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: ##—— Thu Apr 16 14:18:33 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"

# TODO: customize get_google_mobility.py script, add arguments
```

SET UP ENVIORNMENT

Load libraries and set global variables

```
# timestamp start
timestamp()
## ##------ Thu Apr 16 14:18:33 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/css
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_data
Corona_Deaths.source_url <- "https://github.com/CSSEGISandData/COVID-19/raw/master/csse_covid_19_data/c

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

function_name	description
make_long	converts input data to long format (specialized cases)
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.lo
  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","Populat

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: X4.15.20"
```

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

```
## [1] "US total: X4.15.20"
```

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

```
## [1] "World deaths: X4.15.20"
```

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

```
## [1] "World total: X4.15.20"
```

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"
```

```r
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	85	85
Albania	85	85
Algeria	85	85
Andorra	85	85
Angola	85	85
Antigua and Barbuda	85	85

	death	total
Argentina	85	85
Armenia	85	85
Australia	680	680
Austria	85	85
Azerbaijan	85	85
Bahamas	85	85
Bahrain	85	85
Bangladesh	85	85
Barbados	85	85
Belarus	85	85
Belgium	85	85
Belize	85	85
Benin	85	85
Bhutan	85	85
Bolivia	85	85
Bosnia and Herzegovina	85	85
Botswana	85	85
Brazil	85	85
Brunei	85	85
Bulgaria	85	85
Burkina Faso	85	85
Burma	85	85
Burundi	85	85
Cabo Verde	85	85
Cambodia	85	85
Cameroon	85	85
Canada	1275	1275
Central African Republic	85	85
Chad	85	85
Chile	85	85
China	2805	2805
Colombia	85	85
Congo (Brazzaville)	85	85
Congo (Kinshasa)	85	85
Costa Rica	85	85
Cote d'Ivoire	85	85
Croatia	85	85
Cuba	85	85
Cyprus	85	85
Czechia	85	85
Denmark	255	255
Diamond Princess	85	85
Djibouti	85	85
Dominica	85	85
Dominican Republic	85	85
Ecuador	85	85
Egypt	85	85
El Salvador	85	85
Equatorial Guinea	85	85
Eritrea	85	85
Estonia	85	85
Eswatini	85	85

	death	total
Ethiopia	85	85
Fiji	85	85
Finland	85	85
France	935	935
Gabon	85	85
Gambia	85	85
Georgia	85	85
Germany	85	85
Ghana	85	85
Greece	85	85
Grenada	85	85
Guatemala	85	85
Guinea	85	85
Guinea-Bissau	85	85
Guyana	85	85
Haiti	85	85
Holy See	85	85
Honduras	85	85
Hungary	85	85
Iceland	85	85
India	85	85
Indonesia	85	85
Iran	85	85
Iraq	85	85
Ireland	85	85
Israel	85	85
Italy	85	85
Jamaica	85	85
Japan	85	85
Jordan	85	85
Kazakhstan	85	85
Kenya	85	85
Korea, South	85	85
Kosovo	85	85
Kuwait	85	85
Kyrgyzstan	85	85
Laos	85	85
Latvia	85	85
Lebanon	85	85
Liberia	85	85
Libya	85	85
Liechtenstein	85	85
Lithuania	85	85
Luxembourg	85	85
Madagascar	85	85
Malawi	85	85
Malaysia	85	85
Maldives	85	85
Mali	85	85
Malta	85	85
Mauritania	85	85
Mauritius	85	85

	death	total
Mexico	85	85
Moldova	85	85
Monaco	85	85
Mongolia	85	85
Montenegro	85	85
Morocco	85	85
Mozambique	85	85
MS Zaandam	85	85
Namibia	85	85
Nepal	85	85
Netherlands	425	425
New Zealand	85	85
Nicaragua	85	85
Niger	85	85
Nigeria	85	85
North Macedonia	85	85
Norway	85	85
Oman	85	85
Pakistan	85	85
Panama	85	85
Papua New Guinea	85	85
Paraguay	85	85
Peru	85	85
Philippines	85	85
Poland	85	85
Portugal	85	85
Qatar	85	85
Romania	85	85
Russia	85	85
Rwanda	85	85
Saint Kitts and Nevis	85	85
Saint Lucia	85	85
Saint Vincent and the Grenadines	85	85
San Marino	85	85
Sao Tome and Principe	85	85
Saudi Arabia	85	85
Senegal	85	85
Serbia	85	85
Seychelles	85	85
Sierra Leone	85	85
Singapore	85	85
Slovakia	85	85
Slovenia	85	85
Somalia	85	85
South Africa	85	85
South Sudan	85	85
Spain	85	85
Sri Lanka	85	85
Sudan	85	85
Suriname	85	85
Sweden	85	85
Switzerland	85	85

	death	total
Syria	85	85
Taiwan*	85	85
Tanzania	85	85
Thailand	85	85
Timor-Leste	85	85
Togo	85	85
Trinidad and Tobago	85	85
Tunisia	85	85
Turkey	85	85
Uganda	85	85
Ukraine	85	85
United Arab Emirates	85	85
United Kingdom	935	935
Uruguay	85	85
US	85	85
US_state	276760	276760
Uzbekistan	85	85
Venezuela	85	85
Vietnam	85	85
West Bank and Gaza	85	85
Western Sahara	85	85
Yemen	85	85
Zambia	85	85
Zimbabwe	85	85

```r
# 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
```

```r
##------------------------------------------

# 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

# 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 3: Number of longitudinal datapoints (total/death) per state

Var1	Freq
Alabama	1519
Alaska	214
Arizona	461
Arkansas	1429
California	1875
Colorado	1445
Connecticut	268
Delaware	99
Diamond Princess	30
District of Columbia	31
Florida	1781
Georgia	3608
Grand Princess	31
Guam	31
Hawaii	165
Idaho	680
Illinois	1756
Indiana	2145
Iowa	1624
Kansas	1167
Kentucky	1829
Louisiana	1615
Maine	401
Maryland	689
Massachusetts	511
Michigan	1781
Minnesota	1492
Mississippi	1984
Missouri	1782
Montana	593
Nebraska	752
Nevada	277
New Hampshire	304
New Jersey	731
New Mexico	550
New York	1713
North Carolina	2236
North Dakota	574

Var1	Freq
Northern Mariana Islands	29
Ohio	1969
Oklahoma	1303
Oregon	820
Pennsylvania	1653
Puerto Rico	31
Rhode Island	178
South Carolina	1213
South Dakota	770
Tennessee	2099
Texas	3774
Utah	460
Vermont	390
Virgin Islands	31
Virginia	2567
Washington	1201
West Virginia	768
Wisconsin	1419
Wyoming	430

```
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$ca
```

ANALYSIS

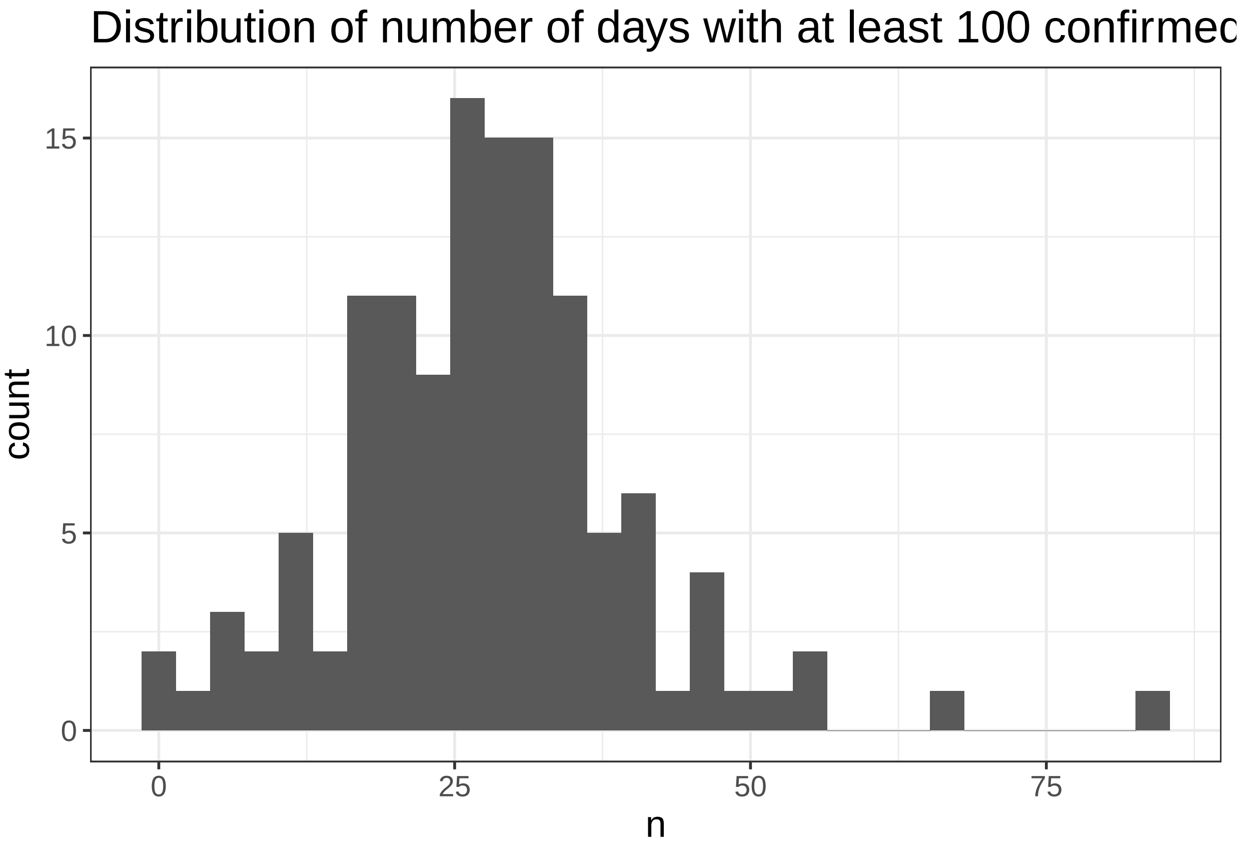
Q1: What is the trend in cases, mortality across geopgraphical regions?

