# MIS503 - Final Project

## Pierce, Michaela

### Zillow Home Value Index Analysis

library(readr)  
library(tidyr)  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(ggplot2)

### Wake County Home Sales

**a. What have been the overall trends in Wake County Home Values?** The home values have increased over the years.

**b. There were dips in home values in the past 10 years. What years did these occur?**  
2013 - 2021

**c. Based on the analysis, where would be the least expensive area to purchase home? Most expensive area?**  
Least Expensive: Zebulon ; Most expensive: Cary

**d. What has happened to the overall property values in Apex and Cary in 2023?**  
Apex and Cary’s overall property values did not increase in 2023.

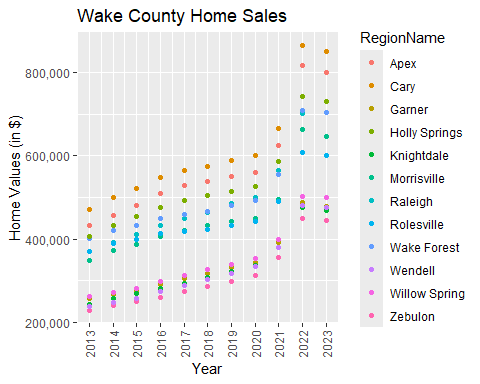
# load data  
Sales <- read\_csv("SingleFamilyResidenceSales.csv")

## Rows: 22275 Columns: 287  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (4): RegionName, State, Metro, CountyName  
## dbl (283): RegionID, 1/31/2000, 2/29/2000, 3/31/2000, 4/30/2000, 5/31/2000, ...  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

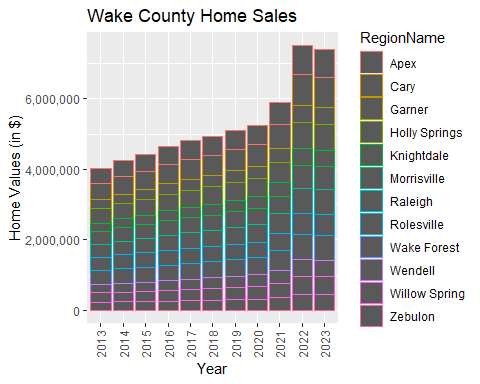
# Create the WakeCountySales tibble  
WakeCountySales <- Sales %>%  
 filter(State == "NC" & CountyName == "Wake County") %>%  
 select(RegionName, State, CountyName, Metro,  
 `5/31/2013` = `5/31/2013`,  
 `5/31/2014` = `5/31/2014`,  
 `5/31/2015` = `5/31/2015`,  
 `5/31/2016` = `5/31/2016`,  
 `5/31/2017` = `5/31/2017`,  
 `5/31/2018` = `5/31/2018`,  
 `5/31/2019` = `5/31/2019`,  
 `5/31/2020` = `5/31/2020`,  
 `5/31/2021` = `5/31/2021`,  
 `5/31/2022` = `5/31/2022`,  
 `5/31/2023` = `5/31/2023`)  
  
# Rename date columns to only include year  
colnames(WakeCountySales)[5:16] <- gsub("5/31/", "", colnames(WakeCountySales)[5:16])  
  
# View the tibble  
# View(WakeCountySales)

# Tidy the data  
TidyWakeCountySales <- WakeCountySales %>%  
 pivot\_longer(cols = matches("^\\d{4}$"),   
 names\_to = "YR",   
 values\_to = "ZHVI")   
  
# View the data  
# View(TidyWakeCountySales)

# Convert YR to numeric  
TidyWakeCountySales$YR <- as.numeric(as.character(TidyWakeCountySales$YR))  
  
# Create scatter plot  
scatter\_plot <- ggplot(TidyWakeCountySales, aes(x = YR, y = ZHVI, color = RegionName)) +  
 geom\_point() +  
 scale\_x\_continuous(name = "Year", breaks = unique(TidyWakeCountySales$YR)) + # Show every year  
 scale\_y\_continuous(name = "Home Values (in $)", labels = scales::comma) + # Add commas to y-axis labels  
 labs(title = "Wake County Home Sales",  
 x = "Year",  
 y = "Home Value (ZHVI)") +  
 theme(axis.text.x = element\_text(angle = 90, vjust = 0.5))   
  
