BRIEFING (FIVE MINUTES) ON US VIRAL OUTBREAK : The Spread of COVID-19 in the US as of March 31, 2020 (Source: CDC*, USA)

Steve Ampah
BiostatsReportAutomation@gmail.com

March 31, 2020

Contents

List of Figures

1 Project Question

Assume you are the only data scientist or biostatistician among a team of 30, dedicated to working on the current viral outbreak in the US. Your input at every meeting is to provide your team with the current number of reported cases and deaths from this outbreak as reported on the CDC website. You always have 5 to 10 minutes to brief your team of these numbers, and they expect an accompanying short report of at least 5 sentenses describing these numbers, a heat map of US showing the intensity of spread of COVID-19, and possibly a bar plot showing order of states in decreasing order of reported cases.

Note no modeling of cases is expected just a summary of these numbers as reported at CDC website. Using either R, SAS or Python build a report that supports your 5 minutes briefing at your usual daily meeting. It's expected that when needed report be reproducible and be able to update old report with new numbers as and when reported cases change on CDC website.

US VIRAL OUTBREAK BRIEFING (Source: CDC*, USA):

The Spread of COVID-19 in the US as of March 31, 2020

As of March 31, 2020 there has been a total of 163,539 COVID-19 confirmed cases in the US, of which 2919 were transmited through close contact relations and 1,042 travel related while the remaining 159,578 are under investigation for method of transmission. Total deaths is on the increase and currently stands at 1.7 % (2860) of total confirmed cases.

Spread across states

Confirmed cases of this outbreak across states range 0 to 67131, with New York having the highest number of cases while 4 states (American Samoa, Marshall Islands, Micronesia and Palau) have no reported cases as of now. Arranging states in decreasing order of impact of COVID-19 outbreak revealed the top 15 (25%) states have reported cases that ranged 1933 and 67131 while the bottom 15 states have cases less than 174. With median reportd cases currently at 576, implied 29 states have cases below this mark, Figure 1, 2 and 3.

