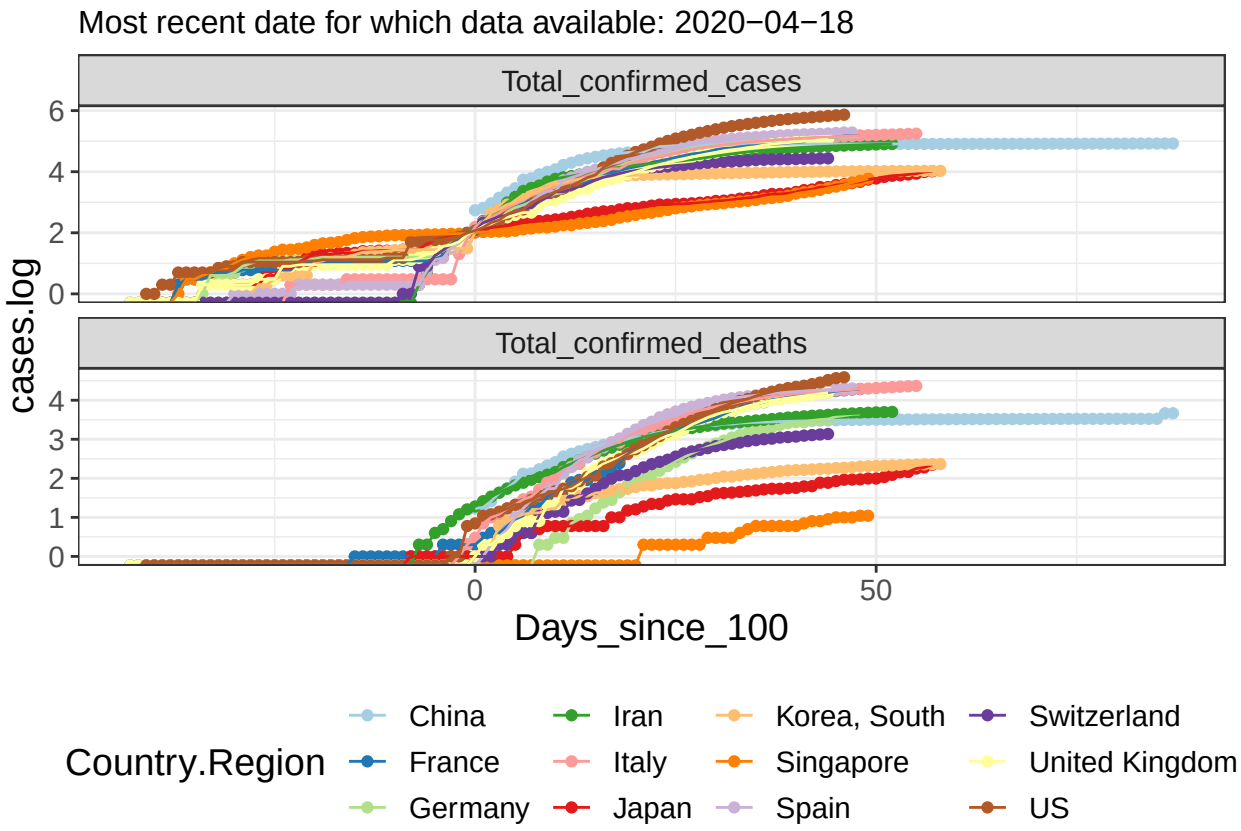
```
melt_arg_list<-list(variable.name = "case_type",value.name = "cases",measure.vars = c("Total_confirmed_
melt_arg_list$data=NULL


melt_arg_list$data=select(Corona_Cases.world,-ends_with(match = "log"))
Corona_Cases.world.long<-do.call(melt,melt_arg_list)
melt_arg_list$data=select(Corona_Cases.UScity,-ends_with(match = "log"))
Corona_Cases.UScity.long<-do.call(melt,melt_arg_list)
melt_arg_list$data=select(Corona_Cases.US_state,-ends_with(match = "log"))
Corona_Cases.US_state.long<-do.call(melt,melt_arg_list)

Corona_Cases.world.long$cases.log<-log(Corona_Cases.world.long$cases,10)
Corona_Cases.US_state.long$cases.log<-log(Corona_Cases.US_state.long$cases,10)
Corona_Cases.UScity.long$cases.log<-log(Corona_Cases.UScity.long$cases,10)


# what is the current death and total case load for US? For world? For states?
#-absolute
#-log

# what is mortality rate (US, world)
#-absolute

#how is death and case correlated? (US, world)
#-absolute
```

```
#Corona_Cases.US<-filter(Corona_Cases,Country.Region=="US" & Total_confirmed_cases>0)
#Corona_Cases.US.case100<-filter(Corona_Cases.US, Days_since_100>=0)
# linear model parameters
#(model_fit<-lm(formula = Total_confirmed_cases.log~Days_since_100,data= Corona_Cases.US.case100 ))

#(slope<-model_fit$coefficients[2])
#(intercept<-model_fit$coefficients[1])

# Correlation coefficient
#cor(x = Corona_Cases.US.case100$Days_since_100,y = Corona_Cases.US.case100$Total_confirmed_cases.log)

##----------------------------------------
## Plot World Data
##----------------------------------------
# Timestamp for world
timestamp_plot.world<-paste("Most recent date for which data available:",max(Corona_Cases.world$Date))#


# Base template for plots
baseplot.world<-ggplot(data=NULL,aes(x=Days_since_100,col=Country.Region))+
  default_theme+
  scale_color_brewer(type = "qualitative",palette = "Paired")+
  ggtitle(paste("Log10 cases over time,",timestamp_plot.world))+
  theme(legend.position = "bottom",plot.title = element_text(size=12))


##////////////////////////
### Plot Longitudinal cases
```

```
(Corona_Cases.world.long.plot<-baseplot.world+
    geom_point(data=Corona_Cases.world.long,aes(y=cases))+
    geom_line(data=Corona_Cases.world.long,aes(y=cases))+
    facet_wrap(~case_type,scales = "free_y",ncol=1)+
    ggtitle(timestamp_plot.world)
    )
```



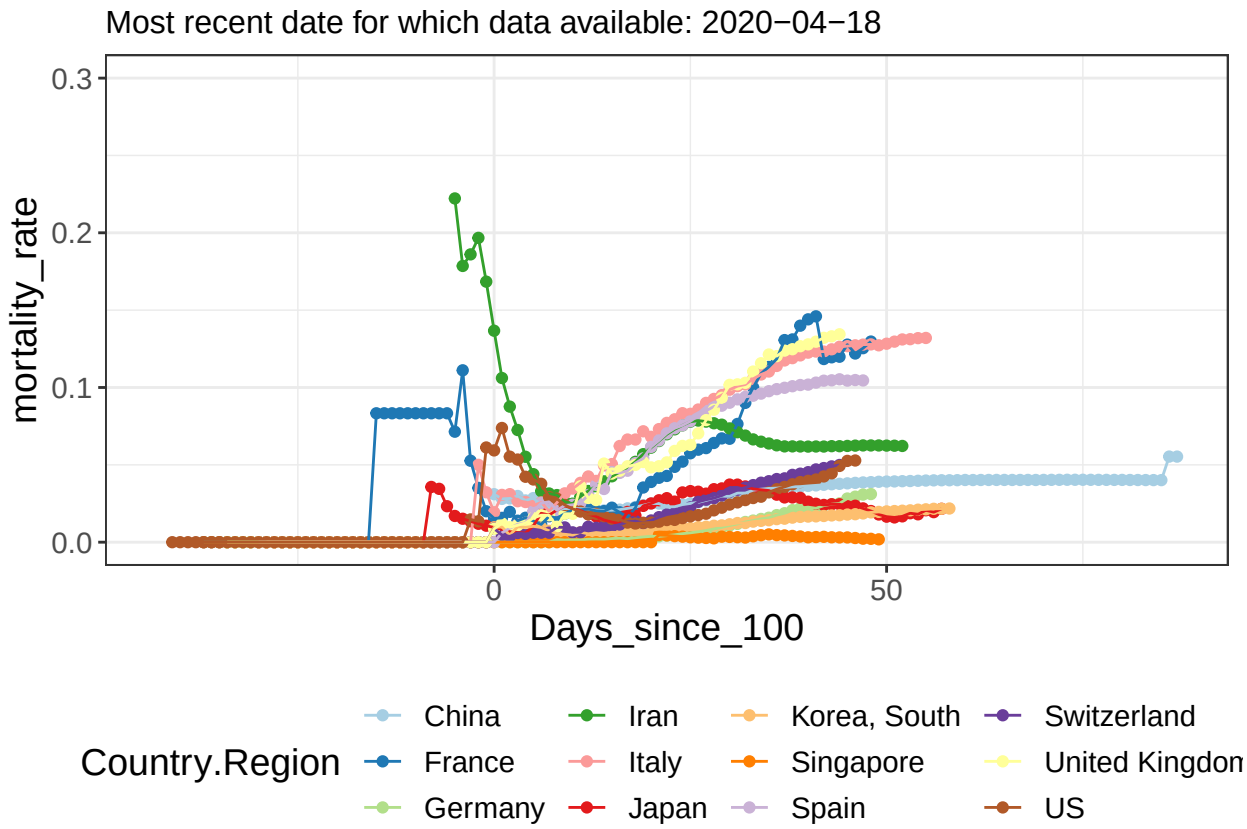

```
(Corona_Cases.world.loglong.plot<-baseplot.world+
    geom_point(data=Corona_Cases.world.long,aes(y=cases.log))+
    geom_line(data=Corona_Cases.world.long,aes(y=cases.log))+
    facet_wrap(~case_type,scales = "free_y",ncol=1)+
    ggtitle(timestamp_plot.world))
```