# Display the scatter plot  
print(scatter\_plot)



# Create stacked bar graph  
stacked\_bar <- ggplot(TidyWakeCountySales, aes(x = factor(YR), y = ZHVI, color = RegionName)) +  
 geom\_col() +  
 scale\_x\_discrete(name = "Year", labels = as.character(unique(TidyWakeCountySales$YR))) + # show all years on x-axis  
 scale\_y\_continuous(name = "Home Values (in $)", labels = scales::comma) + # Add commas to y-axis labels  
 labs(title = "Wake County Home Sales",  
 y = "Total Home Value (ZHVI)") + # Corrected the y-axis label  
 theme(axis.text.x = element\_text(angle = 90, vjust = 0.5))  
   
  
# Display stacked bar graph  
print(stacked\_bar)



### NC Rental Market

**a. What has been the overall trend in the rental market around the state? Are there any cities that have not followed this trend?**  
The rental market has increased over the years. Fayetteville remained consistent until 2021 where it spiked up.

**b. Where is the most expensive city to rent in? Least expensive?**  
Most Expensive: Asheville ; Least Expensive: Fayetteville

**c. You are trying to decide between Wilmington and Asheville. Which market has the lowest rent?**  
Wilmington has the lowest rent.

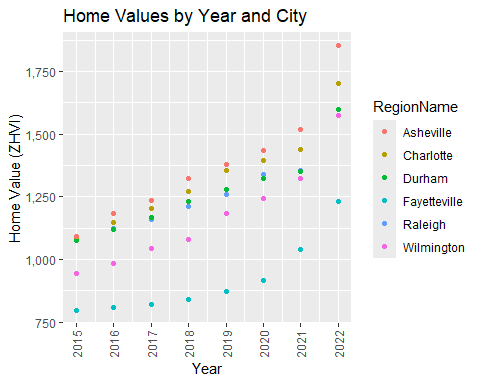
# load data  
Rentals <- read\_csv("SingleFamilyResidenceRental.csv")

## Rows: 3503 Columns: 107  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (4): RegionName, State, Metro, CountyName  
## dbl (103): RegionID, 1/31/2015, 2/28/2015, 3/31/2015, 4/30/2015, 5/31/2015, ...  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

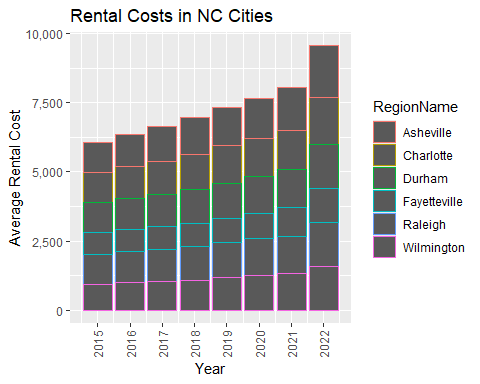
# Asheville, Charlotte, Durham, Fayetteville, Raleigh, and Wilmington  
NC\_RentalCities <- c("Asheville", "Charlotte", "Durham", "Fayetteville", "Raleigh", "Wilmington")  
  
  
# Filter data  
Rentals <- Rentals %>%  
 filter(RegionName %in% NC\_RentalCities,  
 State == "NC") %>%  
 select(RegionName, State, matches("^1/31/20\\d{2}$"))  
  
# Rename date columns to only include the year  
colnames(Rentals)[3:10] <- gsub("1/31/", "", colnames(Rentals)[3:10])  
  
# View the tibble  
# View(Rentals)

# Tidy the data  
Tidy\_Rentals <- Rentals %>%  
 pivot\_longer(cols = matches("^20\\d{2}$"),  
 names\_to = "YR",  
 values\_to = "ZHVI")  
  
# View the tibble  
# View(Tidy\_Rentals)

# Create scatter plot  
scatter\_plot <- ggplot(Tidy\_Rentals, aes(x = as.numeric(YR), y = ZHVI, color = RegionName)) +  
 geom\_point() +  
 scale\_x\_continuous(name = "Year", breaks = unique(as.numeric(Tidy\_Rentals$YR))) +   
 scale\_y\_continuous(name = "Home Value (ZHVI)", labels = scales::comma) +  
 labs(title = "Home Values by Year and City",  
 x = "Year",  
 y = "Average Rentals Cost") +  
 theme(axis.text.x = element\_text(angle = 90, vjust = 0.5))   
  
