COVID19 Final Project

Jared, Matt, Megan

Spring 2022

knitr::opts\_chunk$set(echo = TRUE,  
 fig.width=10,   
 fig.height=6,   
 fig.align = "center")  
  
 #Load the needed package(s) here  
pacman::p\_load(tidyverse, COVID19,gridExtra, usmap, rpart, rpart.plot, caret, caTools)  
cov19 <- covid19(country = c("US"), level = 2, start = "2020-12-01", end = "2021-06-01")

## We have invested a lot of time and effort in creating COVID-19 Data Hub, please cite the following when using it:  
##   
## Guidotti, E., Ardia, D., (2020), "COVID-19 Data Hub", Journal of Open  
## Source Software 5(51):2376, doi: 10.21105/joss.02376.  
##   
## A BibTeX entry for LaTeX users is  
##   
## @Article{,  
## title = {COVID-19 Data Hub},  
## year = {2020},  
## doi = {10.21105/joss.02376},  
## author = {Emanuele Guidotti and David Ardia},  
## journal = {Journal of Open Source Software},  
## volume = {5},  
## number = {51},  
## pages = {2376},  
## }  
##   
## To hide this message use 'verbose = FALSE'.

#Renamed the administrative\_area\_level\_2 to state just for convenience and added three percentage variables  
cov19 <- rename(cov19, state = administrative\_area\_level\_2) %>%  
 mutate(deathRate = (deaths/confirmed)\*100,  
 infectionRate = (confirmed/population)\*100,  
 vaccineRate = (vaccines/population)\*100, na.rm = TRUE)  
  
# Change the default theme below:  
theme\_set(theme\_bw())

The next section is data cleaning

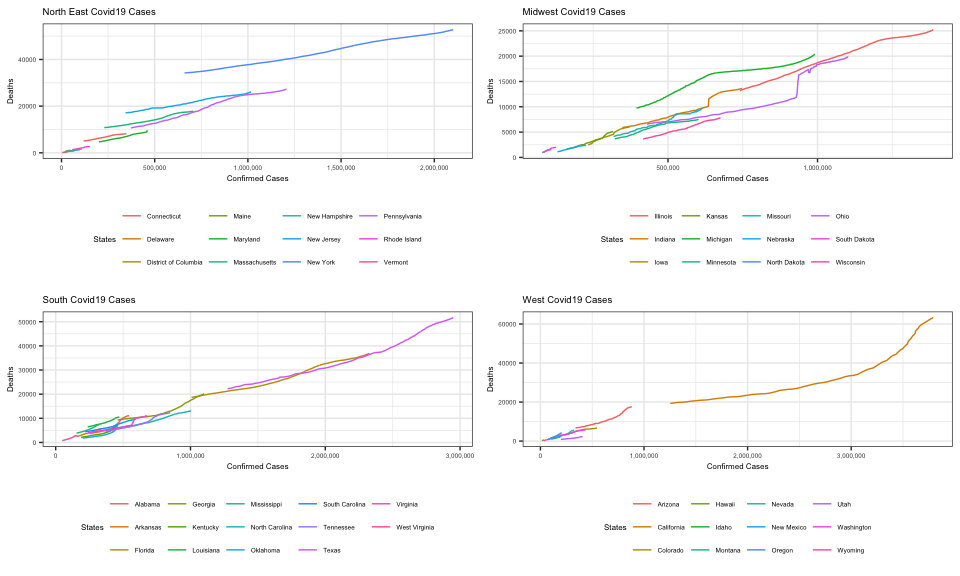
# total count of cov19 at last day of data  
cov19LastDay <- cov19 %>%  
 filter(date == "2021-06-01")  
# Cov19 by region  
cov19South <- cov19 %>%   
 filter(state %in%   
 c('Alabama', 'Florida', 'Georgia', 'Mississippi',   
 'South Carolina','Arkansas', 'Louisiana', 'Oklahoma',   
 'Texas', 'Kentucky', 'North Carolina', 'Tennessee', 'Virginia', 'West Virginia'))  
cov19NorthE <- cov19 %>%   
 filter(state %in%   
 c('Connecticut', 'Maine', 'Massachusetts', 'New Hampshire',   
 'Rhode Island', 'Vermont', 'New Jersey', 'New York', 'Pennsylvania',  
 'Delaware', 'District of Columbia','Maryland'))  
cov19MidW <- cov19 %>%  
 filter(state %in%  
 c('Iowa', 'Kansas', 'Missouri', 'Nebraska', 'North Dakota',   
 'South Dakota', 'Illinois', 'Indiana', 'Michigan', 'Minnesota', 'Ohio', 'Wisconsin'))  
cov19West <- cov19 %>%  
 filter(state %in%  
 c('California', 'Hawaii', 'Nevada','Oregon', 'Washington','Arizona',   
 'Colorado', 'Idaho', 'Montana', 'New Mexico', 'Utah', 'Wyoming'))

