Visualization of Real Estate in Frisco (2021-2023)

Su Lin Goh, Veronica Kopper, Jack Macdonald, Xiaoyan Zhang

Abstract

This paper explores the dynamics influencing single-family homes between the year 2021 to March 2023 in the cities of Frisco, Carrollton, and Little Elm regions experiencing significant population growth. By using a dataset that has been downloaded from redfin.com, it contains spatial coordinates and detailed property characteristics. Our study seeks to uncover the various factors that affect the real estate scene in these fast-growing suburbs. The objective is to equip those involved in real estate transactions with practical knowledge to make well-informed decisions.

This study employs data analytics techniques to explain the spatial patterns of single-family home prices across the cities previously mentioned. Utilizing the R programming language, and packages like ggplot2 (Wickham 2016), the dataset is visualized through various techniques, including bar plot, box plot, scatter plot, heat map, correlation matrix, and a dynamic geospatial visualization map. The dynamic map was created into a web application using the Shiny package (Wickham 2020). Additionally, a linear graph built with basic R code (Murrell 2016), is incorporated to illustrate the population growth in the Frisco area, showcasing its significant impact on the heightened demand for single-family homes and the subsequent shifts in market equilibrium prices.

Introduction

In the real estate market arena, making well-informed decisions is crucial for executing intelligent investments. Housing serves as the primary conduit for wealth accumulation for the typical household, given its feasibility for acquisition through debt (Yiu 2023). According to OECD data from 2022, housing wealth, constituting an average of 50.4% of household total wealth, is a significant component. Furthermore, mortgage debt stands out as the predominant portion, representing 68.6% on average of total household debt. This underscores the profound impact of housing and monetary policies on wealth distribution within the market.

Our analysis centers on the growing cities of Frisco, Carrollton, and Little Elm, where strategic urban planning, economic vitality, and visionary leadership have fueled its evolution from a historic agricultural hub to a thriving modern city. With its strategic location and commitment to quality of life, Frisco earned the distinction of being the nation's fastest-growing city by 1990. With a population that surged from about 120,000 in 2011 to surpass 200,000 today, Frisco's economic narrative reflects a strategic urban metamorphosis. In 2023, the population in Frisco increased to 230,794. The housing units are 85,242 and 59,641 are built as single-family units (according to Frisco Texas demographics).

The purpose of the study is to enhance comprehension of the Frisco housing market, this study transcends conventional numerical analyses. Utilizing data visualization techniques, particularly employing the R programming language, to demonstrate the intricate relationship between home location, sizes, number of rooms and bathrooms, amenities, and other features, all of which contribute to the pricing dynamics of each property. Additionally, the study explores the impact of population growth as an influential variable. The primary objective is to employ data visualization tools to showcase distinct price variations based on diverse property characteristics, thereby providing valuable insights for both buyers and sellers to make well-informed decisions in the real estate market.

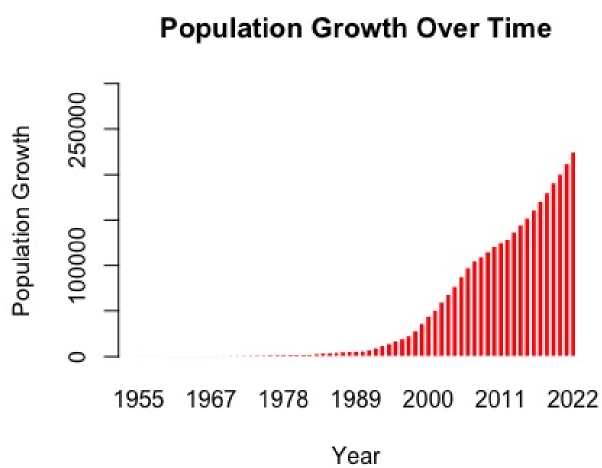
Literature review

Alberto Cairo (2012) emphasizes that infographics and data visualizations are tools designed to enlighten and inform. This literature review delves into the intricate dynamics of factors influencing housing prices, drawing insights from recent studies and analyses. It incorporates the long-term investment perspective, focusing on Frisco, Texas, as a case study.

- Savva (2018) identifies population growth, economic indicators like stock returns, and inflation as key drivers of increasing housing prices. Construction costs are also associated with housing prices.
- Zoning plays a dominant role in inflating house prices (Choudhury, 2012). Clark's (2023) study on Frisco highlights the significance of transparent land-use plans, streamlined permitting processes, and a developer-friendly approach.
- The city's housing market accommodates its rising population, making it an attractive option for those seeking stable and enduring investments. Frisco's appeal as a long-term investment housing market is noteworthy. Home to significant attractions, including the Dallas Cowboys, and PGA of America. Headquarters, Frisco RoughRiders, Frisco Heritage Museum, Museum of the

American Railroad, Sci-Tech Discovery Center, and the Karla Siddhi Hanuman Temple (VictorSteffen.com)

The comprehensive exploration of factors influencing housing prices, from the immediate impact as monetary policy to the enduring significance of zoning policies and long-term investment perspectives in cities like Frisco. As Cairo (2012) suggests, utilizing infographics and data visualizations can enhance understanding and decision-making in this intricate landscape, providing valuable insights for policymakers and the public.



(Dataset from https://www.friscotexas.gov/1454/Demographics)

Furthermore, the economic growth has helped the housing development demand growth, especially with the single-family units, and it is located in the Dallas metropolitan area, which has made it a convenient place to live. In addition, the investments in roads, parks, and public facilities also make it a comfortable place to live in. Frisco is a strong and growing place to perform well in the future.

Data collection

We collected the Frisco demographics dataset from the Frisco Texas website - https://www.friscotexas.gov/1454/Demographics. This dataset shows the population in Frisco from 1955 to 2022.

	A	B	C	D
1	Year	Population	PGrowth	
2	1955	552	0.00	
3	1956	599	0.09	
4	1958	647	0.08	
5	1959	656	0.01	
6	1960	751	0.15	
7	1961	802	0.07	
8	1962	832	0.04	
9	1963	856	0.03	
10	1964	898	0.05	
11	1965	975	0.09	
12	1966	1017	0.04	
13	1967	1061	0.04	
14	1968	1232	0.16	
15	1969	1262	0.02	
16	1970	1359	0.08	
17	1971	1400	0.03	
18	1972	1495	0.07	
19	1973	1573	0.05	
20	1974	1620	0.03	

As for the housing dataset we collected it from the Redfin Corporation website - https://www.redfin.com/. It is a technology-powered real estate company and also the US number one real estate brokerage site which is based in Seattle.

1	SALE TYPE	SOLD DATE	PROPERTY T	ADDRESS	CITY	STATE OR P	ZIP OR POS	PRICE	BEDS	BATHS	LOCATION	SQUARE FEI	LOT SIZE	YEAR BUILT	DAYS ON M	$/SQUARE I	HOA/MONT	STATUS
2	PAST SALE	November-	Single Fami	2017 Jenni	Carrollton	TX	75007	375000	3	2	Woodlake (	1821	8059	1974		206		Sold
3	PAST SALE	April-25-20	Single Fami	2115 Lymi	Carrollton	TX	75007	389900	4	2	Hollyridge.	1706	7841	1979		229		Sold
4	PAST SALE	September	Single Fami	1116 Dent	Carrollton	TX	75007	608000	3	3	Cambridge	2721	9191	1986		223	6	Sold
5	PAST SALE	September	Single Fami	1731 Big C	Carrollton	TX	75007	410000	3	2	Nob Hill Ph	1822	7492	1984		225		Sold
6	PAST SALE	August-9-2	Single Fami	3311 Hillpa	Carrollton	TX	75007	598000	5	3.5	Timbercree	3141	9845	1989		190		Sold
7	PAST SALE	May-20-20	Single Fami	2047 Ash H	Carrollton	TX	75007	400000	3	2	Woodlake :	1818	8756	1974		220		Sold
8	PAST SALE	May-9-202	Single Fami	3146 Oak H	Carrollton	TX	75007	325000	3	2	Woodlake :	1506	7928	1974		216		Sold
9	PAST SALE	December-	Single Fami	4016 Provi	Carrollton	TX	75007	420000	3	2.5	High Ridge	2028	10454	1985		207		Sold
10	PAST SALE	March-16-2	Townhouse	4112 Woo	Carrollton	TX	75007	395000	3	2.5	Quail Creek	1996	4704	2004		198	300	Sold
11	PAST SALE	May-31-20	Single Fami	3902 Furne	Carrollton	TX	75007	375000	3	2	Carillon Hil	1883	9191	1984		199		Sold
12	PAST SALE	August-5-2	Single Fami	3835 Alto /	Carrollton	TX	75007	377000	3	2	Carillon Hil	1247	8189	1984		302		Sold
13	PAST SALE	May-13-20	Single Fami	1765 Flow	Carrollton	TX	75007	575000	4	2.5	Moore Farr	2986	5227	2004		193	50	Sold
14	PAST SALE	April-8-202	Single Fami	1124 Yorks	Carrollton	TX	75007	499900	4	2.5	Nottinghar	2386	8973	1978		210		Sold
15	PAST SALE	March-28-2	Single Fami	2904 Woo	Carrollton	TX	75007	310000	3	2	Woodlake 1	1580	8625	1973		196		Sold
16	PAST SALE	March-15-2	Single Fami	1805 Kensi	Carrollton	TX	75007	305000	3	2	Rosemeade	1862	8494	1977		164		Sold
17	PAST SALE	May-20-20	Single Fami	1529 Pawn	Carrollton	TX	75007	450000	4	2.5	Villages Of I	2338	5881	1995		192	48	Sold
18	PAST SALE	August-29-	Single Fami	1004 Oxfor	Carrollton	TX	75007	540000	4	3	Nottinghar	2231	9087	1978		242		Sold
19	PAST SALE	May-5-202	Single Fami	3106 Andre	Carrollton	TX	75007	439000	4	2.5	Timbercree	2442	7536	1991		180		Sold