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_log_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`.
```



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

Table 5: Sorted number of days with at least 100 confirmed cases

Country.Region	n
US_state	6692
China	85
Diamond Princess	66
Korea, South	56
Japan	55
Italy	53
Iran	50
Singapore	47
France	46
Germany	46
Spain	45
US	44
Switzerland	42
United Kingdom	42
Belgium	41
Netherlands	41
Norway	41
Sweden	41
Austria	39
Malaysia	38
Australia	37
Bahrain	37

Country.Region	n
Denmark	37
Canada	36
Qatar	36
Iceland	35
Brazil	34
Czechia	34
Finland	34
Greece	34
Iraq	34
Israel	34
Portugal	34
Slovenia	34
Egypt	33
Estonia	33
India	33
Ireland	33
Kuwait	33
Philippines	33
Poland	33
Romania	33
Saudi Arabia	33
Indonesia	32
Lebanon	32
San Marino	32
Thailand	32
Chile	31
Pakistan	31
Luxembourg	30
Peru	30
Russia	30
Ecuador	29
Slovakia	29
South Africa	29
United Arab Emirates	29
Armenia	28
Colombia	28
Croatia	28
Mexico	28
Panama	28
Serbia	28
Taiwan*	28
Turkey	28
Argentina	27
Bulgaria	27
Latvia	27
Algeria	26
Costa Rica	26
Dominican Republic	26
Hungary	26
Uruguay	26
Andorra	25
Bosnia and Herzegovina	25

Country.Region	n
Jordan	25
Lithuania	25
Morocco	25
New Zealand	25
North Macedonia	25
Vietnam	25
Albania	24
Cyprus	24
Malta	24
Moldova	24
Brunei	23
Burkina Faso	23
Sri Lanka	23
Tunisia	23
Ukraine	22
Azerbaijan	21
Ghana	21
Kazakhstan	21
Oman	21
Senegal	21
Venezuela	21
Afghanistan	20
Cote d'Ivoire	20
Cuba	19
Mauritius	19
Uzbekistan	19
Cambodia	18
Cameroon	18
Honduras	18
Nigeria	18
West Bank and Gaza	18
Belarus	17
Georgia	17
Bolivia	16
Kosovo	16
Kyrgyzstan	16
Montenegro	16
Congo (Kinshasa)	15
Kenya	14
Niger	13
Guinea	12
Rwanda	12
Trinidad and Tobago	12
Paraguay	11
Bangladesh	10
Djibouti	8
El Salvador	7
Guatemala	6
Madagascar	5
Mali	4
Congo (Brazzaville)	1
Jamaica	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_col
kable(sufficient_data,caption = paste0("Top ",max_colors," countries with sufficient data"))
```

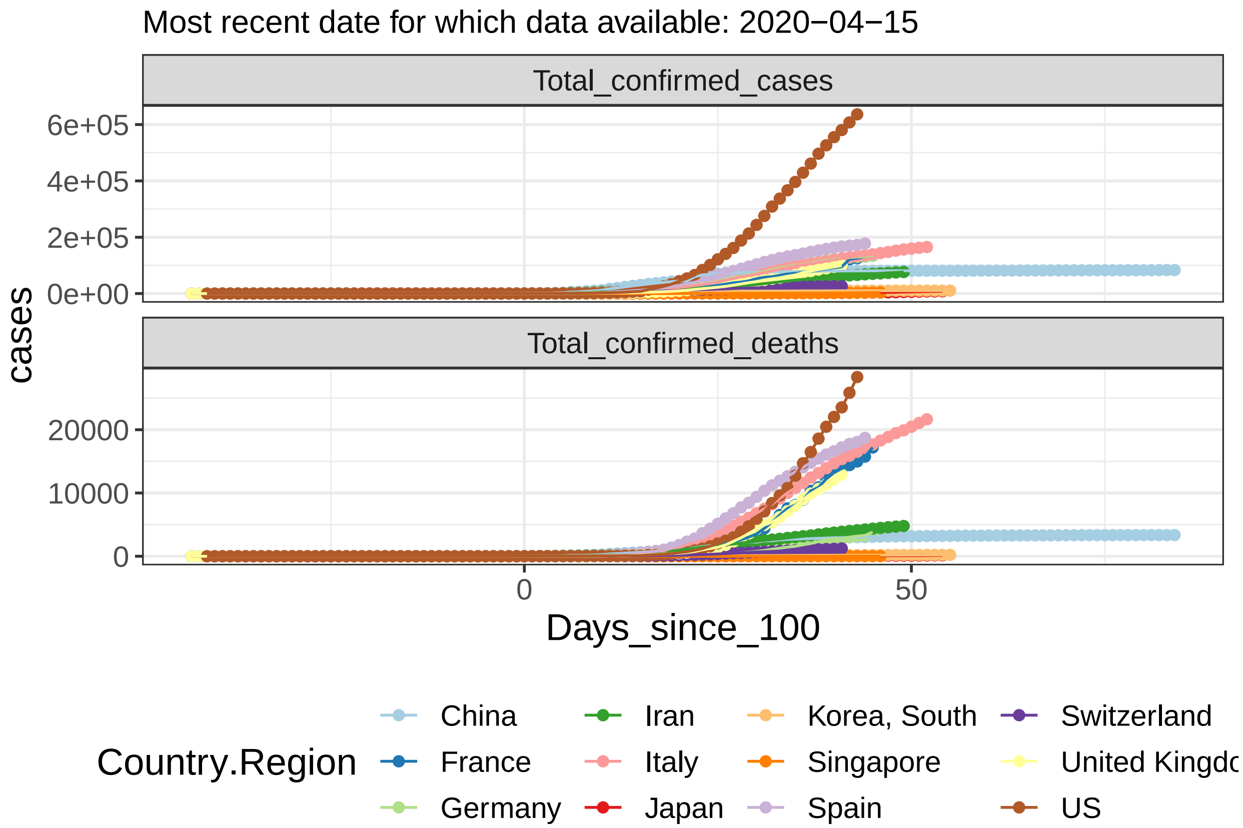
Table 6: Top 12 countries with sufficient data

Country.Region	n
China	85
Korea, South	56
Japan	55
Italy	53
Iran	50
Singapore	47
France	46
Germany	46
Spain	45
US	44
Switzerland	42
United Kingdom	42