The next section is graphing

# North East plot  
northEastPlot <- ggplot(cov19NorthE, aes(x = confirmed, y = deaths, color = state))+  
 geom\_line()+  
 labs(title = "North East Covid19 Cases",  
 x = "Confirmed Cases",  
 y = "Deaths",  
 color = "States")+  
 theme(text=element\_text(size=6),  
 legend.position="bottom")+  
 scale\_x\_continuous(labels = scales::comma)  
  
# South Plot  
southPlot <- ggplot(cov19South, aes(x = confirmed, y = deaths, color = state))+  
 geom\_line()+  
 labs(title = "South Covid19 Cases",  
 x = "Confirmed Cases",  
 y = "Deaths",  
 color = "States")+  
 theme(text=element\_text(size=6),  
 legend.position="bottom")+  
 scale\_x\_continuous(labels = scales::comma)  
# Midwest plot  
midWestPlot <- ggplot(cov19MidW, aes(x = confirmed, y = deaths, color = state))+  
 geom\_line()+  
 labs(title = "Midwest Covid19 Cases",  
 x = "Confirmed Cases",  
 y = "Deaths",  
 color = "States")+  
 theme(text=element\_text(size=6),  
 legend.position="bottom")+  
 scale\_x\_continuous(labels = scales::comma)  
# West plot  
westPlot <- ggplot(cov19West, aes(x = confirmed, y = deaths, color = state))+  
 geom\_line()+  
 labs(title = "West Covid19 Cases",  
 x = "Confirmed Cases",  
 y = "Deaths",  
 color = "States")+  
 theme(text=element\_text(size=6),  
 legend.position="bottom")+  
 scale\_x\_continuous(labels = scales::comma)

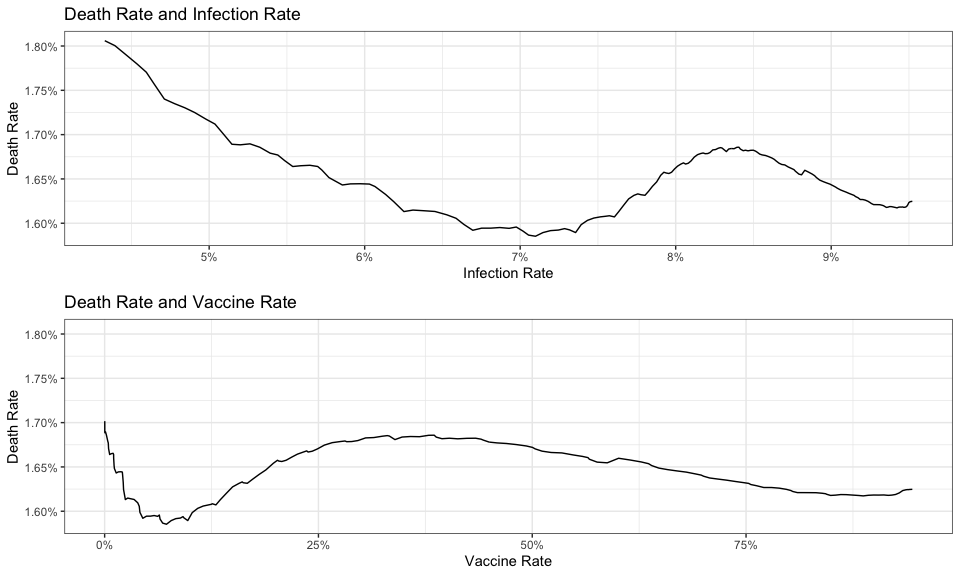
Grid arrange of the 4 plots above

grid.arrange(northEastPlot,midWestPlot, southPlot,westPlot, ncol = 2)



