Jared Yu

STA 141A

HW 2

1) One of the major errors that were worth correcting right away were the differences in the spelling for the county names. Certain observations had a lowercase for the word 'county' in their name, so these had to be changed to all uppercase. Another error with the county variable was with certain observations being mislabeled 'San Franciscoe…' rather than 'San Francisco County.'

   The zip codes were tested by checking the range with the maximum and minimum. It is apparent that all the observations are within the mandatory '9----' format, so no errors were found. Additionally, it was double checked that all of the zip codes were a length of 5 digits.

   After examining a boxplot and histogram of the prices, it seems that there were some extreme values that exist as outliers beyond the rest of the data as highly expensive homes. After sorting the homes by prices, it seems that the greatest observation is not too distant compared to the other homes, despite being $7 million. After searching some of the homes on Google, it was determined that some of the prices were accurate. There was however a home that was valued quite high that didn't match with the link on the real estate website. Therefore, one of the homes with a price at $5.4 million had the price set to NA.

   For the bedroom variable, there was an observation that seemed significantly larger than the rest at a value of 28. After searching for this variable online, it became apparent that it was an apartment complex, which would logically have so many bedrooms. The other high number bedroom observations had similar characteristics, one was an 8-bedroom multi-family home, and the other was a quadruplex which had 12-bedrooms. Therefore, these observations are considered normal and part of the data.

   For the lot size variable, a histogram and boxplot were done to look at the general shape of the data. It seems that there was one extreme outlier with a lot size significantly greater than the rest. After checking the observation online, the address didn't appear quite nearly so large, so the value was set to NA. The same was done for another two more observations.

   For the building size variable, a boxplot and histogram were created to examine the shape and range of the observations. Most observations were shown to have the correct value shown, however one observation was shown to apparently have a '1' added to its value, inflating the observation's effect on others in the graph. Due to this likely error, the value was set to NA.

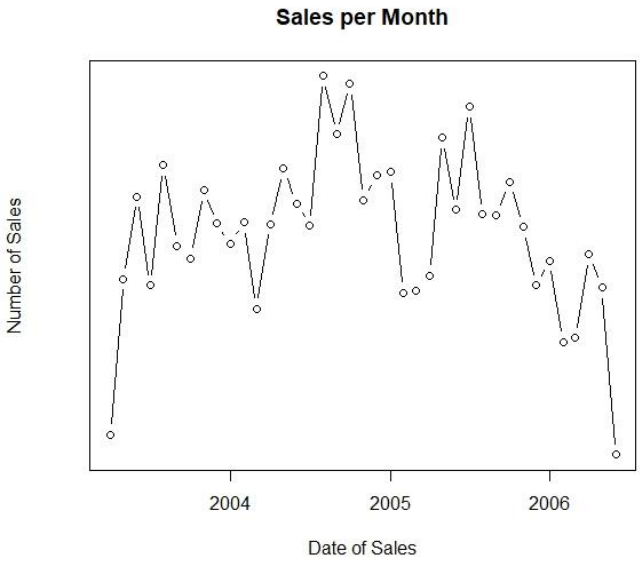
   To examine the 'year' variable, the number of NA's was checked, and apparently 17% of the observations don't have a year available. Additionally, after sorting the variable there were several variables which weren't possible as a year for the date of construction for a home. Some values were too large, others were too small. Therefore, any observation within a certain range such as less than 1800 or greater than 2020 had their values set to NA.

For the ‘date’ variable, observations such as the year, month, and day were checked to see if any of them were out of any possible range of a real date variable. When plotting the data, it was apparent that homes weren’t often purchased around the beginning of each year.

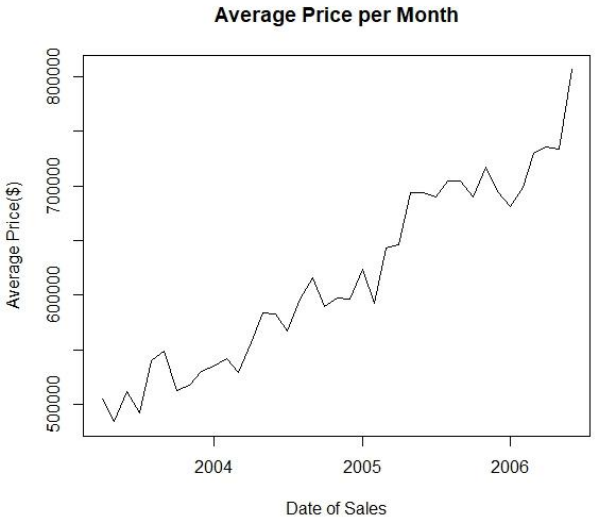
The ‘long’ variable which represents the longitude of the homes has an obvious outlier in the boxplot. The observation was shown to have both longitude and latitude at 0, so a change was made where any longitude greater than -120 was set to NA. The same situation was shown for ‘lat’, so the latitude went through a similar change where any latitude less than 30 was set to NA.

When dealing with data types, two changes that I felt were practical were changing the ‘zip’ to a factor level. This is due to zip-codes being values where arithmetic operations aren’t useful. Another change was switching ‘year’ to a numeric type, since these are data where arithmetic operations can be useful.

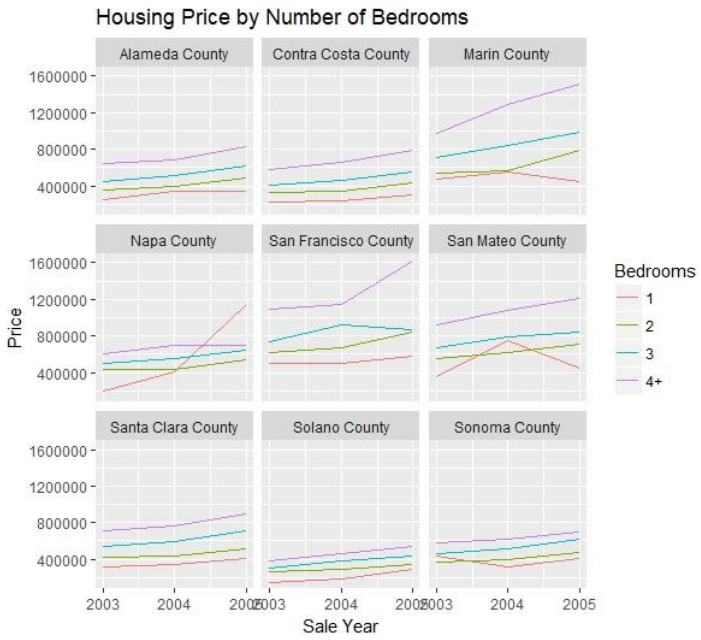
2) The sort function was used to find the closest and furthest date. For the first sale (2003-04-27) and last sale (2006-06-04), there is a total of 1,134 days difference between the two dates. The lubridate package was used to find the difference. The sort function was used again for the ‘year’ variable to find the nearest and furthest date for the home construction years. The oldest date is 1923, and the closest date is 2005 for a difference of 82 years.
3) This first plot shows the number of sales over time.