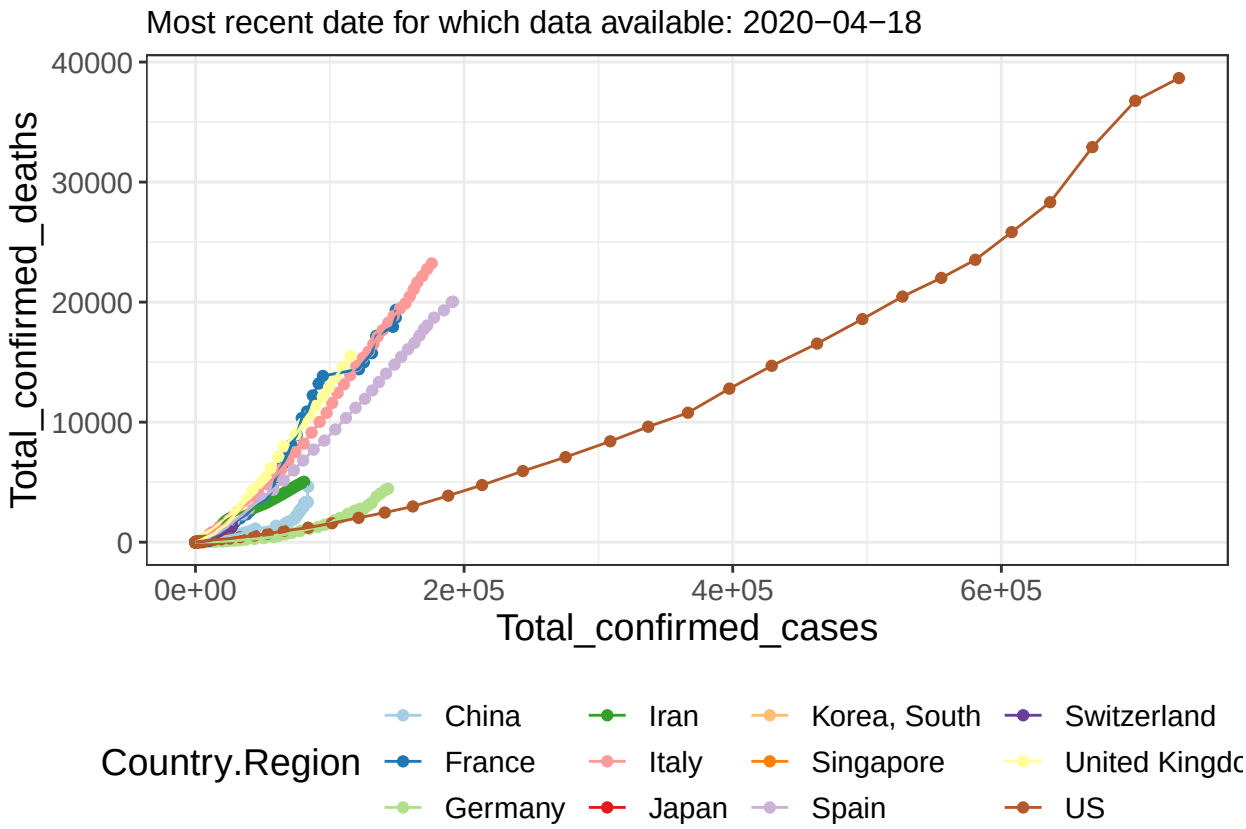

```
##//////////////////////////
### Plot Longitudinal mortality rate

(Corona_Cases.world.mortality.plot<-baseplot.world+
    geom_point(data=Corona_Cases.world,aes(y=mortality_rate))+
    geom_line(data=Corona_Cases.world,aes(y=mortality_rate))+
    ylim(c(0,0.3))+
    ggtitle(timestamp_plot.world))
```

```
## Warning: Removed 100 rows containing missing values (geom_point).
```

```
## Warning: Removed 100 row(s) containing missing values (geom_path).
```




```
##//////////////////////////
### Plot death vs total case correlation

(Corona_Cases.world.casecor.plot<-ggplot(Corona_Cases.world,aes(x=Total_confirmed_cases,y=Total_confirme
  geom_point()+
  geom_line()+
  default_theme+
  scale_color_brewer(type = "qualitative",palette = "Paired")+
  ggtitle(paste("Log10 cases over time,",timestamp_plot.world))+
  theme(legend.position = "bottom",plot.title = element_text(size=12))+
    ggtitle(timestamp_plot.world))
```




```
### Write polots

write_plot(Corona_Cases.world.long.plot,wd = results_dir)
```

```
## [1] "/Users/stevensmith/Projects/coronavirus/results/Corona_Cases.world.long.plot.png"
```

```
write_plot(Corona_Cases.world.loglong.plot,wd = results_dir)
```

```
## [1] "/Users/stevensmith/Projects/coronavirus/results/Corona_Cases.world.loglong.plot.png"
```

```
write_plot(Corona_Cases.world.mortality.plot,wd = results_dir)
```

```
## Warning: Removed 100 rows containing missing values (geom_point).
```

```
## Warning: Removed 100 row(s) containing missing values (geom_path).
```

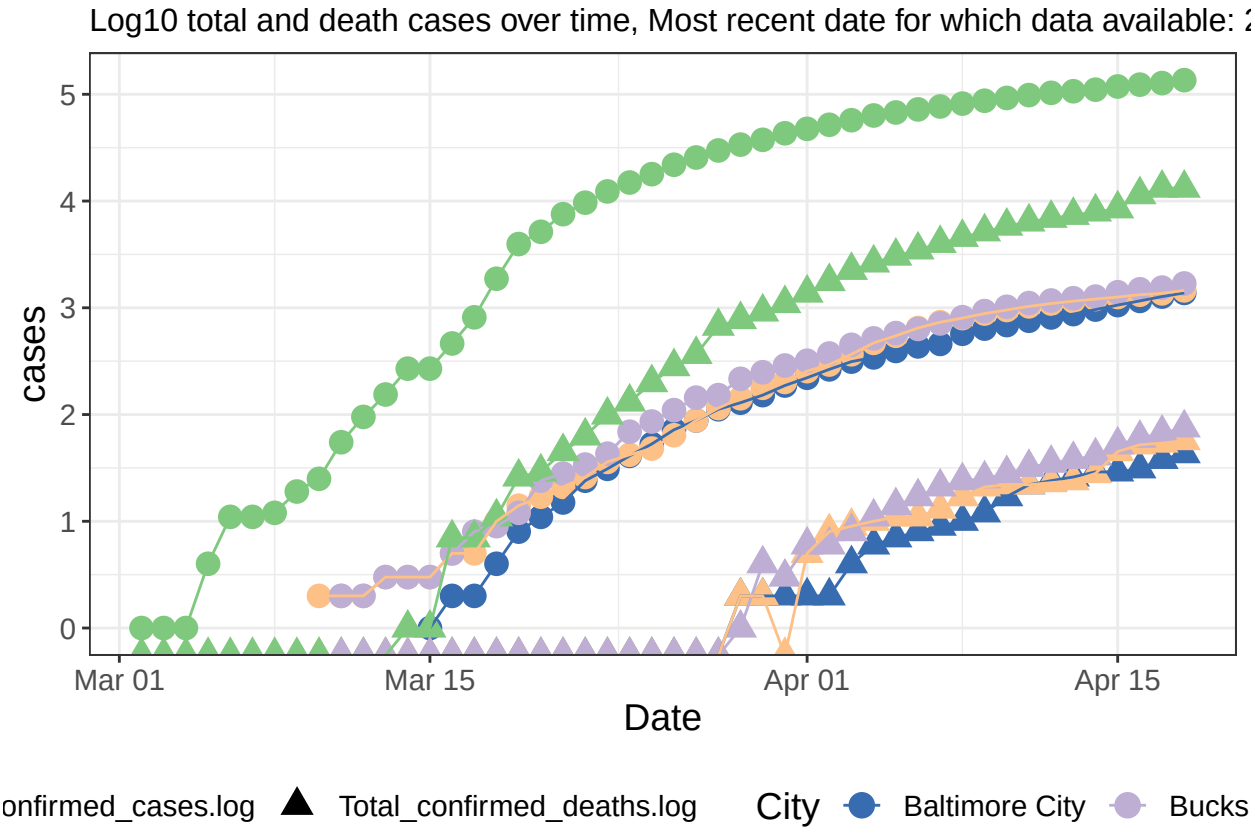
```
## [1] "/Users/stevensmith/Projects/coronavirus/results/Corona_Cases.world.mortality.plot.png"
```

```
write_plot(Corona_Cases.world.casecor.plot,wd = results_dir)
```

```
## [1] "/Users/stevensmith/Projects/coronavirus/results/Corona_Cases.world.casecor.plot.png"
```