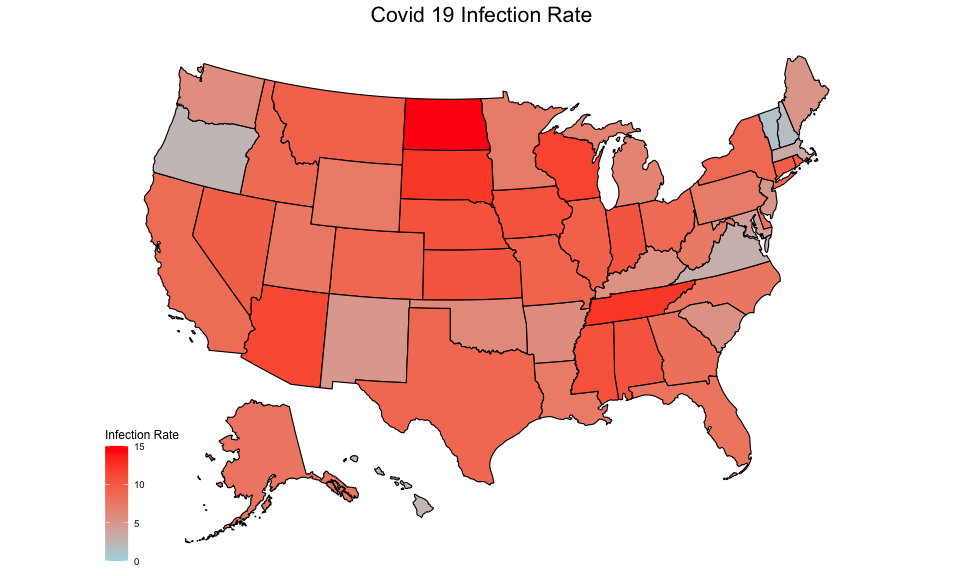
Plots for vaccine rate vs death rate and infection rate vs death rate with a grid arrange

# plot for death rate vs vaccine rate for each day  
cov19Average <- cov19 %>% group\_by(date) %>%  
 summarise(mean(infectionRate, na.rm = TRUE), mean(deathRate, na.rm = TRUE), mean(vaccineRate, na.rm = TRUE))  
  
deathVaccineRatesPlot <- ggplot(cov19Average, aes(x = cov19Average$`mean(vaccineRate, na.rm = TRUE)`, y = cov19Average$`mean(deathRate, na.rm = TRUE)`))+  
 geom\_line()+  
 labs(x = "Vaccine Rate",  
 y = "Death Rate",  
 title = "Death Rate and Vaccine Rate")+  
 scale\_y\_continuous(label = scales::percent\_format(scale = 1, accuracy = .01))+  
 scale\_x\_continuous(label = scales::percent\_format(scale = 1))  
  
deathInfectionRatesPlot <- ggplot(cov19Average, aes(x = cov19Average$`mean(infectionRate, na.rm = TRUE)`, y = cov19Average$`mean(deathRate, na.rm = TRUE)`))+  
 geom\_line()+  
 labs(x = "Infection Rate",  
 y = "Death Rate",  
 title = "Death Rate and Infection Rate")+  
 scale\_y\_continuous(label = scales::percent\_format(scale = 1, accuracy = .01))+  
 scale\_x\_continuous(label = scales::percent\_format(scale = 1, accuracy = 1))  
  