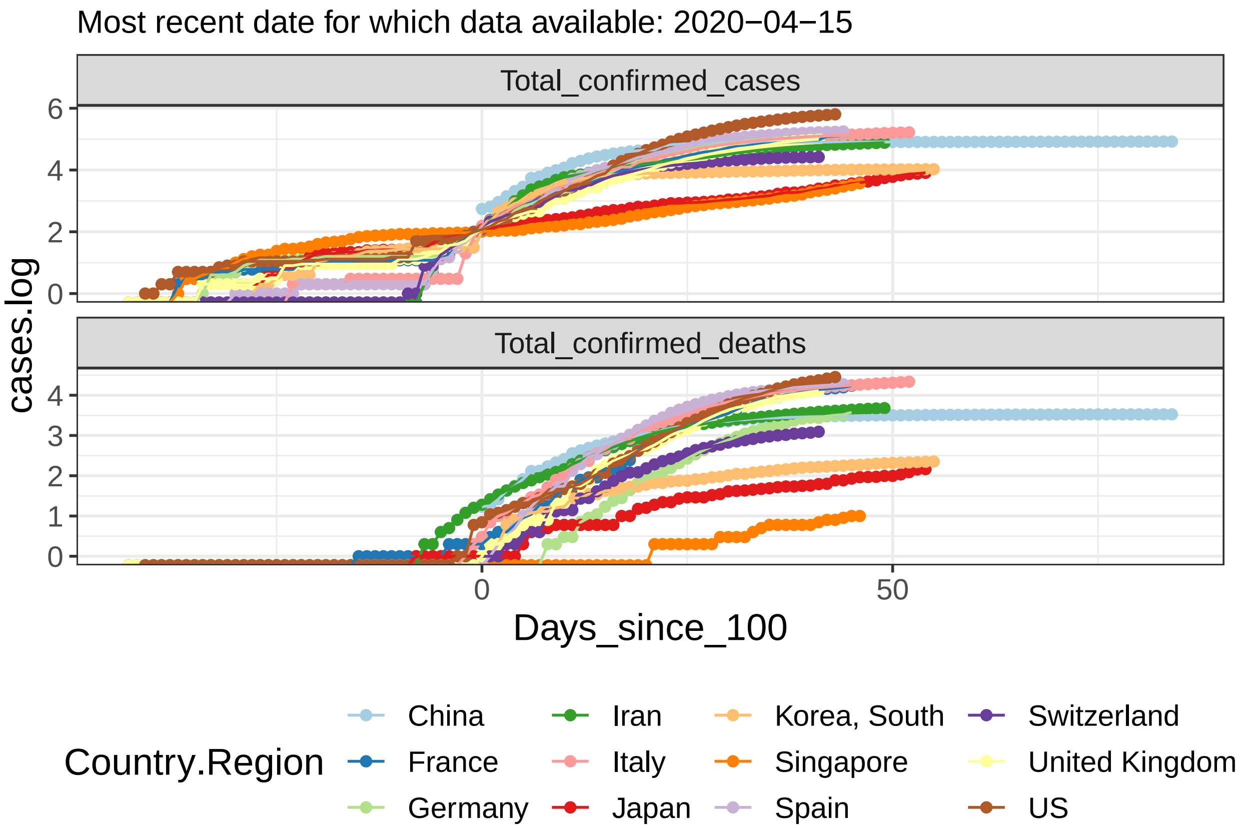
```
Corona_Cases.world<-filter(Corona_Cases,Country.Region %in% c(sufficient_data$Country.Region))


  #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","Ne

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



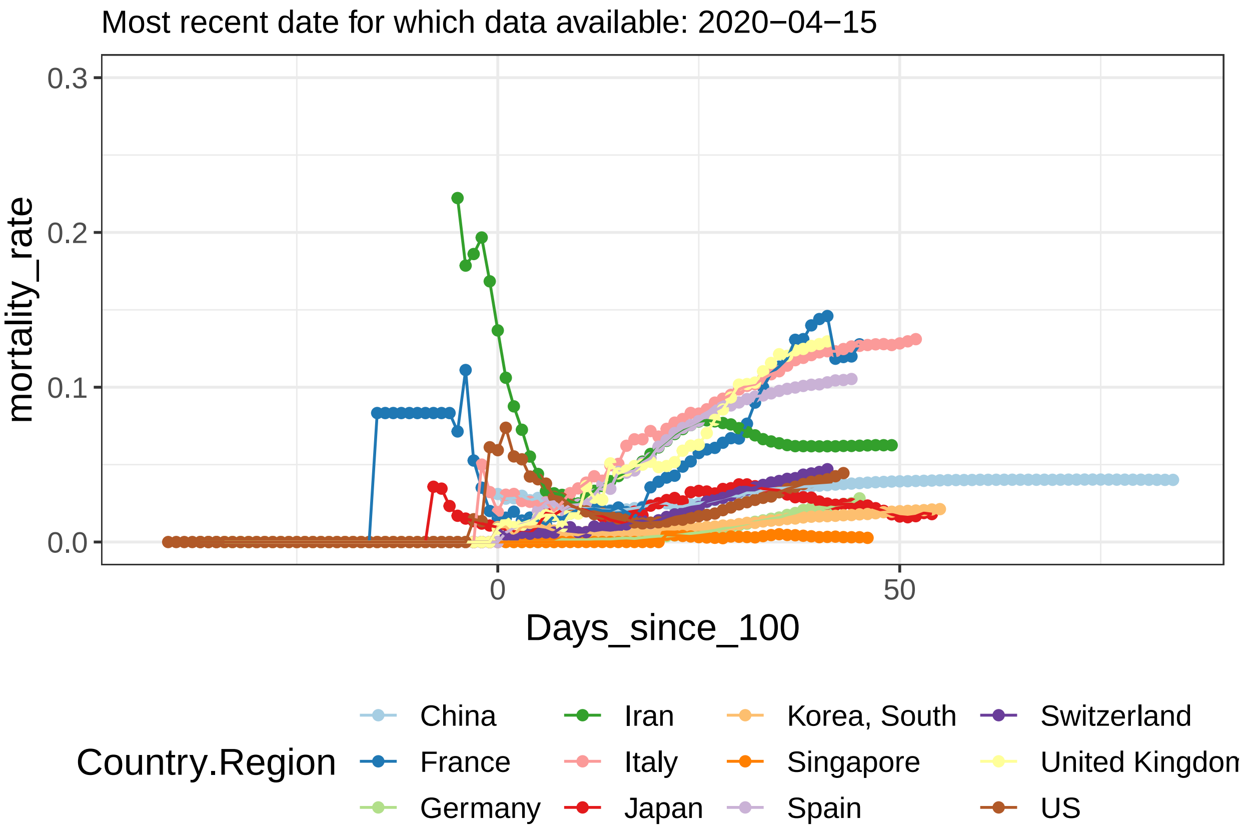
```
(Corona_Cases.world.loglong.plot<-baseplot.world+
    geom_point(data=Corona_Cases.world.long,aes(y=cases.log))+
    geom_line(data=Corona_Cases.world.long,aes(y=cases.log))+
    facet_wrap(~case_type,scales = "free_y",ncol=1)+
    ggtitle(timestamp_plot.world))
```



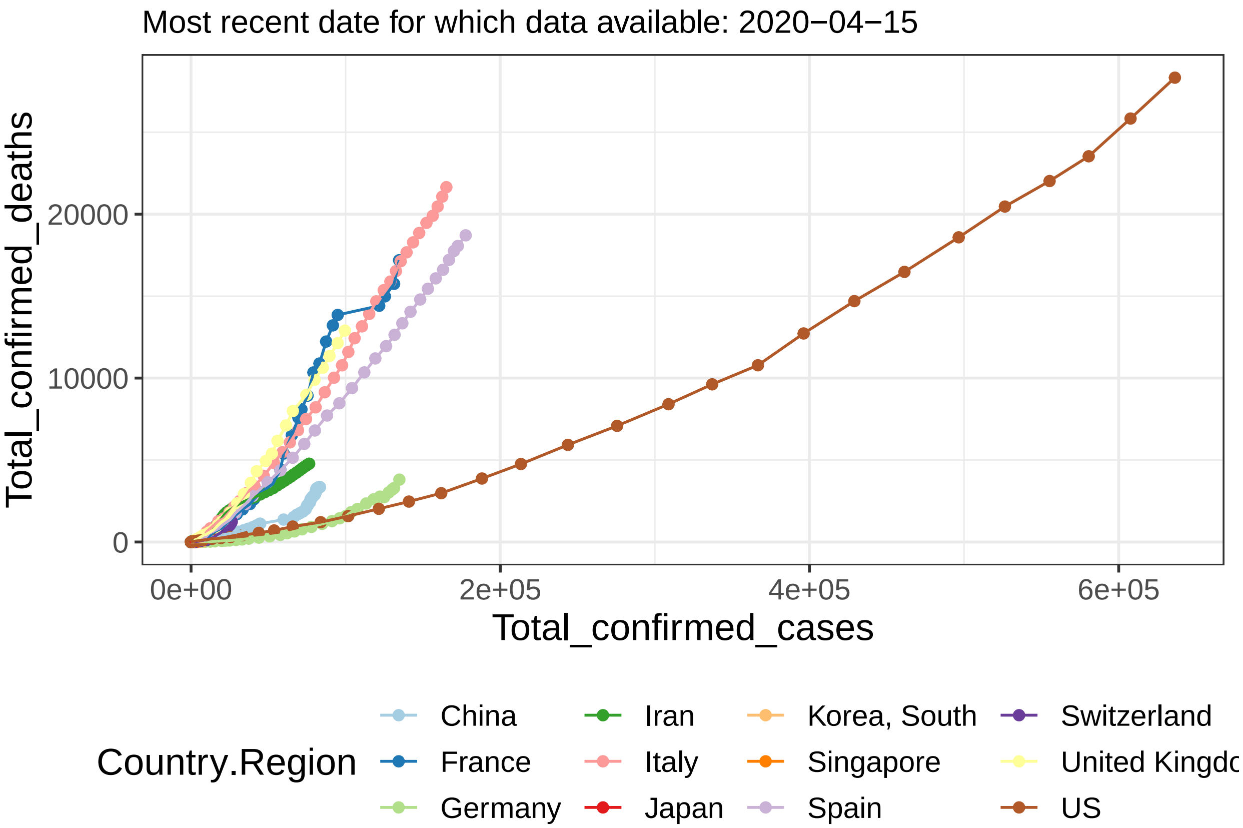
```
##/////////////////////////
### Plot Longitudinal mortality rate

