# 

# 

# 

# **The Effect of COVID on the Housing Market**

*Did COVID impact housing prices or days on market?*

IST 687 - Final Project

Adrian Valles

Suihin Wong

# Business Questions

The business question we set out to determine was how did COVID affect the housing market. In particular, we wanted to determine how COVID affected the housing prices and did it affect how fast houses were selling.

The goal of this project was to answer the following questions to help real estate investors determine which areas are worth investing in or which areas will have the when compared to before or during COVID.

1. How did COVID affect real estate prices?
2. Did real estate prices increase, decrease or remain the same before, during, and after COVID?
3. Did COVID have any effect on how long houses stayed on the market? Did COVID have an impact on buyers’ willingness to buy?

# Data Acquisition

The data was acquired from Realtor.com.

Realtor.com is a trusted website that is used by real estate professionals across the united states. Realtor.com provides a database of historical data as far back as 2016. The database provides a database with data by month, year, county, and state.

**Source:** [**https://www.realtor.com/research/data/**](https://www.realtor.com/research/data/)

# Data Cleaning

To clean the data the first part we addressed was the time periods. In the dataset we hat real estate data from 2016. Since we were only interested in comparing real estate date before, during, and after COVID we created a time frame for our dataset.

The time frame we created includes:

**Before COVID:** January 2019 to February 2020

**During COVID:** March 2020 to January 2021

**After COVID:** January 2021 to August 2021

The second step to cleaning our data included our treatment of N/A and removing other unnecessary information.

We found that our columns with N/A were in columns that were not helpful to answer our business questions. These columns included a percentage change calculator of the listings in a given county for a given state. As a result, we decided to remove all columns that were calculating a percentage change and therefore it eliminated any N/As.

#check NAs

sum(is.na(newdata))

summary(newdata)

**#return NA and some columns. Pick the columns that we want to work for this dataset**

**b=c("price\_increased\_count\_yy","price\_increased\_count\_mm","median\_listing\_price\_per\_square\_foot\_mm", "median\_listing\_price\_per\_square\_foot\_yy", "median\_square\_feet\_yy", "median\_square\_feet\_mm","pending\_listing\_count\_mm","pending\_listing\_count\_yy",'price\_reduced\_count\_mm','price\_reduced\_count\_yy')**

**newdata <- newprojectdata[, !(colnames(newprojectdata) %in% b), drop = FALSE] # picking the columns we want.**

**beforeCovid <- newprojectdata[which(newprojectdata$month\_date\_yyyymm < 202003, newprojectdata$month\_date\_yyyymm > 201900),]**

**duringCovid <- newprojectdata[which(newprojectdata$month\_date\_yyyymm < 202102, newprojectdata$month\_date\_yyyymm > 202002),]**

**afterCovid <- newprojectdata[which(newprojectdata$month\_date\_yyyymm > 202101),]**

# Analysis

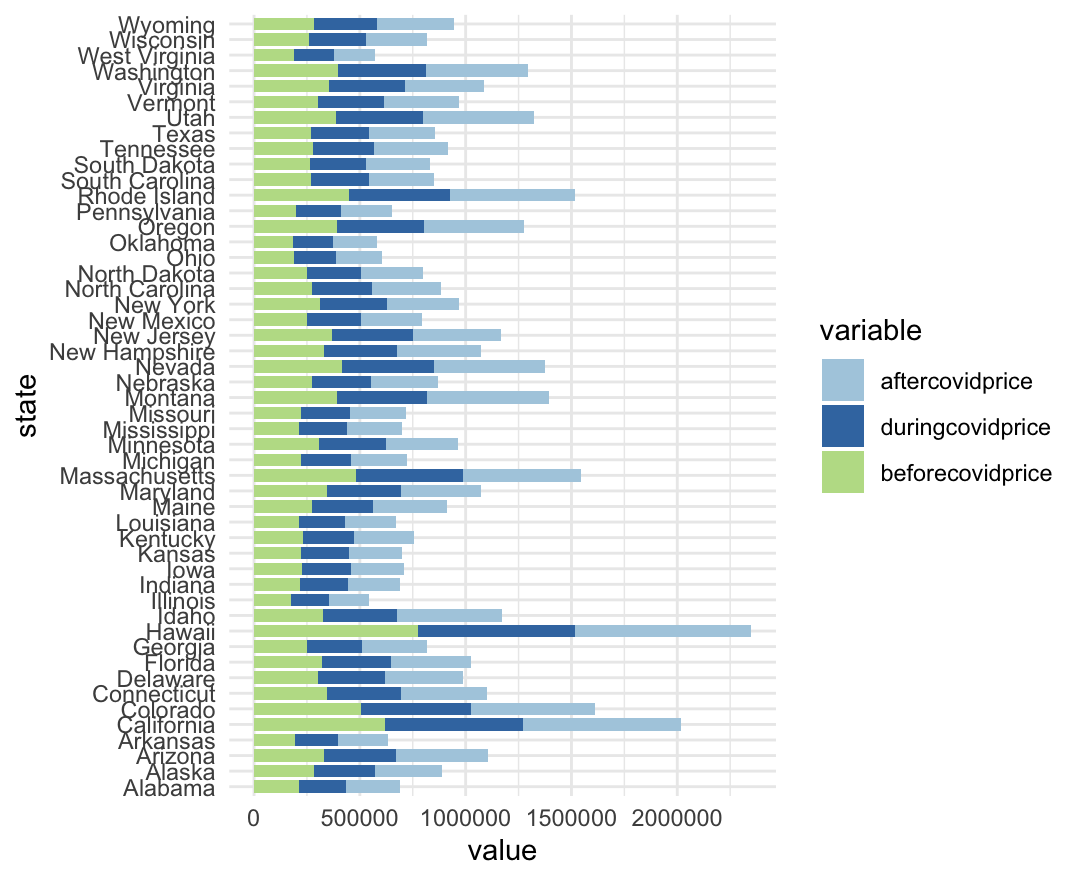
To gauge the changes in the average median prices across our different time periods (before, during, and after) we created a bar chart.

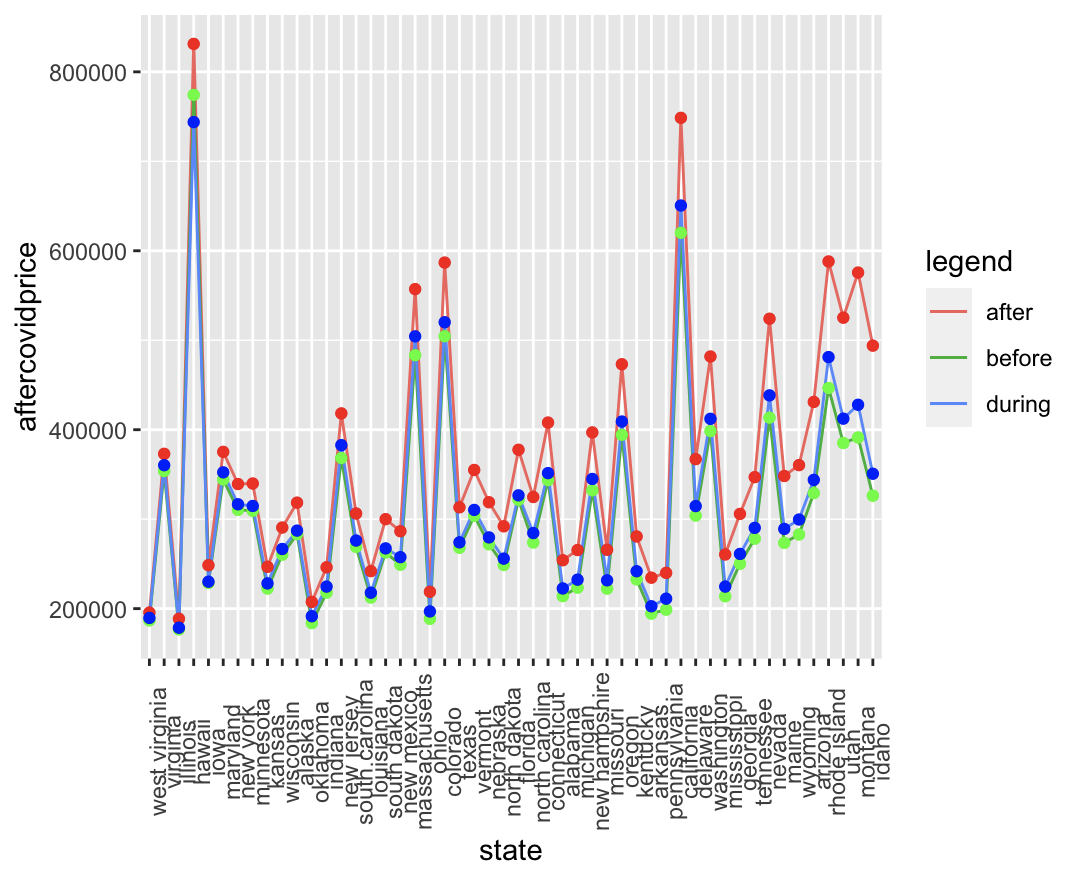
The bar chart shows the average median price for each state at each time period.

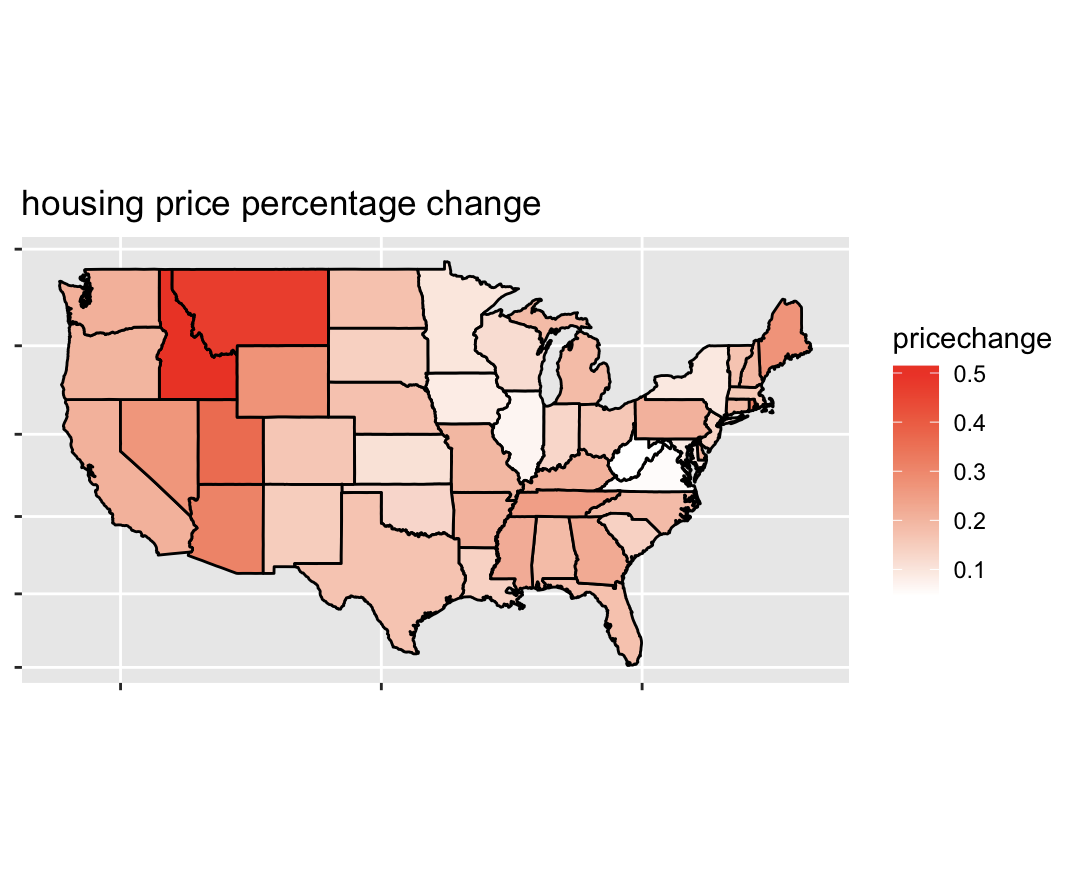
Based on our analysis we concluded that the average median price for each state continues to increase during COVID and after COVID. This suggests that despite what some people might believe housing prices in highly populated areas did not have a negative effect on housing prices in each state.