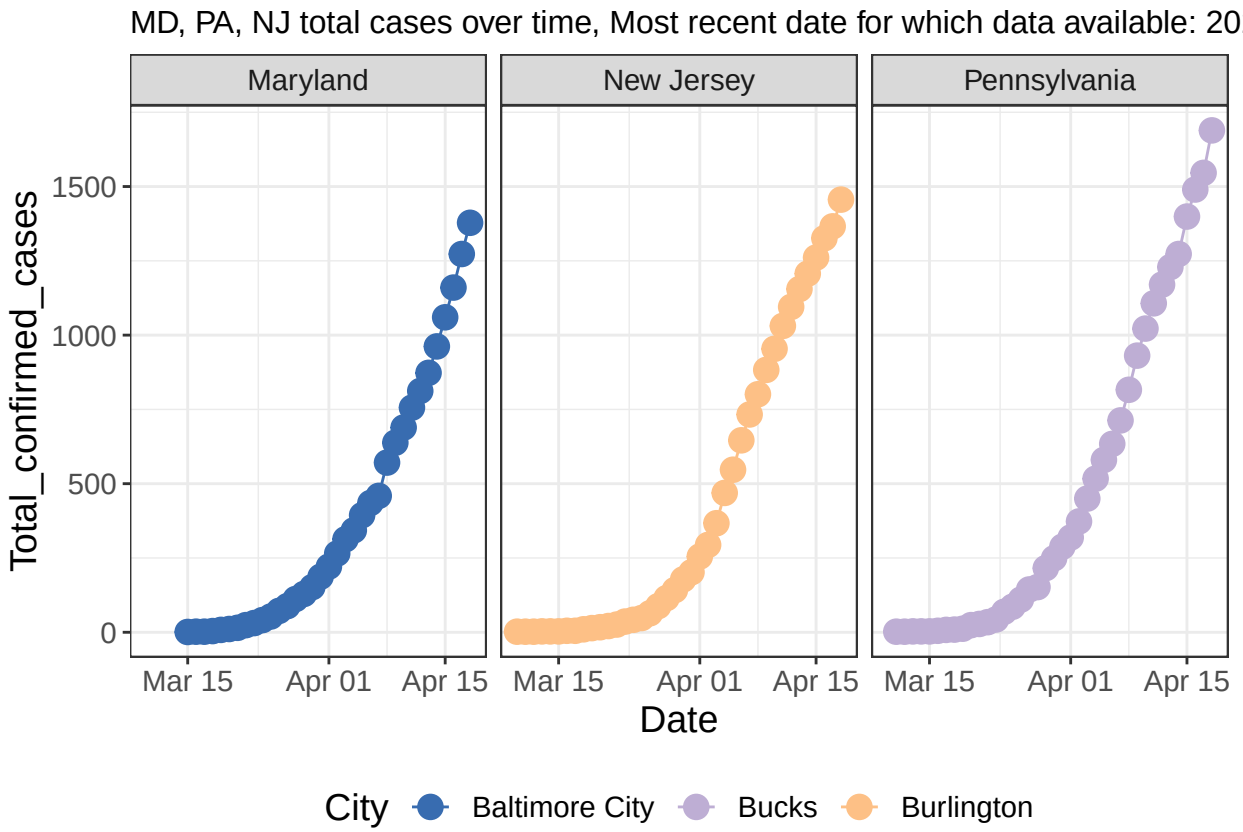
```
##----------------------------------------
## Plot US State Data
##----------------------------------------

baseplot.US<-ggplot(data=NULL,aes(x=Days_since_100_state,col=case_type))+
  default_theme+
  facet_wrap(~Province.State)+
  ggtitle(paste("Log10 cases over time,",timestamp_plot.world))
```

```
Corona_Cases.US_state.long.plot<-baseplot.US+geom_point(data=Corona_Cases.US_state.long,aes(y=cases.log
##----------------------------------------
## Plot US City Data
##----------------------------------------

Corona_Cases.US.plotdata<-filter(Corona_Cases.US_state,Province.State %in% c("Pennsylvania","Maryland",
                                 City %in% c("Bucks","Baltimore City", "New York","Burlington") &
                                 Total_confirmed_cases>0)
timestamp_plot<-paste("Most recent date for which data available:",max(Corona_Cases.US.plotdata$Date))#

city_colors<-c("Bucks"='#beaed4',"Baltimore City"='#386cb0', "New York"='#7fc97f',"Burlington"='#fdc086

##////////////////////////
### Plot death vs total case correlation

(Corona_Cases.city.loglong.plot<-ggplot(melt(Corona_Cases.US.plotdata,measure.vars = c("Total_confirmed
  geom_point(size=4)+
    geom_line()+
  default_theme+
  #facet_wrap(~case_type)+
    ggtitle(paste("Log10 total and death cases over time,",timestamp_plot))+
theme(legend.position = "bottom",plot.title = element_text(size=12))+
    scale_color_manual(values = city_colors))
```

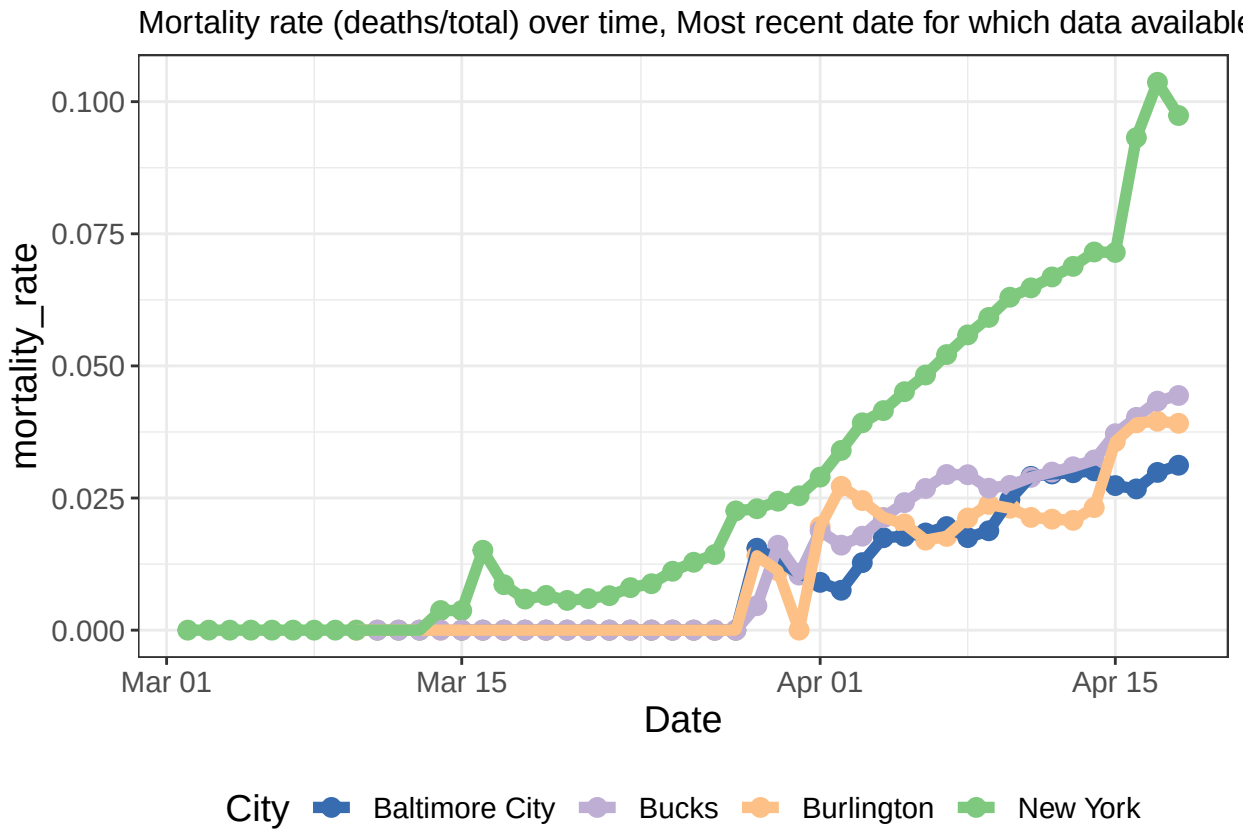



```
(Corona_Cases.city.long.plot<-ggplot(filter(Corona_Cases.US.plotdata,Province.State !="New York"),aes(x=
  geom_point(size=4)+
  geom_line()+
```

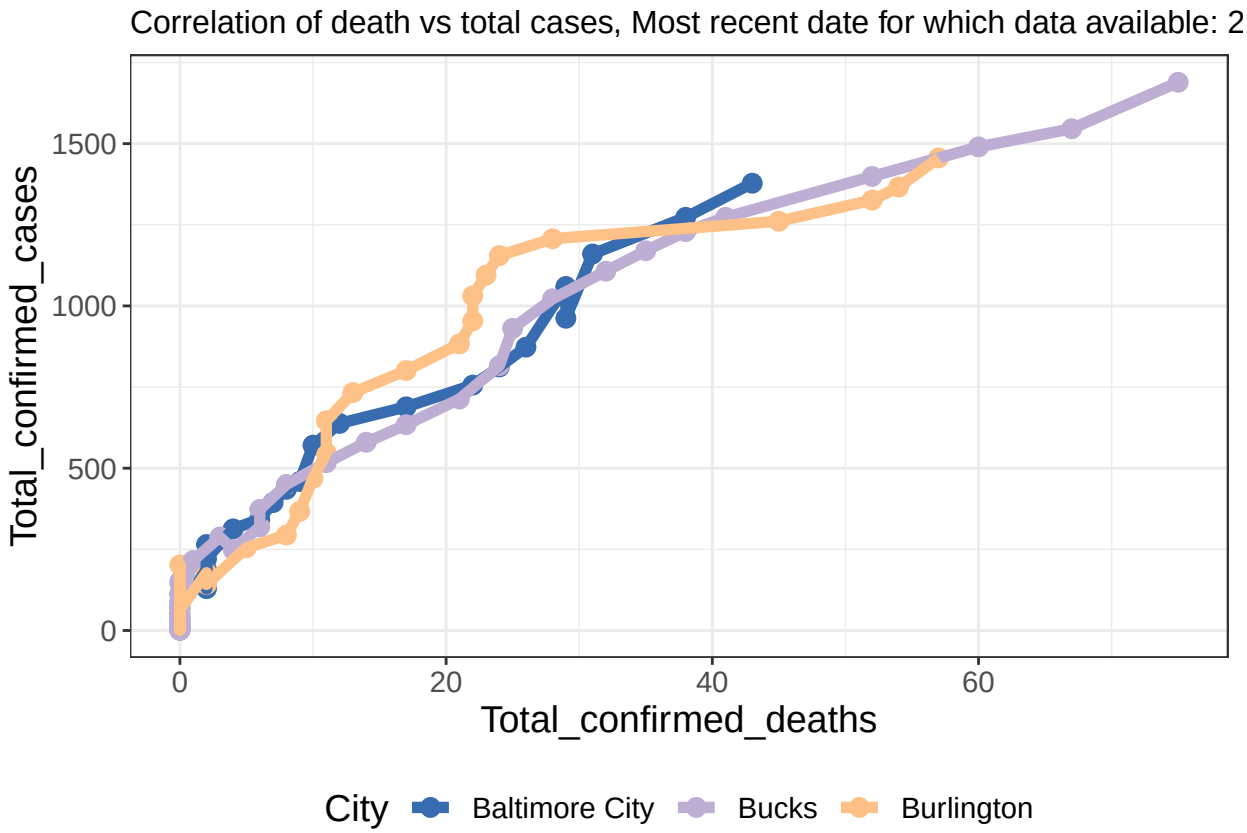
```
  default_theme+
  facet_grid(~Province.State,scales = "free_y")+
  ggtitle(paste("MD, PA, NJ total cases over time,",timestamp_plot))+
  theme(legend.position = "bottom",plot.title = element_text(size=12))+
  scale_color_manual(values = city_colors))
```