  
# Grid Arrange of two graphs  
grid.arrange(deathInfectionRatesPlot, deathVaccineRatesPlot)



Map of US with infection as the value

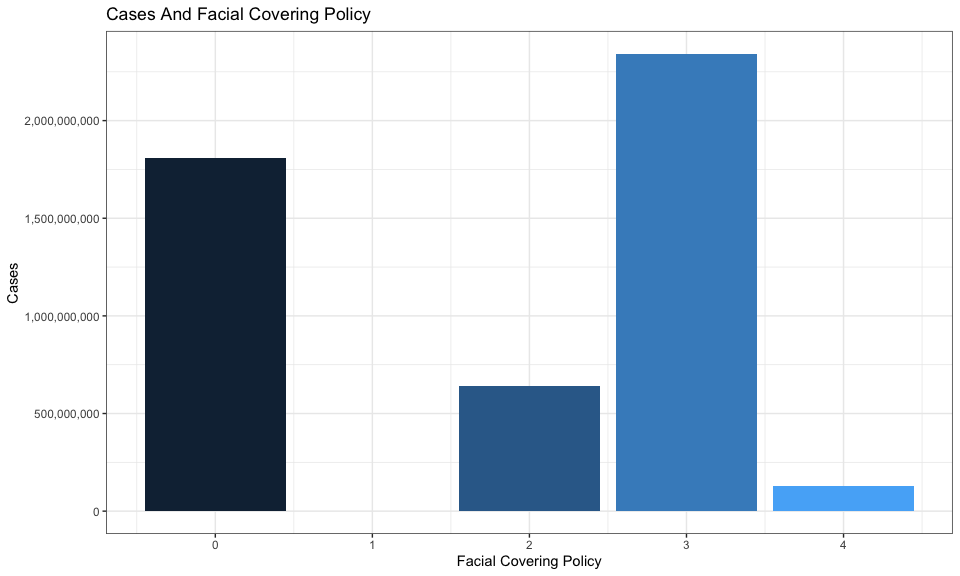
#filter out non states  
  
cov19StatesOnly <- cov19 %>% group\_by(state) %>% filter(state != "American Samoa", state != "Puerto Rico", state != "Northern Mariana Islands", state != "District of Columbia", state != "Guam", state != "Virgin Islands")  
  
## Map of covid cases  
  
plot\_usmap(data = cov19StatesOnly,  
 values = 'infectionRate',  
 regions = 'state',  
 color = "black")+  
 scale\_fill\_continuous(low = 'light blue', high = 'red', limits = c(0,15))+  
 labs(fill = "Infection Rate",  
 title = "Covid 19 Infection Rate")+  
 theme(plot.title = element\_text(hjust = .5, size = 16))



Plot on infection rate with face coverings and without face coverings

infectionBarChartData <- cov19 %>%   
 group\_by(facial\_coverings) %>%   
 filter(facial\_coverings >= -4, na.rm = TRUE)  
  
for(i in 1:nrow(infectionBarChartData)){  
 if(infectionBarChartData$facial\_coverings[i] <= 0){  
 infectionBarChartData$facial\_coverings[i] <- 0  
 }  
}  
  
infectionBarChartData <- infectionBarChartData %>%   
 summarise(sum(confirmed))  
  
infectionBarChart <- ggplot(infectionBarChartData,   
 aes(x = facial\_coverings, y = infectionBarChartData$`sum(confirmed)`))+  
 geom\_col(aes(fill = facial\_coverings))+  
 scale\_y\_continuous(labels = scales::comma)+  
 labs(x = "Facial Covering Policy",  
 y = "Cases",  
 title = "Cases And Facial Covering Policy")+  
 theme(legend.position = "none")+  
 scale\_x\_continuous(breaks = seq(-5,5,1))  
   
  
  
  
  
infectionBarChart

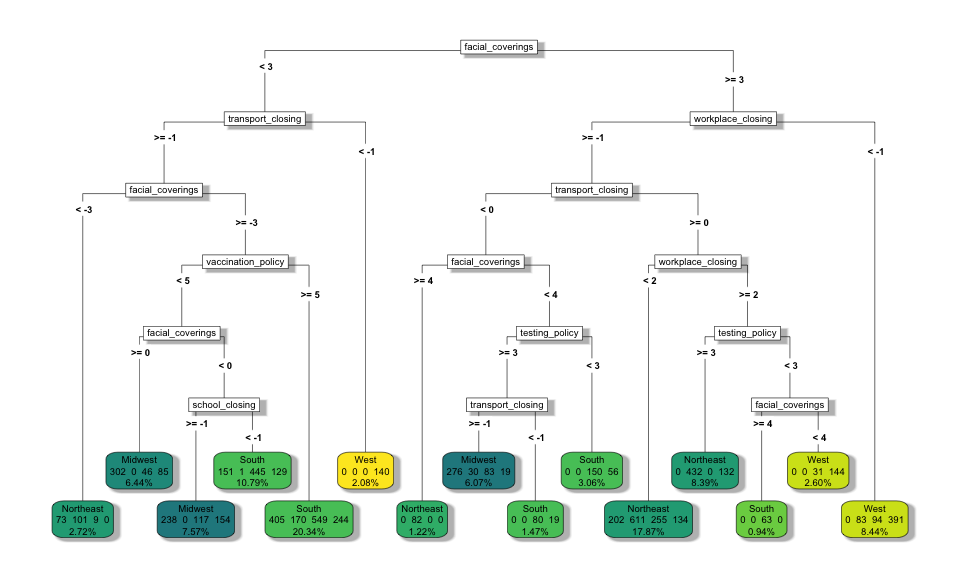
## Warning: Use of `infectionBarChartData$`sum(confirmed)`` is discouraged. Use  
## `sum(confirmed)` instead.



cov19StatesOnly <- cov19StatesOnly %>% mutate(region = NA)  
  
West <- c('California', 'Hawaii', 'Nevada','Oregon', 'Washington','Arizona', 'Colorado', 'Idaho', 'Montana', 'New Mexico', 'Utah', 'Wyoming')  
  