The bar chart does show us that the increase from during COVID to after COVID is typically exponentially higher than the increase from before COVID to after COVID. Our analysis would suggest that COVID did slow down the growth of housing prices but was not sufficient to reverse the trend.

Below are two different visualizations we used to show the changes in housing prices across the different time periods for each state.

****

****

****

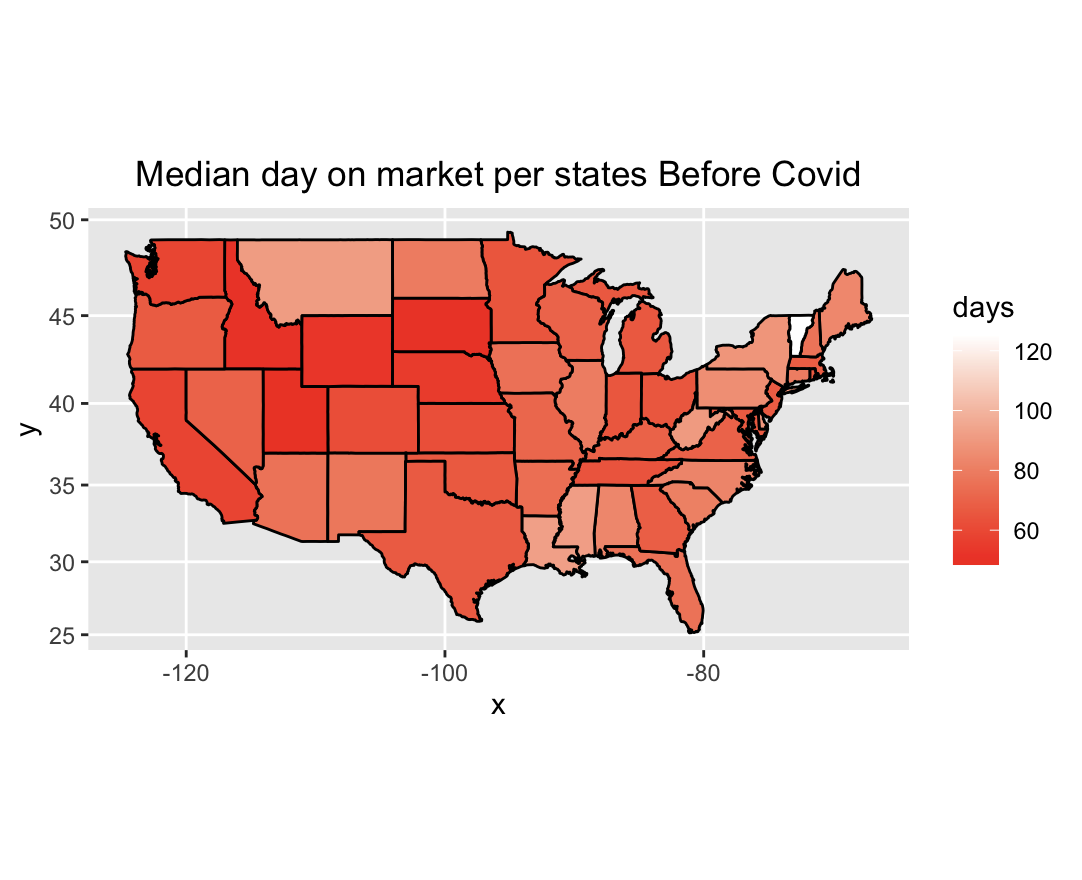
****

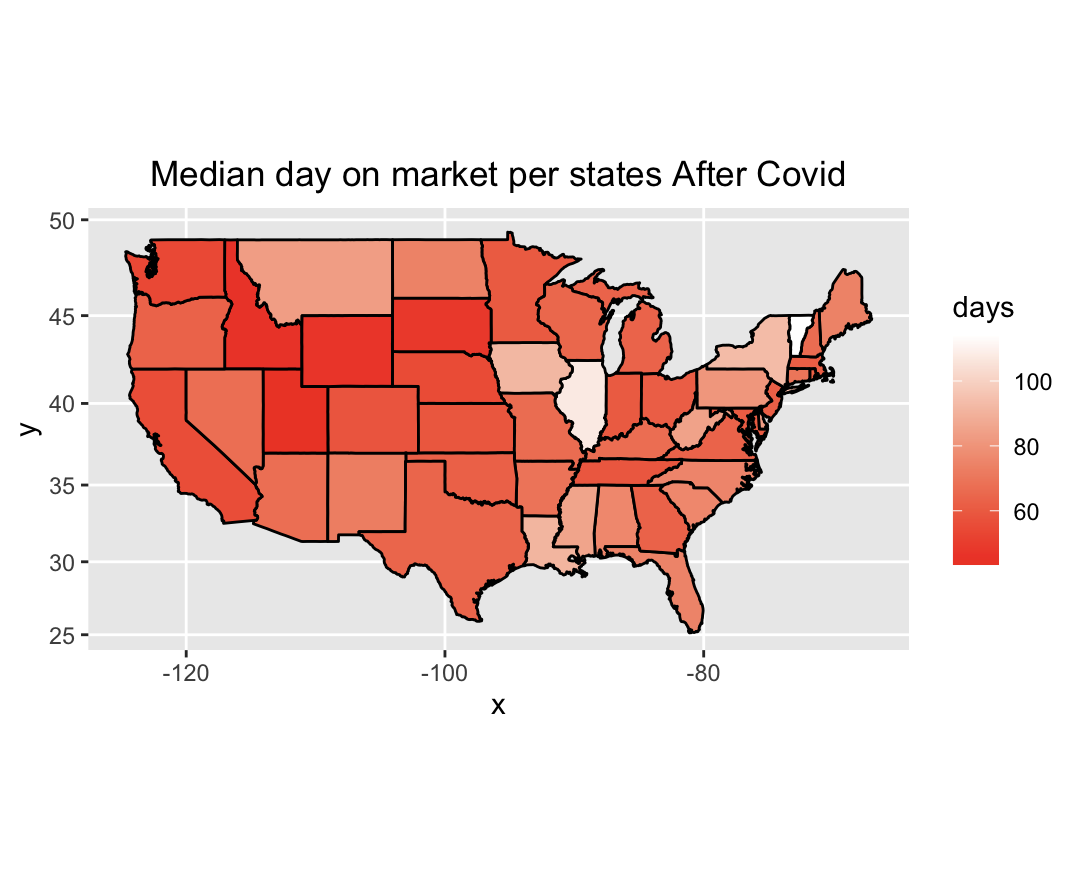
We also analyzed the median days on market to determine if there was any change in the median days on market during, before, and after COVID.

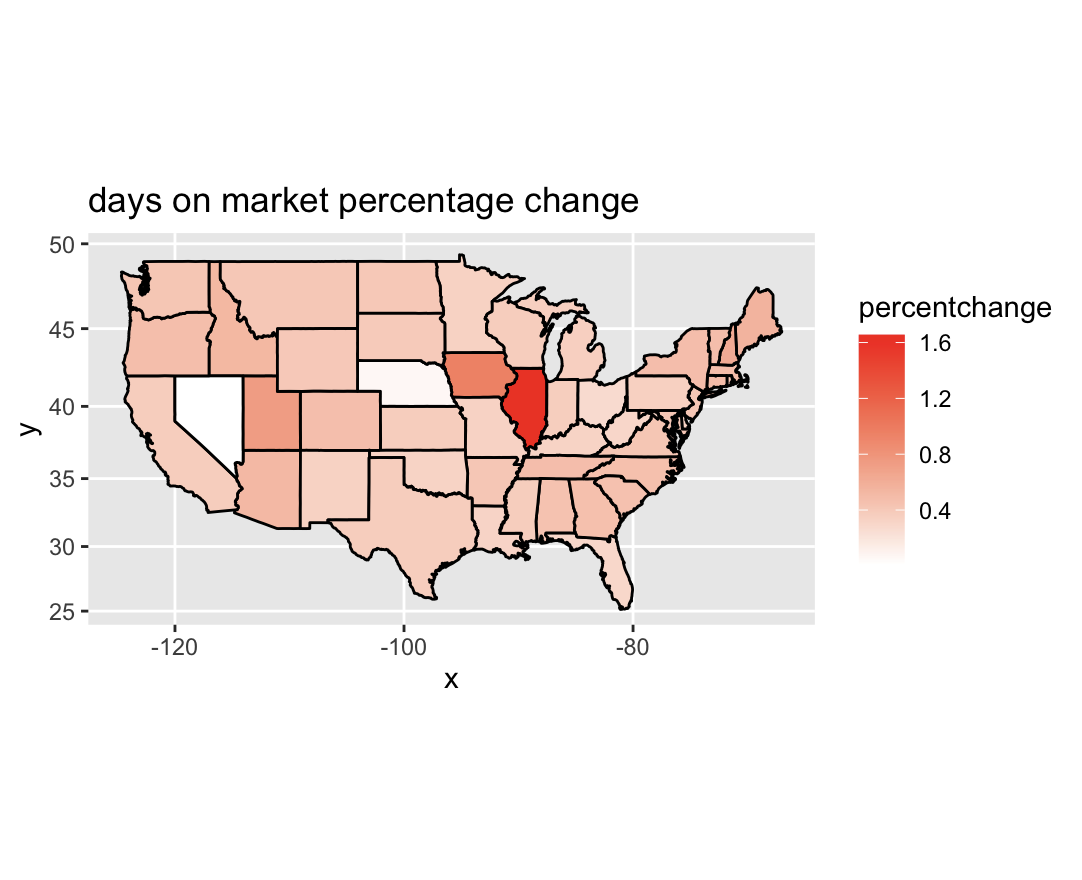
The data shows that days on market changed significantly before and after COVID. The impact was the opposite of the media housing price. With days on market decreasing substantially after COVID. With days on market decreasing after COVID this is a sign that buyers are eager to buy and that housing market is experiencing more, and faster activity than before COVID.

