**####Packages**

**library(tidyverse)**

**library(usmap)**

**library(ggplot2)**

**library(plotly)**

**library(ggiraph)**

**library(scales)**

**library(skimr)**

**library(summary)**

usdata <- read.csv("https://covidtracking.com/data/download/all-states-history.csv", header=T)

usdata

colnames(usdata)

attach(usdata)

date <- as.Date(date)

**Today <- subset(usdata, date==max(date))**

**Today**

library(ggplot2)

**ggplot(Today, aes(x=state, y=positive)) +**

**geom\_bar(stat="identity", fill="steelblue")+**

**geom\_text(aes(label=positive), vjust=-0.3, size=2.5)**

**ggplot(Today, aes(x=state, y=death)) +**

**geom\_bar(stat="identity", fill="red")+**

**geom\_text(aes(label=positive), vjust=-0.3, size=2.5)**

library(usmap)

plot\_usmap(data = Today, values = "death", color = "red") +

scale\_fill\_continuous(

low = "grey", high = "red", name = "COVID Deaths Per State", label = scales::comma

) + theme(legend.position = "right")

December <- subset(usdata,date >= "2020-12-01" & date <= max(date))

December

library(ggiraph)

##Positive and deaths

Dec\_postives <- ggplot(December, aes( x = date, y = positive, color = death) ) + theme\_minimal()

Dec\_postives1 <- Dec\_postives + geom\_point\_interactive(aes(tooltip = state), size = 2)

ggiraph(code = print(Dec\_postives1), width = .7)

##Positive and hospitalized

Dec\_hospitalized <- ggplot(December, aes( x = date, y = positive, color = hospitalized) ) + theme\_minimal()

Dec\_hospitalized1 <- Dec\_hospitalized + geom\_point\_interactive(aes(tooltip = state), size = 2)

ggiraph(code = print(Dec\_hospitalized1), width = .7)