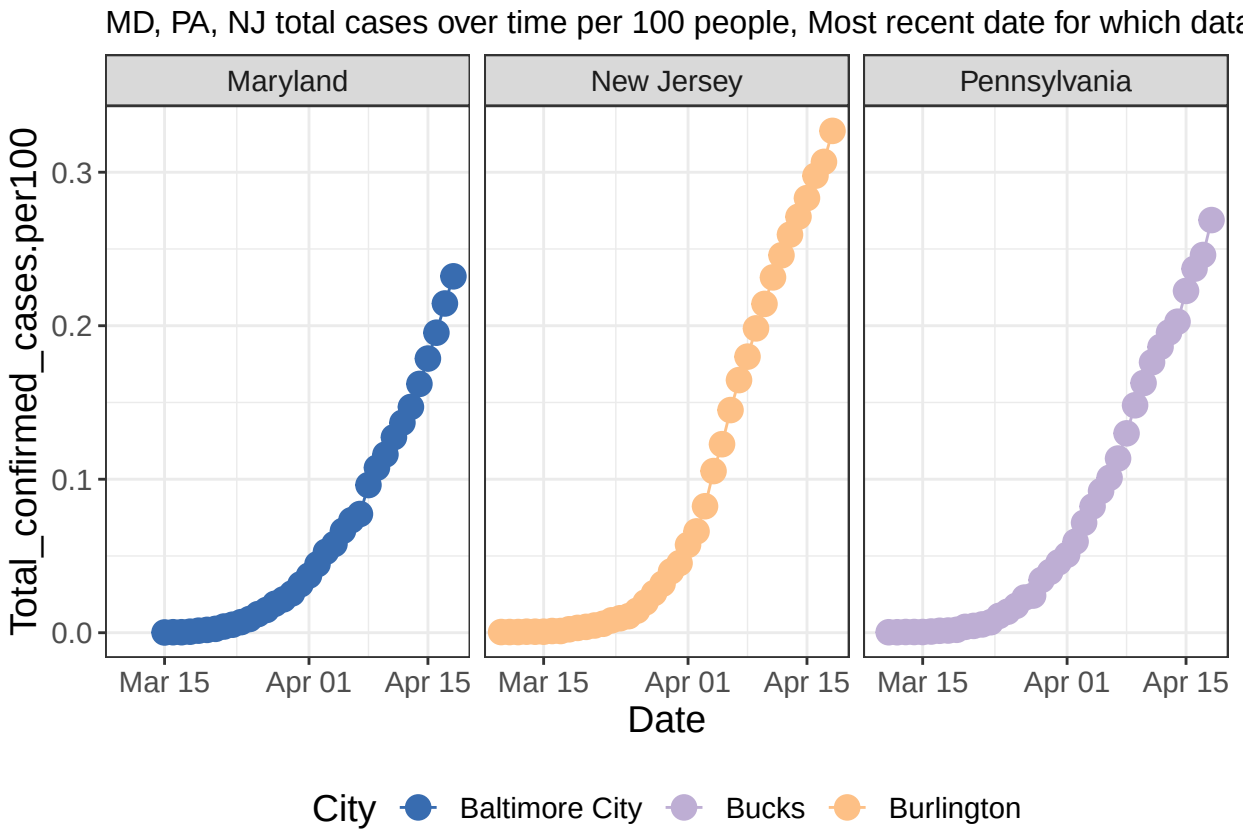

```
(Corona_Cases.city.mortality.plot<-ggplot(Corona_Cases.US.plotdata,aes(x=Date,y=mortality_rate,col=City)
  geom_point(size=3)+
  geom_line(size=2)+
  default_theme+
  ggtitle(paste("Mortality rate (deaths/total) over time,",timestamp_plot))+
  theme(legend.position = "bottom",plot.title = element_text(size=12))+
  scale_color_manual(values = city_colors))
```




```
(Corona_Cases.city.casecor.plot<-ggplot(filter(Corona_Cases.US.plotdata,Province.State !="New York"),aes
  geom_point(size=3)+
  geom_line(size=2)+
  default_theme+
  ggtitle(paste("Correlation of death vs total cases,",timestamp_plot))+
  theme(legend.position = "bottom",plot.title = element_text(size=12))+
  scale_color_manual(values = city_colors))
```




```
(Corona_Cases.city.long.normalized.plot<-ggplot(filter(Corona_Cases.US.plotdata,Province.State !="New Y
  geom_point(size=4)+
  geom_line()+
  default_theme+
  facet_grid(~Province.State)+
  ggtitle(paste("MD, PA, NJ total cases over time per 100 people,",timestamp_plot))+
  theme(legend.position = "bottom",plot.title = element_text(size=12))+
  scale_color_manual(values = city_colors))
```




```
write_plot(Corona_Cases.city.long.plot,wd = results_dir_custom)
```

```
## [1] "/Users/stevensmith/Projects/coronavirus/results/custom/Corona_Cases.city.long.plot.png"
```

```
write_plot(Corona_Cases.city.loglong.plot,wd = results_dir_custom)
```

```
## [1] "/Users/stevensmith/Projects/coronavirus/results/custom/Corona_Cases.city.loglong.plot.png"
```

```
write_plot(Corona_Cases.city.mortality.plot,wd = results_dir_custom)
```

```
## [1] "/Users/stevensmith/Projects/coronavirus/results/custom/Corona_Cases.city.mortality.plot.png"
```

```
write_plot(Corona_Cases.city.casecor.plot,wd = results_dir_custom)
```

```
## [1] "/Users/stevensmith/Projects/coronavirus/results/custom/Corona_Cases.city.casecor.plot.png"
```

```
write_plot(Corona_Cases.city.long.normalized.plot,wd = results_dir_custom)
```

```
## [1] "/Users/stevensmith/Projects/coronavirus/results/custom/Corona_Cases.city.long.normalized.plot.pn
```

Q2: What is the predicted number of cases?

What is the prediction of COVID-19 based on model thus far?

Additional questions:

WHy did it take to day 40 to start a log linear trend? How long will it be till x number of cases? When will the plateu happen? Are any effects noticed with social distancing? Delays

```
##----------------------------------------
## Prediction and Prediction Accuracy
##----------------------------------------
```

```
# What is the predict # of cases for the next few days?
# How is the model performing historically?

# Formula for # of cases by x days
paste0("log10_total_cases = ",slope,"*days + ",intercept)
paste0("total_cases = 10^(",slope,"*days + ",intercept,")")
#Days untill... cases:
# 2.5k, 5k and 1M:
paste0("2.5k cases is ",(log(2.5E5,10) - intercept)/slope," days")
paste0("5k cases is ",(log(5E5,10)- intercept)/slope," days")
paste0("1M cases is ",(log(1E6,10)- intercept)/slope," days")

head(filter(Corona_Cases.raw,Country.Region=="US"))
today_num<-max(Corona_Cases.US$Days_since_100)
predicted_days<-today_num+c(1,2,3,7)

#mods = dlply(mydf, .(x3), lm, formula = y ~ x1 + x2)
#today:
Corona_Cases.US[Corona_Cases.US$Days_since_100==(today_num-1),]
Corona_Cases.US[Corona_Cases.US$Days_since_100==today_num,]
Corona_Cases.US$type<-"Historical"
names(Corona_Cases)

Corona_Cases_wprediction<-rbind.fill(Corona_Cases.US,data.frame(Code="USA",type="MAR26_prediction",predi

Corona_Cases.US.prediction<-Corona_Cases_wprediction
prediction_values<-prediction_model(m=slope,b=intercept,days = predicted_days)$Total_confirmed_cases

histoical_model<-data.frame(date=today_num,m=slope,b=intercept)

# model for previous y days
historical_model_predictions<-data.frame(day_x=NULL,Days_since_100=NULL,Total_confirmed_cases=NULL,Total
for(i in c(1,2,3,4,5,6,7,8,9,10)){
  #i<-1
day_x<-today_num-i # 1, 2, 3, 4
day_x_nextweek<-day_x+c(1,2,3)
model_fit_x<-lm(data = filter(Corona_Cases.US.case100,Days_since_100 < day_x),formula = Total_confirmed_
prediction_day_x_nextweek<-prediction_model(m = model_fit_x$coefficients[2],b = model_fit_x$coefficients
prediction_day_x_nextweek$type<-"Predicted"
acutal_day_x_nextweek<-filter(Corona_Cases.US,Days_since_100 %in% day_x_nextweek) %>% select(c(Days_sinc
acutal_day_x_nextweek$type<-"Historical"
historical_model_predictions.i<-data.frame(day_x=day_x,rbind(acutal_day_x_nextweek,prediction_day_x_next
historical_model_predictions<-rbind(historical_model_predictions.i,historical_model_predictions)
}

historical_model_predictions.withHx<-rbind.fill(historical_model_predictions,data.frame(Corona_Cases.US
historical_model_predictions.withHx$Total_confirmed_cases.log2<-log(historical_model_predictions.withHx$
#TODO: fix case_type.. are we predicting deaths too?
#TODO: better analysis of death rate!
(historical_model_predictions.plot<-ggplot(historical_model_predictions.withHx,aes(x=Days_since_100,y=To
    geom_point(size=3)+
    default_theme+
    theme(legend.position = "bottom")+
```

```
      #geom_abline(slope = slope,intercept =intercept,lty=2)+
    #facet_wrap(~case_type,ncol=1)+
    scale_color_manual(values = c("Historical"="#377eb8","Predicted"="#e41a1c")))
write_plot(historical_model_predictions.plot,wd=results_dir)
```

```
##----------------------------------------
## filter input_data1
##----------------------------------------
input_data1.filter<-fitler(input_data1,col1=="foo")
##----------------------------------------

##----------------------------------------
## sub question 1
##----------------------------------------
table(input_data1.filter$col<5)
##----------------------------------------

##----------------------------------------
## sub question 2
##----------------------------------------
table(input_data1.filter$col<10)
##----------------------------------------

##----------------------------------------
## plot data
##----------------------------------------
(input_data1.filter.plot<-ggplot(input_data1.filter,aes(x=col1,y=col2.log))+
   geom_point()+
   default_plot_theme)
write_plot(input_data1.filter.plot,wd=results_dir)
##----------------------------------------
results_dir
```