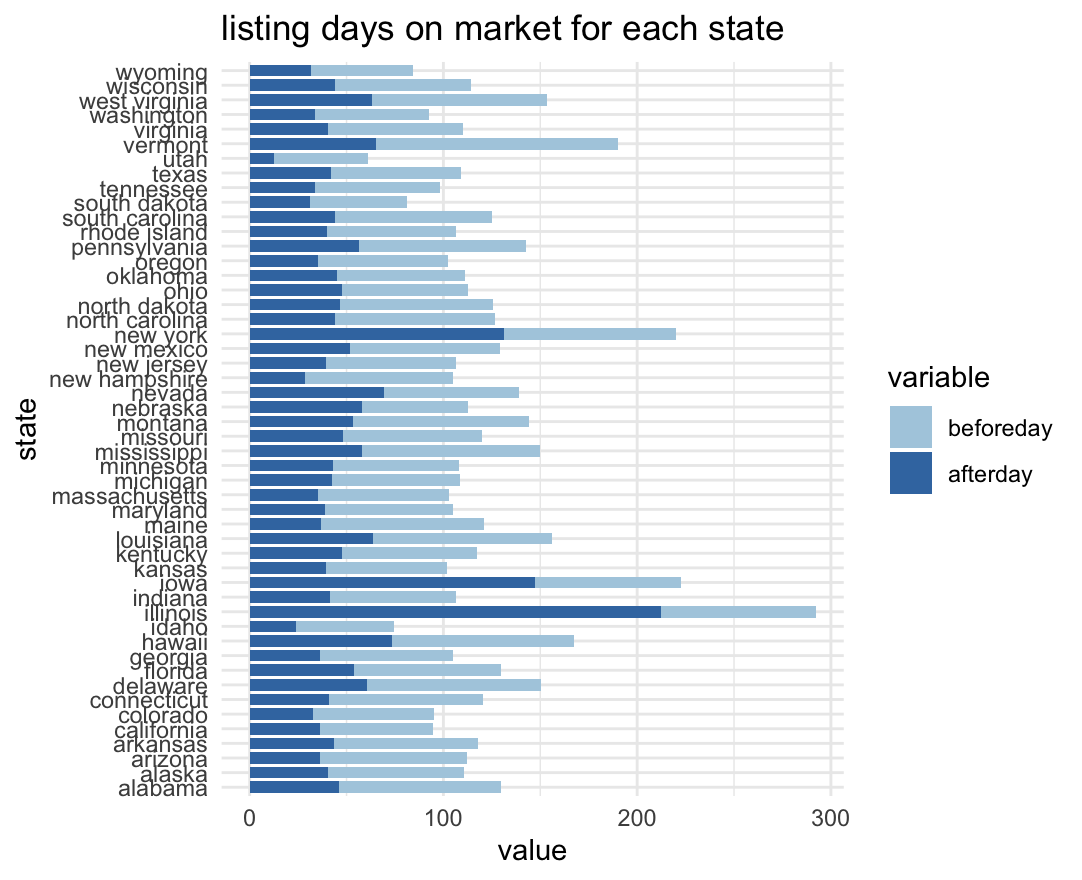
The bar chart shows that in most states the media days on market after COVID is approximately half of what it was before COVID.

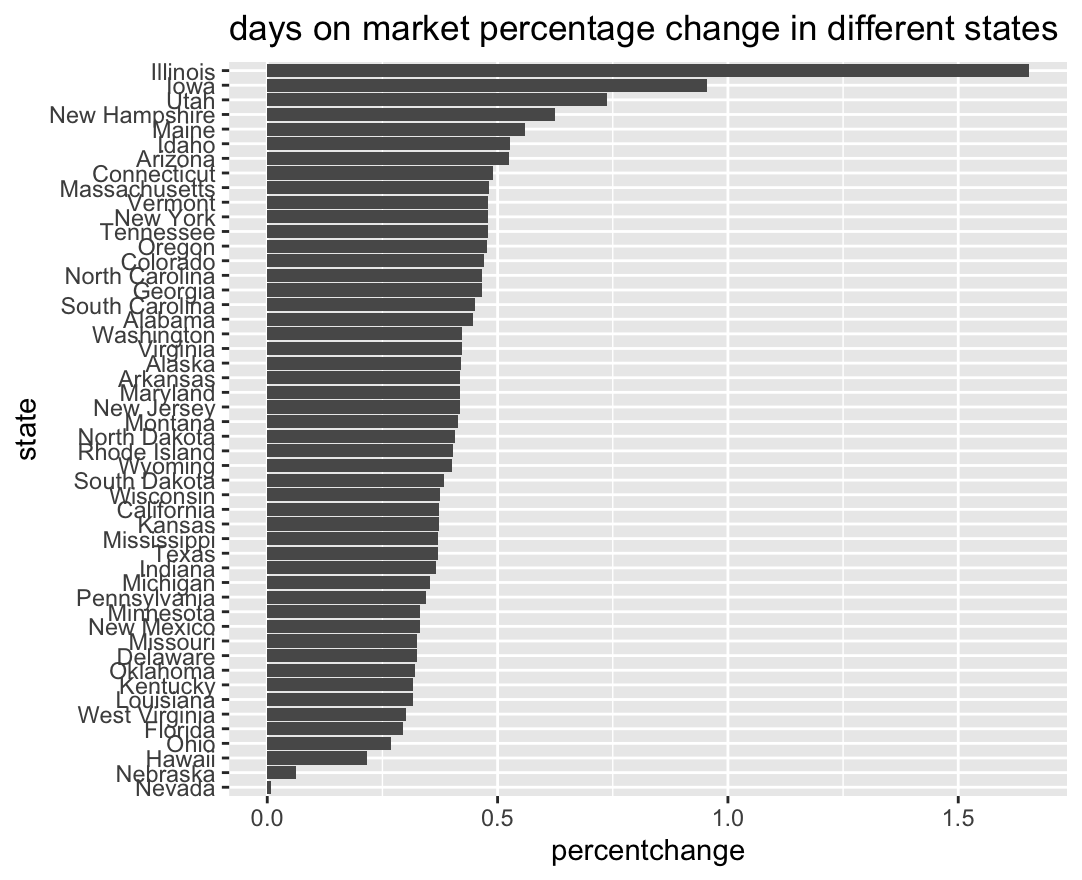
The percentage change shows the reduction in days on market from before to after COVID. The average percentage change in the days on market was -43.2% from before to after COVID.

****

****

****

****

****

# R code for Visualizations

#create bar plot price change for different state in different time period

ggplot(data=t3,aes(y=state,x=value,fill=variable)) + geom\_bar(width = 0.8,stat='identity') + theme(axis.text.x= element\_text(angle=90)) + scale\_fill\_brewer(palette="Paired") + theme\_minimal()

#create line plot for median price for before,during and after covid

p1 <- ggplot(data=t2,aes(x=state)) +geom\_line(aes(y=aftercovidprice,group=1,color='after'))

p1 <- p1 + geom\_line(aes(y=beforecovidprice,group=1,color='before')) +geom\_line(aes(y=duringcovidprice,group=1,color='during'))

p1 <- p1 + geom\_point(aes(y=aftercovidprice),color='red') +geom\_point(aes(y=beforecovidprice),color='green') +geom\_point(aes(y=duringcovidprice),color='blue')

p1 <- p1 + theme(axis.text.x= element\_text(angle=90))

p1 + labs(color='legend')

#create the data.frame to plot

gdd$stateName <- state.name[match(gdd$state,state.abb)]

gdd <- newprojectdata[c('month\_date\_yyyymm','median\_days\_on\_market','state','county')]

gdd$state <- gsub(" ","",gdd$state, fixed = TRUE)

beforeCovidd <- gdd[which(gdd$month\_date\_yyyymm < 202003, gdd$month\_date\_yyyymm > 201900),]

duringCovidd <- gdd[which(gdd$month\_date\_yyyymm < 202102, gdd$month\_date\_yyyymm > 202002),]

afterCovidd <- gdd[which(gdd$month\_date\_yyyymm > 202101),]

# before covid days on market

days <- tapply(beforeCovidd$median\_days\_on\_market, beforeCovidd$stateName,mean)

state<-row.names(days)

daysonmarket <- data.frame(state,days)

daysonmarket$state <- tolower(daysonmarket$state)

beforecoviddplot <- ggplot(daysonmarket, aes(map\_id = state)) +geom\_map(map=us,aes(fill=days),color='black')

beforecoviddplot <- beforecoviddplot + expand\_limits(x=us$long,y=us$lat) + coord\_map() + scale\_fill\_gradient(low='red',high='white')

beforecoviddplot <- beforecoviddplot + ggtitle('Median day on market per states Before Covid')

beforecoviddplot <- beforecoviddplot + theme(plot.title = element\_text(hjust = 0.5))

beforecoviddplot

# order by % of price change between before and after covide

t4$state <- factor(t4$state, levels = t4$state[order(t4$pricechange)])

p1 <- ggplot(data=t4,aes(x=state)) +geom\_line(aes(y=aftercovidprice,group=1,color='after'))

p1 <- p1 + geom\_line(aes(y=beforecovidprice,group=1,color='before')) +geom\_line(aes(y=duringcovidprice,group=1,color='during'))

p1 <- p1 + geom\_point(aes(y=aftercovidprice),color='red') +geom\_point(aes(y=beforecovidprice),color='green') +geom\_point(aes(y=duringcovidprice),color='blue')

p1 <- p1 + theme(axis.text.x= element\_text(angle=90))

p1 + labs(color='legend')

# price percentage change in different state

t4<-t2

t4$pricechangeafter <- (t4$aftercovidprice/t4$duringcovidprice)-1

t4$pricechangeduring <- (t4$duringcovidprice/t4$beforecovidprice)-1

t4$pricechange <- (t4$aftercovidprice/t4$beforecovidprice)-1

melt(t4[,c(1,5:7)],id='state')

t4$state <- tolower(t4$state)

c <- ggplot(t4[,c(1,7)], aes(map\_id = state)) + geom\_map(map=us,aes(fill=pricechange),color='black')

c <- c + expand\_limits(x=us$long,y=us$lat) + coord\_map() + scale\_fill\_gradient(low='white',high='red')

c + theme(axis.title.x = element\_blank(), axis.title.y = element\_blank())

#Price percentage change in different state before and after covid

pricechange <-t4[order(t4$pricechange),]

plot\_price <- pricechange[,c(1,7)]

plot\_price$state <- factor(plot\_price$state, levels = plot\_price$state[order(plot\_price$pricechange)])

ggplot(data = plot\_price, aes(x=pricechange,y=state)) + geom\_bar(stat="identity")

#listing days on market for each state

a1 <- melt(day\_table[1:3], id='state')

str(day\_table)

ggplot(data=a1,aes(y=state,x=value,fill=variable)) + geom\_bar(width = 0.8,stat='identity') + theme(axis.text.x= element\_text(angle=90)) + scale\_fill\_brewer(palette="Paired") + theme\_minimal() + ggtitle('listing days on market for each state')