The plot seems to indicate that the greatest spike in sales happened around the middle-end of 2004. Additionally, it’s apparent again as it was during the initial overview of the ‘date’ variable, that during the beginning of each year the sales will begin to dip. It’s still showing the same indication with the month as the category from where the chart separates the dates. The home sales also seem to dip quite strongly after 2005, however the data is merely ending around this time so it’s not quite indicative of any sharp drop.



The second plot here shows the average price of homes per month. Here a sharp rise in home prices are visible from 2003 – 2006. It seems that during this period a bubble in the real estate prices were beginning to form due to the extreme growth in housing prices over such a short period of time. After prices nearly double from an average of half a million for a home to $800,000 for a home, nearly doubling in less than 5 years, it wouldn't make sense economically speaking for the people of this place to continue to be able to buy and sell at such a high rate. The income of people would likely begin to stop being capable of paying such prices for homes, and certain financial bubbles such as the subprime mortgage crisis may ensue.

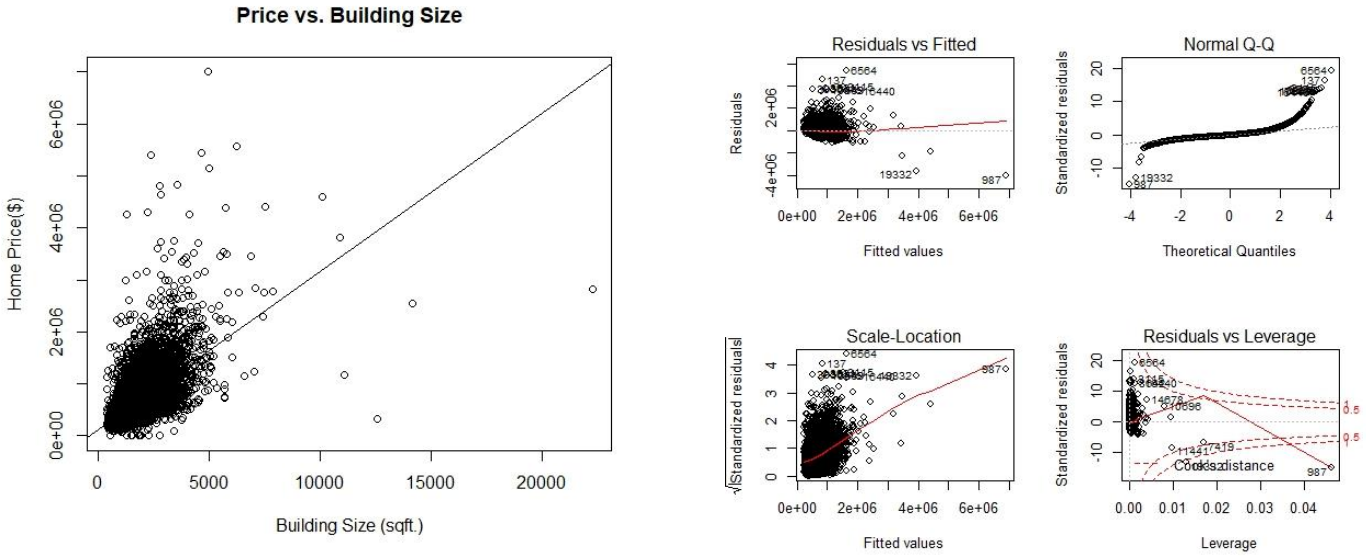


4) First the table function was used to see the range of bedrooms that the dataset seems to have. The number of bedrooms went from 1-28, with few being greater than 6. Another edit was made to the data from question (5), where the observations with county labeled as "Alpine County" were changed. Most counties follow a pattern where the larger the number of bedrooms means that the price of a home will be larger. Also, Marin County, San Francisco County, and San Mateo County seemed to experience the greatest growth in home price over the 3-year period.

5) Initially a table function was used to analyze each row of cities compared with counties to see if there were any duplicates. However, due to the size of the dataset, it seems that this method is time-consuming and prone to making errors. For this reason, a line of code was used to determine whether any duplicates existed. This line used several functions such as which(), rowSums(), table(), and unique(). This piece of code shows that San Francisco and Vallejo seem to have more than one county associated with this 'city' variable. After using View() to examine these subsets, it's clear that there are 19 observations that have the city 'San Francisco' while having the county 'Alpine County' also. By doing a quick Google search of the addresses, these 19 observations were shown to clearly originate from San Francisco County, and so their 'county' variables were changed to 'San Francisco County' instead. Additionally, 2 observations

from Vallejo were shown to originate also from Napa County. However, the city of Vallejo is part of Solano County. Doing a quick Google search on these addresses showed that these observations are from the city 'American Canyon.' However, this difference wasn't really in anyway changing the data significantly, so no change took place.

6) Below is a scatter plot of building size against price. A regression line is drawn through the data, where the building's size is being used to try and predict the price of the home. The points in the scatter plot starts on the bottom left, and begin to spread out to the right, however the direction varies from going to the top right versus just going upwards. So, the prices of homes seem to grow faster than the size of the homes. The next plot is created by plotting the summary() function of the lm() model that predicts the price with the building size.