(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_confirm
  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)
```

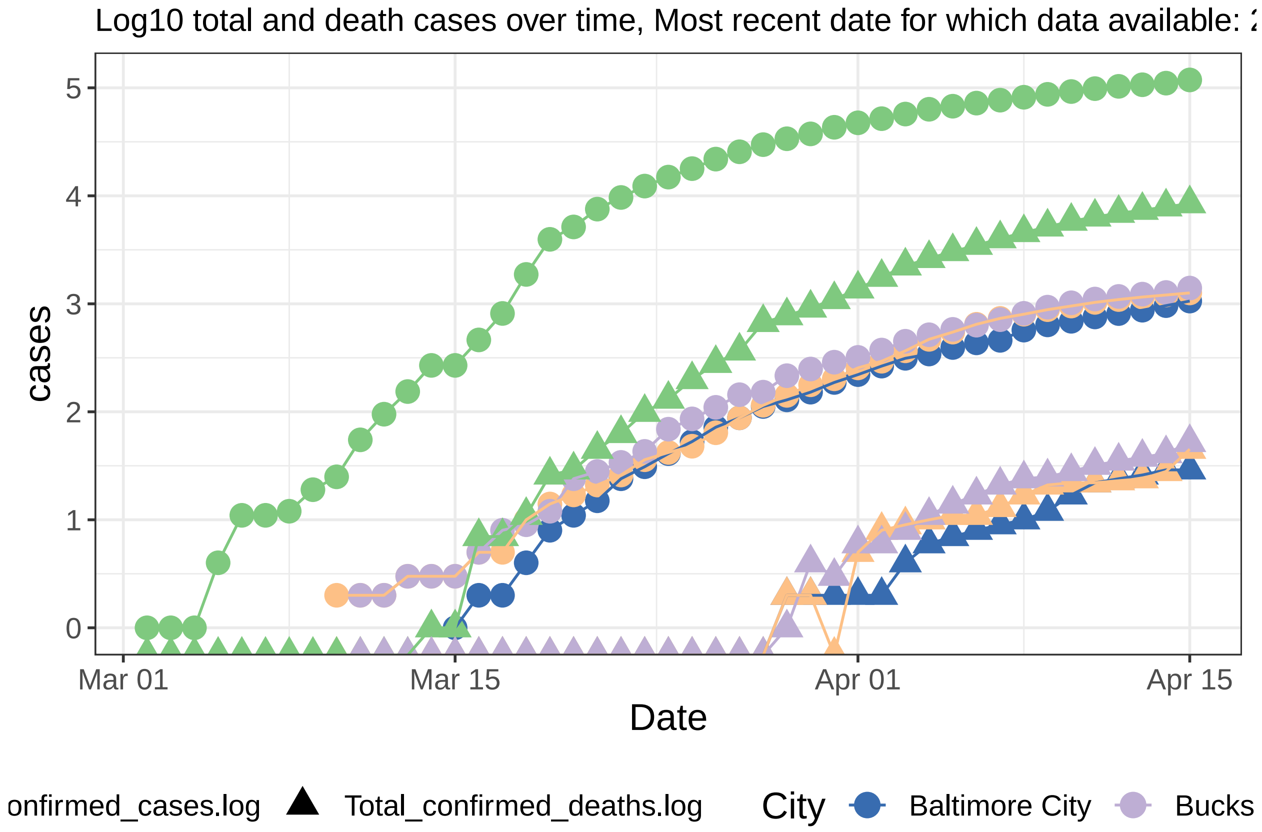
```
## Warning: Removed 100 rows containing missing values (geom_point).

## 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)
```

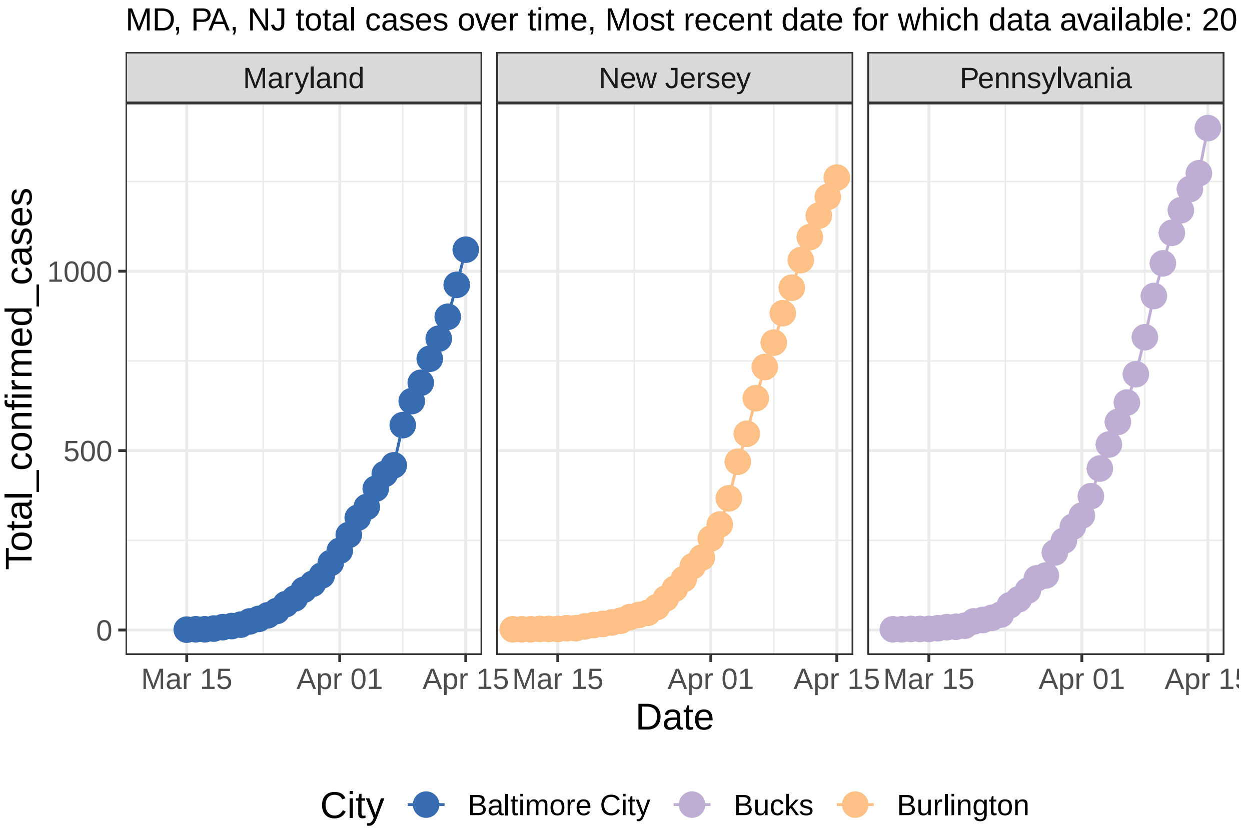
```
## [1] "/Users/stevensmith/Projects/coronavirus/results/Corona_Cases.world.casecor.plot.png"
```

```
##-----------------------------------------
## 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))
```

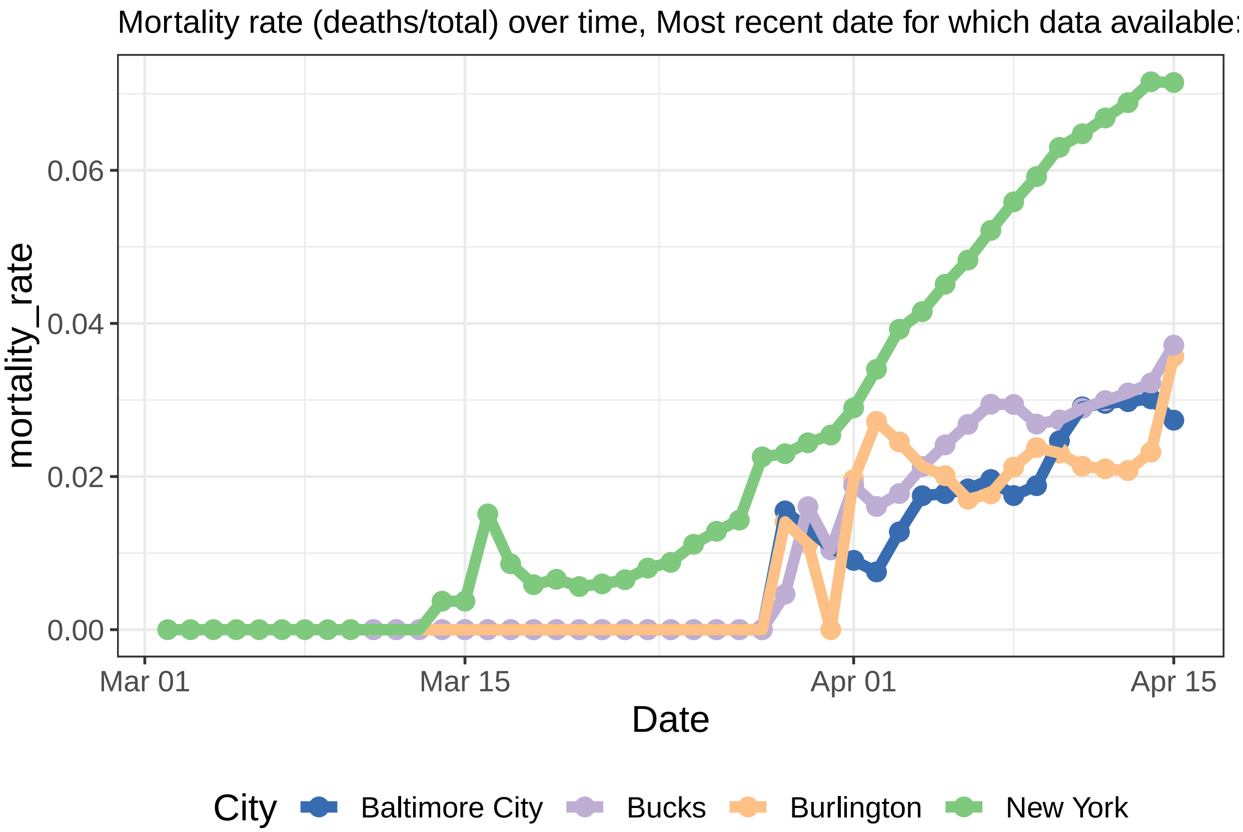
```
Corona_Cases.US_state.long.plot<-baseplot.US+geom_point(data=Corona_Cases.US_state.long,aes(y=cases.log
##------------------------------------------
## Plot US City Data
##-----------------------------------------