# 

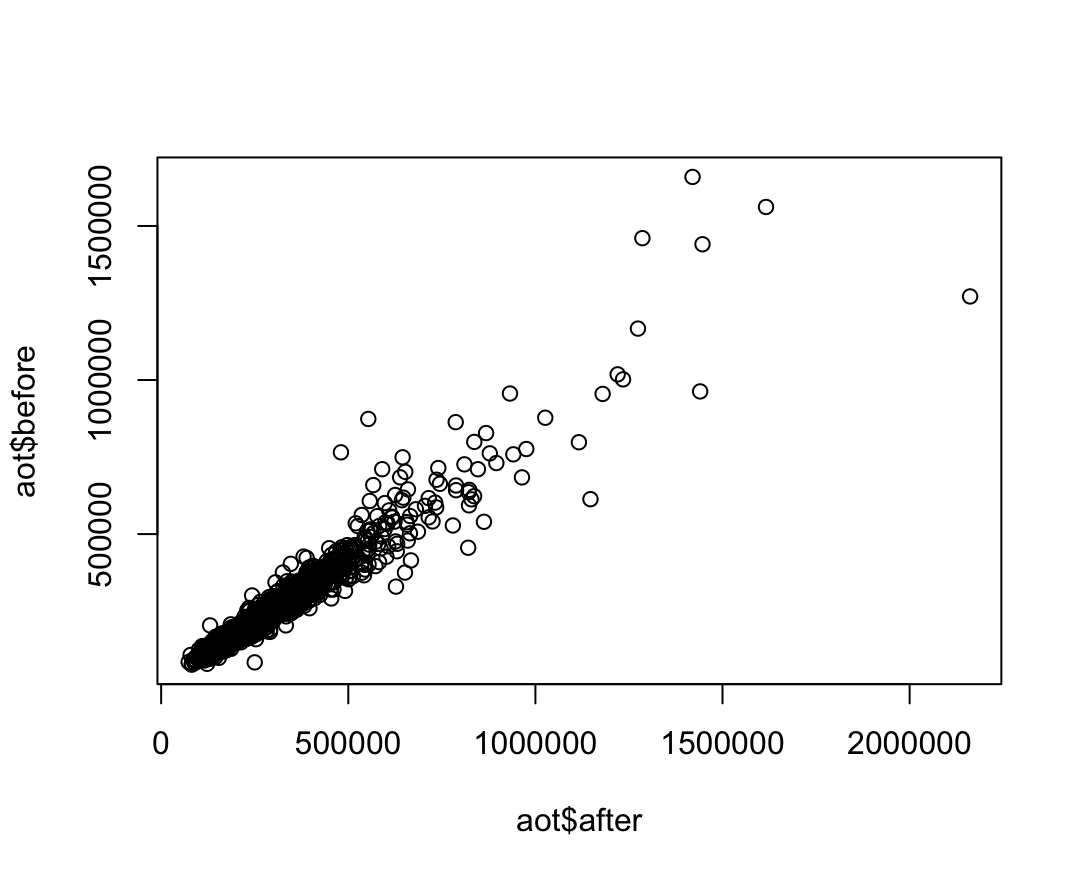
# Modeling

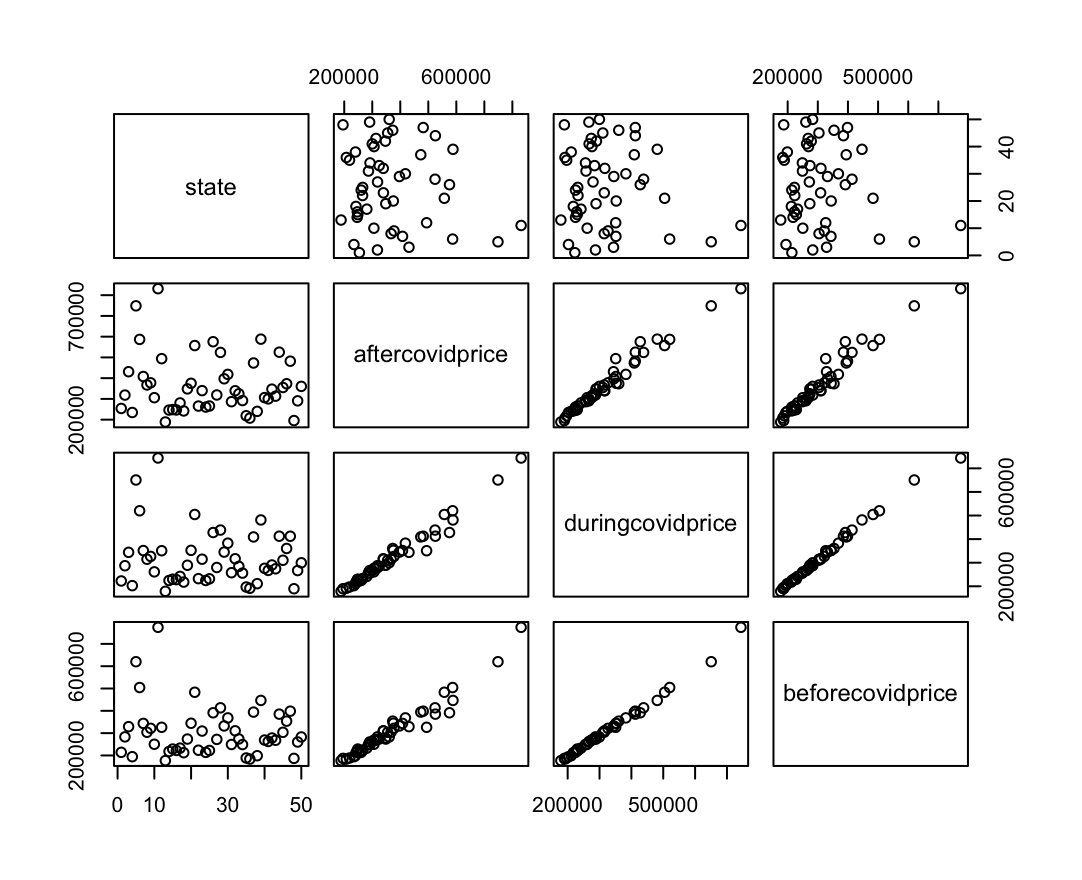
To create our model we used a linear model after plotting our variables against each other to determine the correlation between them. What we found is that the prices before, after, and during COVID were positively correlated and had a linear relationship. As a result, we decided to test run linear regression models on our data.

The tables below show the linear relationship and positive correlation between the price before, after, and during.

**​​#create plot to see the correlation between before and after covid for price**

**plot(aot$after,aot$before)**





To determine if the listing prices before and after COVID were significantly different we used a T-test.

The results of the t-test show that we can reject the null hypothesis that the media listing price are the same.

This tells us that the listing price before and after COVID are statistically different and allow us to conclude tha there is something affecting these prices that is not chance.

To determine if the days on market before and after COVID were statistically different we also used a t-test. The t-test result shows that days on market before COVID is statistically significant higher than days on market after COVID

To create a predictive model of the listing price we created a KSVM liner model. After attempting multiple linear regression models including the linear model, SVM, and KSVM models we determine that the KSVM model best fit our data.

The reason that the KSVM model was the mode that generated the smallest squared mean error. The SVM model was the least accurate and the linear model was a close second with a squared error that is only less than $5k greater than the KSVM model.

With the KSVM model it allows us to predict the listing price of a home using the month, year, county name, median days on market, median square feet, new listing count and the total listing count.

| Squared Mean Error Of Models | | |
| --- | --- | --- |
| Linear Model | SVM | KSVM |
| 36,745.11 | 162,470 | 32,068.57 |

**#t-test for housing price**

**Welch Two Sample t-test**

**data: a and b**

**t = 17.266, df = 9953, p-value < 2.2e-16**

**alternative hypothesis: true difference in means is not equal to 0**

**95 percent confidence interval:**

**44041.1 55321.7**

**sample estimates:**

**mean of x mean of y**

**340208.8 290527.4**

**#t-test for days on market**

**Welch Two Sample t-test**

**data: day\_table$afterday and day\_table$beforeday**

**t = -4.4051, df = 66.793, p-value = 0.00003917**

**alternative hypothesis: true difference in means is not equal to 0**

**95 percent confidence interval:**

**-31.67510 -11.92036**

**sample estimates:**

**mean of x mean of y**

**51.33136 73.12909**

**# LINEAR MODEL:**

**projectlm <- lm(median\_listing\_price~month\_date\_yyyymm + county\_name + median\_days\_on\_market+median\_square\_feet+new\_listing\_count+total\_listing\_count, data=trainData)**