This Frisco housing dataset which we downloaded has 27 variables and 5640 observations and the following variables are: sold a month, city, price, beds, baths, square feet, lot size, year build, listing price, latitude, and longitude. It shows houses built from 1902 till 2023 and houses sold between March 2021 to February 2023. We format the sold month in the dataset as the first date of each month. All properties are single-family houses in our dataset. We check the missing value of all variables and drop the house which has 1 bed with a house price over 1 million. The house price is right-skewed, so we do a log transfer to the price. Another variable we need to mention is the lot size, the lot size was more than 30000 square feet, and less than 2000 square feet were imputed with median lot size.

Data analysis: Charts and Tables

For any project, it is necessary to get a full understanding of the data and the variables. To prevent garbage in and garbage out, we checked all the variables, the qualities, and the relationship between variables. Those data analysis includes: frequency analysis, histograms, trends and scattered plots, summary statistics, missing values and imputation, outliers, new variable derivation, etc.

Frequency analysis

A frequency check is a very important starting point for understanding the variables, especially categorical variables. For this project, counts like houses segmented by zip code, variables like the number of baths, number of bedrooms, etc. are checked.

Example tables are shown below:

2	3	4	5	6	7
117	1034	1439	896	54	1

*Table SEQ Table * ARABIC 1*
Table1. Number of Beds

1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6	6.5	7	7.5	9	10
4	4	987	519	393	611	460	321	77	104	43	12	1	3	1	1

*Table SEQ Table * ARABIC*
2. Number of Baths

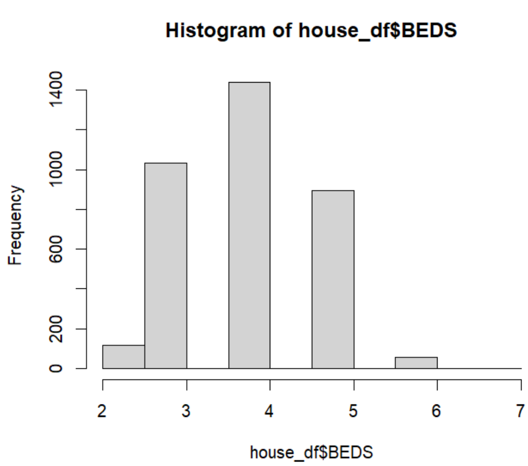
75007	75033	75034	75035	75036	75056	75068	75071	75072	75078	78373
350	783	488	1208	598	9	93	4	2	5	1

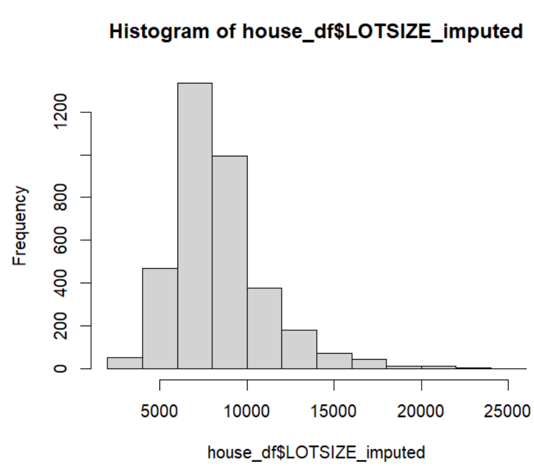
*Table SEQ Table * ARABIC 3.*
Number of houses by zip

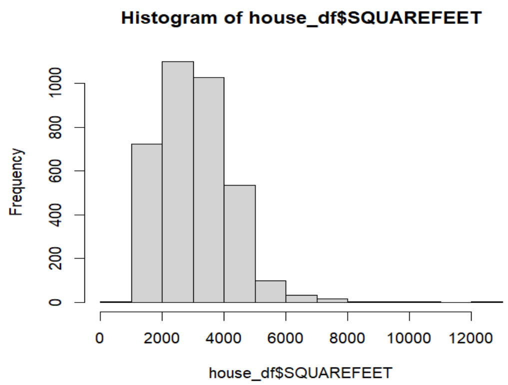
Table 1 tells us that most of the house has 2-5 bedrooms. Table 2 shows that in the data majority of the houses have 2 to 4.5 bathrooms while there're very few houses have more than 7 bathrooms. Table 3 show the house in this data are mainly in the zip codes: 75033, 75034, 75025, and 76036, in which zip code 75035 has most of the houses. There's some of the houses are in nearby Frisco which is due to data scraping, and they are not excluded since they are not material and will not affect the whole analysis.

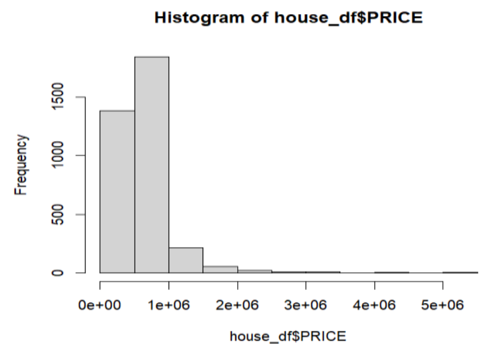
Histogram, Trend, and Scatter Plot

Plots could provide visualization of the data, we could know how the data is distributed based on the histogram, how the variable value tends to be by time from the trend plot, and how the two variables are correlated with each other from the scatter plot. These plots are very straightforward for understanding the data and the variables. In this project, we plot the histograms for variables like price, living square feet, lot size imputed, number of bedrooms, etc. Trend plots are generated for the price by month, bedroom, and so on, and scatter plots of price by living square feet. Here are the plots and their explanations:

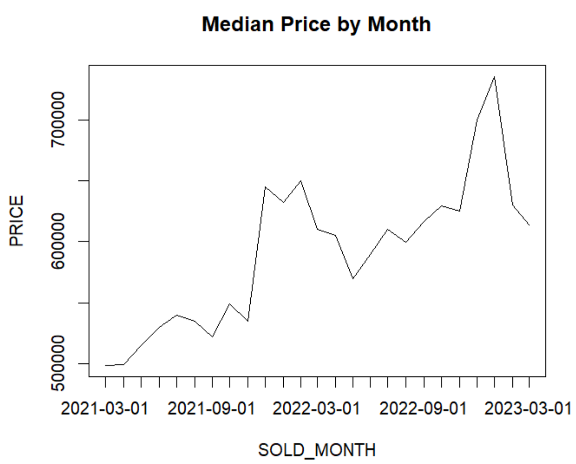


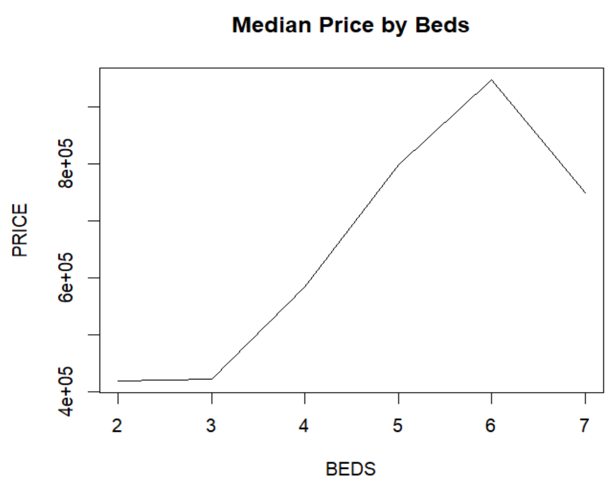






From the histogram of house price, and living square feet, we can see both of them are right skewed which means we might need to do a log transformation for those variables, especially for linear regression, while the bedroom plot is somehow bell-shaped, no transformation is needed.