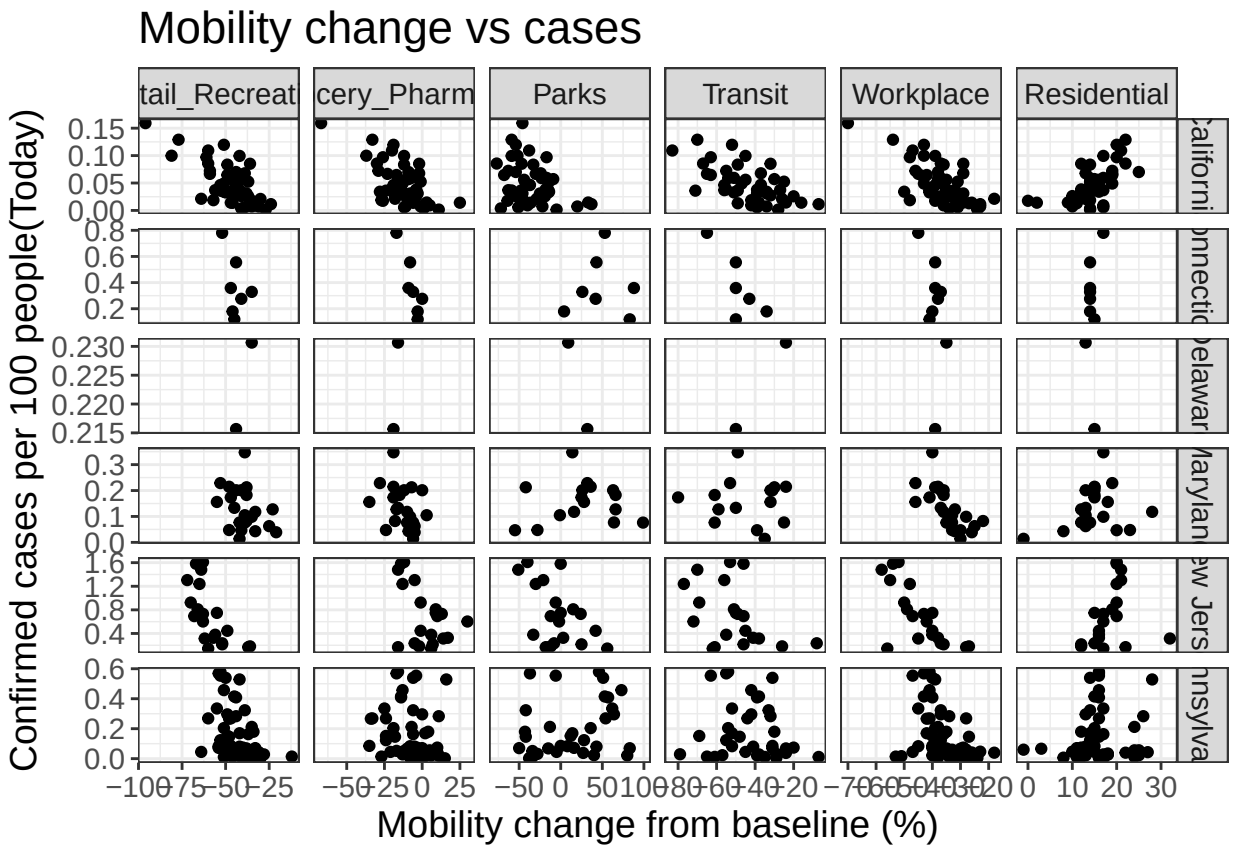
Q3: What is the effect on social distancing, descreased mobility on case load?

Load data from Google which compoutes % change in user mobility relative to baseline for * Recreation
* Workplace
* Residence
* Park
* Grocery

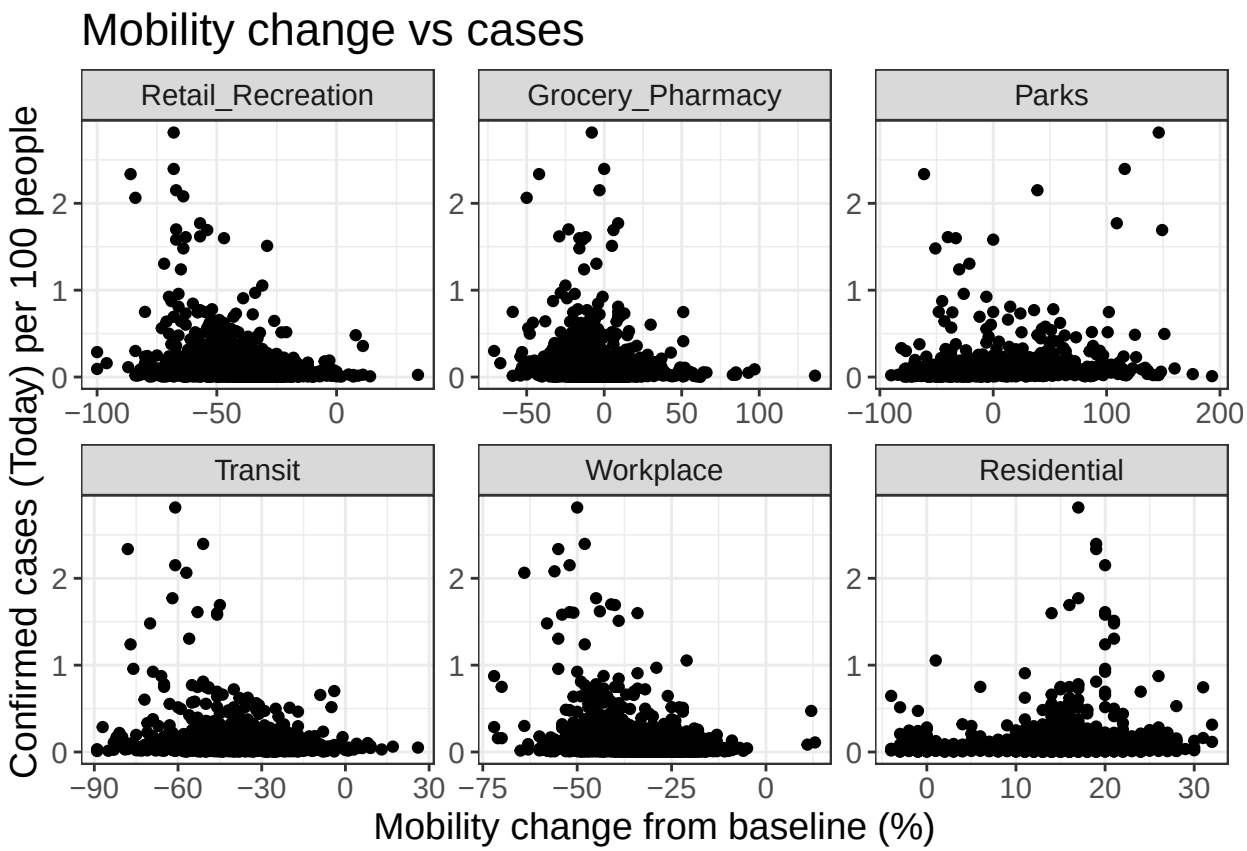
Data from https://www.google.com/covid19/mobility/

```
# See pre-processing section for script on gathering mobility data

# UNDER DEVELOPMENT

mobility<-read.csv("/Users/stevensmith/Projects/MIT_COVID19/mobility.csv",header = T,stringsAsFactors =
#mobility$Retail_Recreation<-as.numeric(sub(mobility$Retail_Recreation,pattern = "%",replacement = ""))
#mobility$Workplace<-as.numeric(sub(mobility$Workplace,pattern = "%",replacement = ""))
#mobility$Residential<-as.numeric(sub(mobility$Residential,pattern = "%",replacement = ""))

##----------------------------------------
## Show relationship between mobility and caseload
##----------------------------------------
```

```
mobility$County<-gsub(mobility$County,pattern = " County",replacement = "")
Corona_Cases.US_state.mobility<-merge(Corona_Cases.US_state,plyr::rename(mobility,c("State"="Province.S

#Corona_Cases.US_state.tmp<-merge(metadata,Corona_Cases.US_state.tmp)
# Needs to happen upsteam, see todos
#Corona_Cases.US_state.tmp$Total_confirmed_cases.perperson<-Corona_Cases.US_state.tmp$Total_confirmed_c
mobility_measures<-c("Retail_Recreation","Grocery_Pharmacy","Parks","Transit","Workplace","Residential"

plot_data<-filter(Corona_Cases.US_state.mobility, Date.numeric==max(Corona_Cases.US_state$Date.numeric)
plot_data$value<-as.numeric(gsub(plot_data$value,pattern = "%",replacement = ""))
plot_data<-filter(plot_data,!is.na(value))

(mobility.plot<-ggplot(filter(plot_data,Province.State %in% c("Pennsylvania","Maryland","New Jersey","Ca
  facet_grid(Province.State~variable,scales = "free")+
  xlab("Mobility change from baseline (%)")+
  ylab(paste0("Confirmed cases per 100 people(Today)"))+
  default_theme+
  ggtitle("Mobility change vs cases"))
```




```
(mobility.global.plot<-ggplot(plot_data,aes(y=Total_confirmed_cases.per100,x=value))+geom_point()+
  facet_wrap(~variable,scales = "free")+
  xlab("Mobility change from baseline (%)")+
  ylab(paste0("Confirmed cases (Today) per 100 people"))+
  default_theme+
  ggtitle("Mobility change vs cases"))
```




```
plot_data.permobility_summary<-ddply(plot_data,c("Province.State","variable"),summarise,cor=cor(y =Total

kable(plot_data.permobility_summary,caption = "Ranked per-state mobility correlation with total confirme
```