**summary(projectlm)**

**lm\_pred1 <- predict(projectlm,testData)**

**# root mean squared error for linear model**

**lm\_table1<- data.frame(testData[,4],lm\_pred1)**

**colnames(lm\_table1) <- c("test","Pred")**

**sqrt(mean((lm\_table1$test-lm\_table1$Pred)^2))**

**# create Linear Model for house price**

**Call:**

**lm(formula = median\_listing\_price ~ month\_date\_yyyymm + county\_name +**

**median\_days\_on\_market + median\_square\_feet + new\_listing\_count +**

**total\_listing\_count, data = trainData)**

**Residuals:**

**Min 1Q Median 3Q Max**

**-777103 -12187 -587 11172 948459**

**Coefficients:**

**Estimate Std. Error t value Pr(>|t|)**

**(Intercept) -44790226.0295 634748.8297 -70.564 < 2e-16 \*\*\***

**month\_date\_yyyymm 222.1822 3.1413 70.730 < 2e-16 \*\*\***

**county\_nameada, id 264200.0866 10489.7168 25.187 < 2e-16 \*\*\***

**county\_nameadams, co 251663.3300 10188.1246 24.702 < 2e-16 \*\*\***

**county\_nameadams, il -11807.7638 10243.1656 -1.153 0.249029**

**county\_nameadams, pa 122385.0452 9639.3626 12.696 < 2e-16 \*\*\***

**county\_nameaiken, sc 71506.8037 9874.9960 7.241 4.61e-13 \*\*\***

**county\_namealachua, fl 121575.9446 9761.1569 12.455 < 2e-16 \*\*\***

**county\_namealamance, nc 120361.5811 9864.0935 12.202 < 2e-16 \*\*\***

**county\_namealameda, ca 696731.4369 10071.5983 69.178 < 2e-16 \*\*\***

**county\_namealbany, ny 147623.0872 9991.6829 14.775 < 2e-16 \*\*\***

**county\_namealbemarle, va 347674.9204 9808.0972 35.448 < 2e-16 \*\*\***

**county\_namealexandria city, va 494120.1856 9473.8052 52.156 < 2e-16 \*\*\***

**county\_nameallegan, mi 148602.4820 9851.1933 15.085 < 2e-16 \*\*\***

**county\_nameallegany, md -38149.8653 9739.6599 -3.917 9.00e-05 \*\*\***

**county\_nameallegheny, pa 152393.2179 9933.9874 15.341 < 2e-16 \*\*\***

**county\_nameallen, in 58792.8785 9982.0859 5.890 3.93e-09 \*\*\***

**county\_nameallen, oh -19593.7469 9746.8638 -2.010 0.044417 \***

**county\_nameanchorage, ak 183131.6962 10584.9599 17.301 < 2e-16 \*\*\***

**county\_nameanderson, sc 167158.8132 10529.1994 15.876 < 2e-16 \*\*\***

**county\_nameanderson, tn 81678.3824 9467.5600 8.627 < 2e-16 \*\*\***

**county\_nameandroscoggin, me 81313.1262 10245.3907 7.937 2.19e-15 \*\*\***

**county\_nameangelina, tx 87243.9051 10396.8138 8.391 < 2e-16 \*\*\***

**county\_nameanne arundel, md 310027.0390 9625.6083 32.209 < 2e-16 \*\*\***

**county\_nameanoka, mn 162268.2771 9774.4894 16.601 < 2e-16 \*\*\***

**county\_namearapahoe, co 274430.2964 10499.7574 26.137 < 2e-16 \*\*\***

**county\_namearlington, va 583575.3757 9990.0730 58.416 < 2e-16 \*\*\***

**county\_namearmstrong, pa 22395.5033 9789.4407 2.288 0.022164 \***

**county\_nameascension, la 120503.6800 9554.4277 12.612 < 2e-16 \*\*\***

**county\_nameashland, oh 6540.3431 9462.5875 0.691 0.489461**

**county\_nameashtabula, oh -1476.0253 10104.3068 -0.146 0.883861**

**county\_nameathens, oh 52526.6091 10606.3267 4.952 7.39e-07 \*\*\***

**county\_nameatlantic, nj 151544.5796 9897.4127 15.312 < 2e-16 \*\*\***

**county\_nameaugusta, va 137792.9939 10409.3069 13.237 < 2e-16 \*\*\***

**county\_nameautauga, al 55029.1740 9484.4132 5.802 6.65e-09 \*\*\***

**county\_namebaldwin, al 222282.5787 10387.4380 21.399 < 2e-16 \*\*\***

**county\_namebaltimore city, md 62627.4600 10015.6543 6.253 4.11e-10 \*\*\***

**county\_namebaltimore, md 155067.4346 10179.8386 15.233 < 2e-16 \*\*\***

**county\_namebannock, id 91617.6305 10007.8292 9.155 < 2e-16 \*\*\***

**county\_namebarnstable, ma 503397.1809 10112.4941 49.780 < 2e-16 \*\*\***

**county\_namebarrow, ga 107356.2182 9477.2297 11.328 < 2e-16 \*\*\***

**county\_namebarry, mi 75667.3235 9640.9873 7.849 4.42e-15 \*\*\***

**county\_namebartholomew, in 75738.1977 10295.5863 7.356 1.96e-13 \*\*\***

**county\_namebartow, ga 108328.5326 9649.5573 11.226 < 2e-16 \*\*\***

**county\_namebastrop, tx 138349.2590 10245.8454 13.503 < 2e-16 \*\*\***

**county\_namebaxter, ar 75521.6600 9655.8207 7.821 5.49e-15 \*\*\***

**county\_namebay, fl 189469.0593 9696.8904 19.539 < 2e-16 \*\*\***

**county\_namebay, mi -29838.0601 10111.0782 -2.951 0.003171 \*\***

**county\_namebeaufort, nc 110955.5457 10244.9203 10.830 < 2e-16 \*\*\***

**county\_namebeaufort, sc 293185.2365 9689.2145 30.259 < 2e-16 \*\*\***

**county\_namebeaver, pa 74163.3600 9926.8154 7.471 8.29e-14 \*\*\***

**county\_namebedford, va 167677.9632 10270.6050 16.326 < 2e-16 \*\*\***

**county\_namebelknap, nh 180080.9500 9545.6362 18.865 < 2e-16 \*\*\***

**county\_namebell, tx 71420.2685 10146.9483 7.039 2.01e-12 \*\*\***

**county\_namebelmont, oh -21865.6759 9740.3295 -2.245 0.024788 \***

**county\_namebenton, ar 141331.2322 9532.7593 14.826 < 2e-16 \*\*\***

**county\_namebenton, or 282791.6186 9859.6671 28.682 < 2e-16 \*\*\***

**county\_namebenton, wa 227218.0787 9406.0517 24.157 < 2e-16 \*\*\***

**county\_namebergen, nj 572759.7066 10620.1021 53.932 < 2e-16 \*\*\***

**county\_nameberkeley, sc 147757.6323 10788.8935 13.695 < 2e-16 \*\*\***

**county\_nameberkeley, wv 85875.5095 9973.7806 8.610 < 2e-16 \*\*\***

**county\_nameberks, pa 83104.8444 10418.3730 7.977 1.58e-15 \*\*\***

**county\_nameberkshire, ma 268315.0503 9650.4192 27.803 < 2e-16 \*\*\***

**county\_namebernalillo, nm 141620.4182 10065.1150 14.070 < 2e-16 \*\*\***

**county\_nameberrien, mi 179292.3361 9755.4705 18.379 < 2e-16 \*\*\***

**county\_namebexar, tx 161385.5685 10777.5363 14.974 < 2e-16 \*\*\***

**county\_namebibb, ga -711.9110 9754.3664 -0.073 0.941820**

**county\_nameblack hawk, ia 23520.6829 10113.3557 2.326 0.020044 \***

**county\_nameblair, pa 21612.1659 10031.2655 2.154 0.031215 \***

**county\_nameblount, al 37145.0314 10241.1556 3.627 0.000287 \*\*\***

**county\_nameblount, tn 153454.9775 9882.1853 15.528 < 2e-16 \*\*\***

**county\_nameblue earth, mn 100711.1477 9873.1491 10.201 < 2e-16 \*\*\***

**county\_namebonneville, id 120729.0859 9633.9930 12.532 < 2e-16 \*\*\***

**county\_nameboone, in 220192.2877 10463.2906 21.044 < 2e-16 \*\*\***

**county\_nameboone, ky 179610.0248 9995.6816 17.969 < 2e-16 \*\*\***

**county\_nameboone, mo 88561.7867 10251.9458 8.639 < 2e-16 \*\*\***

**county\_namebossier, la 72701.5285 9854.0278 7.378 1.67e-13 \*\*\***

**county\_nameboulder, co 515160.0879 9750.6820 52.833 < 2e-16 \*\*\***

**county\_namebowie, tx 20274.2934 9641.6267 2.103 0.035497 \***

**county\_nameboyd, ky -27418.0805 9846.4725 -2.785 0.005365 \*\***

**county\_namebradford, pa 17168.4289 9967.1857 1.722 0.084996 .**

**county\_namebradley, tn 84697.2660 9983.2919 8.484 < 2e-16 \*\*\***

**county\_namebrazoria, tx 120055.8857 10177.7260 11.796 < 2e-16 \*\*\***

**county\_namebrazos, tx 123747.0268 10426.8834 11.868 < 2e-16 \*\*\***

**county\_namebrevard, fl 171562.0148 11234.7059 15.271 < 2e-16 \*\*\***

**county\_namebristol, ma 263657.1560 9488.2661 27.788 < 2e-16 \*\*\***

**county\_namebristol, ri 391834.7600 10107.7031 38.766 < 2e-16 \*\*\***

**county\_namebronx, ny 304005.6484 10190.9017 29.831 < 2e-16 \*\*\***

**county\_namebroome, ny -8593.3866 9646.2032 -0.891 0.373017**

**county\_namebroomfield, co 371922.0740 10790.3370 34.468 < 2e-16 \*\*\***

**county\_namebroward, fl 319507.6648 13369.5255 23.898 < 2e-16 \*\*\***

**county\_namebrown, wi 138830.6534 9649.3614 14.388 < 2e-16 \*\*\***

**county\_namebrunswick, nc 202640.7869 9539.5654 21.242 < 2e-16 \*\*\***

**county\_namebuchanan, mo -33085.0252 10249.7067 -3.228 0.001249 \*\***

**county\_namebucks, pa 314055.8606 9454.4298 33.218 < 2e-16 \*\*\***

**county\_namebuffalo, ne 40061.2988 10149.5584 3.947 7.94e-05 \*\*\***

**county\_namebullitt, ky 114031.2437 9852.8938 11.573 < 2e-16 \*\*\***

**county\_namebulloch, ga 28295.4538 9850.1777 2.873 0.004076 \*\***

**county\_namebuncombe, nc 285092.9263 10423.7706 27.350 < 2e-16 \*\*\***

**county\_nameburke, nc 88771.4157 10402.6315 8.534 < 2e-16 \*\*\***

**county\_nameburleigh, nd 143058.2302 9975.5951 14.341 < 2e-16 \*\*\***

**county\_nameburlington, nj 124630.3294 10299.7958 12.100 < 2e-16 \*\*\***

**county\_namebutler, ks 39778.4085 9868.6760 4.031 5.58e-05 \*\*\***

**county\_namebutler, oh 113232.1609 9887.8079 11.452 < 2e-16 \*\*\***

**county\_namebutler, pa 242729.7595 10031.8750 24.196 < 2e-16 \*\*\***

**county\_namebutte, ca 232038.6491 9553.9464 24.287 < 2e-16 \*\*\***

**county\_namecabarrus, nc 125426.0869 10266.3352 12.217 < 2e-16 \*\*\***

**county\_namecabell, wv -15753.2567 9738.8830 -1.618 0.105773**

**county\_namecache, ut 159014.2388 9831.9979 16.173 < 2e-16 \*\*\***

**county\_namecaddo, la 19236.8654 9748.8398 1.973 0.048481 \***

**county\_namecalcasieu, la 70503.1398 9981.2204 7.064 1.68e-12 \*\*\***

**county\_namecaldwell, nc 72352.8838 9741.8060 7.427 1.16e-13 \*\*\***

**county\_namecalhoun, al -9333.5268 9457.1208 -0.987 0.323689**

**county\_namecalhoun, mi 2365.3759 10404.9302 0.227 0.820168**

**county\_namecalumet, wi 55898.6171 9375.8172 5.962 2.53e-09 \*\*\***

**county\_namecalvert, md 213077.1607 10256.6578 20.775 < 2e-16 \*\*\***

**county\_namecambria, pa -48837.1894 9865.2581 -4.950 7.47e-07 \*\*\***

**county\_namecamden, ga 98140.9726 9974.2028 9.839 < 2e-16 \*\*\***

**county\_namecamden, nj 