The figure median price by month shows that the price is going up over time with seasonality. The figure median price by bed shows the price increase when the number of beds increases.

Summary Statistics

Similar to histograms, summary statistics can provide us with the exact values of the variable distribution, especially the numeric variables. From summary statistics, we could know percentiles at 0%, 1%,10%, 25%, 50%,75%, 90%, 95%, 99% and 100%, the minimum (0% percentile) and maximum (100% percentile), the mean and median (50% percentile) of the variable. The median house price is $575000, the median number of beds is 4, the median number of baths is 3, the median lot size is 7841 square feet, and the median living square feet is 2927. All variables including the target variable are all right skewed except the number of beds. The lot size has outliers, for example, 0% has 1 square foot, and 100% has 290806560 square feet.

Missing Value and Outlier

Checking missing values and outliers can provide us with a solid understanding of what we need to do for the variables accordingly. As for this project, we will focus on data visualization, the data have missing values and outliers could have a huge impact on the results, so we need to treat the missing value. For our data, for example, the variable price has one missing value, so we have to drop that observation. Also, there are some outliers for lot size, we drop the house with 1 feet square feet data and impute the 290806560 square feet as 30000 square feet.

Derive New Variables

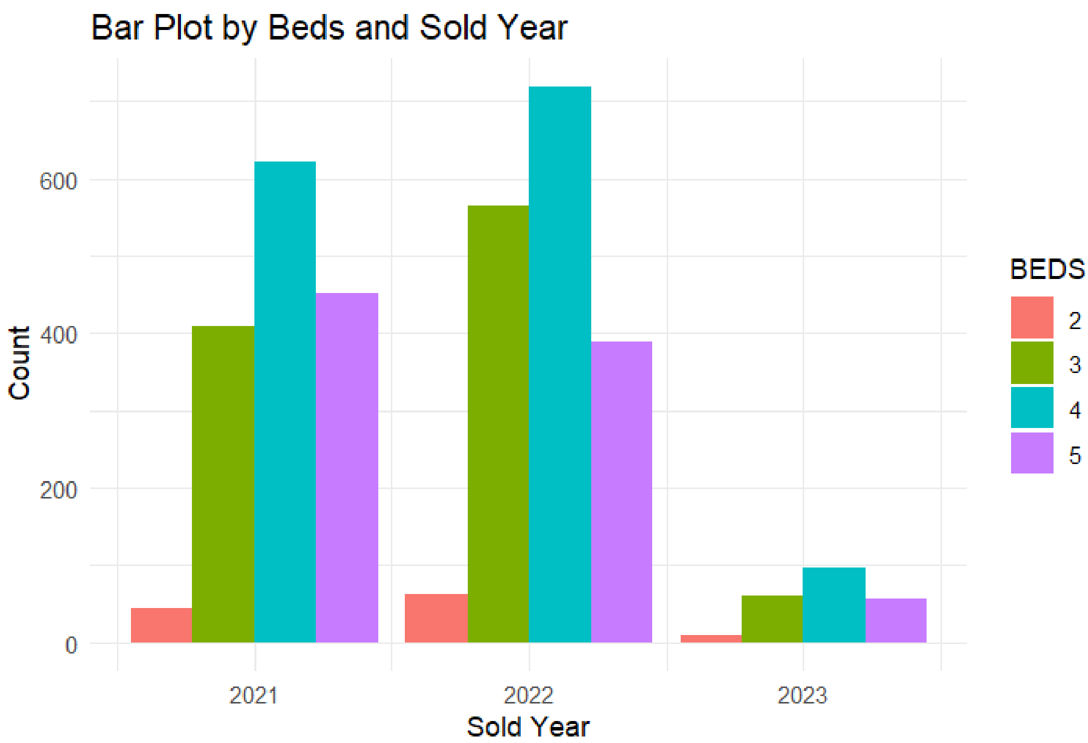
New derived variables sometimes can capture information that cannot be captured by existing variables. Age of house, living square feet imputed, month indicated for the sold month, those variables are not in our data, while it is necessary to create them. because some of the variables are crucial, such as the age of the house.

Visualization results and analysis

- Bar plot
- Box Plot
- Scatter Plot
- Heat Map
- Bubble Plot
- Correlation Matrix

```
################################### bar plot by beds and sold year#############
#filter the beds >1 and beds<=5
house_df1<-house_df[(house_df$BEDS >1) & (house_df$BEDS<=5) , ]
house_df1$BEDS<-factor(house_df1$BEDS)

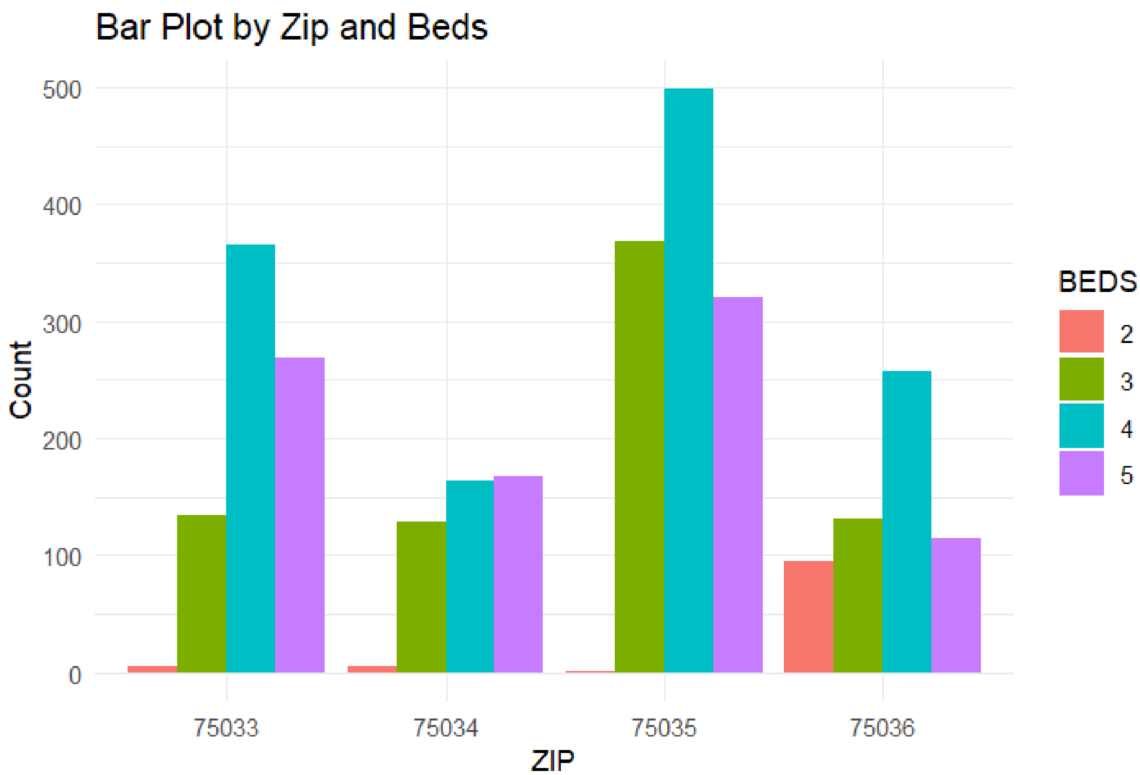
ggplot(house_df1, aes(x= sold_year, fill= BEDS)) +
        geom_bar( position= "dodge")+
        labs(title="Bar Plot by Beds and Sold Year", x="Sold Year", y="Count")+
        theme_minimal()
```