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))
```



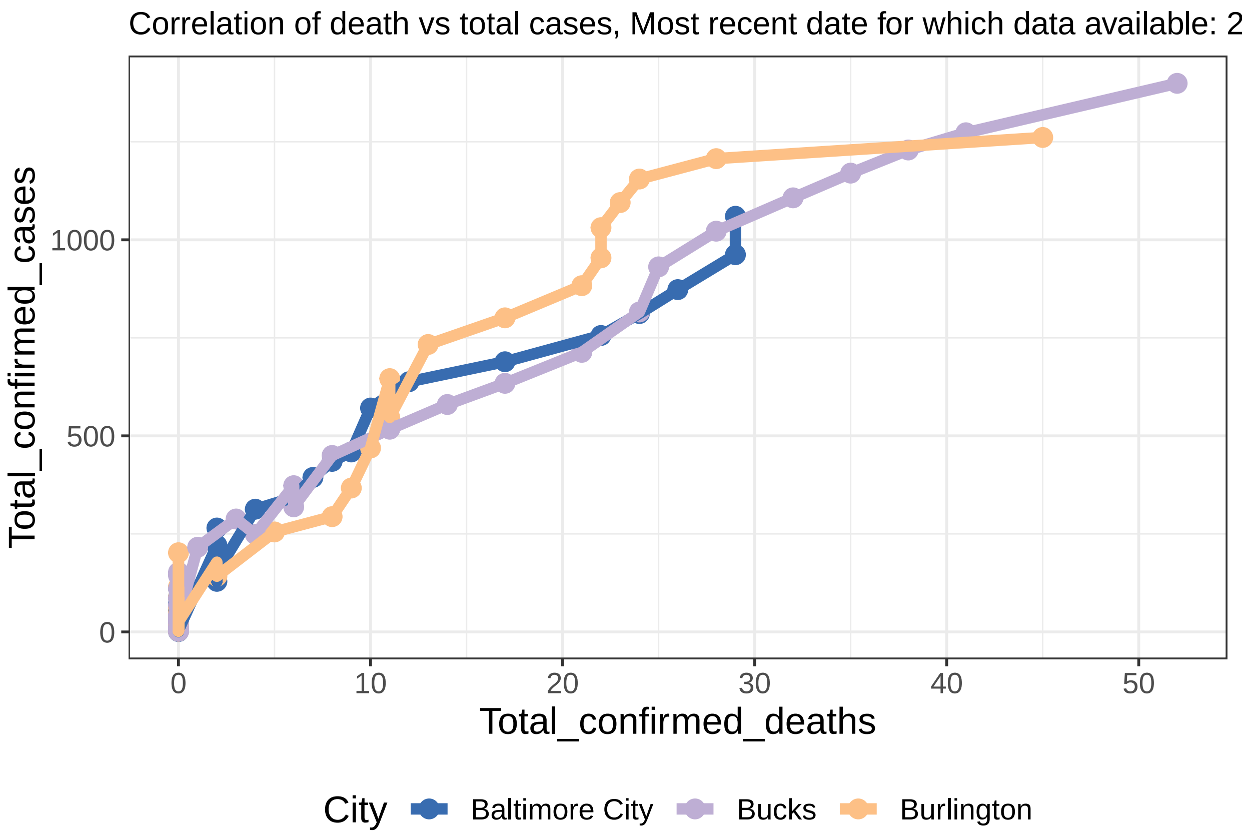
```
(Corona_Cases.city.long.plot<-ggplot(filter(Corona_Cases.US.plotdata,Province.State !="New York"),aes(x=
  geom_point(size=4)+
  geom_line()+
  default_theme+
  facet_grid(~Province.State,scales = "free_y")+
  ggtitle(paste("MD, PA, NJ total cases over time,",timestamp_plot))+
  theme(legend.position = "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))
```



```
#write_plot(Corona_Cases.US.log.plot,wd=results_dir_custom)
#write_plot(Corona_Cases.US.plot,wd=results_dir_custom)
#write_plot(Corona_Cases.tristate.plot,wd=results_dir_custom)


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"
```

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
```

```r
##------------------------------------------

# 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$coefficients[2],b = model_fit_x$coefficient
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
```

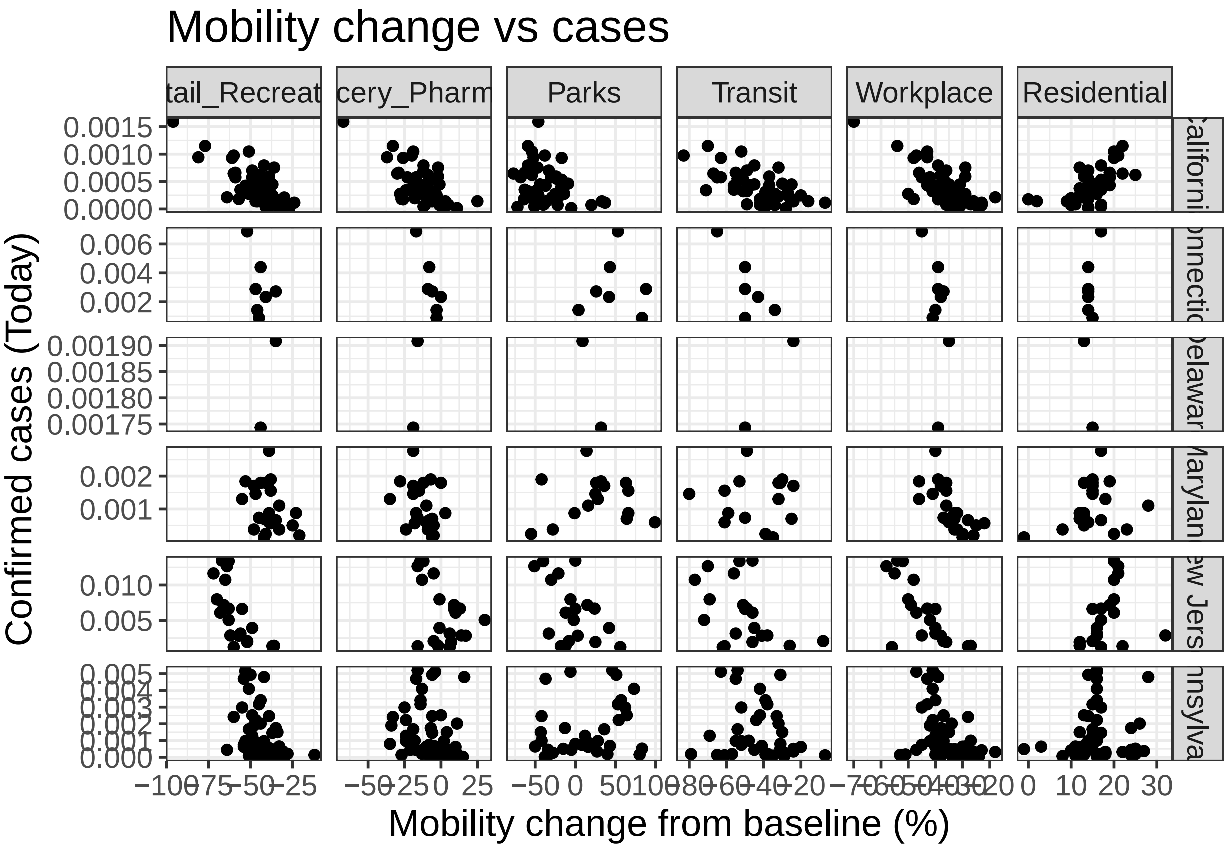
Q3: What is the effect on social distancing, descreased mobility on case load?