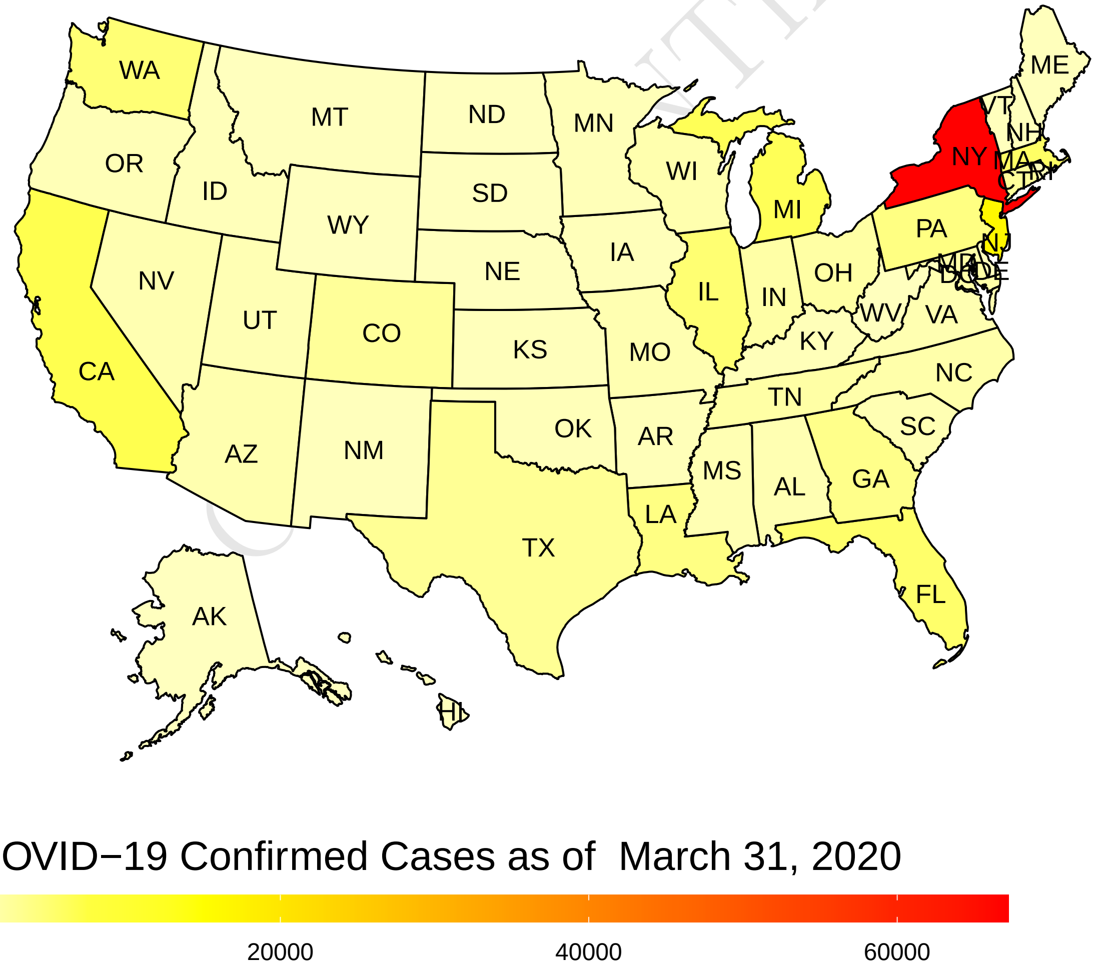

Figure 1: A heat map of US showing the spread of COVID-19 across states

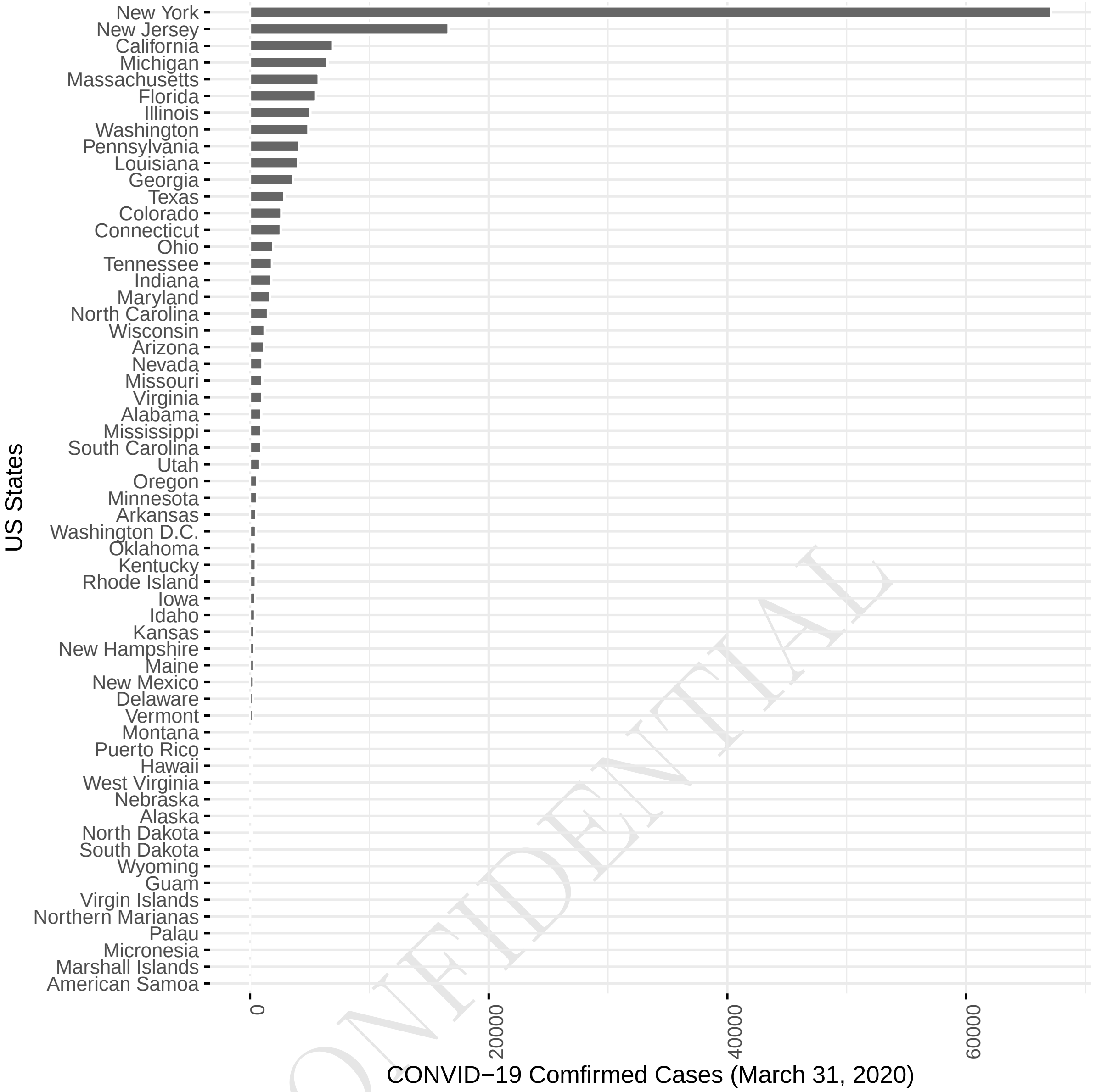

Figure 2: A barplot showing US states in decreasing order of outbreak, with other US terrotories depicting minimal or no spread of COVID-19

Zooming in on Figure 2 above as:

a) Top 30 states with worse outbreak

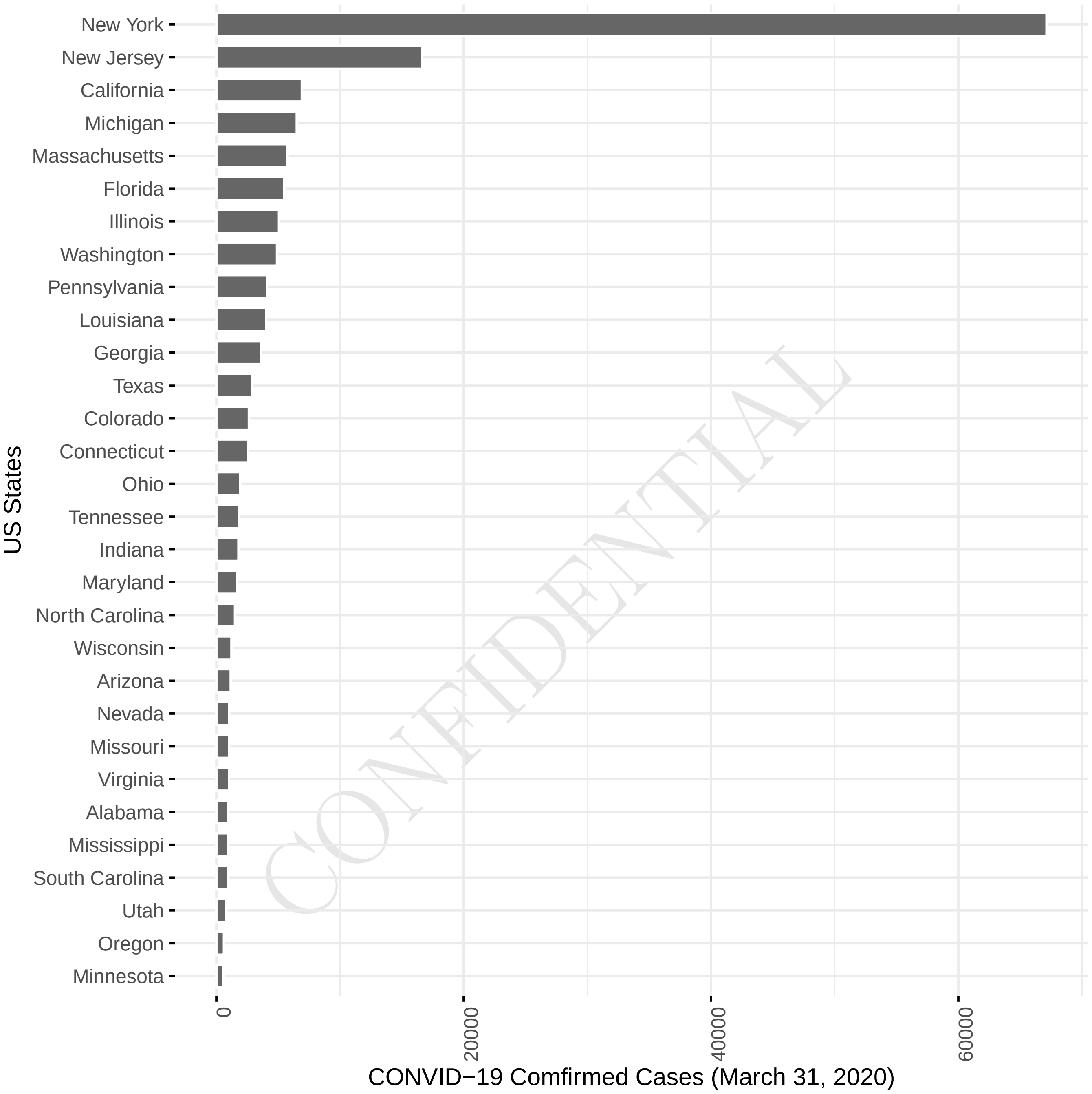

Figure 3: Top 30 states with higher impact by the COVID19 pandemic

b) Bottom 29 states with bad outbreak

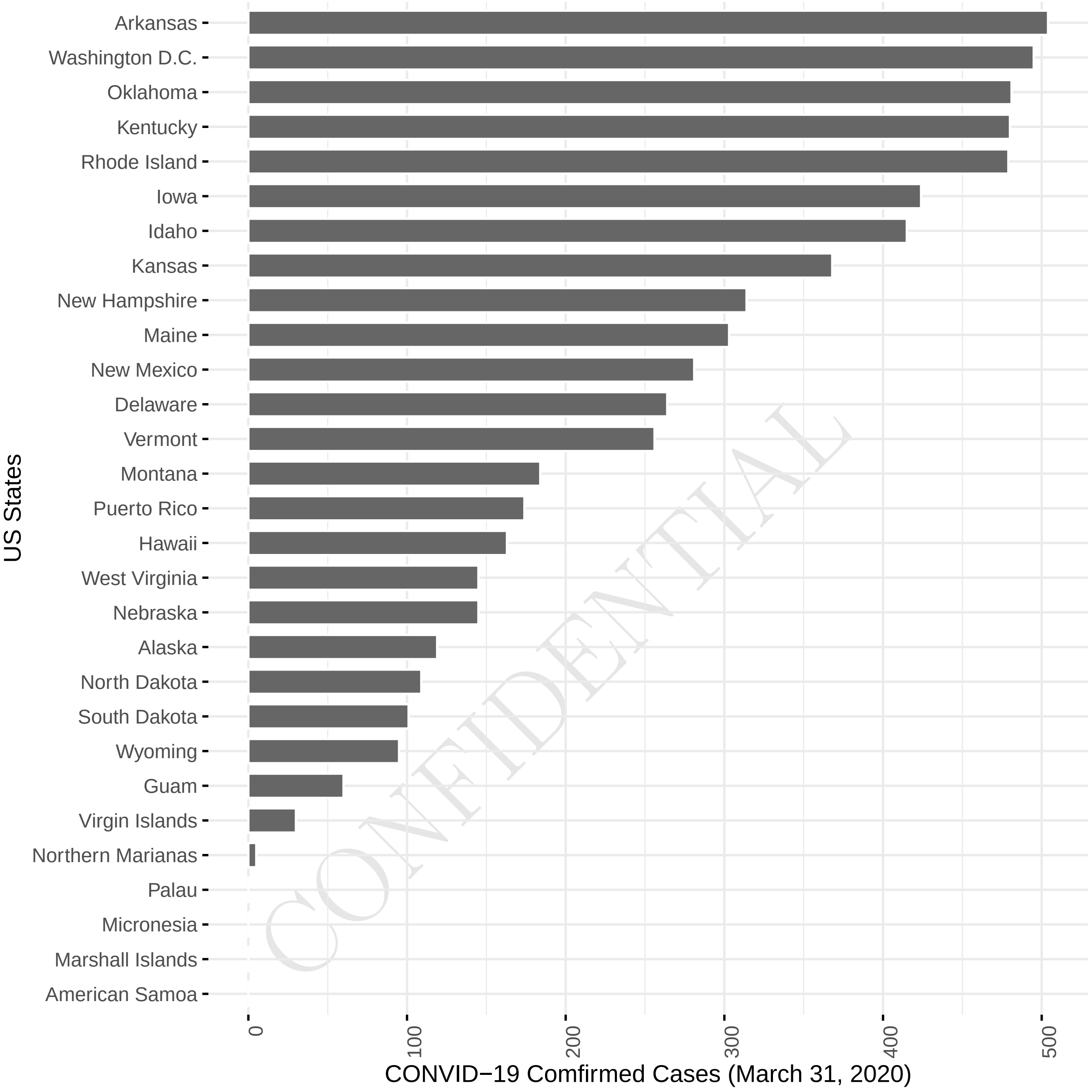

Figure 4: The bottom 29 states less impacted by the COVID19 pandemic

Reference

* https://www.cdc.gov/coronavirus/2019-ncov/cases-updates/

2 R Code for Report

```
## ----title,echo=F,results="hide",include=F-----------------------------------

options(scipen=9999)
library(rvest)
  library(stringr)
    ht=read_html("https://www.cdc.gov/coronavirus/2019-ncov/cases-updates/cases-in-us.html")

#Current Date
current.date=ht%>%html_node(".text-red")%>%html_text()
  current.Dt=word(current.date,start=2,end=-1)
    ncurrent.date1 =as.Date(current.Dt,"%b %d, %y")
      ncurrent.date=as.numeric(ncurrent.date1 )




## ----chunk1, results="hide", include=F,message=F,comment=F,eval=T,warning=F----
library(knitr)
  library(plyr)
   library(ggplot2)
     library(dplyr)

#install.packages("pdftools")
# library(pdftools)
#install.packages("tidyverse")

#devtools::install_github("r-lib/xml2")
library(rvest)
   library(stringr)
     library(tidyr)

# Set true to provide information on whether CDC has updated reported casse
  # as of last report update
  ReportStatus=TRUE



## ----chunk2, results="hide", include=F,message=F,comment=F--------------------
ht=read_html("https://www.cdc.gov/coronavirus/2019-ncov/cases-updates/cases-in-us.html")
#Current Date
  current.date=ht%>%html_node(".text-red")%>%html_text()
   first_date=word(current.date,start=2,end=-1)


#Specify first Date,
# Date data was first read from the CDC website
#first_date="March19,2020"
firstDtn=as.Date(first_date,"%b%d,%y")

#Reload all the holders here

#Initalize a date holder to hold all future read in dates
Dateholder<-NULL
   wDateholder<-NULL
     Tableholder<-NULL
        convidDatholder<-NULL
#Casesholder<-NULL
#Deathsholder<-NULL

Dateholder[1]<-firstDtn
  wDateholder[1]<-current.Dt
    Month=months(firstDtn)
      Day=substr(as.Date(Dateholder[1],origin="1970-01-01"), start=9,stop=10 )
      Year=data.table::year(firstDtn)
         wDateholder[1]=paste(Month," ",Day,", ",Year,sep="")
#as.Date(Dateholder[[1]],"1970-01-01")
getwd()->path
nam.data=dir(paste(path,"/currentData",sep=""))
  datline=readLines(paste(path,"/currentData/",nam.data[1],sep=""))
    #N=length(datline)

col=length(datline)/59
row0=NULL
N=length(datline)/col  # Why col because each state has four data entries

for(i in c(1: N)){
     if(i==1){
        tmp=datline
             }
              row1=tmp[1:col]
              row0= rbind(row0,row1)
              tmp=tmp[-c(1:col)]

               if(i==N){
                 outdat=data.frame(row0,DataUpdate=1,UpDateDt=as.Date(
                     Dateholder[1],origin="1970-01-01") )
                 names(outdat)=c("State","Cases","MethodSpread","DataUpdateN","UdateDt")
                       }
               }


#Cleaning data