This bar plot shows the number of bedrooms in houses sold between 2021 to 2023. The houses with 4 beds show the highest number for all the three years. Houses with 3 beds and 5 beds are also quite demanding compared to 2-bed units.

```
#################### bar plot by count and zipcode #########################
# convert the zip to string
house_df$ZIP <- as.character(house_df$ZIP)
# filter the zipcode to 75033,75034, 75035 and 75036
desired_zips <- c("75033", "75034", "75035", "75036")

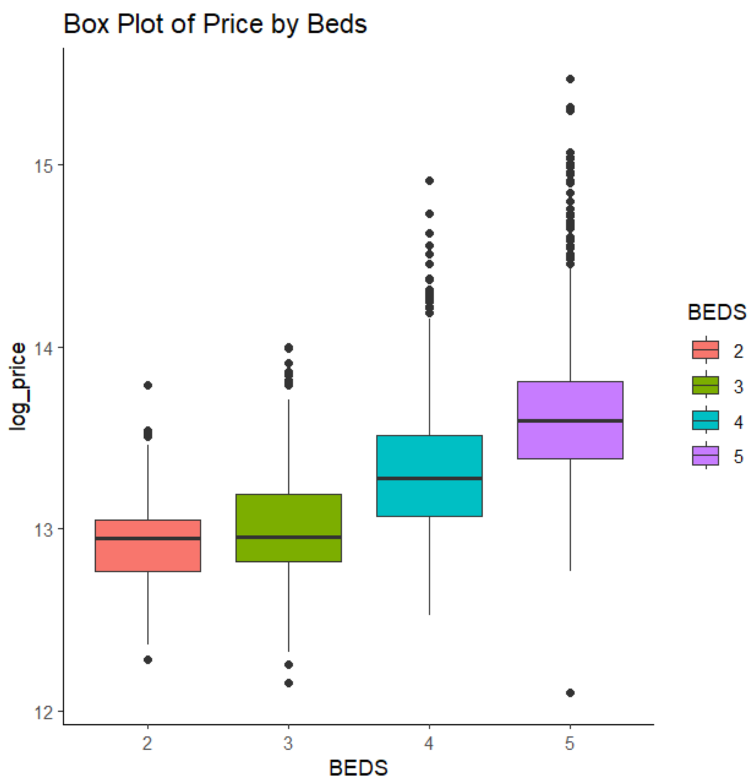
# Filter rows with the specified zip codes
house_df2 <- subset(house_df, ZIP %in% desired_zips)
house_df2<-house_df2[(house_df2$BEDS >1) & (house_df2$BEDS<=5) , ]
house_df2$BEDS<- factor(house_df2$BEDS)
ggplot(house_df2, aes(x= ZIP, fill= BEDS)) +
  geom_bar( position= "dodge")+
  labs(title="Bar Plot by Zip and Beds", x="ZIP", y="Count")+
  theme_minimal()
```



This bar plot shows the number of bedrooms in houses sold based on zip codes. Zip code 75035 is the most demanding compared to the other three zip codes. The second most demanding is 75033. As for the number of beds, the four-bed unit is the most popular.

```
############################## box plot ################################
# log trans to price
house_df$log_price <- log(house_df$PRICE)
#filter the beds >1 and beds<=5
house_df1<-house_df[(house_df$BEDS >1) & (house_df$BEDS<=5) , ]
house_df1$BEDS<-factor(house_df1$BEDS)

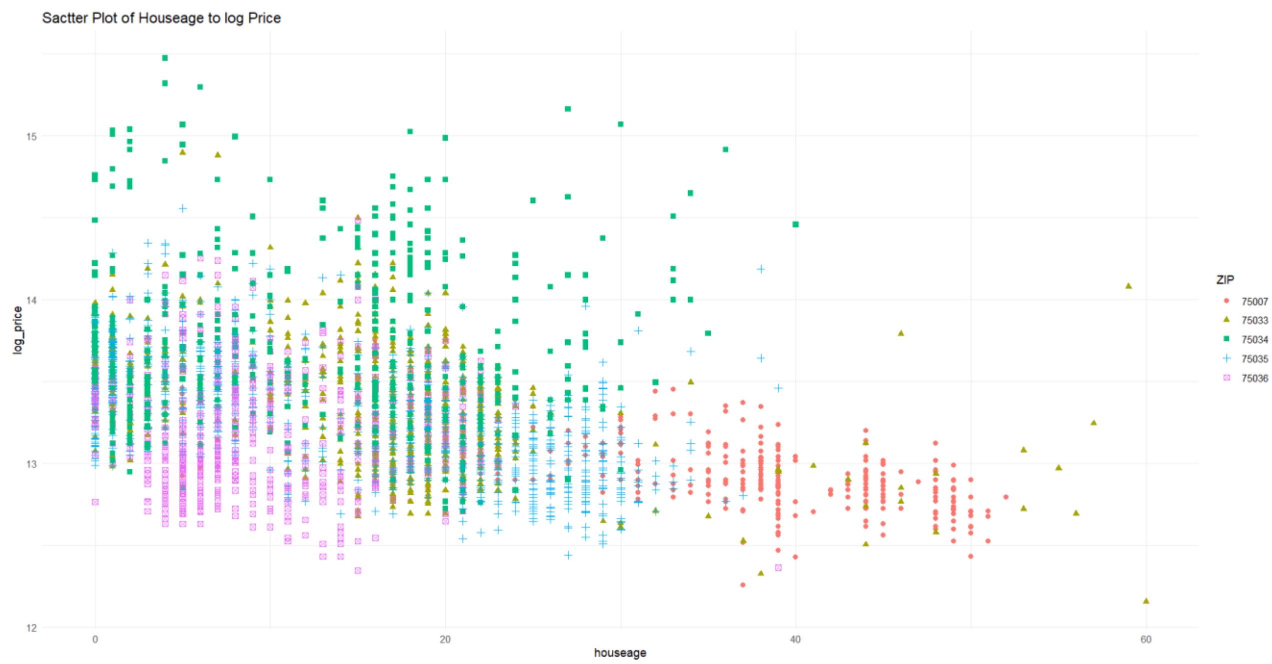
ggplot(house_df1, aes(x= BEDS, y= log_price)) +
  geom_boxplot(fill="gray")+
  labs(title="Box Plot of Price by Beds",x="BEDS", y = "log_price")+
  theme_classic()
# Change  automatically color by groups
bp <- ggplot(house_df1, aes(x=BEDS, y=log_price, fill=BEDS)) +
  geom_boxplot()+
  labs(title="Box Plot of Price by Beds",x="BEDS", y = "log_price")
bp + theme_classic()
```



This box plot compares the price of houses to the number of beds. As you can see, the prices of houses increase due to the increase in the number of beds.

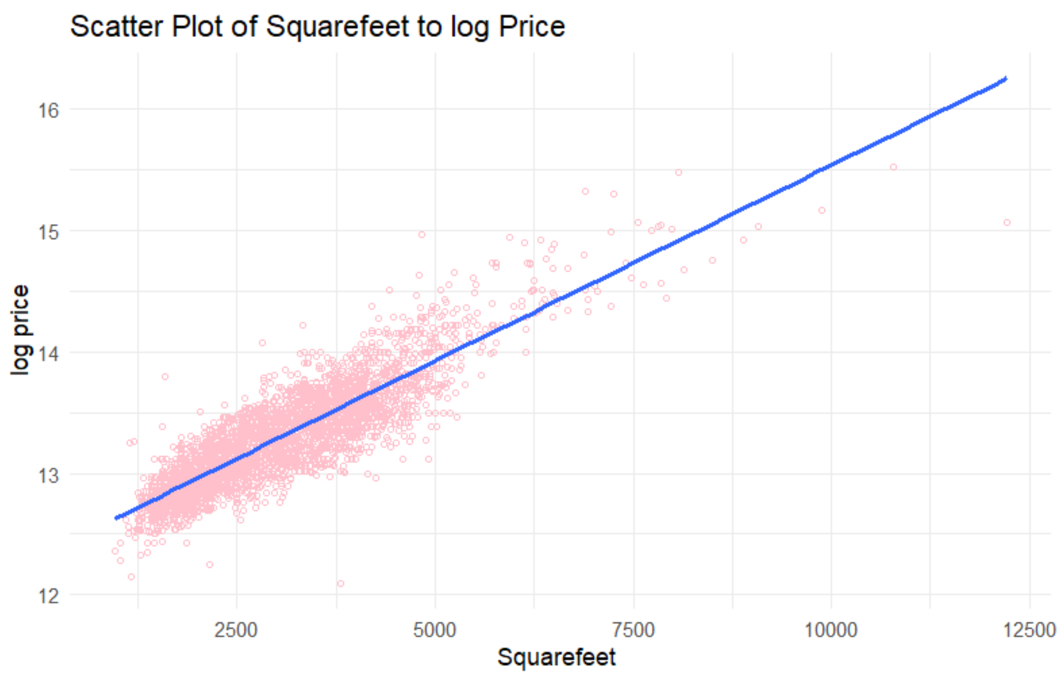
```
########################## price to house age ############################
# filter house age<=60
house_df3<- house_df[house_df$houseage<=60, ]
# filter the zipcode to 75007, 75033,75034, 75035 and 75036
desired_zips1 <- c("75033", "75034", "75035", "75036","75007")
house_df3 <- subset(house_df3, ZIP %in% desired_zips1)

# Change point shapes by the zip
ggplot(house_df3, aes(x=houseage, y=log_price, shape=ZIP)) +
  geom_point()
# Change point shapes and colors
ggplot(house_df3, aes(x=houseage, y=log_price, shape=ZIP, color=ZIP)) +
  geom_point()
# Change point shapes, colors and sizes
ggplot(house_df3, aes(x=houseage, y=log_price, color=ZIP, shape=ZIP,)) +
  geom_point(size=2) +
  labs(title="Sactter Plot of Houseage to log Price") +
  theme_minimal()
```