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

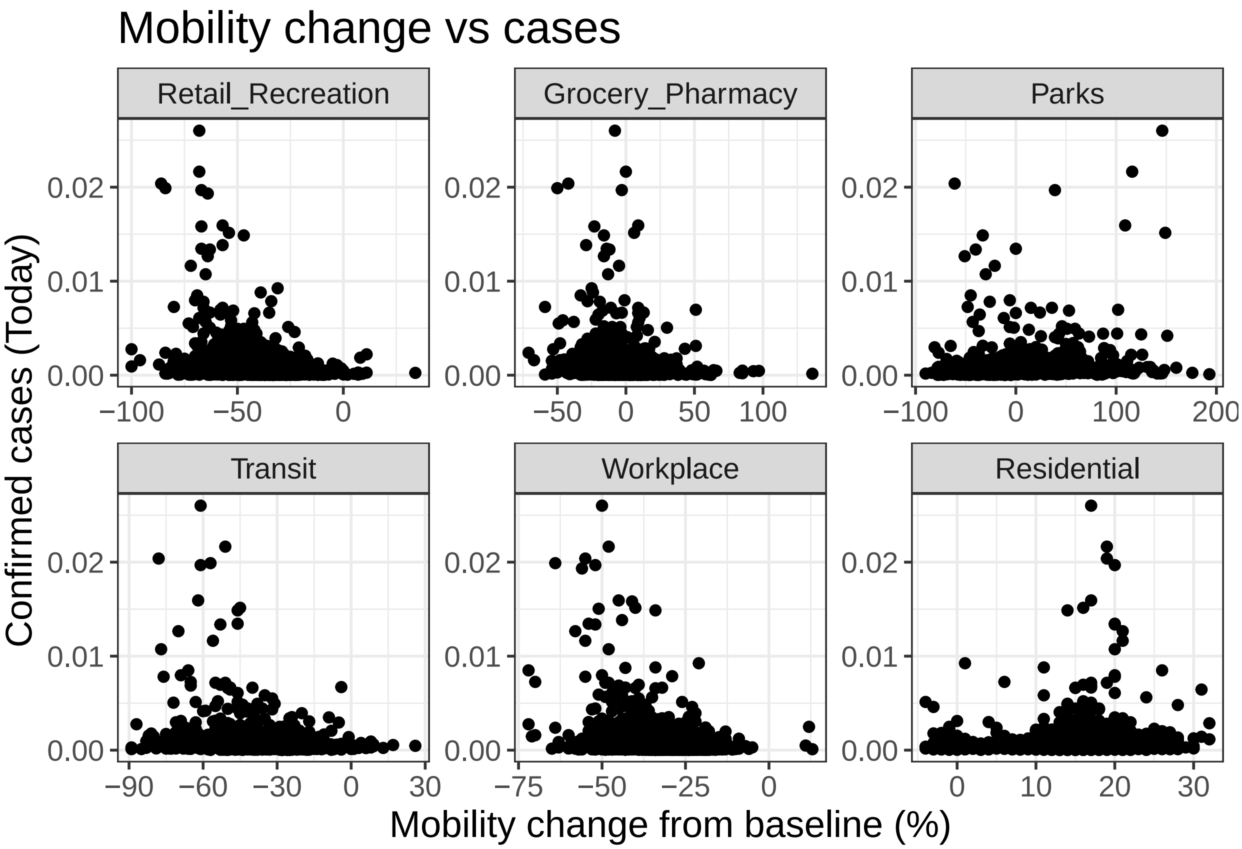
Data from https://www.google.com/covid19/mobility/

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

# UNDER DEVELOPMENT
# TODO convert % to numeric in mobility data
# TODO standardize headers in mobility data
# TODO standardize counties in mobility data to JHU source
# TODO normalize case load to population for mobility data
# TODO automate get_mobility.py script so most recent data is availble
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.tmp<-merge(Corona_Cases.US_state,plyr::rename(mobility,c("State"="Province.State"

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_ca
mobility_measures<-c("Retail_Recreation","Grocery_Pharmacy","Parks","Transit","Workplace","Residential"