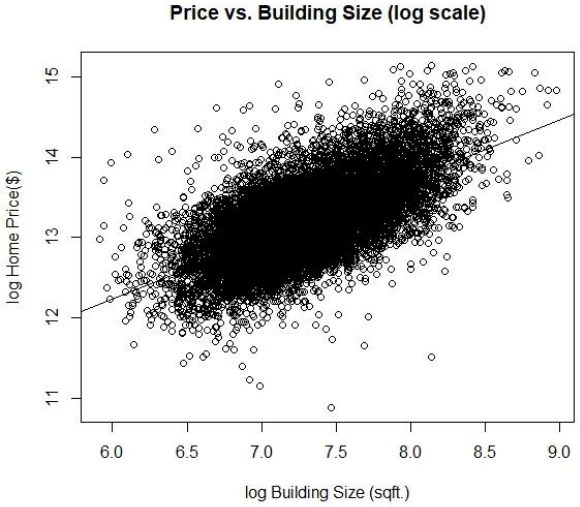


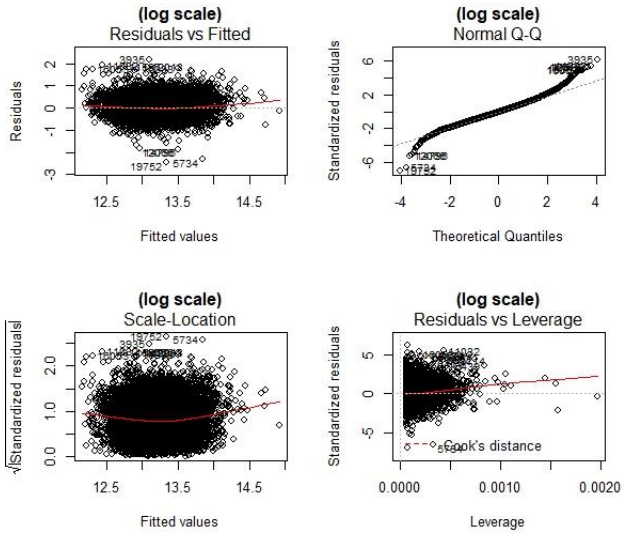
The 4 plots to the right of the scatter plot are different diagnostic tests to test the linear regression model. The plot to the top left labeled 'Residuals vs Fitted' ideally should be fairly spread out with no pattern. However, in this case the data is grouped to the left and along the middle to upper region. The variability is highly associated with the dependent variable. Therefore, there is some sort of non-linear relationship being left out of the model and being unexplained.

The 'Normal Q-Q' plot shows the normality of the residuals in the data. In the above plot, the residuals seem to be somewhat right-skewed along with being heavy-tailed. The 'Scale-Location' plot tests for equal variance, and through this plot it's possible to see that the data is heteroscedastic.

The 'Residuals vs Leverage' plot shows outliers that are influential to the linear regression plot. Here the plot seems to indicate that there are a few outliers which are acting as extreme observations which may be affecting the linear regression diagnostic. In the plot it shows that observation '987' is quite distant, which is an apartment complex consisting of 28 bedrooms. This observation is possibly worth excluding due to it being from an apartment sale, and not necessarily relate to the purpose of the San Francisco Bay Area housing price dataset (which may be referring to homes rather than entire apartments). Another outlier which is close to the Cook's distance is a home that is highly priced, but the

number of bedrooms is ‘NA.’ So this observation could also be temporarily dropped. Another observation is the observation indexed at 11441 which is a large apartment complex like ‘987.’





Above is a plot of the log scale version of the previous plot. This was accomplished by using the log() function on both the x and y axes of the data. The reason for taking the log of both sides was due to the scale of the dataset that is being used currently. The home prices go up to the millions of dollars, and the x-axis goes up to 20,000. The issue with the scale is that most of the dataset is grouped in the bottom left, which signifies that most of the data is on the lower range of the scale. Due to this grouping on the lower end of the range for both x and y axes, the linear regression diagnostic on the data can be affected due to this uneven distribution along a certain scale. Additionally, certain outliers were excluded from this subset. The data includes only ‘bsqft’ < 10,000 and ‘price’ < 4,000,000.

The scatterplot for this data appears much more linear due to the scale of the x and y axes being much more scaled to the area of where the data grouped within the dataset. Drawing the regression line through this data appears much more linear and capable of predicting new observations.

With the diagnostics plot, the ‘Residuals vs Fitted’ seems much more spread out than previously. Although the data doesn’t yet seem quite random, the variability has improved over the previous data. The ‘Normal Q-Q’ plot seems to have also improved slightly over the previous plot. This time the tails are lighter, and the skew doesn’t seem as noticeable in comparison to before. This time the ‘Scale-Location’ plot seems more normal than previously, so the variance appears to be more equal than before. The data after logging both the x and y axes seems to follow a homoscedastic trend in this instance according to the ‘Scale-Location’ plot.

The last plot, the ‘Residuals vs Leverage’ this time doesn’t have observations which are acting as extremes which are adversely affecting the linear regression model. The difference of there not being influential cases whereas in the first diagnostic there were several is interesting. The plot with the subset data and the logarithmic variables has changed the data such that there are no more extreme outliers from the point of view of the Cook’s distance.

After having done the modifications to the dataset, the linear regression model shows that the two variables work in having one predict the other. The variable 'bsqft' seems to be adequate at predicting the variable 'price.'

7) To compute the hypothesis test for $H_0: \beta_{bsqft} \geq \beta_{lsqft}$ vs. $H_1: \beta_{bsqft} < \beta_{lsqft}$, it is simpler to change both variables into one by performing the following changes to both sides.

$H_0: \beta_{bsqft} - \beta_{lsqft} \geq 0$, and $H_1: \beta_{bsqft} - \beta_{lsqft} < 0$.

Next let $\beta = \beta_{bsqft} - \beta_{lsqft}$.

So H_0 and H_1 become $H_0: \beta \geq 0$ and $H_1: \beta < 0$.

To perform the test statistic, $\frac{\beta - 0}{\sqrt{Var(\beta)}}$

Where it is assumed that $Var(\hat{\beta}_{bsqft} - \hat{\beta}_{lsqft}) = Var(\hat{\beta}_{bsqft}) + Var(\hat{\beta}_{lsqft})$.