#strsplit(outdat$State,sep="Click")

outdat1=outdat%>%separate(col=State, into=c("myState", "trush"),sep="Click")%>%select(-trush)%>%separate(col=Cases,into=c("Case","Max"),sep="to")

outdat2=outdat1%>%tbl_df%>%transmute(State=myState,Range=NA,Cases=as.numeric(ifelse(!is.na(Max), Max,ifelse(Case=="None","0",Case))),Transmission=MethodSpread)
```

```
## ----chunk3, results="hide", include=F,message=F,comment=F--------------------
#library(mapproj)
#dt1 <- as.data.table(copy(state.x77))
#dt1$state <- tolower(rownames(state.x77))
#dt1 <- dt1[,.(state, Population)]
# only need state name and variable to plot in the input file:

#Try this
suppressPackageStartupMessages({
library(ggplot2)
library(maps)
library(usmap)
library(data.table)
#library(ggsn) # for scale bar `scalebar`
#install.packages("ggrepel")
library(ggrepel) # if need to repel labels
})



statepop2=statepop%>%transmute(abbr,State=full,pop_2015,fips)
uscovid=left_join(outdat2,statepop2,by="State")

#uscovid%>%print(n=Inf)
uscovid19=uscovid%>%mutate(State_abbr=ifelse(State=="Washington D.C.","DC",
                                   ifelse(State=="American Samoa", "AS",
                                    ifelse(State=="Marshall Islands","MH",
                                      ifelse(State=="Micronesia","FM",
                                         ifelse(State=="Northern Marianas","MP",
                                            ifelse(State=="Guam","GU",abbr) ))))))

#Use in built map data from usmap R Package
us_map <- usmap::us_map() # used to add map scale
#library(rms)
dt5=uscovid19%>%transmute(state=tolower(State),State,Cases)
   pp1=usmap::plot_usmap(data = dt5, values = "Cases", labels = T)+
      labs(fill = '')

pp2=pp1 + theme(legend.position = "bottom", legend.title=element_text(size=17),
        legend.text=element_text(size=10))
    ppp2=pp2+scale_fill_gradientn(colours=rev(heat.colors(10)),na.value="grey90"
                         , guide = guide_colourbar(barwidth = 30,
                                                   barheight = 0.8,
                                               #put legend title on top of legend
                                               title.position = "top")) +
  labs(fill = paste("COVID-19 Confirmed Cases as of ",wDateholder[1],sep=" "),
x = "Longitude", y = "Latitude")


pdf(paste(path,"/fighold/heatfig1.pdf",sep=""))
ppp2
dev.off()




## ----barplot,results="hide", echo=F,include=F---------------------------------
#install.packages("ggpubr")
library(ggpubr)
dt6=dt5%>%transmute(States=toupper(state),State,Cases,Group=ifelse(state%in%c("american samoa",
                                                                    "guam" ,"maine",
                                                                    "micronesia" ,
                                                                    "palau" ,
                                                                     "puerto rico" ,
                                                                    "marshall islands"