# Display the scatter plot  
print(scatter\_plot)



# Create stacked bar graph  
stacked\_bar <- ggplot(Tidy\_Rentals, aes(x = as.numeric(YR), y = ZHVI, color = RegionName)) +  
 geom\_col() +  
 scale\_x\_continuous(name = "Year", breaks = unique(as.numeric(Tidy\_Rentals$YR))) +   
 scale\_y\_continuous(name = "Average Rental Cost", labels = scales::comma) +  
 labs(title = "Rental Costs in NC Cities",  
 x = "Year",  
 y = "Average Rental Cost") +  
 theme(axis.text.x = element\_text(angle = 90, vjust = 0.5))   
  
# Display the stacked bar graph  
print(stacked\_bar)



### Home Values in Select Markets

**a. According to the results, which market has the lowest median price (represented as horizontal bar in box plot)?**  
Charlotte-Concord-Gastonia

**b. The violin plot will show density meaning the wider the plot is, the more observations occur within that area. Which market has the most density around the median value of homes?**  
Wilmington

**c. The box plot will also show outliers in the various markets. Which metro area had the largest outlier (i.e., the highest value home sold in the past 10 years)?**  
Raleigh-Cary and Asheville

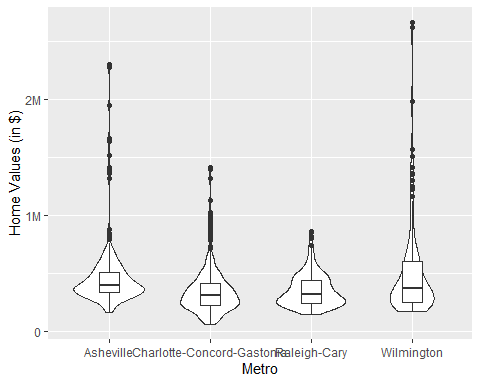
# Asheville, Charlotte-Concord-Gastonia, Raleigh-Cary and Wilmington  
NCHomeSales\_Cities <- c("Asheville", "Charlotte-Concord-Gastonia", "Raleigh-Cary", "Wilmington")  
  
# Create the NCHomeSales tibble  
NCHomeSales <- Sales %>%  
 filter(State == "NC" & Metro %in% NCHomeSales\_Cities) %>%  
 select(RegionName, State, Metro,  
 `5/31/2013` = `5/31/2013`,  
 `5/31/2014` = `5/31/2014`,  
 `5/31/2015` = `5/31/2015`,  
 `5/31/2016` = `5/31/2016`,  
 `5/31/2017` = `5/31/2017`,  
 `5/31/2018` = `5/31/2018`,  
 `5/31/2019` = `5/31/2019`,  
 `5/31/2020` = `5/31/2020`,  
 `5/31/2021` = `5/31/2021`,  
 `5/31/2022` = `5/31/2022`,  
 `5/31/2023` = `5/31/2023`)  
  
# Rename date columns to only include year  
colnames(NCHomeSales)[4:ncol(NCHomeSales)] <- gsub("5/31/", "", colnames(NCHomeSales)[4:ncol(NCHomeSales)])  
  
# View the tibble  
# View(NCHomeSales)

# Tidy the data.  
TidyNCHomeSales2 <- NCHomeSales %>%  
 gather(key = "Year", value = "ZHVI", -RegionName, -State, -Metro) %>%  
 mutate(Year = gsub("\\.", "/", Year))   
  
# Group by Metro  
NCHomeSales\_grouped <- TidyNCHomeSales2 %>%  
 group\_by(Metro)

ggplot(NCHomeSales\_grouped, aes(x = Metro, y = ZHVI)) +  
 geom\_violin(width = 0.7) + # Adjust the width of the violin plot  
 geom\_boxplot(width = 0.2) + # Adjust the width of the box plot  
 scale\_y\_continuous(breaks = c(0, 1000000, 2000000), labels = c("0", "1M", "2M")) +  
 labs(x = "Metro", y = "Home Values (in $)") +  
 theme(axis.text.x = element\_text(angle = 0))

## Warning: Removed 18 rows containing non-finite outside the scale range  
## (`stat\_ydensity()`).

## Warning: Removed 18 rows containing non-finite outside the scale range  
## (`stat\_boxplot()`).



# Relocation Home Value Comparison

**a. Based on your analysis, which city’s housing is most affordable? Least affordable?**  
Most affordable city: Houston ; Least affordable city: New York

**b. Which cities saw the largest change in prices over the past 5 years? Which city has remained more consistent (i.e., no huge swings up or down in home values)?**  
City with largest change in prices over the past 5 years: Denver ; City that has remained consistent over the past 5 years: Chicago

**c. Which cities saw a decline in value during 2023 and which cities remained consistent?**  
Denver saw a decline in home values during 2023.New York had a very small decline. ; Chicago and Houston’s home values remained consistent during 2023.