This scatter plot shows the differences based on prices, house age, and zip code. The houses in zip codes 75034 and 75036 are less than 20 years old. As for zip code 75035, the houses are between 0 to 35 years old. Houses in zip code 75007 aged between 30 to 50 years and prices are lower compared to other zip codes. Most new houses are built in zip codes 75034, 75035, and 75036. Houses are at higher prices in zip code 75034, but you can also find some in zip codes 75033 and 75035.

```
############################# scatter plot ################################
########################### price and living square feet ###################
ggplot(house_df,  aes(x=SQUAREFEET,y=log_price)) +
  geom_point(color = "PINK", size = 1, shape = 21) +
  labs(
    title = "Scatter Plot of Squarefeet to log Price",
    x = "Squarefeet ",
    y = "log price"
  ) +
  geom_smooth(method = "lm", se = FALSE)+
  theme_minimal()
########################### price to house age ############################
# filter house age<=60
house_df3<- house_df[house_df$houseage<=60, ]
# filter the zipcode to 75007, 75033,75034, 75035 and 75036
desired_zips1 <- c("75033", "75034", "75035", "75036","75007")
house_df3 <- subset(house_df3, ZIP %in% desired_zips1)
```



This scatter plot shows square feet to price. Most houses are built between 1250 to 5000 square feet. As the square feet of a house increase, the price also increases.

```
###################### heat map ############################################

house_df4 <- house_df[!is.na(house_df$SQUAREFEET), ]
house_df4<- house_df4[!is.na(house_df4$LOTSIZE_imputed), ]

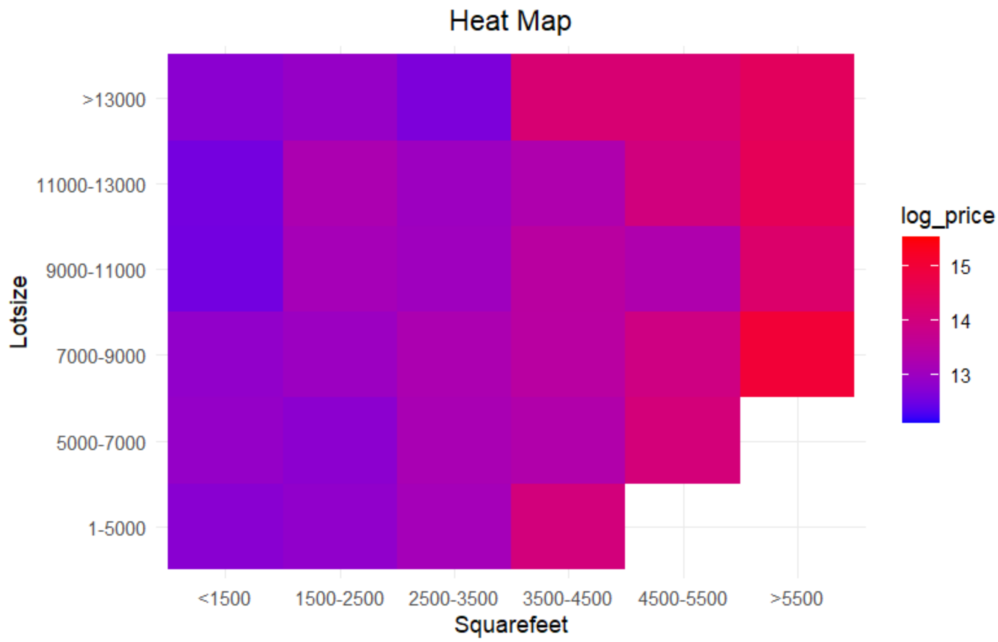
# there is 60 house with square feet> 6000
Sqr_breaks <- c(960,1500, 2500, 3500, 4500, 5500,12300)

# Define your custom bin labels
Sqr_labels <- c( "<1500","1500-2500", "2500-3500", "3500-4500","4500-5500",">5500")

# Create a new variable "bin_var" with custom bins and labels
house_df4$bin_var1 <- cut(house_df4$SQUAREFEET, breaks = Sqr_breaks, labels = Sqr_labels)
# there is 27 house with lot size >25000 sqrt
lot_breaks<- c(2170,5000,7000,9000,11000,13000,25300)
lot_labels<- c("1-5000","5000-7000","7000-9000","9000-11000","11000-13000",">13000")
house_df4$bin_var2 <- cut(house_df4$LOTSIZE_imputed, breaks = lot_breaks, labels = lot_labels)

ggplot(house_df4, aes(x = house_df4$bin_var1, y = house_df4$bin_var2  , fill = log_price))  +
  geom_tile() +
  scale_fill_gradient(low = "#0000FF", high = "#FF0000") +
  xlab("Squarefeet") +  # Add x-axis label
  ylab("Lotsize") +   # Add y-axis label

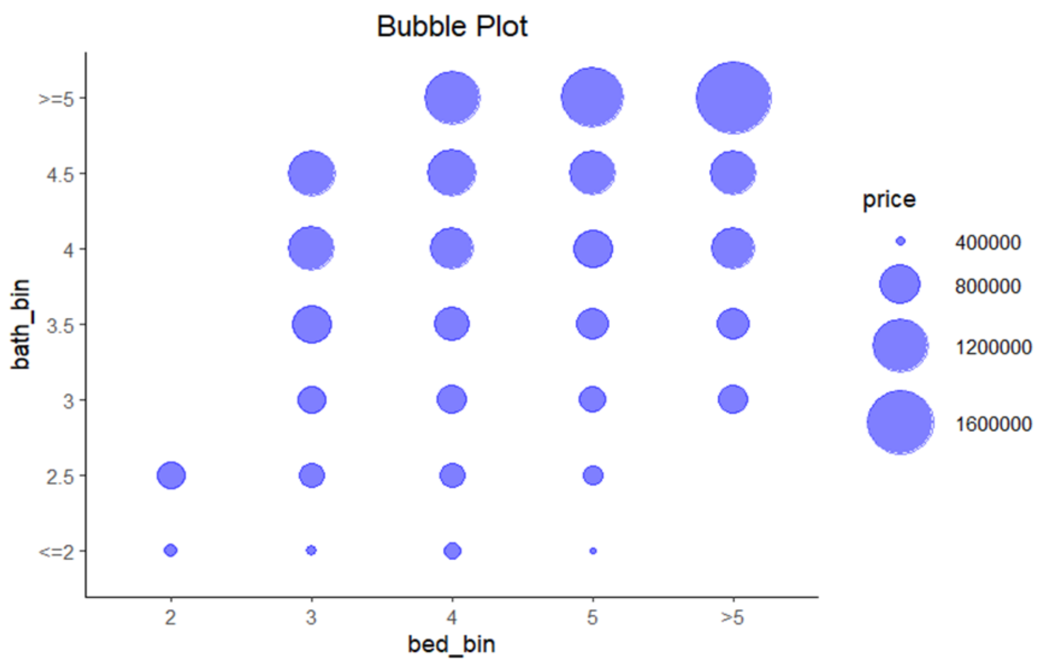
  theme_minimal()+ggtitle("Heat Map")+
  theme(plot.title = element_text(hjust = 0.5))
```