Table 8: Ranked per-state mobility correlation with total confirmed cases

Province.State	variable	cor	median_change
Alaska	Transit	-1.0000000	-63.0
Delaware	Retail_Recreation	1.0000000	-39.5
Delaware	Grocery_Pharmacy	1.0000000	-17.5
Delaware	Parks	-1.0000000	20.5
Delaware	Transit	1.0000000	-37.0
Delaware	Workplace	1.0000000	-37.0
Delaware	Residential	-1.0000000	14.0
Hawaii	Parks	0.9996005	-72.0
Hawaii	Transit	0.9938946	-89.0
Alaska	Residential	0.9672579	13.0
Vermont	Parks	0.9211184	-35.5
South Dakota	Parks	0.9083557	-26.0
New Hampshire	Parks	0.9079658	-20.0
Connecticut	Grocery_Pharmacy	-0.8839067	-6.0
Utah	Retail_Recreation	-0.8777268	-39.0
Utah	Workplace	-0.8730312	-35.0
Hawaii	Grocery_Pharmacy	0.8548709	-34.0
Massachusetts	Workplace	-0.8510178	-39.0
Alaska	Grocery_Pharmacy	-0.8475738	-7.0

Province.State	variable	cor	median_change
Utah	Grocery_Pharmacy	-0.8119480	-4.0
Rhode Island	Workplace	-0.7837848	-39.5
Connecticut	Transit	-0.7818403	-50.0
Hawaii	Retail_Recreation	0.7799437	-56.0
Utah	Transit	-0.7340335	-18.0
New Mexico	Parks	0.7333501	-31.5
New Jersey	Workplace	-0.7189288	-44.0
North Dakota	Retail_Recreation	-0.7133506	-43.5
California	Retail_Recreation	-0.7008942	-44.0
Kansas	Parks	0.6984014	72.0
Maryland	Workplace	-0.6933217	-35.0
Massachusetts	Retail_Recreation	-0.6888126	-44.0
California	Workplace	-0.6855722	-36.0
New York	Workplace	-0.6665410	-34.5
Utah	Residential	-0.6636286	12.0
Vermont	Grocery_Pharmacy	-0.6627273	-25.0
New Jersey	Retail_Recreation	-0.6572450	-62.5
North Dakota	Parks	-0.6569088	-34.0
Maine	Transit	-0.6452735	-50.0
California	Grocery_Pharmacy	-0.6290696	-12.0
Connecticut	Residential	0.6286051	14.0
New York	Retail_Recreation	-0.6277159	-46.0
California	Residential	0.6223247	14.0
Rhode Island	Residential	-0.6018789	18.5
California	Transit	-0.6012098	-42.0
Montana	Workplace	-0.5912397	-40.5
West Virginia	Parks	0.5806560	-27.0
Massachusetts	Grocery_Pharmacy	-0.5757763	-7.0
Alaska	Workplace	-0.5627265	-34.0
Rhode Island	Retail_Recreation	-0.5578342	-45.0
Nevada	Transit	-0.5549183	-20.0
Nevada	Retail_Recreation	-0.5435421	-43.0
Connecticut	Workplace	-0.5427897	-39.0
Utah	Parks	-0.5357441	13.5
Montana	Transit	-0.5330115	-41.0
Idaho	Workplace	-0.5161789	-29.5
New Jersey	Parks	-0.5126491	-6.0
Montana	Retail_Recreation	-0.5050534	-51.0
Kansas	Grocery_Pharmacy	-0.5019759	-14.0
Maine	Workplace	-0.4994685	-30.0
Montana	Parks	-0.4986153	-58.0
Maine	Parks	0.4814967	-31.0
New Jersey	Grocery_Pharmacy	-0.4721005	2.5
Minnesota	Parks	0.4679127	-9.0
Idaho	Transit	-0.4573213	-30.0
Montana	Residential	0.4544538	14.0
Connecticut	Retail_Recreation	-0.4514328	-45.0
Massachusetts	Transit	-0.4394980	-45.0
Arizona	Grocery_Pharmacy	-0.4388694	-15.0
Pennsylvania	Workplace	-0.4376781	-36.0
Vermont	Residential	0.4371329	11.5
Arkansas	Parks	-0.4357575	-12.0

Province.State	variable	cor	median_change
New Mexico	Residential	0.4293836	13.5
New York	Parks	0.4269244	20.0
Idaho	Grocery_Pharmacy	-0.4268925	-4.0
Rhode Island	Parks	0.4214581	52.0
New Jersey	Transit	-0.4210584	-50.5
New York	Transit	-0.4188354	-48.0
Michigan	Workplace	-0.4032723	-40.0
Montana	Grocery_Pharmacy	-0.4023952	-16.0
Colorado	Residential	0.3995965	14.0
Pennsylvania	Retail_Recreation	-0.3960269	-45.0
Florida	Parks	-0.3924315	-43.0
Colorado	Workplace	-0.3915247	-39.0
Idaho	Retail_Recreation	-0.3791311	-41.0
Illinois	Transit	-0.3787347	-31.0
Virginia	Retail_Recreation	-0.3774475	-35.0
Vermont	Retail_Recreation	0.3657732	-57.0
Oregon	Parks	0.3648752	16.5
Alabama	Workplace	-0.3617157	-29.0
New Mexico	Grocery_Pharmacy	-0.3589887	-11.5
Arizona	Transit	0.3585495	-38.0
Maryland	Grocery_Pharmacy	-0.3578786	-10.0
Nebraska	Grocery_Pharmacy	-0.3535899	0.0
Virginia	Transit	-0.3511034	-33.0
Rhode Island	Grocery_Pharmacy	0.3499457	-7.5
New Mexico	Retail_Recreation	-0.3486574	-42.5
North Dakota	Grocery_Pharmacy	-0.3466084	-9.5
Colorado	Retail_Recreation	-0.3465570	-44.0
Maryland	Retail_Recreation	-0.3440714	-39.0
Alaska	Retail_Recreation	0.3438643	-39.0
Minnesota	Transit	-0.3392734	-28.5
Colorado	Parks	-0.3375568	2.0
Texas	Transit	0.3359341	-42.0
Maine	Grocery_Pharmacy	-0.3353545	-10.5
South Dakota	Transit	-0.3320080	-40.0
Mississippi	Parks	0.3294420	-25.0
Washington	Transit	-0.3280112	-33.5
California	Parks	-0.3227182	-38.0
Arizona	Residential	0.3203019	13.0
Florida	Residential	0.3187491	14.0
Idaho	Parks	0.3165578	-22.0
Colorado	Grocery_Pharmacy	-0.3162537	-17.0
Florida	Transit	-0.3162362	-49.0
Arkansas	Retail_Recreation	-0.3115122	-30.0
Colorado	Transit	-0.3102405	-36.0
Wisconsin	Transit	-0.3080295	-23.5
Kansas	Retail_Recreation	-0.3046916	-39.0
Virginia	Workplace	-0.3025107	-31.5
Iowa	Residential	-0.2982065	13.0
Maine	Retail_Recreation	-0.2950187	-41.5
New York	Grocery_Pharmacy	-0.2923414	8.0
North Dakota	Workplace	0.2890914	-33.5
Arizona	Retail_Recreation	-0.2860134	-42.5

Province.State	variable	cor	median_change
New Jersey	Residential	0.2836027	18.0
Mississippi	Grocery_Pharmacy	-0.2814229	-8.0
Florida	Workplace	-0.2790047	-33.0
Pennsylvania	Parks	0.2776938	13.0
Virginia	Grocery_Pharmacy	-0.2733318	-8.0
Tennessee	Retail_Recreation	-0.2717198	-29.5
Arkansas	Residential	0.2716868	12.0
New Hampshire	Grocery_Pharmacy	-0.2674760	-6.0
Oregon	Residential	0.2673395	10.5
Illinois	Workplace	-0.2645037	-30.0
Indiana	Grocery_Pharmacy	-0.2608181	-5.5
Georgia	Grocery_Pharmacy	-0.2588658	-10.0
Iowa	Workplace	-0.2571085	-29.0
Maryland	Residential	0.2565506	15.0
New Hampshire	Residential	-0.2470032	14.0
Massachusetts	Residential	0.2459572	15.0
South Carolina	Residential	0.2418585	12.0
West Virginia	Retail_Recreation	0.2393618	-38.5
Michigan	Grocery_Pharmacy	-0.2352822	-11.0
Texas	Residential	-0.2335482	15.0
Texas	Parks	0.2330747	-42.0
North Carolina	Retail_Recreation	-0.2310385	-33.0
Hawaii	Workplace	-0.2210727	-46.0
Pennsylvania	Grocery_Pharmacy	-0.2202814	-6.0
Rhode Island	Transit	-0.2191759	-56.0
Michigan	Retail_Recreation	-0.2190196	-53.0
Alabama	Residential	0.2169850	11.0
Nevada	Residential	0.2168990	17.0
Georgia	Retail_Recreation	-0.2145387	-41.0
Kentucky	Workplace	-0.2105673	-34.0
Washington	Workplace	-0.2061162	-38.0
West Virginia	Grocery_Pharmacy	-0.2057990	-6.0
Oklahoma	Residential	0.2057739	15.0
Georgia	Workplace	-0.2055303	-33.5
New Hampshire	Retail_Recreation	-0.2054313	-41.0
Washington	Parks	0.2050328	-3.5
Alabama	Transit	-0.2025601	-36.5
Nebraska	Residential	0.2020503	14.0
Wisconsin	Parks	0.2005435	51.5
Oklahoma	Retail_Recreation	0.1999559	-31.0
Oklahoma	Grocery_Pharmacy	0.1982537	-0.5
Tennessee	Grocery_Pharmacy	-0.1980654	6.0
Virginia	Residential	0.1970695	14.0
Alabama	Grocery_Pharmacy	-0.1943145	-2.0
Kentucky	Parks	0.1916883	28.5
Wisconsin	Workplace	-0.1892001	-31.0
South Dakota	Retail_Recreation	-0.1887724	-38.5
Michigan	Parks	0.1867940	30.0
Oklahoma	Workplace	-0.1815528	-30.0
South Carolina	Retail_Recreation	-0.1769429	-35.0
South Carolina	Parks	-0.1735369	-23.0
South Carolina	Workplace	0.1725488	-30.0