Therefore, the test statistic can be rewritten as $\frac{\beta - 0}{\sqrt{Var(\hat{\beta}_{bsqft}) + Var(\hat{\beta}_{lsqft})}}$

Using the following line of code 'summary(lm(price~bsqft+lsqft,house))$coefficients,' it is possible to find the Standard Error for both 'bsqft' and 'lsqft'. Squaring both values and taking the square root of the sum it is possible to determine the value of the denominator for the test statistic. The estimates of both are also used for the numerator. From the table the following outputs are utilized:

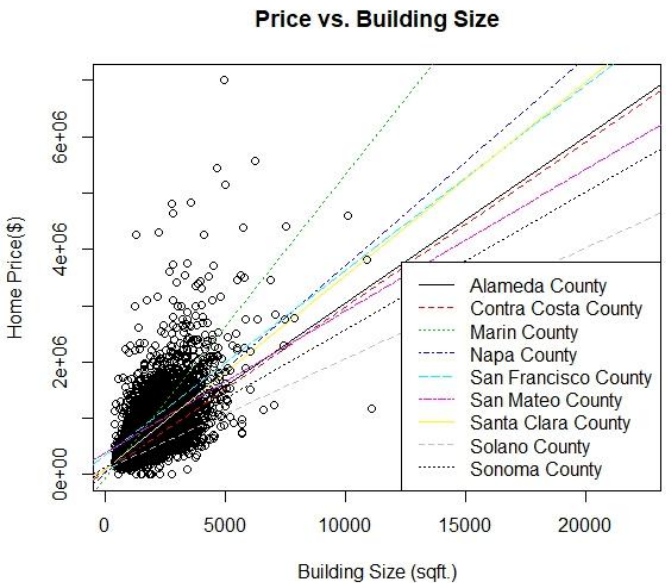
	Estimate	Std. Error
'bsqft'	306.6	3.021
'lsqft'	-0.0005951	0.000823

The value then equates to the following:

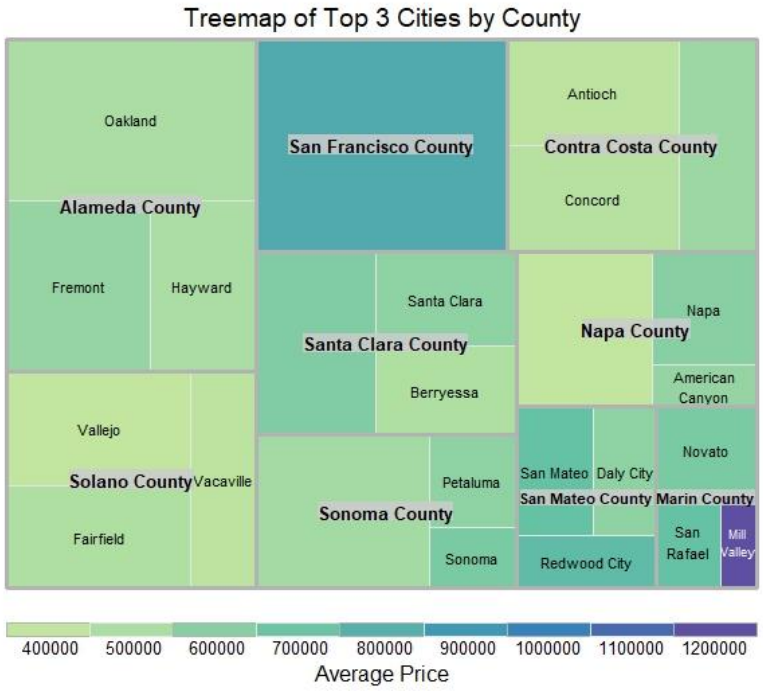
$$\frac{306.6 - (-0.0005951)}{\sqrt{9.126442}} = \frac{306.6006}{3.021} = 101.4898$$

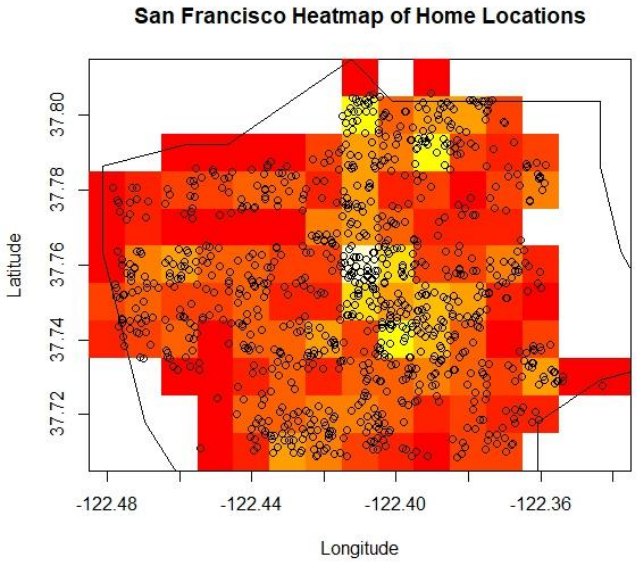
Our alternative hypothesis test follows a left-tailed test, and since the test statistic of 101.4898 is a positive number we fail to reject the null hypothesis. Therefore, the conclusion from this hypothesis test is that we are unable to state from the alternative hypothesis that the 'lsqft' is a greater coefficient than 'bsqft.'

8) Below is a plot of regression lines by county, where the housing price is trying to be predicted by building size. Through this plot it is evident that the county variable does greatly affect the prediction power of building size for housing price. The reason for this is that the slopes are different for each of the counties, therefore there are different effects depending on which county the homes are from. So, when trying to predict price based on building size, it would make sense to conclude that county is a confounding variable.