**####Months**

**library(plotly)**

**March <- subset(usdata, date=="2020-03-31")**

**April <- subset(usdata, date=="2020-04-30")**

**May <- subset(usdata, date=="2020-05-31")**

**June <- subset(usdata, date=="2020-06-30")**

**July <- subset(usdata, date=="2020-07-31")**

**August <- subset(usdata, date=="2020-08-30")**

**September <- subset(usdata, date=="2020-09-30")**

**October <- subset(usdata, date=="2020-10-31")**

**November <- subset(usdata, date=="2020-11-30")**

**Months.data <- list(March,April,May,June,July,August,September,October,November)**

**for (i in Months.data){**

**Today$hover <- with(Today, paste( state, '<br>', "Death", death, '<br>',"Hospitalized", hospitalized, "<br>",**

**"Negative", negative, '<br>',"Positive", positive))**

**# give state boundaries a white border**

**l <- list(color = toRGB("white"), width = 2)**

**# specify some map projection/options**

**g <- list(**

**scope = 'usa',**

**projection = list(type = 'albers usa'),**

**showlakes = TRUE,**

**lakecolor = toRGB('white')**

**)**

**fig <- plot\_geo(Today, locationmode = 'USA-states')**

**fig <- fig %>% add\_trace(**

**z = ~death, text = ~hover, hoverinfo = 'text',locations = ~state,**

**color = ~death, colors = 'Blues'**

**)**

**fig <- fig %>% colorbar(title = "Deaths")**

**fig <- fig %>% layout(**

**title = 'COVID Data Deaths',**

**geo = g**

**)**

**fig**

**#####Updated Code for Months**

**library(plotly)**

**March <- subset(usdata, date=="2020-03-31")**

**April <- subset(usdata, date=="2020-04-30")**

**May <- subset(usdata, date=="2020-05-31")**

**June <- subset(usdata, date=="2020-06-30")**

**July <- subset(usdata, date=="2020-07-31")**

**August <- subset(usdata, date=="2020-08-30")**

**September <- subset(usdata, date=="2020-09-30")**

**October <- subset(usdata, date=="2020-10-31")**

**November <- subset(usdata, date=="2020-11-30")**

**i<-(March)**

**i$hover <- with(i, paste( state, '<br>', "Death", death, '<br>',"Hospitalized", hospitalized, "<br>",**

**"Negative", negative, '<br>',"Positive", positive))**

**# give state boundaries a white border**

**l <- list(color = toRGB("white"), width = 2)**

**# specify some map projection/options**

**g <- list(**

**scope = 'usa',**

**projection = list(type = 'albers usa'),**

**showlakes = TRUE,**

**lakecolor = toRGB('white')**

**)**

**fig <- plot\_geo(i, locationmode = 'USA-states')**

**fig <- fig %>% add\_trace(**

**z = ~death, text = ~hover, hoverinfo = 'text',locations = ~state,**

**color = ~death, colors = 'Reds'**

**)**

**fig <- fig %>% colorbar(title = "Deaths")**

**fig <- fig %>% layout(**

**title = 'COVID Data Deaths',**

**geo = g**

**)**

**fig**

**cases.bar <- ggplot(i, aes(x=state, y=positive)) +**

**geom\_bar(stat="identity", fill="steelblue")+**

**geom\_text(aes(label=positive), nudge\_y = 1500, size=2.5)**

**cases.bar + coord\_flip()**

**summary(i)**

**summary(lm(i$recovered~ i$positive))**

**Rates**

**month.list <- rbind(March,April,May,June,July,August,September,October,November)**

**death.rate <- (month.list$death/month.list$positive)**

**recovery.rate <- (month.list$recovered/month.list$positive)**

**icu.rate <- (month.list$inIcuCurrently/month.list$positive)**

**total.tests <- (month.list$totalTestResults)**

**i$hover <- with(i, paste(state, '<br>', "Deaths:", death, '<br>', "Death Rate:", death.rate, '<br>', "Hospitalized:", hospitalizedCurrently, '<br>',**

**"Positive:", positive, '<br>', "Recovery Rate:", recovery.rate, '<br>', "Current ICU Rate:", icu.rate, '<br>',**

**"Total Tests Performed:", total.tests))**

**###########12/10/2020#############**

**#######Pulling In Data and Setting Objects(Sim)**

**usdata <- read.csv("https://covidtracking.com/data/download/all-states-history.csv", header=T)**

**######Cleaning Data(Replace)**

**usdata[is.na(usdata)] = 0**

**usdata[usdata=="NA"] <- 0**

**usdata[usdata==" "] <- 0**

**library(plotly)**

**March <- subset(usdata, date=="2020-03-31")**

**April <- subset(usdata, date=="2020-04-30")**

**May <- subset(usdata, date=="2020-05-31")**

**June <- subset(usdata, date=="2020-06-30")**

**July <- subset(usdata, date=="2020-07-31")**

**August <- subset(usdata, date=="2020-08-30")**

**September <- subset(usdata, date=="2020-09-30")**

**October <- subset(usdata, date=="2020-10-31")**

**November <- subset(usdata, date=="2020-11-30")**

**i<-(March)**

**########Rates(Jon)**

**death.rate <- percent((i$death/i$positive))**

**recovery.rate <- percent((i$recovered/i$positive))**

**icu.rate <- percent((i$inIcuCurrently/i$positive))**

**total.tests <- (i$totalTestResults)**

**#######Map(Jean)**

**i$hover <- with(i, paste(state, '<br>', "Deaths:", death, '<br>', "Death Rate:", death.rate, '<br>', "Hospitalized:", hospitalizedCurrently, '<br>',**

**"Positive:", positive, '<br>', "Recovery Rate:", recovery.rate, '<br>', "Current ICU Rate:", icu.rate, '<br>',**

**"Total Tests Performed:", total.tests))**

**# give state boundaries a white border**

**l <- list(color = toRGB("white"), width = 2)**

**# specify some map projection/options**

**g <- list(**

**scope = 'usa',**

**projection = list(type = 'albers usa'),**

**showlakes = TRUE,**

**lakecolor = toRGB('white')**

**)**

**fig <- plot\_geo(i, locationmode = 'USA-states')**

**fig <- fig %>% add\_trace(**

**z = ~death, text = ~hover, hoverinfo = 'text',locations = ~state,**

**color = ~death, colors = 'Reds'**

**)**

**fig <- fig %>% colorbar(title = "Deaths")**

**fig <- fig %>% layout(**

**title = 'COVID Data Deaths',**

**geo = g**

**)**

**fig**

**######Bar Chart**

**cases.bar <- ggplot(i, aes(x=state, y=positive)) +**

**geom\_bar(stat="identity", fill="red")+**

**geom\_text(aes(label=positive), nudge\_y = 1500, size=2.5)**

**cases.bar + coord\_flip()**

**#########Statistics(Tom) run Recovery Rate & ICU**

**lm(usdata$positive~ usdata$state + usdata$death)**

**#Death Rate**

**ggplot(usdata, aes(x=state, y=death/recovered)) +**

**geom\_point(shape=1)+**

**geom\_smooth(method = lm)**

**#Recovery Rate**

**ggplot(usdata, aes(x=state, y=recovered/positive)) +**

**geom\_point(shape=1)+**

**geom\_smooth(method = lm)**

**# Current ICU Rate**

**ggplot(usdata, aes(x=date, y=inIcuCurrently/positive)) +**

**geom\_point(shape=1)+**

**geom\_smooth(method = lm)**

**# Recovery Rate of Total Tests**

**ggplot(usdata, aes(x=state, y=recovered/totalTestResults)) +**

**geom\_point(shape=1)+**

**geom\_smooth(method = lm)**

**# Positive Cases to Total Tests**

**ggplot(usdata, aes(x=state, y=positive/totalTestResults)) +**

**geom\_point(shape=1)+**

**geom\_smooth(method = lm)**