Midwest <- c('Iowa', 'Kansas', 'Missouri', 'Nebraska', 'North Dakota',   
 'South Dakota', 'Illinois', 'Indiana', 'Michigan', 'Minnesota', 'Ohio', 'Wisconsin')  
  
NorthEast <- c('Connecticut', 'Maine', 'Massachusetts', 'New Hampshire',   
 'Rhode Island', 'Vermont', 'New Jersey', 'New York', 'Pennsylvania',  
 'Delaware', 'District of Columbia','Maryland')  
  
South <- c('Alabama', 'Florida', 'Georgia', 'Mississippi',   
 'South Carolina','Arkansas', 'Louisiana', 'Oklahoma',   
 'Texas', 'Kentucky', 'North Carolina', 'Tennessee', 'Virginia', 'West Virginia')  
  
for(i in 1:nrow(cov19StatesOnly)){  
 for(j in 1:length(West)){  
 if(cov19StatesOnly$state[i] == West[j]){  
 cov19StatesOnly$region[i] <- "West"  
 }  
 }  
 for(j in 1:length(NorthEast)){  
 if(cov19StatesOnly$state[i] == NorthEast[j]){  
 cov19StatesOnly$region[i] <- "Northeast"  
 }  
 }  
 for(j in 1:length(South)){  
 if(cov19StatesOnly$state[i] == South[j]){  
 cov19StatesOnly$region[i] <- "South"  
 }  
 }  
 for(j in 1:length(Midwest)){  
 if(cov19StatesOnly$state[i] == Midwest[j]){  
 cov19StatesOnly$region[i] <- "Midwest"  
 }  
 }  
}  
  
cov19Tree <- cov19StatesOnly %>% select(testing\_policy, vaccination\_policy, facial\_coverings, workplace\_closing, school\_closing, transport\_closing, region)  
split <- sample.split(cov19Tree$region, SplitRatio = 0.75)  
trainingCov19 <- subset(cov19Tree, subset = split, select = -state)  
testingCov19 <- subset(cov19Tree, subset = !split, select = -state)  
  
testingCov19$region <- as.factor(testingCov19$region)  
  
cov19DT <- rpart(formula = region ~ .,  
 data = trainingCov19,  
 method = "class",  
 parms = list(split = "information"),  
 minsplit = 0,   
 minbucket = 0,  
 cp = -1)  
  
  
  
cov19DTPruned <- prune(cov19DT,  
 cp = .0095)  
  
cov19DTPruned$cptable %>%   
 data.frame() %>%   
 filter(xerror < min(xerror) + 0.006795436)

## CP nsplit rel.error xerror xstd  
## 10 0.0095 14 0.5666112 0.5811823 0.008411887

rpart.plot(cov19DTPruned,   
 digits = 4,  
 fallen.leaves = TRUE,  
 type = 5,   
 extra = 101,  
 box.palette = 'BlGnYl',  
 shadow.col = 'gray',  
 legend.x = NA)



cov19\_pred <- predict(object = cov19DTPruned,  
 newdata = testingCov19,  
 type = "class")  
  
cm\_cov19 <-   
 confusionMatrix(data = cov19\_pred,   
 reference = testingCov19$region,  
 positive = c("Midwest", "Northeast","West", "South"),  
 dnn = c('predicted', 'actual'))   
  
cm\_cov19

## Confusion Matrix and Statistics  
##   
## actual  
## predicted Midwest Northeast South West  
## Midwest 269 11 74 92  
## Northeast 87 405 79 98  
## South 193 62 451 132  
## West 0 25 36 227  
##   
## Overall Statistics  
##   
## Accuracy : 0.6033   
## 95% CI : (0.5827, 0.6236)  
## No Information Rate : 0.2856   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.4682   
##   
## Mcnemar's Test P-Value : < 2.2e-16   
##   
## Statistics by Class:  
##   
## Class: Midwest Class: Northeast Class: South Class: West  
## Sensitivity 0.4900 0.8052 0.7047 0.4135  
## Specificity 0.8954 0.8481 0.7583 0.9639  
## Pos Pred Value 0.6031 0.6054 0.5382 0.7882  
## Neg Pred Value 0.8440 0.9377 0.8653 0.8351  
## Prevalence 0.2450 0.2245 0.2856 0.2450  
## Detection Rate 0.1200 0.1807 0.2012 0.1013  
## Detection Prevalence 0.1990 0.2985 0.3739 0.1285  
## Balanced Accuracy 0.6927 0.8266 0.7315 0.6887