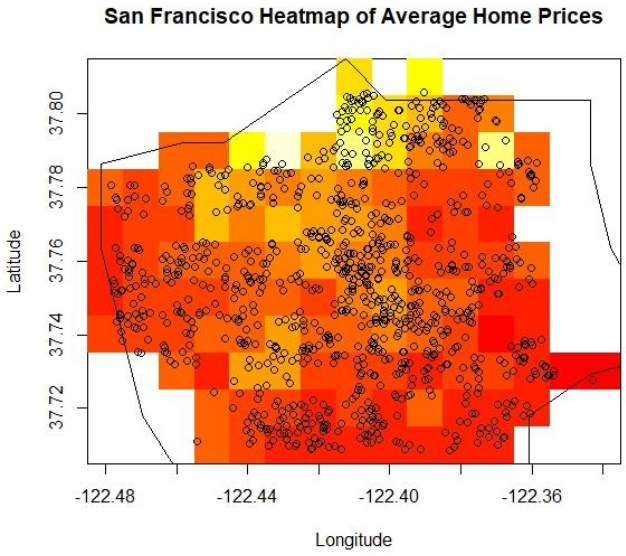


9) Below is a treemap of the top 3 cities according to the county that they are from. The first step was to properly subset the data so that the top 3 cities in terms of number of sales were found. The average price for these cities was later added. The counties with the most sales are shown by their relative sizes in comparison to the other counties. Alameda seems to be the largest county in comparison to the rest, taking the largest portion of the treemap. Most of the homes from Alameda County that are sold are from Oakland, the second most from Fremont, and third most are from Hayward. San Francisco County is an interesting observation, since the only city apparently is the city of San Francisco, making it the city with the most homes sold overall. It is also one of the more expensive cities for the average price of a home. However, the most expensive city is Mill Valley, where the average price is close to $1,200,000 per home.





10) Below is a heatmap of home locations within San Francisco. The black line represents the border of the city of San Francisco. The matrix shows that the greatest number of homes that are sold within the city come from the center around (37.76,-122.41). The number of homes sold seems to spread out from this center part of the city, with some density also appearing in the northern part of the city. The eastern and western borders don't seem to have as many real estate properties being sold either. It is also possible that there aren't as many buildings in these areas, so the number of homes sold would be fewer since those are areas along the beach. Additionally, points were added of the locations of the homes. The most dense collection of points is in the center.



To the left is another heatmap which represents the average price of homes that are sold in an area. This map seems to tell a different story than the previous map where most homes were sold around the center of the city. However, in this heatmap it becomes apparent that the most expensive homes are sold in the northern part of the city. The center of the city seems to have some cheaper prices, which makes sense since the downtown part of the city would likely have cheaper more affordable homes as well since there is a mixture of wealthy and poor people living there. Additionally, points were added of the locations of the homes. It seems that there are several homes in the southern part of San Francisco which are sold typically at lower prices.

Resources:

Other students: Bailey Wang, Minh Truong, Tiffany Chen

https://stackoverflow.com/questions/15680350/plot-a-histogram-without-zero-values-in-r

https://stackoverflow.com/questions/16194212/how-to-suppress-warnings-globally-in-an-r-script

op

http://ggplot2.tidyverse.org/reference/labs.html

https://www.statmethods.net/management/subset.html

https://stackoverflow.com/questions/6081439/changing-column-names-of-a-data-frame

http://www.rstudio.com/wp-content/uploads/2015/02/data-wrangling-cheatsheet.pdf

https://stackoverflow.com/questions/13967063/remove-duplicated-rows

https://stackoverflow.com/questions/27766054/getting-the-top-values-by-group

https://bookdown.org/lyzhang10/lzhang_r_tips_book/how-to-plot-data.html#creating-treemaps

https://www.r-graph-gallery.com/236-custom-your-treemap/

https://stackoverflow.com/questions/9617348/reshape-three-column-data-frame-to-matrix-long-to-wide-format