                                                                    ),"Other US Territories","US States"))
pdf(paste(path,"/fighold/barfig2.pdf",sep=""))
ggbarplot(dt6, x = "State", y = "Cases",
          fill = "grey40",           # change fill color by mpg_level
          color = "white",            # Set bar border colors to white
          palette = "jco",            # jco journal color palett. see ?ggpar
          sort.val = "asc",          # Sort the value in descending order
          sort.by.groups = FALSE,     # Don't sort inside each group
          x.text.angle = 90,          # Rotate vertically x axis texts
          ylab = paste("CONVID-19 Comfirmed Cases (",
                       wDateholder[1], ")",sep=""),
          legend.title = "United States",
          rotate = T,
          xlab="US States",
          ggtheme = theme_minimal()
          )
dev.off()


#arrange states in decreasing order of cases
N.1=nrow(dt6)
dt6.top=dt6%>%arrange(desc(Cases))%>%slice(1:30)
uq1=dt6.top%>%slice(15)%>%select(Cases)%>%unlist%>%as.vector

lq1=dt6%>%arrange(desc(Cases))%>%slice(45)%>%select(Cases)%>%unlist%>%as.vector
dt6.bottom=dt6%>%arrange(desc(Cases))%>%slice(31:N.1)


pdf(paste(path,"/fighold/barfig2a.pdf",sep=""))
ggbarplot(dt6.top, x = "State", y = "Cases",
          fill = "grey40",           # change fill color by mpg_level
          color = "white",            # Set bar border colors to white
          palette = "jco",            # jco journal color palett. see ?ggpar
          sort.val = "asc",          # Sort the value in descending order
          sort.by.groups = FALSE,     # Don't sort inside each group
          x.text.angle = 90,          # Rotate vertically x axis texts
          ylab = paste("CONVID-19 Comfirmed Cases (",
                       wDateholder[1], ")",sep=""),
          legend.title = "United States",
          rotate = T,
          xlab="US States",
          ggtheme = theme_minimal()
          )
dev.off()

pdf(paste(path,"/fighold/barfig2b.pdf",sep=""))
ggbarplot(dt6.bottom, x = "State", y = "Cases",
          fill = "grey40",           # change fill color by mpg_level
          color = "white",            # Set bar border colors to white
          palette = "jco",            # jco journal color palett. see ?ggpar
          sort.val = "asc",          # Sort the value in descending order
          sort.by.groups = FALSE,     # Don't sort inside each group
          x.text.angle = 90,          # Rotate vertically x axis texts
          ylab = paste("CONVID-19 Comfirmed Cases (",
                       wDateholder[1], ")",sep=""),
          legend.title = "United States",
          rotate = T,
          xlab="US States",
          ggtheme = theme_minimal()
          )
dev.off()

# Read in the Total number of deaths, and cases and include in summary


ht=read_html("https://www.cdc.gov/coronavirus/2019-ncov/cases-updates/cases-in-us.html")
#step 1: Read first the website URL