This heat map shows the relationship between lot size and square feet in houses. Houses with larger lot sizes and square feet are more expensive. The red color shows square feet to the price. Most houses are built between 1250 to 5000 square feet. As the square feet increase, and lot sizes increase, so does the price increases. Some people like to have a bigger house on a larger piece of land. But some prefer to have a bigger lot of land with a decent size of a house.

```
############################  Bubble Plot  ####################################
house_df5 <- house_df

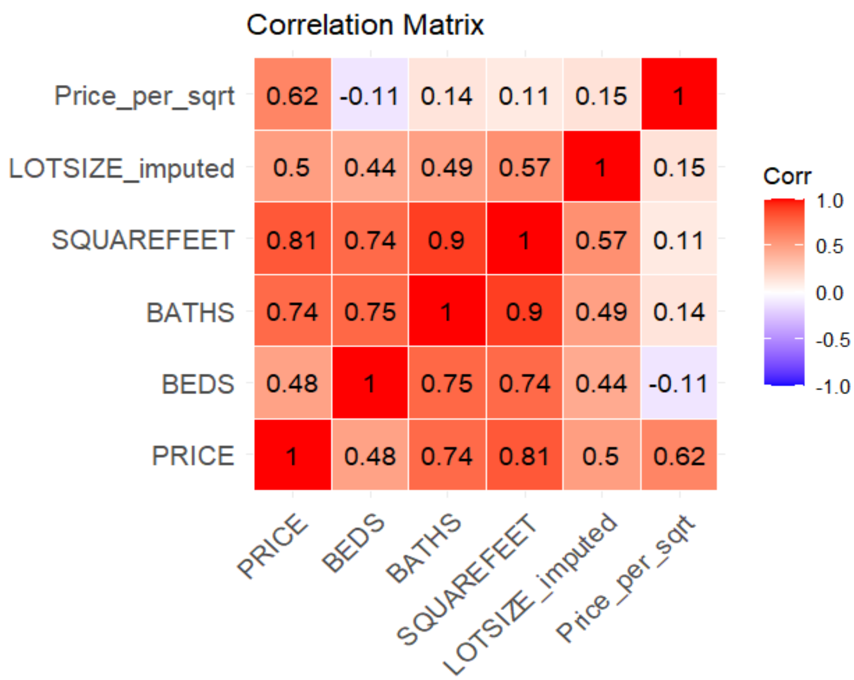
#set bin for beds
beds_breaks<- c(1,2,3,4,5,7)
beds_lab<- c("2","3","4","5",">5")
house_df5$bed_bin <- cut(house_df5$BEDS, breaks = beds_breaks, labels = beds_lab)
#set bin for bath
bath_breaks <- c(0,2,2.5,3,3.5,4,4.5,10)
bath_lab <- c("<=2","2.5","3","3.5","4","4.5",">=5")
house_df5$bath_bin <- cut(house_df5$BATHS, breaks = bath_breaks, labels = bath_lab)
agg_bbp<- aggregate(PRICE ~ bed_bin + bath_bin, data = house_df5, FUN = mean)
ggplot(agg_bbp, aes(x= bed_bin, y=bath_bin,
                     size=PRICE))+
  geom_point(alpha=0.5, color="#0000FF")+
  scale_size(range=c(1,15), name=" price") +
  theme_classic()+ggtitle("Bubble Plot")+
  theme(plot.title = element_text(hjust = 0.5))
```



This bubble plot shows the relationship between the number of beds, the number of bathrooms, and the price of houses. When the number of beds and baths increases, the price of the house increases.

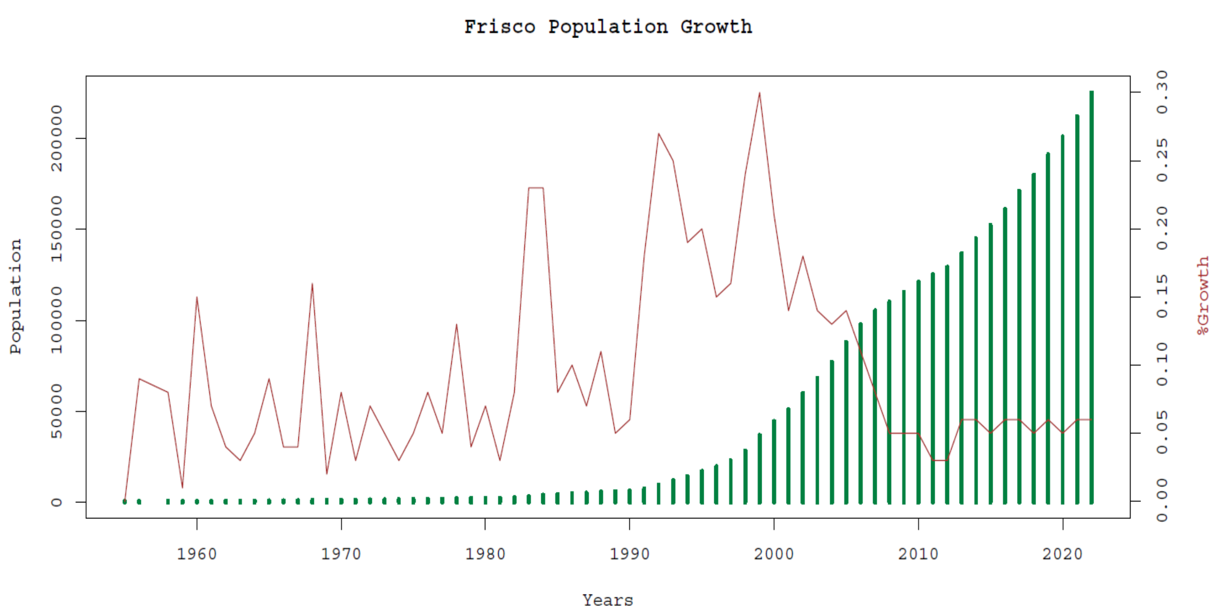
```
############################correlation matrix ################################
house_df6<- house_df[ ,c("PRICE","BEDS","BATHS","SQUAREFEET","LOTSIZE_imputed",
                         "Price_per_sqrt")]
# Create a ggplot2-based correlation plot

correlation_matrix <- cor(house_df6)
ggcorrplot(correlation_matrix,
           title ="Correlation Matrix",
           #type = "lower",
           lab = TRUE,
          # method = "circle",  # Use circles to represent correlations
           outline.color = "white",  # Set the color of circle outlines
           ggtheme = theme_minimal())  # Use a minimal theme
```



This matrix correlation shows the relationship of price, beds, baths, square feet, lot size imputed, and price per square foot. Positive correlations show that when one variable increases, the other variable tends to increase and are usually represented by warm colors, such as red or orange, for example, PRICE and SQUAREFEET show a positive correlation. Negative correlations show when one variable increases, the other variable tends to decrease and they are usually represented by cool colors, such as blue, for example, BEDS and Price_per_sqrt show a negative correlation.

Frisco's Population Growth by Percentage



YEAR	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014
Population	44592	50959	59929	68398	77186	87953	97817	105295	110047	115427	121078	125054	129186	136827	144822
PGrowth	0.21	0.14	0.18	0.14	0.13	0.14	0.11	0.08	0.05	0.05	0.05	0.03	0.03	0.06	0.06

YEAR	2015	2016	2017	2018	2019	2020	2021	2022
Population	152264	160917	171028	179860	191011	200727	212022	225060
PGrowth	0.05	0.06	0.06	0.05	0.06	0.05	0.06	0.06

According to the literature review, there is a positive correlation between population growth and the increase in house pricing, so the population growth of Frisco was added to our study. It is important to note that in 2004 the North Dallas Tollway extension to Frisco was built and as shown in the graph the red line represents the percentage growth it was when we saw a higher increase in population percentage.

As for the code used to visualize the population growth, we utilize the built-in R code, to employ the techniques learned in *R graphics* by Murrell, Pau.

```
par(family = "mono")

par(mar = c(5,4,4,4) + 0.3)

plot(FP$YEAR, FP$Population,
     main = "Frisco Population Growth",
     xlab = "Years",
     ylab = "Population",
     type = "h",
     col = "blue")

par(new = TRUE)
plot(FP$YEAR, FP$PGrowth,
     type = "l",
     col = "red",
     axes = FALSE, xlab = "")

axis(side = 4, at = pretty(range(FP$PGrowth)))
mtext("%Growth", side = 4, line = 3)


# Add line for population growth percentage on the right y-axis
par(new = TRUE)
plot(FriscoP$YEAR, FriscoP$GrowthPercentage,
     type = "l",
     col = "red",
     axes = FALSE,
     xlab = "",
     ylab = "")

# Add y-axis label for population growth percentage
axis(side = 4, at = pretty(FriscoP$GrowthPercentage),
     labels = pretty(FriscoP$GrowthPercentage))
mtext("Population Growth Percentage", side = 4, line = 3)
```

Shiny App

Finally, to allow people to interact with the housing data in an intuitive format, an application was built using Shiny. As many apps do, this went through a few versions before reaching the final build. In an effort to involve some of the concepts used in class, research was conducted to see how concepts used in the “flexdashboard” library could be integrated into a shiny app. While flexdashboard itself could not be used due to limitations in rMarkdown's integration of shiny components, other options can produce similar results. The first option discovered was Quarto's new developments in a type of document called dashboard. Similar to flexdashboard, quarto's dashboard allows the user to control elements of the user interface to create different layouts for their pages. This led to the first iteration of the housing sales app.

```
---
title: "tester"
format: dashboard
server: shiny
---

```{r}
#| context: setup
library(shiny)library(ggplot2)
library(leaflet)

house <- house <- read.csv("~/Desktop/app/HouseDemo/Copy of house_data.csv")
col <-
c("PRICE","ADDRESS","CITY","STATE.OR.PROVINCE","ZIP.OR.POSTAL.CODE","SOLD.DATE","BE
DS","BATHS","SQUARE.FEET","LOT.SIZE","YEAR.BUILT","LATITUDE","LONGITUDE","URL..SEE.
https...www.redfin.com.buy.a.home.comparative.market.analysis.FOR.INFO.ON.PRICING."
)
house <- house[col]
```

{.sidebar}