http://www.statisticshowto.com/homoscedasticity/

http://seankross.com/2016/02/29/A-Q-Q-Plot-Dissection-Kit.html

http://data.library.virginia.edu/diagnostic-plots/

http://www.statisticshowto.com/one-tailed-test-or-two/

https://www.stat.berkeley.edu/classes/s133/Lr-a.html

Code Appendix:

```
library(treemap) # libraries
library(lubridate)
library(lattice)
library(MASS)
library(stringr)
library(tidyverse)
library(data.table)
library(dplyr)
library(plyr)
library(maps)
```

```r
library(ggplot2)

setwd("C:/Users/qizhe/Desktop/STA 141A/HW2") # set working directory

house <- readRDS("housing.rds") # give a name to the data

# 1
### beginning of question 1 code
# change variables to proper types
str(house)
house$zip <- as.factor(house$zip)
house$year <- as.numeric(house$year)

# Fix county to County using stringr's str_replace()
table(house$county) # lowercase counties
house$county <- gsub(house$county, pattern = "county", replacement = "County
") # fix county to County
house$county <- gsub(house$county, pattern = "Franciscoe", replacement = "Fra
ncisco") # fix Franciscoe to Francisco

# possible values to plot: zip, price, br, lsqft, bsqft, year, date, long, la
t
# zip
max(as.numeric(house$zip), na.rm = T) # all are within the required 9----
min(as.numeric(house$zip), na.rm = T)
table(house$zip)
any(sapply(house$zip, function(x) nchar(as.character(x))) != 5, na.rm=TRUE) #
 no zip code not in 5 digits

# price
hist(house$price, xlab = "House Price", main = "Histogram of Home Prices")
boxplot(house$price, ylab = "House Price", main = "Boxplot of Home Prices")
max(house$price, na.rm = T)
sort(house$price, decreasing = T)[1:10] # the maximum is not too far from the
 rest of the data
sort(house$price, decreasing = F)[1:10] # certain values are 0, so they shoul
d be set to NA
house <- house %>% # set unlikely value to NA
  mutate(price = ifelse(price == 0, NA, price))
house[which(house$price == 7000000),]
# https://www.zillow.com/homedetails/16-Skyland-Way-Ross-CA-94957/19273018_zp
id/
# Some information such as price seem accurate, however other information is
different than
# what is listed. The owner can easily alter the home over time so these othe
r differences are ignored.
house[which(house$price == 5574000),]
# https://www.zillow.com/homedetails/138-Gilmartin-Dr-Tiburon-CA-94920/192617
46_zpid/
# According to the website, the price seems accurate
```

```r
house[which(house$price == 5434000),]
# https://www.zillow.com/homedetails/11-Southwood-Ave-Ross-CA-94957/19273220_zpid/
# Again the information seems accurate
house[which(house$price == 5400000),]
# https://www.redfin.com/CA/Mill-Valley/43-Park-Ave-94941/home/911182
# This price actually seems incorrect according to this listing
house <- house %>% # set unlikely value to NA
  mutate(price = ifelse(price == 5400000, NA, price))

# br
hist(house$br, xlab = "Bedrooms", main = "Number of Bedrooms")
max(house$br, na.rm = T)
sort(house$br, decreasing = T)[1:10] # one is considerably larger than the others
sort(house$br, decreasing = F)[1:10]
house[which(house$br == 28),]
# https://www.redfin.com/CA/Redwood-City/75-Duane-St-94062/home/1741841
# the website shows that it's an apartment complex
house[which(house$br == 16),]
# https://www.redfin.com/CA/Rohnert-Park/909-Kirsten-Ct-94928/home/2544629
# the website shows that it's only an 8br multi-family home
house[which(house$br == 12),]
# https://www.movoto.com/oakland-ca/8290-macarthur-blvd-oakland-ca-94605/pid_rrw8uaf38g/
# the website shows that one is a 12 br quadruplex, so these sizes are possible

# lsqft
hist(house$lsqft, xlab = "Lot Square feet", main = "Histogram of Home Lot Size")
boxplot(house$lsqft, main = "Boxplot Lot Size of Home")
max(house$lsqft, na.rm = T)
sort(house$lsqft, decreasing = T)[1:10]
sort(house$lsqft, decreasing = F)[1:10]
house[which(house$lsqft == 313632000),]
# https://www.zillow.com/homedetails/2118-Peppertree-Way-APT-2-Antioch-CA-94509/18316652_zpid/
# The property doesn't look so large, likely an error
house[which(house$lsqft == 65340000),]
# https://www.redfin.com/CA/Walnut-Creek/1904-Ptarmigan-Dr-94595/unit-1/home/888404
# The property doesn't look so large, likely an error
house[which(house$lsqft == 51399998),]
# https://www.trulia.com/p/ca/san-francisco/439-greenwich-st-9-san-francisco-ca-94133--2083094246
# The property is only on a small area of a street so there's an error
house[which(house$lsqft == 25),]
# https://www.zillow.com/homedetails/318-Shirley-St-Graton-CA-95444/15830352_zpid/
```

```r
house <- house %>% # set unlikely value to NA
  mutate(lsqft = ifelse(street == "318 Shirley Street", NA, lsqft))
house[which(house$lsqft == 30),]
# https://www.redfin.com/CA/San-Francisco/39-Scott-St-94117/home/1349981
# Actually shown as 29 lsqft
house[which(house$lsqft == 100),]
# https://www.zillow.com/homedetails/648-Ridgewood-Ave-Mill-Valley-CA-94941/94645334_zpid/
# Shown to be an error
house <- house %>% # set unlikely value to NA
  mutate(lsqft = ifelse(street == "648 Ridgewood Avenue", NA, lsqft))

# bsqft
hist(house$bsqft, xlab = "Building Size in Sqft.", main = "Histogram of Building Size")
boxplot(house$bsqft, ylab = "Building Size in Sqft.", main = "Boxplot of Building Size")
max(house$bsqft, na.rm = T)
sort(house$bsqft, decreasing = T)[1:10]
sort(house$bsqft, decreasing = F)[1:10]
house[which(house$bsqft == 22266),]
# Same as before, it's an apartment complex
house[which(house$bsqft == 14149),]
# https://www.redfin.com/CA/San-Francisco/1010-Gough-St-94109/home/979662
# The website show's that it's a multi-family home with this actual bsqft
house[which(house$bsqft == 12582),]
# https://www.redfin.com/CA/Oakland/1009-E-22nd-St-94606/home/1398101
# The link shows that it's only 2,582 bsqft, there's an extra '1' added
house <- house %>% # set unlikely value to NA
  mutate(bsqft = ifelse(bsqft == 12582, NA, bsqft))
house[which(house$bsqft == 11064),]
# Same as before, it's a multi-family home with correct bsqft
house[which(house$bsqft == 10898),]
# https://www.redfin.com/CA/Oakland/4609-Rising-Hill-Ct-94619/home/1820925
# The link also has the same information
house[which(house$bsqft == 370),]