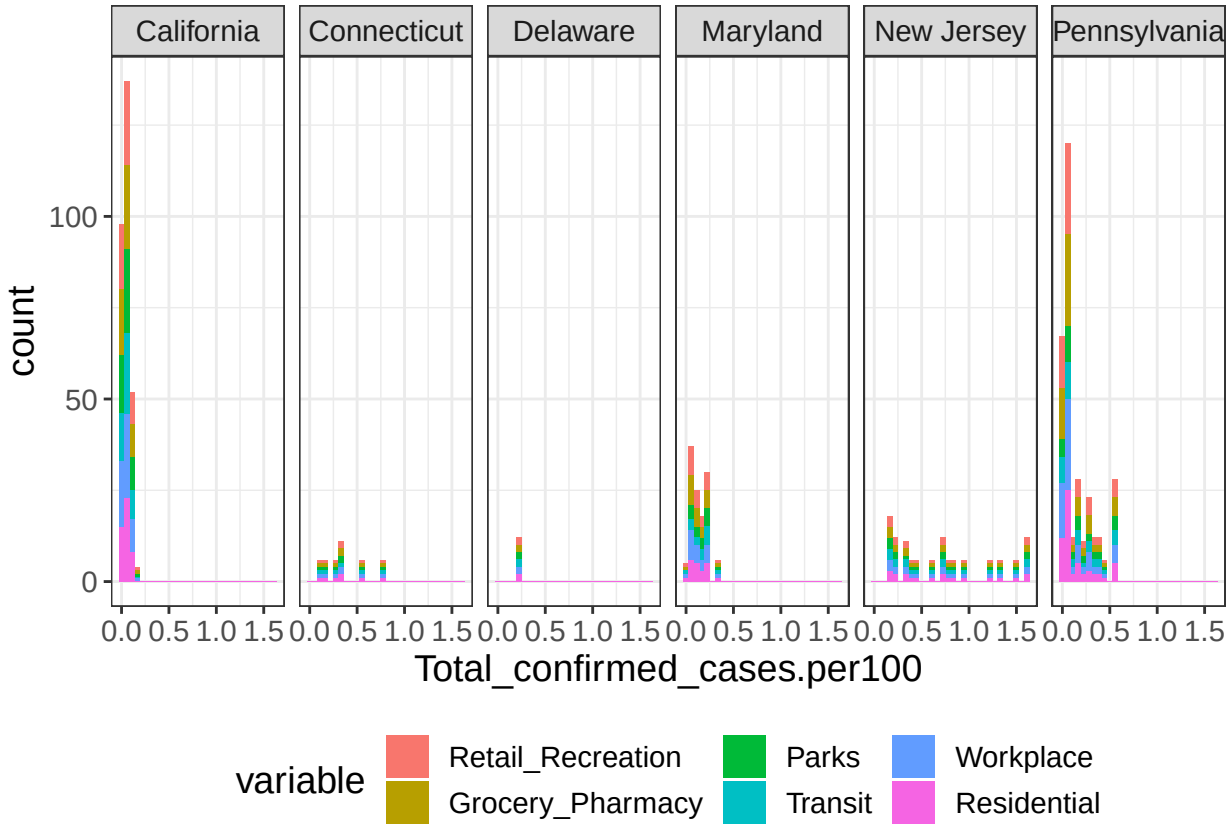
Province.State	variable	cor	median_change
North Carolina	Transit	0.1715866	-32.0
Arizona	Workplace	-0.1708676	-35.0
Missouri	Transit	-0.1702439	-23.0
Kentucky	Residential	0.1687039	12.0
Florida	Grocery_Pharmacy	-0.1686546	-14.0
Illinois	Residential	0.1682843	14.0
Tennessee	Workplace	-0.1642679	-31.0
Indiana	Retail_Recreation	-0.1625665	-38.0
Oregon	Grocery_Pharmacy	0.1620246	-7.0
West Virginia	Workplace	0.1615039	-32.5
Nevada	Workplace	-0.1543206	-40.0
Pennsylvania	Transit	-0.1528964	-41.5
Arkansas	Workplace	-0.1526153	-26.0
Arizona	Parks	0.1523548	-44.5
South Dakota	Grocery_Pharmacy	0.1521924	-9.0
Mississippi	Workplace	-0.1520576	-33.0
Alabama	Parks	0.1458533	-1.0
Tennessee	Residential	0.1456665	11.5
Idaho	Residential	-0.1442904	11.0
Wisconsin	Residential	-0.1436485	14.0
New Hampshire	Transit	-0.1431295	-57.0
Ohio	Transit	0.1428536	-28.0
Minnesota	Retail_Recreation	0.1397189	-41.0
Florida	Retail_Recreation	-0.1383406	-43.0
Nebraska	Transit	0.1366502	-11.5
Texas	Workplace	0.1330952	-31.0
Tennessee	Parks	0.1305974	10.5
Wisconsin	Grocery_Pharmacy	0.1301286	-1.5
Vermont	Workplace	-0.1256542	-43.0
Georgia	Residential	-0.1242744	13.0
Pennsylvania	Residential	0.1227599	15.0
Nebraska	Retail_Recreation	-0.1192319	-37.5
Oklahoma	Parks	-0.1177523	-23.0
Illinois	Retail_Recreation	-0.1174560	-40.0
Maine	Residential	-0.1161859	11.0
Hawaii	Residential	-0.1103339	19.0
Michigan	Residential	0.1073488	15.0
Ohio	Residential	0.1063771	14.0
Indiana	Workplace	-0.1044123	-34.0
Washington	Retail_Recreation	-0.1032064	-42.0
Washington	Residential	0.1030546	13.0
Ohio	Workplace	-0.0988450	-35.0
New Hampshire	Workplace	-0.0985649	-37.0
New Mexico	Workplace	-0.0968695	-34.0
Illinois	Parks	0.0957077	26.5
Wisconsin	Retail_Recreation	-0.0947990	-44.0
Illinois	Grocery_Pharmacy	-0.0938853	2.0
Kansas	Transit	-0.0893923	-26.5
Virginia	Parks	0.0856265	6.0
Maryland	Parks	0.0816607	27.0
New York	Residential	0.0776829	17.5
Oregon	Transit	-0.0771341	-28.0

Province.State	variable	cor	median_change
Kentucky	Transit	0.0755873	-31.0
New Mexico	Transit	0.0739426	-37.0
Texas	Retail_Recreation	-0.0714106	-39.5
North Carolina	Grocery_Pharmacy	0.0703504	1.0
North Carolina	Parks	-0.0698709	7.0
Missouri	Grocery_Pharmacy	-0.0690970	2.0
Minnesota	Grocery_Pharmacy	-0.0682117	-4.0
Kansas	Workplace	-0.0679649	-31.5
Missouri	Retail_Recreation	-0.0674358	-36.5
North Carolina	Residential	0.0673238	13.0
Arkansas	Transit	0.0662700	-27.0
Iowa	Parks	0.0598123	28.5
Georgia	Transit	-0.0597692	-35.0
Arkansas	Grocery_Pharmacy	0.0582494	3.5
Indiana	Residential	0.0567280	12.0
South Carolina	Transit	-0.0564269	-45.0
Ohio	Grocery_Pharmacy	0.0550478	0.0
Maryland	Transit	-0.0523983	-39.0
Massachusetts	Parks	-0.0522342	39.0
West Virginia	Transit	-0.0510671	-45.0
Mississippi	Residential	0.0498867	13.0
Tennessee	Transit	0.0459778	-32.0
Missouri	Workplace	0.0448501	-28.5
Ohio	Retail_Recreation	0.0429404	-36.0
Connecticut	Parks	0.0425459	43.0
Washington	Grocery_Pharmacy	-0.0422355	-7.0
North Dakota	Residential	0.0419220	17.0
Iowa	Retail_Recreation	-0.0411774	-37.0
Michigan	Transit	0.0410998	-46.0
North Dakota	Transit	0.0403932	-48.0
South Carolina	Grocery_Pharmacy	-0.0395685	1.0
Minnesota	Workplace	0.0374852	-33.0
Kansas	Residential	0.0374367	13.0
Indiana	Parks	-0.0358848	29.0
Iowa	Grocery_Pharmacy	-0.0345109	4.0
Georgia	Parks	-0.0336119	-6.0
Ohio	Parks	0.0326422	67.5
Nevada	Parks	-0.0307510	-12.5
West Virginia	Residential	0.0262598	11.0
Nevada	Grocery_Pharmacy	-0.0256321	-11.0
Vermont	Transit	0.0245526	-63.0
Oregon	Workplace	-0.0244824	-32.0
Iowa	Transit	-0.0236381	-25.0
Nebraska	Parks	0.0219267	55.5
Kentucky	Retail_Recreation	0.0210420	-30.0
South Dakota	Residential	0.0208692	15.0
Indiana	Transit	-0.0195968	-29.0
Mississippi	Transit	-0.0174983	-38.5
Texas	Grocery_Pharmacy	-0.0169217	-13.0
Missouri	Parks	0.0161057	0.0
Oregon	Retail_Recreation	0.0142868	-41.0
Kentucky	Grocery_Pharmacy	-0.0137322	4.0