# Filter data  
NationalHomeSales <- Sales %>%  
 filter(RegionName %in% c("Chicago", "Denver", "Houston", "New York"),  
 CountyName %in% c("Cook County", "Denver County", "Harris County", "Queens County")) %>%  
 select(RegionName, State, CountyName, Metro,  
 `5/31/2013`, `5/31/2014`,   
 `5/31/2015`, `5/31/2016`,   
 `5/31/2017`, `5/31/2018`,   
 `5/31/2019`, `5/31/2020`,   
 `5/31/2021`, `5/31/2022`,   
 `5/31/2023`)  
  
# Rename the date columns to only include the year  
colnames(NationalHomeSales)[5:ncol(NationalHomeSales)] <- gsub("5/31/", "", colnames(NationalHomeSales)[5:ncol(NationalHomeSales)])  
  
# View the tibble  
# View(NationalHomeSales)

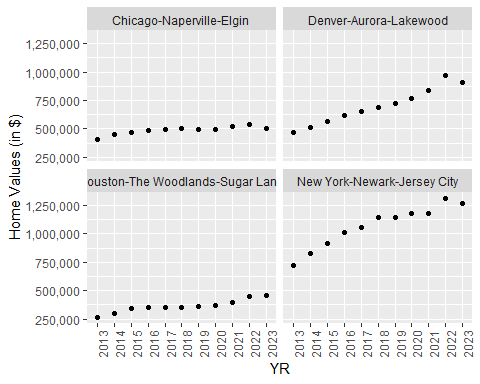
# Tidy the data  
NationalHomeSales\_tidy <- NationalHomeSales %>%  
 pivot\_longer(cols = `2013`:`2023`, names\_to = "YR", values\_to = "ZHVI")  
  
# View the data  
# View(NationalHomeSales\_tidy)

library(scales)

##   
## Attaching package: 'scales'

## The following object is masked from 'package:readr':  
##   
## col\_factor

# Plotting scatterplots for each city separately using facets  
ggplot(NationalHomeSales\_tidy, aes(x = YR, y = ZHVI)) +  
 geom\_point() +  
 facet\_wrap(~Metro) +  
 labs(x = "YR", y = "Home Values (in $)") +  
 scale\_y\_continuous(labels = label\_comma()) + # Add commas to y-axis labels  
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))



### Future Home Values

**a. Which is the only city that is projected to have a decrease in home values in the next 3 months?** No city projected to have a decrease in home values in the next 3 months

**b. If you are only concerned about the largest home value increase (by percentage) in the next 12 months, which city would you choose to relocate to?** Chicago, IL

Projections <- read\_csv("Projections.csv")

## Rows: 895 Columns: 9  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (3): RegionName, RegionType, StateName  
## dbl (5): RegionID, SizeRank, 2024-04-30, 2024-06-30, 2025-03-31  
## date (1): BaseDate  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

# Filter the data for the specified cities and columns  
FutureHomeValues <- Projections %>%  
 filter(RegionName %in% c("Chicago, IL", "Denver, CO", "Houston, TX", "New York, NY")) %>%  
 select(RegionName, `2024-04-30`, `2024-06-30`, `2025-03-31`)  
  
# Rename the columns  
colnames(FutureHomeValues)[2:4] <- c("Current", "ThreeMonths", "TwelveMonths")  
  
# View the tibble  
# View(FutureHomeValues)

# Tidy the data  
FutureHomeValues\_tidy <- FutureHomeValues %>%  
 pivot\_longer(cols = -RegionName, names\_to = "Time", values\_to = "PercentageChange")  
  
# Plotting scatter plot with city as color  
ggplot(FutureHomeValues\_tidy, aes(x = Time, y = PercentageChange, color = RegionName)) +  
 geom\_point() +  
 labs(x = "Time", y = "Percentage Change") +  
 theme(axis.text.x = element\_text(angle = 0))