69421.9859 9523.6156 7.289 3.23e-13 \*\*\***

**county\_namecameron, tx 108582.5087 10418.1944 10.422 < 2e-16 \*\*\***

**county\_namecampbell, ky 171553.9493 9982.7327 17.185 < 2e-16 \*\*\***

**county\_namecampbell, va 73367.8718 10239.7556 7.165 8.05e-13 \*\*\***

**county\_namecanadian, ok 74040.6473 9649.9707 7.673 1.76e-14 \*\*\***

**county\_namecanyon, id 179029.1238 9992.1205 17.917 < 2e-16 \*\*\***

**county\_namecape girardeau, mo 37298.4742 9738.3610 3.830 0.000129 \*\*\***

**county\_namecape may, nj 393042.0842 10195.2147 38.552 < 2e-16 \*\*\***

**county\_namecarbon, pa 41095.4229 9547.8495 4.304 1.68e-05 \*\*\***

**county\_namecarroll, ga 107629.3161 9663.9654 11.137 < 2e-16 \*\*\***

**county\_namecarroll, md 214363.4940 9986.4059 21.466 < 2e-16 \*\*\***

**county\_namecarson city, nv 246703.0559 9854.1776 25.035 < 2e-16 \*\*\***

**county\_namecarter, ok 10599.5989 9636.2028 1.100 0.271356**

**county\_namecarter, tn 39507.0367 9966.6570 3.964 7.40e-05 \*\*\***

**county\_namecarteret, nc 225173.7512 9644.8861 23.346 < 2e-16 \*\*\***

**county\_namecarver, mn 212686.8920 10008.3757 21.251 < 2e-16 \*\*\***

**county\_namecascade, mt 87185.8016 10585.4580 8.236 < 2e-16 \*\*\***

**county\_namecass, mi 74890.5180 10396.8335 7.203 6.09e-13 \*\*\***

**county\_namecass, mo 129356.3177 9551.5033 13.543 < 2e-16 \*\*\***

**county\_namecass, nd 96038.2345 9662.9422 9.939 < 2e-16 \*\*\***

**county\_namecatawba, nc 136940.0159 9558.9910 14.326 < 2e-16 \*\*\***

**county\_namecatoosa, ga 96345.0649 10119.9963 9.520 < 2e-16 \*\*\***

**county\_namecattaraugus, ny -4231.2683 9740.0684 -0.434 0.663989**

**county\_namecayuga, ny 30690.5866 10095.8985 3.040 0.002370 \*\***

**county\_namececil, md 146182.4536 9738.8188 15.010 < 2e-16 \*\*\***

**county\_namecentre, pa 131690.5257 9641.9977 13.658 < 2e-16 \*\*\***

**county\_namecerro gordo, ia 3304.6776 9967.9565 0.332 0.740248**

**county\_namechampaign, il 43115.7152 9644.6928 4.470 7.85e-06 \*\*\***

**county\_namecharles, md 202105.9272 10117.5305 19.976 < 2e-16 \*\*\***

**county\_namecharleston, sc 446092.0759 10019.4981 44.522 < 2e-16 \*\*\***

**county\_namecharlotte, fl 147487.2680 10054.6962 14.668 < 2e-16 \*\*\***

**county\_namecharlottesville city, va 371690.7251 9770.6360 38.042 < 2e-16 \*\*\***

**county\_namechatham, ga 154441.0115 10023.0625 15.409 < 2e-16 \*\*\***

**county\_namechatham, nc 361930.8321 9927.7475 36.456 < 2e-16 \*\*\***

**county\_namechautauqua, ny 13387.1979 10243.0679 1.307 0.191244**

**county\_namechaves, nm 23843.7512 9739.9129 2.448 0.014372 \***

**county\_namechelan, wa 307707.9498 9544.6662 32.239 < 2e-16 \*\*\***

**county\_namechemung, ny -29543.7443 10098.8697 -2.925 0.003444 \*\***

**county\_namecherokee, ga 224832.6958 9883.4864 22.748 < 2e-16 \*\*\***

**county\_namecherokee, ok 25455.8978 10984.1995 2.318 0.020487 \***

**county\_namecherokee, sc 28443.3747 9850.7170 2.887 0.003888 \*\***

**county\_namechesapeake city, va 164457.4254 10129.7230 16.235 < 2e-16 \*\*\***

**county\_namecheshire, nh 122166.8841 9639.8739 12.673 < 2e-16 \*\*\***

**county\_namechester, pa 292665.7019 9945.0265 29.428 < 2e-16 \*\*\***

**county\_namechesterfield, va 187607.3861 9648.2009 19.445 < 2e-16 \*\*\***

**county\_namechippewa, wi 74952.9001 10099.3585 7.422 1.20e-13 \*\*\***

**county\_namechisago, mn 163770.1966 9853.0001 16.621 < 2e-16 \*\*\***

**county\_namechittenden, vt 247268.4176 9855.0469 25.091 < 2e-16 \*\*\***

**county\_namechristian, ky 5512.9521 9547.6807 0.577 0.563667**

**county\_namechristian, mo 115549.5058 9908.5843 11.662 < 2e-16 \*\*\***

**county\_namecitrus, fl 81924.2725 9992.8428 8.198 2.59e-16 \*\*\***

**county\_nameclackamas, or 384478.0805 9720.6996 39.553 < 2e-16 \*\*\***

**county\_nameclallam, wa 281702.3853 9641.3240 29.218 < 2e-16 \*\*\***

**county\_nameclark, in 91507.5531 9549.7036 9.582 < 2e-16 \*\*\***

**county\_nameclark, nv 216858.3595 11583.0097 18.722 < 2e-16 \*\*\***

**county\_nameclark, oh -9739.6626 9867.2812 -0.987 0.323622**

**county\_nameclark, wa 304229.5766 9720.1958 31.299 < 2e-16 \*\*\***

**county\_nameclarke, ga 136832.7402 9550.1677 14.328 < 2e-16 \*\*\***

**county\_nameclay, fl 113753.3331 10123.3866 11.237 < 2e-16 \*\*\***

**county\_nameclay, mn 66169.7718 9565.2862 6.918 4.73e-12 \*\*\***

**county\_nameclay, mo 144805.0348 9876.5583 14.661 < 2e-16 \*\*\***

**county\_nameclayton, ga 27439.9942 10012.4879 2.741 0.006139 \*\***

**county\_nameclearfield, pa -25838.3524 10097.7997 -2.559 0.010511 \***

**county\_nameclermont, oh 179600.9901 9753.8726 18.413 < 2e-16 \*\*\***

**county\_namecleveland, nc 50310.2521 9970.9483 5.046 4.56e-07 \*\*\***

**county\_namecleveland, ok 88909.5830 9319.7250 9.540 < 2e-16 \*\*\***

**county\_nameclinton, ia -29218.7574 10078.3696 -2.899 0.003746 \*\***

**county\_nameclinton, mi 98632.8238 10248.2034 9.624 < 2e-16 \*\*\***

**county\_nameclinton, ny 39858.2337 9542.0934 4.177 2.97e-05 \*\*\***

**county\_namecobb, ga 246045.6277 10872.0055 22.631 < 2e-16 \*\*\***

**county\_namecochise, az 62407.3662 10575.1427 5.901 3.67e-09 \*\*\***

**county\_namecoconino, az 375566.9077 10117.8272 37.119 < 2e-16 \*\*\***

**county\_namecoffee, al 27618.2306 9852.7046 2.803 0.005066 \*\***

**county\_namecoffee, tn 102353.9893 9750.3966 10.497 < 2e-16 \*\*\***

**county\_namecolbert, al 15354.2200 9970.4681 1.540 0.123584**

**county\_namecole, mo 22749.2864 9982.6807 2.279 0.022685 \***

**county\_namecoles, il -58111.2748 10573.2329 -5.496 3.93e-08 \*\*\***

**county\_namecollier, fl 417645.7388 11122.3796 37.550 < 2e-16 \*\*\***

**county\_namecollin, tx 234690.7649 10380.8977 22.608 < 2e-16 \*\*\***

**county\_namecolumbia, fl 77633.8754 9638.4518 8.055 8.43e-16 \*\*\***

**county\_namecolumbia, ga 96150.7136 9815.7424 9.796 < 2e-16 \*\*\***

**county\_namecolumbia, ny 305195.5846 10121.0868 30.154 < 2e-16 \*\*\***

**[ reached getOption("max.print") -- omitted 805 rows ]**

**---**

**Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1**

**Residual standard error: 31510 on 19661 degrees of freedom**

**Multiple R-squared: 0.9698, Adjusted R-squared: 0.9682**

**F-statistic: 628.4 on 1004 and 19661 DF, p-value: < 2.2e-16**

**# SVM**

Call:

svm(formula = median\_listing\_price ~ county\_name + month\_date\_yyyymm + median\_days\_on\_market +

median\_square\_feet + new\_listing\_count + total\_listing\_count, data = trainData, kernel = "radial",

C = 10, cross = 10, prob.model = TRUE)

Parameters:

SVM-Type: eps-regression

SVM-Kernel: radial

cost: 1

gamma: 0.0009950249

epsilon: 0.1

Number of Support Vectors: 16835

10-fold cross-validation on training data:

Total Mean Squared Error: 25929789987

Squared Correlation Coefficient: 0.2258663

Mean Squared Errors:

25733962137 27684231824 24550651018 19177744099 30688401898 25746000376 31384128134 26006548745 22189771039 26135646858

> projectsvmpred <- predict(projectsvm, testData, type= "votes")

> # root mean squared error for svm model

> compTable <- data.frame(testData[,4], projectsvmpred)

> colnames(compTable) <- c("test","Pred")

> #Root Mean Squared Error

> sqrt(mean((compTable$test-compTable$Pred)^2))

[1] 162470

**# KSVM model**

> projectksvm

Support Vector Machine object of class "ksvm"

SV type: eps-svr (regression)

parameter : epsilon = 0.1 cost C = 10

Gaussian Radial Basis kernel function.

Hyperparameter : sigma = 0.0386438576329615

Number of Support Vectors : 4265

Objective Function Value : -894.7964

Training error : 0.004137

Cross validation error : 1119185411

Laplace distr. width : 291854.3

> summary(projectksvm)

Length Class Mode

1 ksvm S4

> ksvm\_pred <- predict(projectksvm, testData)

> summary(ksvm\_pred)

V1

Min. : 60989

1st Qu.: 204565

Median : 273318

Mean : 307602

3rd Qu.: 359928

Max. :2082923

> str(testData[4])

tibble [10,334 × 1] (S3: tbl\_df/tbl/data.frame)

$ median\_listing\_price: num [1:10334] 367000 165000 134900 342450 549500 ...

> # root mean squared error for ksvm model

> compTable <- data.frame(testData[,4], ksvm\_pred[,1])

> colnames(compTable) <- c("test","Pred")

> #Root Mean Squared Error

> sqrt(mean((compTable$test-compTable$Pred)^2))

[1] 32068.57

# Interpretation and Action Steps: Answering the Business Questions

**Housing Prices**

* Using a T-Test we were able to conclude that the price before and after COVID was significantly different.
* While COVID did slow down the rate of how much price increases. After COVID housing prices bounced back harder and faster than before.
* I would recommend that land investors who currently own a home or land should sell their property because prices are appreciating.
* On the other hand, investors looking to invest might want to wait until prices decrease or the market normalizes.

**Days on Market**

* Using a T-Test we also concluded that the median days on market after COVID is significantly different before and after COVID. Based on our findings I would advise land investors that the current real estate market is a seller market and not a buyers market.
* This means that investors can earn a higher return on an investment in a shorter period of time compared to before COVID.