# No information on bsqft online
house[which(house$bsqft == 380),]
# https://www.zillow.com/homedetails/579-Beresford-Ave-Redwood-City-CA-94061/58660857_zpid/
# Information online is quite different
house[which(house$bsqft == 381),]
# https://www.zillow.com/homedetails/1412-Wisner-Dr-Antioch-CA-94509/18305299_zpid/
# Very different information online

# year
class(house$year) # character type, must change to workable format
options(warn=-1) # prevent warnings
# https://stackoverflow.com/questions/15680350/plot-a-histogram-without-zero-
```

```r
values-in-r
table(is.na(house$year)) # over 17% of the observations have an NA for year
(3507 NA's)
# https://stackoverflow.com/questions/16194212/how-to-suppress-warnings-globa
lly-in-an-r-script
options(warn=0) # turn warnings back on
sort(house$year, decreasing = T)[1:10] # Several impossible values at the top
sort(house$year, decreasing = F)[1:20] # Several impossible values at the bot
tom
house <- house %>% # set impossible year values to NA
  mutate(year = ifelse(year < 1800 | year > 2020, NA, year))
table(is.na(house$year)) # 3526 NA's, 19 impossible years

# date
sort(year(house$date))[1:10] # no irregular years/months/days
sort(year(house$date), decreasing = T)[1:10]
sort(month(house$date))[1:10]
sort(month(house$date), decreasing = T)[1:10]
sort(day(house$date))[1:10]
sort(day(house$date), decreasing = T)[1:10]

plot(house$date, xlab = "Date Sold", ylab = "Frequency", main = "Date of Home
 Sales")
# pattern of houses not being sold around the beginning of the year

# long
boxplot(house$long, ylab = "Longitude of Home", main = "Boxplot of Longitudes
") # boxplot
sort(house$long, decreasing = T)[1:10] # examine extreme values
sort(house$long, decreasing = F)[1:10]
house[which(house$long == 0),] # identify the outlier
house <- house %>% # set the outlier to NA
  mutate(long = ifelse(long > -120, NA, long))
boxplot(house$long, ylab = "Longitude of Home", main = "Boxplot of Longitudes
") # data now apppears more regular

# lat
boxplot(house$lat, ylab = "Latitude of Home", main = "Boxplot of Latitudes")
# boxplot
sort(house$lat) # examine extreme values
house[which(house$lat == 0),] # same observation as previously
house <- house %>% # set the outlier to NA
  mutate(lat = ifelse(lat < 30, NA, lat))
boxplot(house$lat, ylab = "Latitude of Home", main = "Boxplot of Latitudes")
# data now apperas more regular

# duplication
sum(duplicated(house)) # there are duplications in the data
house[duplicated(house),]
house[house$street == "1882 48th Avenue", ]
```

```r
house = house[-which(duplicated(house)),] # remove duplciation

### end of question 1 code

# 2
sort(house$date)[1:10] # starts at 4/27/03
sort(house$date, decreasing = T)[1:10] # ends at 6/4/06
first_sale <- ymd("2003-04-27") # convert both to units that can be subtracte
d
last_sale <- ymd("2006-06-04")
last_sale - first_sale # 1134 days difference

class(house$year)
sort(house$year)[1:10] # starts in 1923
sort(house$year, decreasing = T)[1:10] # ends at 2005
2005-1923 # 82 years of home construction

# 3
# order dates
house$date_ym <- format.Date(house$date, "%Y-%m") # formatted date
house$year <- strptime(house$year, "%Y")

date_table <- table(house$date_ym) # table the month-year data
avg_price <- aggregate(price ~ date_ym, house, mean, na.rm = T) # take the av
erage of the prices
avg_price$ym <- paste(avg_price$date_ym, "-01") # include an extra day variab
le to signify the first of the month

date_aggregate <- as.Date(avg_price$ym, "%Y-%m -%d") # change to Date class

plot(x=parse_date_time(names(date_table),"y-m"),y=date_table, type = 'b',
     xlab = "Date of Sales", ylab = "Number of Sales", main = "Sales per Mont
h") # number of sales
plot(date_aggregate, avg_price$price, type = "l", xlab = "Date of Sales",
     ylab = "Average Price($)", main = "Average Price per Month") # average p
rice

# 4
table(as.factor(house$br)) # check bedrooms
house$br <- as.integer(house$br) # change to integer

house <- house %>% # set Alpine County cities to San Francisco (from problem
5)
  mutate(county = ifelse(county == "Alpine County", "San Francisco County", c
ounty))

house$sale_Year <- year(house$date) # subset sale years 2003 - 2005
house05 <- subset(house, sale_Year < 2006) # remove years less beyond 2005
house05$Bedrooms <-cut(house05$br, breaks = c(0,1,2,3,Inf),
                       labels = c("1", "2", "3", "4+")) # set bedroom levels
```

```r
br_avg <- aggregate(price ~ county + Bedrooms + sale_Year, house05, mean, na.rm = T) # take average price
xticks = c(2003, 2004, 2005) # fix x-labels
# http://ggplot2.tidyverse.org/reference/labs.html
ggplot(br_avg, aes(x = sale_Year, y = price, col = Bedrooms)) + geom_line() + # plot the counties
  facet_wrap(~ county) + scale_x_continuous(breaks = xticks) +
  labs(x = "Sale Year", y = "Price") + labs(title = "Housing Price by Number of Bedrooms")

# 5
table(house$city, house$county) # draw a table of city/county table(city,county)
which(rowSums(table(unique(house[c("city","county")])))>1) # duplicates in Vallejo, San Francisco

# https://www.statmethods.net/management/subset.html
alpsf <- subset(house, city == "San Francisco" & county == "Alpine County") # examine the Alpine County data
# 19 observations that have this mistake
vall <- subset(house, city == "Vallejo") # examine the Vallejo data
# 2 errors in vallejo/napa/solano
# https://www.redfin.com/CA/American-Canyon/28-Spikerush-Cir-94503/home/12222859
# http://www.loopnet.com/Listing/15529307/3860-Broadway-Drive-American-Canyon-CA/
# Belvedere/tirburon, belvedere/Tiburon (mislabeled data)

# 6
plot(house$bsqft, house$price, xlab = "Building Size (sqft.)", # scatter plot of price and bsqft
     ylab = "Home Price($)", main = "Price vs. Building Size")
abline(lm(price ~ bsqft, data = house[-1])) # regression line
houselm <- lm(price ~ bsqft, data = house[-1]) # examine the summary
summary(houselm)
par(mfrow=c(2,2)) # allow for 2x2 plots
plot(lm(house$price ~ house$bsqft),id.n=10) # plot the lm diagnostic
# examine points 987, 19332
house[987,] # apartment complex