plot_data<-filter(Corona_Cases.US_state.tmp, Date.numeric==max(Corona_Cases.US_state.tmp$Date.numeric) 
plot_data$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","C
  facet_grid(Province.State~variable,scales = "free")+
  xlab("Mobility change from baseline (%)")+
  ylab(paste0("Confirmed cases (Today)"))+
  default_theme+
  ggtitle("Mobility change vs cases"))
```



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



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

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

Table 7: 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
Alaska	Residential	0.9801495	13.0
Vermont	Parks	0.9380911	-35.5
South Dakota	Parks	0.9145688	-26.0
New Hampshire	Parks	0.8894686	-20.0
Connecticut	Grocery_Pharmacy	-0.8886774	-6.0
Hawaii	Transit	0.8678700	-89.0
Alaska	Grocery_Pharmacy	-0.8522624	-7.0

Province.State	variable	cor	median_change
Utah	Workplace	-0.8257291	-33.0
Hawaii	Parks	0.8240971	-72.0
Massachusetts	Workplace	-0.8160722	-39.0
Connecticut	Transit	-0.7928647	-50.0
Rhode Island	Workplace	-0.7858415	-39.5
North Dakota	Residential	-0.7746267	17.0
New Mexico	Parks	0.7648296	-31.5
Hawaii	Workplace	-0.7535008	-46.0
Utah	Retail_Recreation	-0.7499142	-36.0
Utah	Grocery_Pharmacy	-0.7376920	-3.0
California	Retail_Recreation	-0.7357414	-44.0
New Jersey	Workplace	-0.7256778	-44.0
Kansas	Parks	0.7251101	72.0
California	Workplace	-0.7138498	-36.0
Maryland	Workplace	-0.7086971	-35.0
Utah	Transit	-0.7048700	-18.0
New Jersey	Retail_Recreation	-0.6733613	-62.5
Connecticut	Residential	0.6715324	14.0
Vermont	Grocery_Pharmacy	-0.6701429	-25.0
New York	Workplace	-0.6638950	-34.5
California	Grocery_Pharmacy	-0.6591532	-12.0
Massachusetts	Retail_Recreation	-0.6589582	-44.0
Nevada	Transit	-0.6486644	-20.0
North Dakota	Transit	0.6470030	-48.0
North Dakota	Retail_Recreation	-0.6388331	-43.5
New York	Retail_Recreation	-0.6238109	-46.0
California	Transit	-0.6143508	-42.0
California	Residential	0.6128323	14.0
Rhode Island	Residential	-0.6022688	18.5
Maine	Transit	-0.6014272	-50.0
Connecticut	Workplace	-0.5777085	-39.0
Montana	Retail_Recreation	-0.5764223	-51.0
Montana	Workplace	-0.5719817	-40.5
Maine	Workplace	-0.5693317	-30.0
Montana	Transit	-0.5620388	-41.0
North Dakota	Parks	0.5503056	-34.0
Massachusetts	Grocery_Pharmacy	-0.5380281	-7.0
Rhode Island	Retail_Recreation	-0.5360091	-45.0
Montana	Parks	-0.5303208	-58.0
Idaho	Workplace	-0.5229692	-29.5
Utah	Residential	-0.5197978	12.0
New Jersey	Parks	-0.4977182	-6.0
Hawaii	Residential	0.4967913	19.0
Kansas	Grocery_Pharmacy	-0.4756168	-14.0
Maine	Parks	0.4740260	-31.0
Nebraska	Grocery_Pharmacy	-0.4726213	0.0
Nevada	Retail_Recreation	-0.4700865	-43.0
Minnesota	Parks	0.4690157	-3.5
Connecticut	Retail_Recreation	-0.4661806	-45.0
Idaho	Transit	-0.4630352	-30.0
New Jersey	Grocery_Pharmacy	-0.4604355	2.5
Massachusetts	Transit	-0.4555300	-45.0

Province.State	variable	cor	median_change
Virginia	Transit	-0.4487712	-33.0
Montana	Residential	0.4479566	14.0
Vermont	Residential	0.4430198	11.5
Colorado	Workplace	-0.4394690	-39.0
Virginia	Retail_Recreation	-0.4346862	-35.0
New York	Parks	0.4340375	20.0
Pennsylvania	Workplace	-0.4323959	-36.0
Arkansas	Parks	-0.4305437	-12.0
Idaho	Grocery_Pharmacy	-0.4253249	-4.0
New Jersey	Transit	-0.4196529	-50.5
New Mexico	Residential	0.4193898	13.5
Colorado	Residential	0.4147624	14.0
Rhode Island	Parks	0.4145638	52.0
New York	Transit	-0.4145294	-48.0
Florida	Parks	-0.4096604	-43.0
Michigan	Workplace	-0.4047589	-40.0
North Dakota	Grocery_Pharmacy	-0.4046811	-9.5
Pennsylvania	Retail_Recreation	-0.3978054	-45.0
Oregon	Parks	0.3959356	16.5
Idaho	Retail_Recreation	-0.3879193	-41.0
Arizona	Grocery_Pharmacy	-0.3874411	-15.0
Hawaii	Grocery_Pharmacy	0.3846654	-34.0
Montana	Grocery_Pharmacy	-0.3826009	-16.0
Kansas	Retail_Recreation	-0.3815962	-39.0
Rhode Island	Grocery_Pharmacy	0.3748927	-7.5
Colorado	Transit	-0.3740063	-36.0
Utah	Parks	-0.3716502	0.0
Colorado	Retail_Recreation	-0.3711122	-44.0
Idaho	Parks	0.3652935	-22.0
Illinois	Transit	-0.3576090	-31.0
Arizona	Transit	0.3565089	-38.0
Alaska	Workplace	-0.3561072	-35.0
Maryland	Retail_Recreation	-0.3531858	-39.0
South Dakota	Transit	-0.3529519	-40.0
Vermont	Retail_Recreation	0.3488753	-57.0
Mississippi	Parks	0.3405218	-25.0
Washington	Transit	-0.3356081	-33.5
Colorado	Parks	-0.3338503	2.0
New Mexico	Retail_Recreation	-0.3304166	-42.0
Florida	Transit	-0.3272449	-49.0
Maine	Retail_Recreation	-0.3256381	-41.5
Florida	Residential	0.3219248	14.0
Colorado	Grocery_Pharmacy	-0.3183064	-17.0
Virginia	Workplace	-0.3180398	-32.0
New Hampshire	Grocery_Pharmacy	-0.3180236	-6.0
Texas	Transit	0.3135843	-42.0
Maryland	Grocery_Pharmacy	-0.3084032	-10.0
Arkansas	Retail_Recreation	-0.3077937	-30.0
Alabama	Workplace	-0.3064330	-29.0
Iowa	Residential	-0.3016748	13.0
Kentucky	Parks	0.3009198	28.5
Nevada	Residential	-0.3003301	18.0

Province.State	variable	cor	median_change
California	Parks	-0.2988077	-38.0
Arkansas	Residential	0.2976033	12.0
Arizona	Residential	0.2964677	13.0
North Carolina	Retail_Recreation	-0.2960871	-33.0
Florida	Workplace	-0.2936010	-33.0
Oregon	Residential	0.2899537	10.5
New Jersey	Residential	0.2897533	18.0
Maine	Grocery_Pharmacy	-0.2872983	-10.5
Virginia	Grocery_Pharmacy	-0.2866668	-8.0
Pennsylvania	Parks	0.2858636	13.0
New Mexico	Grocery_Pharmacy	-0.2853571	-12.0
New York	Grocery_Pharmacy	-0.2853045	8.0
South Carolina	Residential	0.2767172	12.0
Indiana	Grocery_Pharmacy	-0.2747970	-5.5
Mississippi	Grocery_Pharmacy	-0.2703852	-8.0
Tennessee	Retail_Recreation	-0.2699664	-30.0
Hawaii	Retail_Recreation	0.2610592	-56.0
Georgia	Grocery_Pharmacy	-0.2598244	-10.0
New Hampshire	Retail_Recreation	-0.2551458	-41.0
Iowa	Workplace	-0.2549528	-29.0
Maryland	Residential	0.2547816	15.0
Nebraska	Residential	0.2501824	14.0
Illinois	Workplace	-0.2487918	-30.0
Wisconsin	Transit	-0.2465065	-23.5
Massachusetts	Residential	0.2421259	15.0
Arizona	Retail_Recreation	-0.2269253	-42.5
Iowa	Parks	0.2242605	28.5
Washington	Workplace	-0.2180272	-38.0
Georgia	Retail_Recreation	-0.2173409	-41.0
Michigan	Retail_Recreation	-0.2168307	-53.0
Kansas	Residential	0.2123547	13.0
Rhode Island	Transit	-0.2117649	-56.0
Pennsylvania	Grocery_Pharmacy	-0.2113410	-6.0
Nebraska	Retail_Recreation	-0.2082028	-37.5
Tennessee	Grocery_Pharmacy	-0.2078832	6.0
Alabama	Residential	0.2062622	11.0
Michigan	Grocery_Pharmacy	-0.2045430	-11.0
Kentucky	Residential	0.2044962	12.0
Georgia	Workplace	-0.2019012	-33.5
Mississippi	Workplace	-0.2010190	-33.0
Florida	Grocery_Pharmacy	-0.1980474	-14.0
Wisconsin	Workplace	-0.1976285	-31.0
North Dakota	Workplace	0.1968640	-33.5
Oklahoma	Residential	0.1932994	15.0
Oklahoma	Workplace	-0.1906996	-30.0
Texas	Residential	-0.1883315	15.0
Oklahoma	Retail_Recreation	0.1873703	-31.0
Tennessee	Parks	0.1844074	10.5
Arizona	Parks	0.1837907	-44.5
Tennessee	Workplace	-0.1827435	-31.0
Wisconsin	Parks	0.1812903	51.5
Missouri	Retail_Recreation	-0.1800298	-37.0

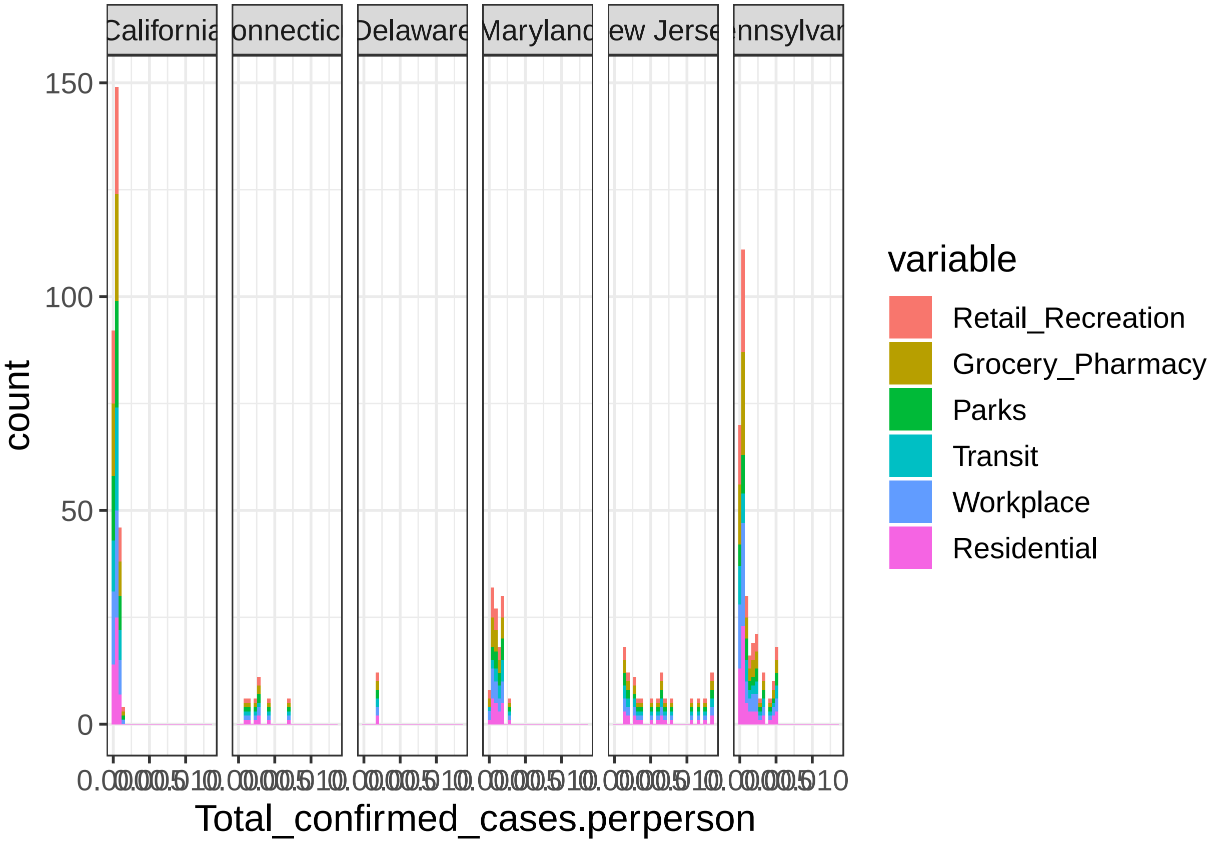
Province.State	variable	cor	median_change
Oklahoma	Grocery_Pharmacy	0.1787751	-0.5
South Dakota	Retail_Recreation	-0.1746248	-38.5
Texas	Parks	0.1736111	-42.0
Kentucky	Workplace	-0.1732501	-34.5
Washington	Parks	0.1724897	-3.5
Indiana	Retail_Recreation	-0.1723565	-38.0
New Hampshire	Residential	-0.1715010	14.0
Minnesota	Workplace	0.1707732	-33.0
North Carolina	Transit	0.1688764	-32.0
Arizona	Workplace	-0.1642294	-35.0
South Carolina	Retail_Recreation	-0.1633272	-35.0
Alabama	Transit	-0.1631734	-36.5
Florida	Retail_Recreation	-0.1604293	-43.0
Alabama	Grocery_Pharmacy	-0.1601563	-2.0
Pennsylvania	Transit	-0.1580547	-41.5
South Carolina	Parks	-0.1571772	-23.0
South Carolina	Workplace	0.1561853	-30.0
Illinois	Residential	0.1534711	14.0
New Hampshire	Workplace	-0.1499564	-37.0
Missouri	Transit	-0.1486519	-23.0
North Carolina	Residential	0.1435933	13.0
South Dakota	Grocery_Pharmacy	0.1408678	-9.0
Nevada	Grocery_Pharmacy	-0.1388614	-6.5
Michigan	Parks	0.1382268	33.0
Vermont	Workplace	-0.1366545	-43.0
Nevada	Workplace	-0.1357693	-40.0
Idaho	Residential	-0.1355995	11.0
Missouri	Grocery_Pharmacy	-0.1329874	2.0
Alabama	Parks	0.1329547	-1.0
Iowa	Transit	-0.1328017	-25.0
Kentucky	Retail_Recreation	-0.1310149	-30.0
Arkansas	Workplace	-0.1292491	-26.0
Minnesota	Retail_Recreation	0.1292107	-41.0
Maine	Residential	-0.1256546	11.0
Nebraska	Transit	0.1250813	-11.5
North Carolina	Parks	0.1243907	7.0
Ohio	Transit	0.1234934	-28.0
Oregon	Grocery_Pharmacy	0.1212809	-7.0
New Mexico	Workplace	-0.1206169	-34.0
Arkansas	Grocery_Pharmacy	0.1204187	3.5
Pennsylvania	Residential	0.1197503	15.0
Virginia	Residential	0.1192923	14.0
Minnesota	Grocery_Pharmacy	-0.1177003	-4.0
Illinois	Grocery_Pharmacy	-0.1170483	2.0
Georgia	Residential	-0.1164178	13.0
Minnesota	Transit	-0.1160218	-28.5
Wisconsin	Residential	-0.1151553	14.0
South Carolina	Transit	-0.1134097	-45.0
Illinois	Retail_Recreation	-0.1127271	-40.0
New Mexico	Transit	0.1111206	-38.0
Wisconsin	Retail_Recreation	-0.1101510	-44.5
Washington	Residential	0.1096651	13.0

Province.State	variable	cor	median_change
Indiana	Workplace	-0.1054354	-34.0
Illinois	Parks	0.1039082	26.5
Mississippi	Residential	0.1033007	13.0
Michigan	Residential	0.1018827	15.0
Alaska	Retail_Recreation	-0.0996743	-35.5
Nebraska	Workplace	-0.0988010	-32.0
Washington	Retail_Recreation	-0.0982868	-42.0
Tennessee	Residential	0.0976437	12.0
Kansas	Transit	-0.0972618	-26.5
Oregon	Transit	-0.0971458	-28.0
Texas	Retail_Recreation	-0.0944977	-39.0
Kansas	Workplace	-0.0935800	-31.0
Virginia	Parks	0.0899091	6.0
Oklahoma	Parks	-0.0898834	-23.0
Ohio	Workplace	-0.0881106	-35.0
Wisconsin	Grocery_Pharmacy	0.0858634	-1.0
Kentucky	Grocery_Pharmacy	-0.0857384	4.5
Maryland	Parks	0.0831430	27.0
Georgia	Transit	-0.0799498	-35.0
New Hampshire	Transit	-0.0761430	-57.0
New York	Residential	0.0749256	17.5
Kentucky	Transit	0.0726273	-31.0
Ohio	Residential	0.0706094	14.0
North Carolina	Grocery_Pharmacy	0.0662695	1.0
Missouri	Workplace	0.0648543	-28.5
North Carolina	Workplace	-0.0628910	-31.0
Iowa	Grocery_Pharmacy	-0.0602787	4.0
Massachusetts	Parks	-0.0584422	39.0
Arkansas	Transit	0.0504261	-27.0
Tennessee	Transit	0.0494191	-32.0
Georgia	Parks	-0.0493484	-6.0
Washington	Grocery_Pharmacy	-0.0417604	-7.0
Michigan	Transit	0.0404771	-46.0
Oregon	Workplace	-0.0389695	-32.0
Nevada	Parks	0.0359711	-12.5
Connecticut	Parks	0.0353629	43.0
South Carolina	Grocery_Pharmacy	-0.0335059	1.0
Texas	Workplace	0.0319843	-31.0
Indiana	Residential	0.0312018	12.0
Ohio	Retail_Recreation	-0.0308137	-36.0
Mississippi	Retail_Recreation	0.0255810	-40.0
Missouri	Parks	-0.0255780	0.5
Vermont	Transit	0.0235674	-63.0
Alabama	Retail_Recreation	-0.0197793	-39.0
Iowa	Retail_Recreation	-0.0196719	-37.0
Ohio	Grocery_Pharmacy	0.0157480	0.0
South Dakota	Residential	0.0150571	15.0
Mississippi	Transit	0.0146552	-38.5
Indiana	Transit	-0.0116150	-29.0
South Dakota	Workplace	0.0098092	-35.0
Oklahoma	Transit	-0.0097138	-26.0
Ohio	Parks	0.0093672	67.5

Province.State	variable	cor	median_change
Missouri	Residential	0.0092517	13.0
Maryland	Transit	0.0073123	-39.0
Indiana	Parks	-0.0056280	29.0
Nebraska	Parks	-0.0037219	55.5
Texas	Grocery_Pharmacy	-0.0023363	-13.5
Minnesota	Residential	-0.0015431	18.0
Oregon	Retail_Recreation	-0.0000870	-41.0
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