# Full RCode

**install.packages("scales")**

**library(scales)**

**projectdata <- RDC\_Inventory\_Core\_Metrics\_County\_History**

**count(newprojectdata)**

**install.packages('caTools')**

**library('caTools')**

**install.packages('corrplot')**

**library(corrplot)**

**library(readr)**

**library(zipcode)**

**library(ggmap)**

**library(ggplot2)**

**library(reshape2)**

**library(e1071)**

**library(arulesViz)**

**newprojectdata <- projectdata[projectdata$month\_date\_yyyymm > 201812,] #date between**

**colnames(new)[colSums(is.na(new)) >0] #col name with NAs**

**sum(is.na(newdata))**

**summary(newdata)**

**a= c("price\_increased\_count\_yy","price\_increased\_count\_mm")**

**new <- subset(newprojectdata, select = a, drop = FALSE)**

**str(new)**

**a = c(17,18,20,21,23,24)**

**new <- newprojectdata[,-a]**

**summary(newdata)**

**colnames(new)**

**b = c("price\_increased\_count\_yy","price\_increased\_count\_mm","median\_listing\_price\_per\_square\_foot\_mm", "median\_listing\_price\_per\_square\_foot\_yy", "median\_square\_feet\_yy", "median\_square\_feet\_mm","pending\_listing\_count\_mm","pending\_listing\_count\_yy",'price\_reduced\_count\_mm','price\_reduced\_count\_yy')**

**newdata <- newprojectdata[, !(colnames(newprojectdata) %in% b), drop = FALSE] # picking the columns we want.**

**sum(is.na(newdata))**

**newprojectdata <- newdata**

**library(dplyr)**

**count(newprojectdata, state)**

**str(newdata)**

**us <- map\_data("state")**

**us**

**newprojectdata$state = unlist(sapply(strsplit(newprojectdata$county\_name, ","),function(x) x[2]))**

**newprojectdata$county = unlist(sapply(strsplit(newprojectdata$county\_name, ","),function(x) x[1]))**

**newprojectdata$state <- toupper(newprojectdata$state)**

**newprojectdata**

**newprojectdata$month = substr(newprojectdata$month\_date\_yyyymm, 1,4)**

**newprojectdata$date = substr(newprojectdata$month\_date\_yyyymm, 5,6)**

**newprojectdata[c('median\_listing\_price','')]**

**a <- ggplot(newprojectdata, aes(y=median\_listing\_price, fill= state, color=)) + geom\_bar()**

**a # count of housing open sales in different state**

**count(newprojectdata,state)**

**count(newprojectdata,state)[order(n),]**

**str(newprojectdata)**

**m = aggregate(newprojectdata$median\_listing\_price,list(newprojectdata$state),mean)**

**colnames(m)**

**newprojectdata[order(newprojectdata$state),]**

**beforeCovid <- newprojectdata[which(newprojectdata$month\_date\_yyyymm < 202003, newprojectdata$month\_date\_yyyymm > 201900),]**

**duringCovid <- newprojectdata[which(newprojectdata$month\_date\_yyyymm < 202102, newprojectdata$month\_date\_yyyymm > 202002),]**

**afterCovid <- newprojectdata[which(newprojectdata$month\_date\_yyyymm > 202101),]**

**v = tapply(beforeCovid$median\_listing\_price, beforeCovid$state, mean)**

**v1 = tapply(duringCovid$median\_listing\_price, duringCovid$state, mean)**

**v2 = tapply(afterCovid$median\_listing\_price, afterCovid$state, mean)**

**v[which.max(v)]**

**v1[which.max(v1)]**

**v2[which.max(v2)]**

**m[which.max(m$x),]**

**plot(c(v[which.max(v)],v1[which.max(v1)],v2[which.max(v2)]))**

**c1=mean(beforeCovid$median\_listing\_price[beforeCovid$state == ' AR'])**

**c2=mean(duringCovid$median\_listing\_price[duringCovid$state == ' AR'])**

**c3= mean(afterCovid$median\_listing\_price[afterCovid$state == ' AR'])**

**plot(c(c1,c2,c3))**

**gd <- newprojectdata[c('month\_date\_yyyymm','median\_listing\_price','state','county')]**

**gd$state <- toupper(gd$state)**

**gd$state <- gsub(" ","",gd$state, fixed = TRUE)**

**gd$stateName <- state.name[match(gd$state,state.abb)]**

**beforeCovid <- gd[which(gd$month\_date\_yyyymm < 202003, gd$month\_date\_yyyymm > 201900),]**

**duringCovid <- gd[which(gd$month\_date\_yyyymm < 202102, gd$month\_date\_yyyymm > 202002),]**

**afterCovid <- gd[which(gd$month\_date\_yyyymm > 202101),]**

**ggplot(gd, aes(x=median\_listing\_price)) + geom\_histogram(binwidth = 10)**

**str(gd)**

**#overall**

**price <- tapply(gd$median\_listing\_price, gd$stateName,sum)**

**state<-row.names(price)**

**stateincome <- data.frame(state,price)**

**stateincome$state <- tolower(stateincome$state)**

**ggplot(stateincome, aes(map\_id = state)) +geom\_map(map=us,aes(fill=price)) +expand\_limits(x=us$long,y=us$lat)+ coord\_map()**

**str(stateincome)**

**#beforecovid**

**beforeprice <- tapply(beforeCovid$median\_listing\_price, beforeCovid$stateName,mean)**

**state<-row.names(beforeprice)**

**stateincome <- data.frame(state,beforeprice)**

**stateincome$state <- tolower(stateincome$state)**

**beforecovidplot <- ggplot(stateincome, aes(map\_id = state)) +geom\_map(map=us,aes(fill=beforeprice),color='black')**

**beforecovidplot <- beforecovidplot + expand\_limits(x=us$long,y=us$lat) + scale\_fill\_gradient(low='red',high='white') + coord\_map()**

**beforecovidplot + ggtitle('Median Housing price per states before Covid')**

**#duringcovid**

**price <- tapply(duringCovid$median\_listing\_price, duringCovid$stateName,mean)**

**state<-row.names(price)**

**stateincome <- data.frame(state,price)**

**stateincome$state <- tolower(stateincome$state)**

**duringcovidplot <- ggplot(stateincome, aes(map\_id = state)) +geom\_map(map=us,aes(fill=price),color='black')**

**duringcovidplot <- duringcovidplot + expand\_limits(x=us$long,y=us$lat)**

**duringcovidplot <- duringcovidplot + scale\_fill\_gradient(low='red',high='white') + coord\_map()**

**duringcovidplot + ggtitle('Median Housing price per states during Covid')**

**#aftercovid**

**options(scipen=10)**

**price <- tapply(afterCovid$median\_listing\_price, afterCovid$stateName,mean)**

**state<-row.names(price)**

**stateincome <- data.frame(state,price)**

**stateincome$state <- tolower(stateincome$state)**

**aftercovidplot <- ggplot(stateincome, aes(map\_id = state)) +geom\_map(map=us,aes(fill=price),color='black')**

**aftercovidplot <- aftercovidplot + expand\_limits(x=us$long,y=us$lat) + coord\_map() + scale\_fill\_gradient(low='red',high='white')**

**aftercovidplot <- aftercovidplot + ggtitle('Median Housing price per states After Covid')**

**aftercovidplot <- aftercovidplot + theme(plot.title = element\_text(hjust = 0.5))**

**aftercovidplot**

**summary(gd)**

**# ANOVA**

**anova1 <- aov(median\_listing\_price~active\_listing\_count,data=newprojectdata)**

**summary(anova1)**

**cor(newprojectdata[c('median\_listing\_price','active\_listing\_count','new\_listing\_count','median\_days\_on\_market','month\_date\_yyyymm')], use='complete.obs',method='spearman')**

**# T-Test**

**#t test on whole US**

**a <- afterCovid$median\_listing\_price**

**b <- beforeCovid$median\_listing\_price**

**t.test(a,b)**

**#t test for days on market**

**t.test(day\_table$afterday, day\_table$beforeday)**

**ap <- data.frame(tapply(afterCovid$median\_listing\_price, afterCovid$stateName,mean))**

**ap$state <- rownames(ap)**

**dp <- data.frame(tapply(duringCovid$median\_listing\_price, duringCovid$stateName,mean))**

**dp$state <- rownames(dp)**

**bp <- data.frame(tapply(beforeCovid$median\_listing\_price, beforeCovid$stateName,mean))**

**bp$state <- rownames(bp)**

**t1 <- merge(ap,dp, by = 'state')**

**t2 <- merge(t1,bp, by='state')**

**str(t2)**

**colnames(t2) <- c('state','aftercovidprice','duringcovidprice','beforecovidprice')**

**plot(t2)**

**# each state median price for different time period**

**str(t2)**

**t4$state <- factor(t4$state, levels = t4$state[order(t4$pricechange)])**

**p1 <- ggplot(data=t4,aes(x=state)) +geom\_line(aes(y=aftercovidprice,group=1,color='after'))**

**p1 <- p1 + geom\_line(aes(y=beforecovidprice,group=1,color='before')) +geom\_line(aes(y=duringcovidprice,group=1,color='during'))**

**p1 <- p1 + geom\_point(aes(y=aftercovidprice),color='red') +geom\_point(aes(y=beforecovidprice),color='green') +geom\_point(aes(y=duringcovidprice),color='blue')**

**p1 <- p1 + theme(axis.text.x= element\_text(angle=90))**

**p1 + labs(color='legend')**

**str(t3)**

**t3 <- melt(t2,id='state')**

**newdata['median\_listing\_price']**

**ggplot(data=t3,aes(y=state,x=value,fill=variable)) + geom\_bar(width = 0.8,stat='identity') + theme(axis.text.x= element\_text(angle=90)) + scale\_fill\_brewer(palette="Paired") + theme\_minimal() + ggtitle('Median housing price per states')**

**# before and after percent change for house price**

**pricechange <-t4[order(t4$pricechange),]**

**plot\_price <- pricechange[,c(1,7)]**

**plot\_price$state <- factor(plot\_price$state, levels = plot\_price$state[order(plot\_price$pricechange)])**

**ggplot(data = plot\_price, aes(x=pricechange,y=state)) + geom\_bar(stat="identity") + ggtitle("housing price percentage change")**

**head(plot\_price)**

**colnames(t4)**

**t4<-t2**

**t4$pricechangeafter <- (t4$aftercovidprice/t4$duringcovidprice)-1**

**t4$pricechangeduring <- (t4$duringcovidprice/t4$beforecovidprice)-1**

**t4$pricechange <- (t4$aftercovidprice/t4$beforecovidprice)-1**

**str(t4)**

**melt(t4[,c(1,5:7)],id='state')**

**t4$state <- tolower(t4$state)**

**c <- ggplot(t4[,c(1,7)], aes(map\_id = state)) + geom\_map(map=us,aes(fill=pricechange),color='black')**

**c <- c + expand\_limits(x=us$long,y=us$lat) + coord\_map() + scale\_fill\_gradient(low='white',high='red')**

**c + theme(axis.title.x = element\_blank(), axis.title.y = element\_blank(), axis.text.y = element\_blank(), axis.text.x=element\_blank()) + ggtitle("housing price percentage change")**

**# Create dataFrame for median days on makret plot**

**rm(gdd)**

**gdd<-data.frame()**

**gdd <- newprojectdata[c('month\_date\_yyyymm','median\_days\_on\_market','state','county')]**

**gdd$state <- gsub(" ","",gdd$state, fixed = TRUE)**

**gdd$stateName <- state.name[match(gdd$state,state.abb)]**

**beforeCovidd <- gdd[which(gdd$month\_date\_yyyymm < 202003, gdd$month\_date\_yyyymm > 201900),]**

**duringCovidd <- gdd[which(gdd$month\_date\_yyyymm < 202102, gdd$month\_date\_yyyymm > 202002),]**

**afterCovidd <- gdd[which(gdd$month\_date\_yyyymm > 202101),]**

**str(gdd)**

**# after covid days on market**

**days <- tapply(afterCovidd$median\_days\_on\_market, afterCovidd$stateName,mean)**

**state<-row.names(days)**

**daysonmarket <- data.frame(state,days)**

**daysonmarket$state <- tolower(daysonmarket$state)**

**aftercoviddplot <- ggplot(daysonmarket, aes(map\_id = state)) +geom\_map(map=us,aes(fill=days),color='black')**

**aftercoviddplot <- aftercoviddplot + expand\_limits(x=us$long,y=us$lat) + coord\_map() + scale\_fill\_gradient(low='red',high='white')**