```{r}
radioButtons("price_range", "Select Price Range",
                   choices = c("$100,000-$200,000",
                               "$200,000-$300,000",
                               "$300,000-$400,000",
                               "$400,000-$500,000",
                               "$500,000-$600,000",
                               "$600,000-$700,000",
                               "$700,000 and Above"),
                   selected = "$100,000-$200,000")
```

```{r}
sliderInput("bedroom_range", "Select Number of Bedrooms",
                  min = min(house$BEDS),
```
```

````
                  max = max(house$BEDS),
                  value = c(min(house$BEDS), max(house$BEDS)),
                  step = 1)
```

```{r}
sliderInput("bathroom_range", "Select Number of Bathrooms",
                  min = min(house$BATHS),
                  max = max(house$BATHS),
                  value = c(min(house$BATHS), max(house$BATHS)),
                  step = 0.5)
```

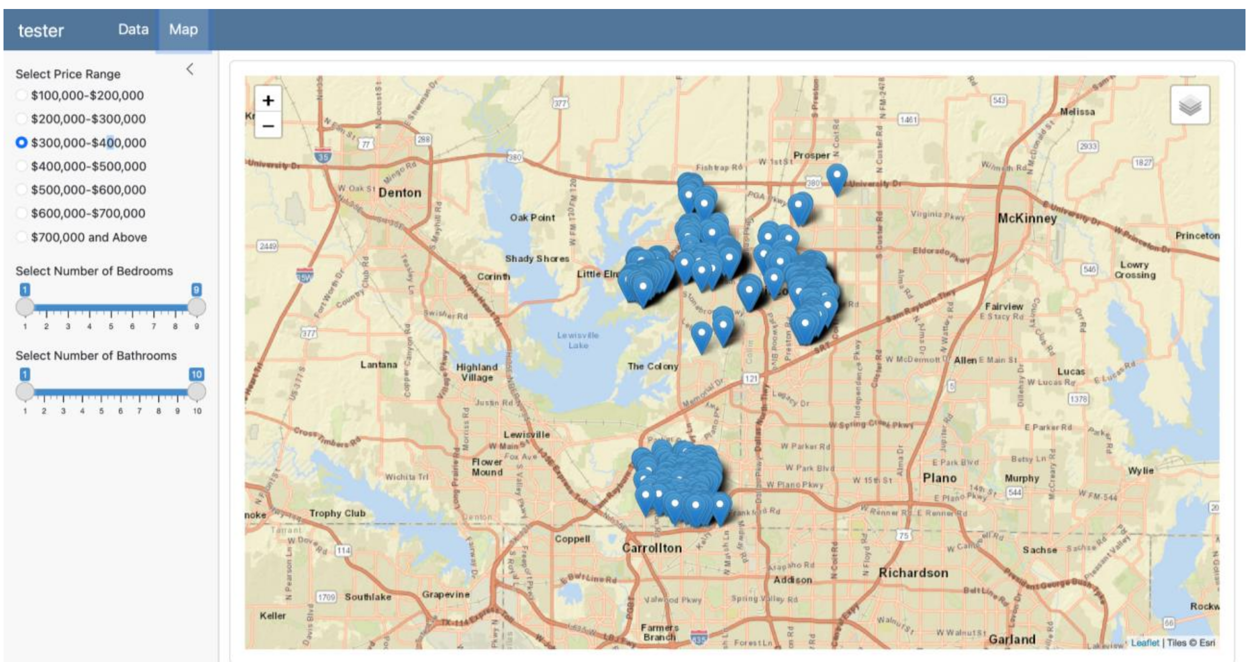
# Data

```{r}
dataTableOutput("filtered_table")
```

# Map
```{r}
leafletOutput("map")
```

```{r}
#| context: server
filtered_data <- reactive({
    price_range <- switch(input$price_range,
                "$100,000-$200,000" = house$PRICE >= 100000 & house$PRICE < 200000,
                "$200,000-$300,000" = house$PRICE >= 200000 & house$PRICE < 300000,
                "$300,000-$400,000" = house$PRICE >= 300000 & house$PRICE < 400000,
                "$400,000-$500,000" = house$PRICE >= 400000 & house$PRICE < 500000,
                "$500,000-$600,000" = house$PRICE >= 500000 & house$PRICE < 600000,
                "$600,000-$700,000" = house$PRICE >= 600000 & house$PRICE < 700000,
                "$700,000 and above"= house$PRICE >= 500000)
    subset(house,
            price_range&
            BEDS  >= input$bedroom_range[1] & BEDS <= input$bedroom_range[2] &
            BATHS >= input$bathroom_range[1] & BATHS <= input$bathroom_range[2]
    )
  })

output$filtered_table <- renderDataTable({
    filtered_data()
  })

output$map <- renderLeaflet({
````

```
  houseMap <- leaflet(data = filtered_data)

  esri <- grep("^Esri", providers, value = TRUE)

for (provider in esri) {
  houseMap <- houseMap %>% addProviderTiles(provider, group = provider)
}

houseMap %>%
  addLayersControl(baseGroups = names(esri),
                   options = layersControlOptions(collapsed = TRUE)) %>%
  addMarkers(data = filtered_data() %>% as.data.frame(),
             lat = ~LATITUDE, lng = ~LONGITUDE,
             popup = ~paste("Address: ", ADDRESS,
                            "<br>Price: $", PRICE,
                            "<br>Bed/Bath: ",BEDS,"/",BATHS,
                            "<br>Date of Sale: ", SOLD.DATE,
                            "<br>House on <a href='",
URL..SEE.https...www.redfin.com.buy.a.home.comparative.market.analysis.FOR.INFO.ON.
PRICING., "' target='_blank'>Redfin</a>"))
})
```



This version first shows the user a table of all of the data in a table and the user can scroll through the houses in the database based on their section of variables. Secondly, if the user wanted to see a map view of the houses, they could select the map in the navigation bar at the top and see a map view of the data. While both of these elements were carried throughout future iterations, this build could not be deployed to GitHub because the shiny app was not supported by a backend. The quarto dashboard format does not currently support uploading a shiny app to Posit's backend for Shiny. This was likely due to how new the dashboard feature is for Quarto. It is a part of development software that is currently in early access development. Released in October of 2023, Quarto Dashboard is a very new software. Quarto stated that a shiny backend was in development but had not been implemented yet. This forced development back to the familiar app development using R and publishing the app to shinyapps.io.