#Step 2: Date of update
#Step 3: Extract total number of cases and deaths, broken that method of transmision
#Step 4: Scrap numbers from html data tables read from the site

#Current Date
current.date=ht%>%html_node(".text-red")%>%html_text()
current.Dt=word(current.date,start=2,end=-1)
current.Dtn=as.Date(current.Dt, "%b %d, %y")

Table <- html_table(ht, fill=T)
wTotal=Table[[1]][4,2]
wTravel= Table[[1]][1,2]
nTravel=gsub(",","",wTravel)
wCloseContact=Table[[1]][2,2]
nCloseContact=gsub(",","",wCloseContact)
wInvestigation=Table[[1]][3,2]
nInvestigation=gsub(",","",wInvestigation)
Total=as.numeric(gsub(",","",wTotal))

#Deaths
Death=ht%>%html_node(".card-body li:nth-child(2)")%>%html_text()
Deaths=gsub(",","",Death)
#word(Deaths,start=3,end=-1)
nDeaths=as.numeric(word(Deaths,start=3,end=-1))

rg=range(dt6$Cases)
med=median(dt6$Cases)
lq=ceiling(quantile(dt6$Cases)[[2]])
uq=ceiling(quantile(dt6$Cases)[[4]])

nZero=dt6%>%filter(Cases==0)%>%summarize(zero=n())%>%unlist%>%as.vector
zeroStates=dt6%>%filter(Cases==0)%>%select(State)%>%unlist%>%as.vector


if(nZero==1)zeroStatesfinal=zeroStates
#zeroStatesfinal=NULL
if(nZero>2){
zeroStatesfinal=paste(paste(zeroStates[-nZero],collapse=", "), " and ", zeroStates[nZero],sep="")
}

#Finding the State with the highest number of cases
highest.state=dt6%>%arrange(desc(Cases))%>%filter(row_number()==1)%>%select(State)%>%unlist%>%as.vector





## ----SaveFirst,echo=F,results="hide",include=F--------------------------------

#prepare data to store for future use
NextData=data.frame(outdat2,wDate=wDateholder[1],Date=Dateholder[1],Tcases=Total,Travel=nTravel,
          nClosect=nCloseContact,nInvest=nInvestigation)


#Readinold data and test last read in date with new
# if new date is greater then append above

oldsingleData=read.csv(paste(path,"/savedData/oldsingleData.csv",sep=""))
  logic=identical(as.numeric(NextData$Cases),as.numeric(oldsingleData$Cases))
oldData=read.csv(paste(path,"/savedData/firstData.csv",sep=""))
lastDate=max(oldData$Date)

if ( (ncurrent.date > lastDate)&(!logic) ){
  cumData=rbind(NextData,oldData[-1])


write.csv(cumData,file= paste(path,"/savedData/firstData.csv",sep=""))
  #Note Start Date is Friday March19, 2020
}

write.csv(NextData,file=paste(path,"/savedData/oldsingleData.csv",sep=""))
```

```
updateStatus=paste("CDC website has not had any updates in reported cases of COVID19 as of last report update on ", wDateholder[1],sep="")

jjj=as.Date(lastDate,origin="1970-01-01")

#Check to inform analyst to update data, copy and create data for next update
if(ncurrent.date > lastDate){
  updateStatus=paste("Today is ",  wDateholder[1], " and there has been updates on CDC website, please consider updaing number of cases across states, global numbers may be corre



}




## ----ReportStatus,echo=F,results="asis",include=T----------------------------

if(ReportStatus==T){
 cat("\\clearpage\n")
  cat("\\noindent \\textbf{\\textcolor{red}{Note: ",updateStatus,"}}\n",sep="" )



}
```