Province.State	variable	cor	median_change
South Dakota	Workplace	0.0108769	-35.0
North Carolina	Workplace	-0.0089566	-31.0
Nebraska	Workplace	-0.0087334	-33.0
Minnesota	Residential	0.0077847	18.0
Missouri	Residential	0.0066887	13.0
Mississippi	Retail_Recreation	0.0048020	-40.0
Alabama	Retail_Recreation	0.0039891	-39.0
Oklahoma	Transit	-0.0037925	-26.0
Alaska	Parks	NA	29.0
District of Columbia	Retail_Recreation	NA	-69.0
District of Columbia	Grocery_Pharmacy	NA	-28.0
District of Columbia	Parks	NA	-65.0
District of Columbia	Transit	NA	-69.0
District of Columbia	Workplace	NA	-48.0
District of Columbia	Residential	NA	17.0

```
# sanity check
ggplot(filter(plot_data,Province.State %in% c("Pennsylvania","Maryland","New Jersey","California","Dela
  facet_grid(~Province.State)+
    default_theme+
  theme(legend.position = "bottom")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



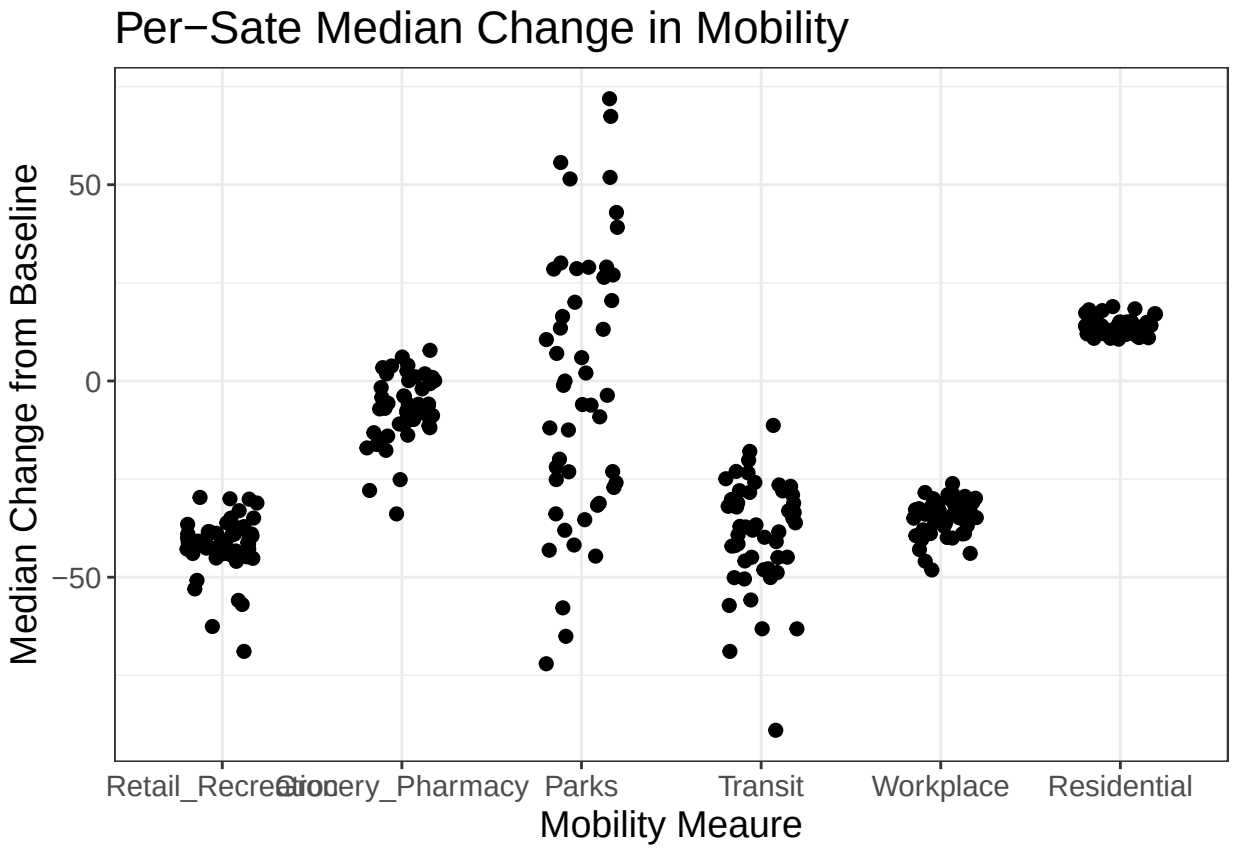

```
write_plot(mobility.plot,wd = results_dir)
```

```
## [1] "/Users/stevensmith/Projects/coronavirus/results/mobility.plot.png"
```

```
write_plot(mobility.global.plot,wd = results_dir)
```

```
## [1] "/Users/stevensmith/Projects/coronavirus/results/mobility.global.plot.png"
```

```
(plot_data.permobility_summary.plot<-ggplot(plot_data.permobility_summary,aes(x=variable,y=median_chang
  geom_jitter(size=2,width=.2)+
  #geom_jitter(data=plot_data.permobility_summary %>% arrange(-abs(median_change)) %>% head(n=15),aes(c
  default_theme+
  ggtitle("Per-Sate Median Change in Mobility")+
  xlab("Mobility Meaure")+
  ylab("Median Change from Baseline"))
```




```
write_plot(plot_data.permobility_summary.plot,wd = results_dir)
```

```
## [1] "/Users/stevensmith/Projects/coronavirus/results/plot_data.permobility_summary.plot.png"
```

DELIVERABLE MANIFEST

The following link to commited documents pushed to github. These are provided as a convienence, but note this is a manual process. The generation of reports, plots and tables is not coupled to the execution of this markdown. ## Report This report, html & pdf

Plots

```
github_root<-"https://github.com/sbs87/coronavirus/blob/master/"
link<-paste0(github_root,"results/Corona_Cases.world.casecor.plot.png")
```

```
section_ref<-'Q3'
plot_handle<-c("Corona_Cases.world.casecor.plot","Corona_Cases.world.long.plot")
name<-"World total & death cases, correlation"
deliverable_manifest<-data.frame(
  name=c("World total & death cases, correlation",
         "World total & death cases, longitudinal"),
  plot_handle=plot_handle,
  link=paste0(github_root,"results/",plot_handle,".png")
)
(tmp<-data.frame(row_out=apply(deliverable_manifest,MARGIN = 1,FUN = function(x) paste(x[1],x[2],x[3],s
```

```
##
## 1 World total & death cases, correlation | Corona_Cases.world.casecor.plot | https://github.com/sbs87
## 2      World total & death cases, longitudinal | Corona_Cases.world.long.plot | https://github.com/sb
```

```
row_out<-apply(tmp, 2, paste, collapse="\t\n")
```

name	handle	link
World total & death cases, correlation	Corona_Cases.world.casecor.plot	https://github.com/sbs87/coronavirus/blob/master/results/Corona_Cases.world.casecor.plot.png
World total & death cases, longitudinal	Corona_Cases.world.long.plot	https://github.com/sbs87/coronavirus/blob/master/results/Corona_Cases.world.long.plot.png

Tables

CONCLUSION

Overall, the trends of COVID-19 cases is no longer in log-linear phase for world or U.S. (but some regions like MD are still in the log-linear phase). Mortality rate (deaths/confirmed RNA-based cases) is >1%, with a range depending on region. Mobility is not a strong indicator of caseload (U.S. data).

See table below for detailed breakdown.

Question	Answer
What is the effect on social distancing, descreased mobility on case load?	There is not a strong apparent effect on decreased mobility (work, grocery, retail) or increased mobility (at residence, parks) on number of confirmed cases, either as a country (U.S.) or state level. California appears to have one of the best correlations, but this is a mixed bag
What is the trend in cases, mortality across geopgraphical regions?	The confirmed total casees and mortality is overall log-linear for most countries, with a trailing off beginning for most (inlcuding U.S.). On the state level, NY, NJ, PA starting to trail off; MD is still in log-linear phase. Mortality and case load are highly correlated for NY, NJ, PA, MD. The mortality rate flucutates for a given region, but is about 3% overall.

END

End: ##—— Sat Apr 18 23:41:03 2020 ——##

Cheatsheet: http://rmarkdown.rstudio.com>

Sandbox

```
# Geographical heatmap!
install.packages("maps")
library(maps)
library
mi_counties <- map_data("county", "pennsylvania") %>%
  select(lon = long, lat, group, id = subregion)
head(mi_counties)

ggplot(mi_counties, aes(lon, lat)) +
  geom_point(size = .25, show.legend = FALSE) +
  coord_quickmap()
mi_counties$cases<-1:2226
name_overlaps(metadata,Corona_Cases.US_state)

tmp<-merge(Corona_Cases.US_state,metadata)
ggplot(filter(tmp,Province.State=="Pennsylvania"), aes(Long, Lat, group = as.factor(City))) +
  geom_polygon(aes(fill = Total_confirmed_cases), colour = "grey50") +
  coord_quickmap()


ggplot(Corona_Cases.US_state, aes(Long, Lat))+
  geom_polygon(aes(fill = Total_confirmed_cases ), color = "white")+
  scale_fill_viridis_c(option = "C")
dev.off()


require(maps)
require(viridis)

world_map <- map_data("world")
ggplot(world_map, aes(x = long, y = lat, group = group)) +
  geom_polygon(fill="lightgray", colour = "white")

head(world_map)
head(Corona_Cases.US_state)
unique(select(world_map,c("region","group"))) %>% filter()

some.eu.countries <- c(
  "US"
)
# Retrievethe map data
some.eu.maps <- map_data("world", region = some.eu.countries)

# Compute the centroid as the mean longitude and lattitude
# Used as label coordinate for country's names
region.lab.data <- some.eu.maps %>%
  group_by(region) %>%
  summarise(long = mean(long), lat = mean(lat))

unique(filter(some.eu.maps,subregion %in% Corona_Cases.US_state$Province.State) %>% select(subregion))
unique(Corona_Cases.US_state$Total_confirmed_cases.log)
ggplot(filter(Corona_Cases.US_state,Date=="2020-04-17") aes(x = Long, y = Lat)) +
```

```
  geom_polygon(aes( fill = Total_confirmed_cases.log))+
  #geom_text(aes(label = region), data = region.lab.data,  size = 3, hjust = 0.5)+
  #scale_fill_viridis_d()+
  #theme_void()+
  theme(legend.position = "none")
library("sf")
library("rnaturalearth")
library("rnaturalearthdata")

world <- ne_countries(scale = "medium", returnclass = "sf")
class(world)
ggplot(data = world) +
    geom_sf()

counties <- st_as_sf(map("county", plot = FALSE, fill = TRUE))
counties <- subset(counties, grepl("florida", counties$ID))
counties$area <- as.numeric(st_area(counties))
#install.packages("lwgeom")
class(counties)
head(counties)
ggplot(data = world) +
    geom_sf(data=Corona_Cases.US_state) +
    #geom_sf(data = counties, aes(fill = area)) +
  geom_sf(data = counties, aes(fill = area)) +
   # scale_fill_viridis_c(trans = "sqrt", alpha = .4) +
    coord_sf(xlim = c(-88, -78), ylim = c(24.5, 33), expand = FALSE)


head(counties)
tmp<-unique(select(filter(Corona_Cases.US_state,Date=="2020-04-17"),c(Lat,Long,Total_confirmed_cases.pe
st_as_sf(map("county", plot = FALSE, fill = TRUE))

join::inner_join.sf(Corona_Cases.US_state, counties)
```

https://stevenbsmith.net