**aftercoviddplot <- aftercoviddplot + ggtitle('Median day on market per states After Covid')**

**aftercoviddplot <- aftercoviddplot + theme(plot.title = element\_text(hjust = 0.5))**

**aftercoviddplot**

**# before covid days on market**

**days <- tapply(beforeCovidd$median\_days\_on\_market, beforeCovidd$stateName,mean)**

**state<-row.names(days)**

**daysonmarket <- data.frame(state,days)**

**daysonmarket$state <- tolower(daysonmarket$state)**

**beforecoviddplot <- ggplot(daysonmarket, aes(map\_id = state)) +geom\_map(map=us,aes(fill=days),color='black')**

**beforecoviddplot <- beforecoviddplot + expand\_limits(x=us$long,y=us$lat) + coord\_map() + scale\_fill\_gradient(low='red',high='white')**

**beforecoviddplot <- beforecoviddplot + ggtitle('Median day on market per states Before Covid')**

**beforecoviddplot <- beforecoviddplot + theme(plot.title = element\_text(hjust = 0.5))**

**beforecoviddplot**

**p1 <- ggplot(data=t2,aes(x=state)) +geom\_line(aes(y=aftercovidprice,group=1,color='after'))**

**p1 <- p1 + geom\_line(aes(y=beforecovidprice,group=1,color='before')) +geom\_line(aes(y=duringcovidprice,group=1,color='during'))**

**p1 <- p1 + geom\_point(aes(y=aftercovidprice),color='red') +geom\_point(aes(y=beforecovidprice),color='green') +geom\_point(aes(y=duringcovidprice),color='blue')**

**p1 <- p1 + theme(axis.text.x= element\_text(angle=90))**

**p1 + labs(color='legend')**

**# after covid days on market**

**days <- tapply(afterCovidd$median\_days\_on\_market, afterCovidd$stateName,mean)**

**state<-row.names(days)**

**daysonmarket <- data.frame(state,days)**

**daysonmarket$state <- tolower(daysonmarket$state)**

**aftercoviddplot <- ggplot(daysonmarket, aes(map\_id = state)) +geom\_map(map=us,aes(fill=days),color='black')**

**aftercoviddplot <- aftercoviddplot + expand\_limits(x=us$long,y=us$lat) + coord\_map() + scale\_fill\_gradient(low='red',high='white')**

**aftercoviddplot <- aftercoviddplot + ggtitle('Median day on market per states After Covid')**

**aftercoviddplot <- aftercoviddplot + theme(plot.title = element\_text(hjust = 0.5))**

**aftercoviddplot**

**# during covid days on market**

**days <- tapply(duringCovidd$median\_days\_on\_market, duringCovidd$stateName,mean)**

**state<-row.names(days)**

**daysonmarket <- data.frame(state,days)**

**daysonmarket$state <- tolower(daysonmarket$state)**

**duringcoviddplot <- ggplot(daysonmarket, aes(map\_id = state)) +geom\_map(map=us,aes(fill=days),color='black')**

**duringcoviddplot <- duringcoviddplot + expand\_limits(x=us$long,y=us$lat) + coord\_map() + scale\_fill\_gradient(low='red',high='white')**

**duringcoviddplot <- duringcoviddplot + ggtitle('Median day on market per states during Covid')**

**duringcoviddplot <- duringcoviddplot + theme(plot.title = element\_text(hjust = 0.5))**

**duringcoviddplot**

**#percentage change on days on market**

**str(before\_day)**

**afterday <- tapply(afterCovidd$median\_days\_on\_market, afterCovidd$stateName,mean)**

**beforeday <- tapply(beforeCovidd$median\_days\_on\_market, beforeCovidd$stateName,mean)**

**after\_day <- data.frame(row.names(afterday),afterday)**

**colnames(after\_day)<- c('state','afterday')**

**before\_day <- data.frame(row.names(beforeday),beforeday)**

**colnames(before\_day)<- c('state','beforeday')**

**day\_table <- merge(before\_day, after\_day, by='state')**

**day\_table$percentchange <- abs((day\_table$afterday/day\_table$beforeday) -1)**

**#plot result for percentage change for days on market before and after covid**

**plot\_day <- day\_table[,c(1,4)]**

**plot\_day$state <- factor(plot\_day$state, levels = plot\_day$state[order(plot\_day$percentchange)])**

**ggplot(data = plot\_day, aes(x=percentchange,y=state)) + geom\_bar(stat="identity") + scale\_fill\_viridis\_c() + ggtitle("days on market percentage change in different states")**

**#listing days on market for each state bar chart**

**a1 <- melt(day\_table[1:3], id='state')**

**str(a1)**

**ggplot(data=a1,aes(y=state,x=value,fill=variable)) + geom\_bar(width = 0.8,stat='identity') + theme(axis.text.x= element\_text(angle=90)) + scale\_fill\_brewer(palette="Paired") + theme\_minimal() + ggtitle('listing days on market for each state')**

**#map plot for days**

**day\_table$state <- tolower(day\_table$state)**

**map\_day <- ggplot(day\_table[,c(1,4)], aes(map\_id = state)) + geom\_map(map=us,aes(fill=percentchange),color='black')**

**map\_day <- map\_day + expand\_limits(x=us$long,y=us$lat) + coord\_map() + scale\_fill\_gradient(low='white',high='red')**

**map\_day + theme(axis.title.x = element\_blank(), axis.title.y = element\_blank(), axis.text.y = element\_blank(), axis.text.x=element\_blank())**

**map\_day + ggtitle("days on market percentage change")**

**# linear model**

**lmdata <- newdata**

**scatter.smooth(x=afterCovid$median\_listing\_price, y=duringCovid$median\_listing\_price,beforeCovid$median\_listing\_price)**

**lmmodel <- lm(formula = median\_listing\_price~median\_days\_on\_market + total\_listing\_count+median\_listing\_price\_per\_square\_foot, data=newdata)**

**summary(lmmodel)**

**cor.data <- select(newdata,-c('county\_name'))**

**#newprojectdata[,c('median\_days\_on\_market','median\_listing\_price','total\_listing\_count','median\_listing\_price\_mm','median\_listing\_price\_yy','active\_listing\_count\_mm')]**

**str(cor.data)**

**cor.data**

**c<-cor(x=cor.data)**

**cor(x= newprojectdata$median\_listing\_price,select(newprojectdata))**

**any(is.na(newdata))**

**corrplot(c, method='number')**

**two.way <- aov(aftercovidprice ~ beforecovidprice + duringcovidprice, data = t2)**

**summary(two.way)**

**ap <- data.frame(tapply(afterCovid$median\_listing\_price, afterCovid$stateName,mean))**

**ap$state <- rownames(ap)**

**dp <- data.frame(tapply(duringCovid$median\_listing\_price, duringCovid$stateName,mean))**

**dp$state <- rownames(dp)**

**bp <- data.frame(tapply(beforeCovid$median\_listing\_price, beforeCovid$stateName,mean))**

**bp$state <- rownames(bp)**

**str(price)**

**t1 <- merge(ap,dp, by = 'state')**

**t2 <- merge(t1,bp, by='state')**

**one.anova <- aov(aftercovidprice ~ beforecovidprice, data = t2)**

**summary(one.anova)**

**str(t2)**

**test1 <- newprojectdata[c('month\_date\_yyyymm','median\_listing\_price','county\_name','median\_days\_on\_market')]**

**#svm**

**packages=c("arulesViz", "kernlab","e1071","gridExtra","ggplot2", "caret", "arules")**

**package.check <- lapply(packages, FUN = function(x) {**

**if (!require(x, character.only = TRUE)) {**

**install.packages(x, dependencies = TRUE)**

**library(x, character.only = TRUE)**

**}**

**})**

**#verify they are loaded**

**svmData <- newdata[c('')]**

**search()**

**# train test split**

**randIndex <- sample(1:dim(newdata)[1])**

**nrow(newdata)**

**cutpoint2\_3 <- floor(2\*nrow(newdata)/3)**

**cutpoint2\_3**

**trainData <- newdata[randIndex[1:cutpoint2\_3],]**

**dim(trainData)**

**head(trainData)**

**testData <- newdata[randIndex[(cutpoint2\_3+1):nrow(newdata)],]**

**str(testData)**

**any(is.na(testData))**

**colnames(trainData)**

**#month\_date\_yyyymm + county\_name + median\_days\_on\_market+median\_square\_feet+new\_listing\_count+total\_listing\_count**

**#Linear model**

**projectlm <- lm(median\_listing\_price~month\_date\_yyyymm + county\_name + median\_days\_on\_market+median\_square\_feet+new\_listing\_count+total\_listing\_count, data=trainData)**

**summary(projectlm)**

**lm\_pred1 <- predict(projectlm,testData)**

**# root mean squared error for linear model**

**lm\_table1<- data.frame(testData[,4],lm\_pred1)**

**colnames(lm\_table1) <- c("test","Pred")**

**sqrt(mean((lm\_table1$test-lm\_table1$Pred)^2))**

**# ksvm model**

**projectksvm <- ksvm(median\_listing\_price~., # set "Ozone" as the target predicting variable; "." means use all other variables to predict "Ozone"**

**data = trainData, # specify the data to use in the analysis**

**kernel = "rbfdot", # kernel function that projects the low dimensional problem into higher dimensional space**

**kpar = "automatic",# kpar refer to parameters that can be used to control the radial function kernel(rbfdot)**

**C = 10, # C refers to "Cost of Constrains"**

**cross = 5, # use 10 fold cross validation in this model**

**prob.model = TRUE # use probability model in this model**

**)**

**projectksvm**

**summary(projectksvm)**

**ksvm\_pred <- predict(projectksvm, testData)**

**summary(ksvm\_pred)**

**str(testData[4])**

**# root mean squared error for ksvm model**

**compTable <- data.frame(testData[,4], ksvm\_pred[,1])**

**colnames(compTable) <- c("test","Pred")**

**#Root Mean Squared Error**

**sqrt(mean((compTable$test-compTable$Pred)^2))**

**# svm**

**projectsvm <- svm(median\_listing\_price~county\_name + month\_date\_yyyymm + median\_days\_on\_market+median\_square\_feet+new\_listing\_count+total\_listing\_count,data=trainData,kernel="radial",C=10,cross=10,prob.model=TRUE)**

**summary(projectsvm)**

**projectsvmpred <- predict(projectsvm, testData, type= "votes")**

**# root mean squared error for svm model**

**compTable <- data.frame(testData[,4], projectsvmpred)**

**colnames(compTable) <- c("test","Pred")**

**#Root Mean Squared Error**

**sqrt(mean((compTable$test-compTable$Pred)^2))**

**#lm**

**colnames(aot) <- c('count\_name','price\_after','price\_before')**

**str(aot)**

**randIndex <- sample(1:dim(aot)[1])**

**nrow(aot)**

**cutpoint2\_3 <- floor(2\*nrow(aot)/3)**

**cutpoint2\_3**

**trainData <- aot[randIndex[1:cutpoint2\_3],]**

**dim(trainData)**

**head(trainData)**

**any(is.na(testData))**

**str(testData)**

**testData <- aot[randIndex[(cutpoint2\_3+1):nrow(aot)],]**

**lm\_p <- lm(price\_after~price\_before, data =trainData)**

**summary(lm\_p)**

**lm\_pred <- predict(lm\_p,testData)**