# https://www.redfin.com/CA/Redwood-City/75-Duane-St-94062/home/1741841
# the website shows that it's an apartment complex
house[19332,]
# https://www.redfin.com/CA/Oakland/1009-E-22nd-St-94606/home/1398101
# The link shows that it's only 2,582 bsqft, there's an extra '1' added
house <- house %>% # set unlikely value to NA
  mutate(bsqft = ifelse(bsqft == 12582, NA, bsqft))
dev.off()

house_normal <- subset(house, bsqft < 10000 & price < 4000000) # subset data according to bsqft and price
```

```r
house_normal$pos_price <- ifelse(house_normal$price <= 0, NA, house_normal$pr
ice) # set 0's to NA for log()
plot(log(house_normal$bsqft), log(house_normal$pos_price), xlab = "log Buildi
ng Size (sqft.)", # scatterplot of log data
     ylab = "log Home Price($)", main = "Price vs. Building Size (log scale)
")
abline(lm(log(price + 1) ~ log(bsqft), data = house_normal[-1])) # regression
 line
houselm2 <- lm(log(price + 1) ~ log(bsqft), data = house_normal[-1]) # examin
e summary
summary(houselm2)
par(mfrow=c(2,2)) # allow for 2x2 plots
plot(lm(log(house_normal$pos_price) ~ log(house_normal$bsqft)), id.n = 10, ma
in = "(log scale)") # plot the lm diagnostic
dev.off()

# 7
# H_0
# B_bsqft >= B_lsqft
# B_bsqft - B_lsqft >= 0
# H_1
# B_bsqft - B_lsqft < 0
# let B = B_bsqft - B_lsqft
# So H_0 is B >= 0
# and H_1 is B < 0
# conclusion and test statistic
# test statistic:
# B - 0 / sqrt(Var(B))
# sqrt(Var(B)) = sqrt(Var(B_bsqft - B_lsqft)) = sqrt(Var(B_bsqft) + Var(B_lsq
ft))
summary(lm(price~bsqft+lsqft,house))$coefficients # examine the std. errors a
nd estimates ofr both variables

# 8
split_counties <- split(house, house$county) # separate data according to cou
nty
lm_split_counties <- lapply(split_counties, function(x) lm(price ~ bsqft, dat
a = x)) # linear model for each county
plot(house$bsqft, house$price, xlab = "Building Size (sqft.)", # plot buildin
g size against prices
     ylab = "Home Price($)", main = "Price vs. Building Size")
sapply(1:9, function(x) abline(coef(lm_split_counties[[x]]), col = x, lty = c
(1:6,1:3)[x] )) # draw reg. line per county
legend("bottomright", names(table(house$county)), col = 1:9, lty = rep_len(1:
6, 9)) # include legend of lines

# 9
house$county <- as.factor(house$county) # change into factor levels
sales_per_city <- t(t(table(house$city))) # get the count of sales per city
sales_per_city <- as.data.frame(sales_per_city) # make the data into a useabl
```

```r
e format
sales_per_city <- sales_per_city[,-2] # remove unecessary column

# https://stackoverflow.com/questions/6081439/changing-column-names-of-a-data
-frame
colnames(sales_per_city) <- c("city", "sales") # add labels

house$county <- as.character(house$county)
cities <- house[,c(1,2)] # create new dataframe with cities and county

# http://www.rstudio.com/wp-content/uploads/2015/02/data-wrangling-cheatshee
t.pdf
cities <- arrange(cities,city) # order by city

sales_per_city <- sales_per_city %>% # combine both through joins
  left_join(cities, by = "city")

# https://stackoverflow.com/questions/13967063/remove-duplicated-rows
sales_per_city <- sales_per_city[!duplicated(sales_per_city),] # drop duplica
tes

# https://stackoverflow.com/questions/27766054/getting-the-top-values-by-grou
p
sales_per_city <- sales_per_city %>% # subset the top 3 per county
  group_by(county) %>%
  top_n(n = 3, wt = sales)

sales_per_city <- sales_per_city %>% # orders counties alphabetically and acc
ording to highest price
  group_by(sales) %>%
  arrange(county)

avg_city <- aggregate(price ~ city, house, mean, na.rm = T) # average price p
er city

sales_per_city <- sales_per_city %>% # combine both through joins
  left_join(avg_city, by = "city")

colnames(sales_per_city) <- c("City", "Sales", "County", "Average Price") # a
dd labels

# https://bookdown.org/lyzhang10/lzhang_r_tips_book/how-to-plot-data.html#cre
ating-treemaps
# https://www.r-graph-gallery.com/236-custom-your-treemap/
treemap(dtf = sales_per_city, # create treemap
        index = c("County", "City"),
        vSize = "Sales", # number of sales
        vColor = "Average Price", # average price
        type = "value",
        palette = "Spectral",
```

```
        border.col = c("grey70", "grey90"),
        fontsize.labels = c(10, 8),
        title = "Treemap of Top 3 Cities by County")

# 10
# 1. A heatmap that is colorized based on the number of houses in a cell
sfhouse <- subset(house, city == "San Francisco") # subset SF
sfhouse$long2 <- round(sfhouse$long, 2) # round values
sfhouse$lat2 <- round(sfhouse$lat, 2)
long_range<-range(sfhouse$long2,na.rm=TRUE) # create overall range for longit
ude/latitude
lat_range<-range(sfhouse$lat2,na.rm=TRUE)
sfhouse$longF <- factor(sfhouse$long2,levels=seq(long_range[1],long_range
[2],.01)) # create factor levels
sfhouse$latF <- factor(sfhouse$lat2,levels=seq(lat_range[1],lat_range[2],.0
1))
sf_matrix <- table(sfhouse$longF,sfhouse$latF) # create table matrix of longi
tude/latitude data
sf_matrix[sf_matrix == 0] <- NA # set 0's to NA (Patrick's advice)
image(x = as.numeric(levels(sfhouse$longF)) + .03, y = as.numeric(levels(sfho
use$latF)), z = sf_matrix, #create heatmap
      xlab = "Longitude", ylab = "Latitude", main = "San Francisco Heatmap of
 Home Locations")
CA_boarder=map('county',plot=FALSE); lines(CA_boarder$x,CA_boarder$y) # draw
border of San Francisco
points(x=sfhouse$long + .03,y=sfhouse$lat)

# 2. A heatmap that is colorized by the average price of the houses within a
cell
longlat_avg <- aggregate(price ~ longF + latF, sfhouse, mean, na.rm = T) # av
erage per longitude/latitude interval
# https://stackoverflow.com/questions/9617348/reshape-three-column-data-frame
-to-matrix-long-to-wide-format
sf_avg_map <- daply(longlat_avg, .(longF,latF), function(x) x$price) # set da
ta into proper matrix
image(x = as.numeric(levels(sfhouse$longF)) + .03, y = as.numeric(levels(sfho
use$latF)), z = sf_avg_map, #create heatmap
      xlab = "Longitude", ylab = "Latitude", main = "San Francisco Heatmap of
 Average Home Prices")
CA_boarder=map('county',plot=FALSE); lines(CA_boarder$x,CA_boarder$y) # draw
border of San Francisco
points(x=sfhouse$long + .03,y=sfhouse$lat)
```