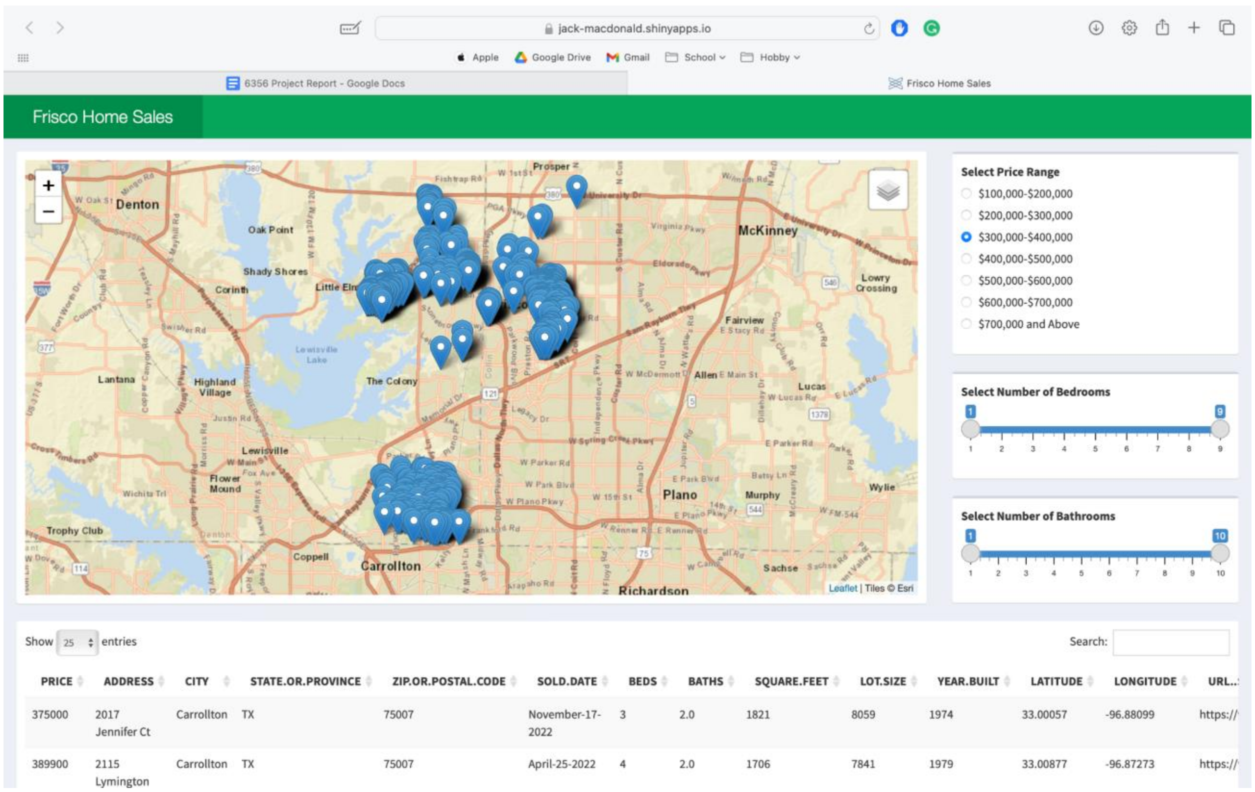
After an easy conversion to the standard methods, we felt that the app was lacking in quality and originality. The dashboard format offered UI elements that were more engaging for the user and this prompted more research. We found a solution in a library called "shinydashboard". This offers the same element control found in flexdashboard and Quarto Dashboard but formats it into the "ui" and "server" systems that shinyapps is used to. This led to the final implementation of the app used for the project.

```
library(shiny)
library(shinydashboard)
library(ggplot2)
library(leaflet)

house <- house <- read.csv("Copy of house_data.csv")

col <- c("PRICE","ADDRESS",
         "CITY","STATE.OR.PROVINCE",
         "ZIP.OR.POSTAL.CODE","SOLD.DATE",
         "BEDS","BATHS","SQUARE.FEET",
         "LOT.SIZE","YEAR.BUILT","LATITUDE",

"LONGITUDE","URL..SEE.https...www.redfin.com.buy.a.home.comparative.market.analysis
.FOR.INFO.ON.PRICING.")
house <- house[col]
```

```
server <- function(input,output){
  filtered_data <- reactive({
    price_range <- switch(input$price_range,
                 "$100,000-$200,000"  = house$PRICE >= 100000 & house$PRICE < 200000,
                 "$200,000-$300,000"  = house$PRICE >= 200000 & house$PRICE < 300000,
                 "$300,000-$400,000"  = house$PRICE >= 300000 & house$PRICE < 400000,
                 "$400,000-$500,000"  = house$PRICE >= 400000 & house$PRICE < 500000,
                 "$500,000-$600,000"  = house$PRICE >= 500000 & house$PRICE < 600000,
                 "$600,000-$700,000"  = house$PRICE >= 600000 & house$PRICE < 700000,
                 "$700,000 and Above" = house$PRICE >= 500000)

subset(house,
           price_range&
             BEDS  >= input$bedroom_range[1] & BEDS <= input$bedroom_range[2] &
             BATHS >= input$bathroom_range[1] & BATHS <= input$bathroom_range[2]
    )
  })

  output$filtered_table <- renderDataTable({
    filtered_data()
  })

  output$map <- renderLeaflet({
    houseMap <- leaflet(data = filtered_data)

    esri <- grep("^Esri", providers, value = TRUE)

    for (provider in esri) {
      houseMap <- houseMap %>% addProviderTiles(provider, group = provider)
    }

    houseMap %>%
      addLayersControl(baseGroups = names(esri),
                       options = layersControlOptions(collapsed = TRUE)) %>%
      addMarkers(data = filtered_data() %>% as.data.frame(),
                 lat = ~LATITUDE, lng = ~LONGITUDE,
                 popup = ~paste("Address: ", ADDRESS,
                                "<br>Price: $", PRICE,
                                "<br>Bed/Bath: ",BEDS,"/",BATHS,
                                "<br>Date of Sale: ", SOLD.DATE,
                     "<br>House on <a href='", URL, "'target='_blank'>Redfin</a>"))

  })
}

header <- dashboardHeader(
```

```
  title = "Frisco Home Sales"
)

body <- dashboardBody(
  fluidRow(
    column(width = 9,
           box(width = NULL, solidHeader = T,
               leafletOutput("map", height = 500)
           )
    ),
    column(width = 3,
      box(width = NULL,
          radioButtons("price_range", "Select Price Range",
                       choices = c("$100,000-$200,000",
                                   "$200,000-$300,000",
                                   "$300,000-$400,000",
                                   "$400,000-$500,000",
                                   "$500,000-$600,000",
                                   "$600,000-$700,000",
                                   "$700,000 and Above"),
                       selected =  "$100,000-$200,000"),
      ),
      box(width = NULL,
          sliderInput("bedroom_range", "Select Number of Bedrooms",
                      min = min(house$BEDS),
                      max = max(house$BEDS),
                      value = c(min(house$BEDS), max(house$BEDS)),
                      step = 1),
      ),
      box(width = NULL,
          sliderInput("bathroom_range", "Select Number of Bathrooms",
                      min = min(house$BATHS),
                      max = max(house$BATHS),
                      value = c(min(house$BATHS), max(house$BATHS)),
                      step = 0.5)
      )
    )
  ),
  fluidRow(
    column(width = 12,
      box(width = NULL, solidHeader = T, style = 'overflow-x: scroll',
          dataTableOutput("filtered_table")
      )
    )
  )
)
```

```
ui <- dashboardPage(
  skin = "green",
  header,
  dashboardSidebar(disable = T),
  body
)

shinyApp(ui, server)
```



This final version behaves similarly to the alpha build where the user can parse through the homes based on their desired options, displays a map view of those houses, and then displays them in a table below the selections. The three variables, PRICE, BEDS, and BATHS, were chosen to parse the data because these are the attributes people are looking for when buying a home, along with location. While other variables are important, like square footage and the year it was built, price, bedrooms, and bathrooms are crucial to buying a home. The library “leaflet” was used to create the mapping portion of the app. It used the included variables for longitude and latitude for the houses to create markers on the map. When a marker is clicked on, it displays some information about the house Including the address and a link to the listing on Redfin, as well as other information about the home. The table is a Data Table in shiny that allows the user to sort the variables and search within the table as well. In the future, the app could be developed to include a connection to a database that can be updated with new sales, or changed to include houses that are currently on the market, instead of previously sold houses. Additionally, the ability for the user to select which variables they want to filter by would broaden the utility of the interface. Finally, the ability to select houses from the table to be displayed on the map would be a great addition the application.

Conclusion

Located within the Dallas- Fort Worth metropolitan area, Frisco is a city growing rapidly, which embodies community involvement and rapid expansion. It is known for thriving sports culture, family friendly neighborhoods and high-quality schools. In 2023, due to the high interest rates and a short-term decline in housing price, potential may withdraw from the market. Both buyers and sellers are locked in to some extent, resulting in low volume in the house market. Mortgage rates are likely to fall at some point in 2024, housing market inventory will increase, and buyers whose demand has been dampened by high interest rates may return to the market.

References:

Clark, Cullum. "Frisco Pro-Growth Housing Policies." *Dallas News,* November 11, 2023, https://www.dallasnews.com/opinion/commentary/2023/11/11/frisco-pro-growth-housing-policies/.

VictorSteffen.com, Brokered by EXP Realty, “Frisco Real Estate Investment”, https://victorsteffen.com/texas-real-estate-investment-property/frisco/

Frisco Texas, Demographics, Frisco Texas website, accessed November 28, 2023, https://www.friscotexas.gov/1454/Demographics

Frisco Texas, Demographics, “Frisco at A Glance 2023”, accessed November 29, 2023, https://www.friscotexas.gov/DocumentCenter/View/4900/2023_At-A-Glance-PDF?bidId=

RedFIN, https://www.redfin.com

Patrick Villanova, CEPF, February 11, 2023, “Safetest Cities in America – 2023 Edition”, https://smartasset.com/data-studies/safest-cities-in-america-2023

APAS intext citation: Yiu, C. Y. "Are Central Banks’ Monetary Policies the Future of

Housing Affordability Solutions." Urban Science 7, no. 1 (2023): 18.

https://doi.org/10.3390/urbansci7010018

Murrell, Paul. 2016. R graphics. CRC Press.

Wickham, Hadley. 2020. Mastering Shiny. O'Reilly

Wickham, Hadley. 2016. Ggplot2: elegant graphics for data analysis. Springer

Bhat, M. R., Jiao, J., & Azimian, A. (2023). "The impact of COVID-19 on home value in major Texas cities." *International Journal of Housing Markets and Analysis,* 16(3), 616-627.

Savva, C. (2018). "Factors Affecting Housing Prices: International Evidence." *Cyprus Economic Policy Review,* 12, 87-96.

Choudhury, A. (2012). Effect of tax rate on zone-dependent housing value. *Journal of Economics and Economic Education Research, 13,* 111.

Cairo, Alberto. 2012. *The Functional Art: An introduction to information graphics and visualization*. New Riders (Cairo, 2012).