```
ggplot(filter(plot_data,Province.State %in% c("Pennsylvania","Maryland","New Jersey","California","Dela
  facet_grid(~Province.State)+
  default_theme
```

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



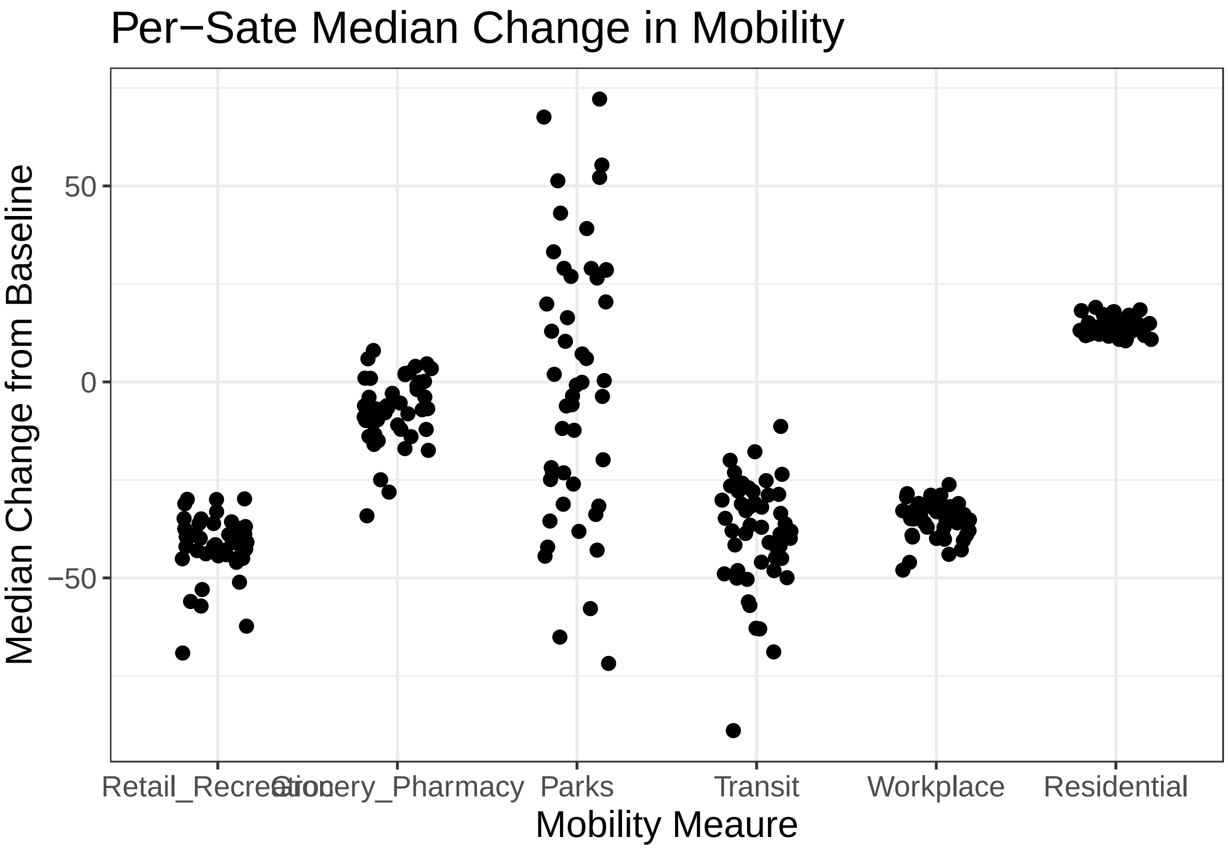
```
write_plot(mobility.plot,wd = results_dir)
```

```
## [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"
```

```
# TODO secondary question: rank greatest to least mobility

(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

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

name	section	name2	link
World total & death cases, correlatio	What is the effect on social distancing, descreased mobility on case load?	Corona_Cases.world.casecor.plot	https://github.com/sbs87/coronavirus/blob/master/results/Corona_Cases.world.casecor.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

Question	Answer
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: ##—— Thu Apr 16 14:19:34 2020 ——##

Cheatsheet: http://rmarkdown.rstudio.com> # TODO * mkdir the results dir if it doesn't exist * make ggplot a dependency for plot.utils?
* automated way of downloading daily data * fix plot_utils, add dataset and documentation * Auto git mv the new data?

Sandbox

```
##TODO:
# 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()
```

smith
research

https://stevenbsmith.net